AIRLINES RESERVATION SYSTEM

Submitted in partial fulfillment of the requirements
for the award of the degree of

Bachelor of Computer Applications

To

Guru Gobind Singh Indraprastha University, Delhi

Guide:

Ms Shilpa Taneja
Assistant Professor(IT)

Submitted by:

1. Harish Singh Bisht (00221102010)
2 . Geetanjali (01121102010)

Institute of Information Technology & Management,
New Delhi – 110058
Batch (2010-2013)

Certificate

We, 1. Harish Singh Bisht 00221102010 & 2. Geetanjali 01121102010 certify that the Minor Project Report (BCA-355) entitled Airlines Reservation system is done by us and it is an authentic work carried out by us at Institute of Information Technology And Management. The matter embodied in this project work has not been submitted earlier for the award of any degree or diploma to the best of my knowledge and belief.

1. Signature of the Student

2. Signature of the Student

Date:

Certified that the Project Report (BCA-355) entitled "Airlines Reservation System" done by the above students is completed under my guidance.

Signature of the Guide
Date:

Name of the Guide:
Designation:

Countersigned

Director

Acknowledgement

We would like to thank our Guide, Ms. Shilpa Taneja, for without her constant presence, support, and trust this project could not have been what it is today. Thank you for being there for us and patiently handling our slow progress.

Also, we would like to thank Ms. Suman Singh, who helped us through the initial hurdles of the project, and was there for us when we faltered.

Sincere and heart-felt thanks

Harish Singh Bisht
(00221102010)

Geetanjali
(01121102010)

TABLE OF CONTENTS

S No	Topic	Page No
1	Certificate	i
2	Acknowledgements	ii
3	List of Tables/Figures/Symbols	iii, iv, v
4	Chapter-1: Introduction	6
	1 Description of Organization	6
	1.1 Introuction	6
	1.2 History of Organization	6
	1.3 Objective of Organization	6
	1.4 Organizational Structure	7
	1.5 Key Result Areas	8
	1.6 Functions	8
	2 Software Requirement Specifications	10
	2.1 Introduction	10
	2.1.1 SRS Used in the project	10
	2.2 Purpose	10
	2.3 Scope	10
	2.4 Definition, acronyms, abbreviations	10
	2.5 References	11
	2.6 Overview	11
	2.7 Overall description of proposed system	11
	2.8 Product Perspective	12
	2.9 System Interfaces	12
	2.10 Interfaces	12
	2.10.1 Hardware Interfaces	12
	2.10.2 Software Interfaces	12
	2.10.3 Communication Interfaces	12
	2.11 Memory Constraints	13
	2.12 Operations	13
	2.13 Site Adaptation Requirement	13
	2.14 Product functions	13
	2.15User Characteristics	13
	2.16 Constraints	13
	2.17 Assumptions and Dependencies	13
	2.18 Apportioning Requirement	14
	2.19 Specific Requirements	14
	2.20 External Interfaces	14
	2.21 User Interfaces	14
	2.22 Hardware Interfaces	15
	2.23 Software interfaces	16
	2.24 Communication Interfaces	16
	2.25 System Features	16
	2.26 Performance Requirements	18

	2.27 Logical Database Requirements	18
	2.27.1 Flight Database Table	18
	2.27.2 Customer & Reservation Database Table	19
	2.27.3 Login Database Table	20
	2.28 Design Constraints	20
	2.29 Software System Attributes	20
	2.30 Other Requirements	20
	3 Gantt chart	21
5	Chapter 2: System Design	24
	2.1 Physical Design	24
	2.1.1 Block Diagram	24
	2.1.2 Use case diagram	25
	2.1.3 Data Flow Diagram	26
	2.1.3.1 DFD 0 Level	26
	2.1.3.1 DFD 1 Level	27
	2.1.4 Entity Relationship Diagram	28
	2.2 Database Design	29
	2.2.1 Login Database	29
	2.2.2 Flight Database	29
	2.2.3 Customer & Reservation Database	30
	2.3 Interface Design	31
	2.3.1 Main Form	31
	2.3.2 Customer information form	33
	2.3.3 Ticket Form	34
	2.3.4 Flight Entry Form	36
6	Chapter-3: Systems Development & Implementation	37
	3.1 Program Development	37
	3.1.1 Login Form	37
	3.1.2 Main Form	41
	3.1.3 All Flight Information Form	49
	3.1.4 Flight Form	54
	3.1.5 Customer Information Form	55
	3.1.6 Ticket Form	60
	3.1.7 Main Form (for cancellation)	63
	3.1.8 Flight Entry Form	66
	3.1.9 Database Treeview Form	78
	3.2 Testing and Debugging	85
	3.2.1 Login Form	86
	3.2.2 Main Form	87
	3.2.3 Customer Information Form	88
7	Chapter-4: Scope of Improvement, Summary and Conclusions	89
	4.1 Objectives of the project	89
	4.2 Scope of Improvement	89

	4.3 Limitations	89
8	Appendices	90

LIST OF TABLES

Table No	Title	Page No
1	Flight Database Table	18
2	Customer & Reservation Database Table	19
3	Login Database Table	20

LIST OF FIGURES

Figure No	Title	Page No
1	Block Diagram	24
2	Use case diagram	25
3	DFD 0 Level	26
4	DFD 1 Level	27
5	Entity Relationship Diagram	28

LIST OF SYMBOLS

S No	Symbol	Nomenclature & Meaning
1	+	Addition
2	-	Subtraction
3	*	Multiplication
4	<	Less then
5	>	Greater then
6	&	Ampersand
7	;	Semicolon
8	:	Colon

Chapter 1: Introduction

1 Description of Organization

1.1 Introduction

The **airline reservations system (ARS)** 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

1.2 History of Organization

The airline reservations system (ARS) 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.

Jet Airways is the largest Indianairline based out of Mumbai, Maharashtra. It operates over 400 flights daily to 76 destinations worldwide. Its main hub is Mumbai, with secondary hubs at Delhi, Kolkata, Chennai, Cochin, Ahmedabad, and Bengaluru. It has an international hub at Brussels Airport, Belgium. Jet Airways is owned by NareshGoyal. Jet Airways's head office is located in the Siroya Centre in Andheri, Mumbai Jet Airways serves 52 domestic destinations and 24 international destinations, a total of 76 in 19 countries across southern Africa, Asia, Europe and North America. Short-haul destinations are served using Boeing 737 Next Generation. ATR 72-500s are used only on domestic regional routes, while long-haul routes are served using its Airbus A330-200 and Boeing 777-300ER aircraft. London, England was the airline's first long-haul destination and was launched in 2005.

1.3 Objective of Organization

Jet Airways will achieve these objectives:-

1. simultaneously ensuring consistent profitability
2. achieving healthy, long-term returns for the investors
3. Providing its employees with an environment for excellence and growth.
4. Providing Comforts in all zones while traveling to the customers.
5. Giving a quick response on any query raised either by employee or by customer.

6. Achieving goal in hanging up with customer in a increasing percentage year by year
7. They are trying to upgrade the concept of domestic airline travel to be a world class domestic airline.

1.4 Organizational Structure

Jet Airways is led by a dynamic, extremely talented and experianced team:

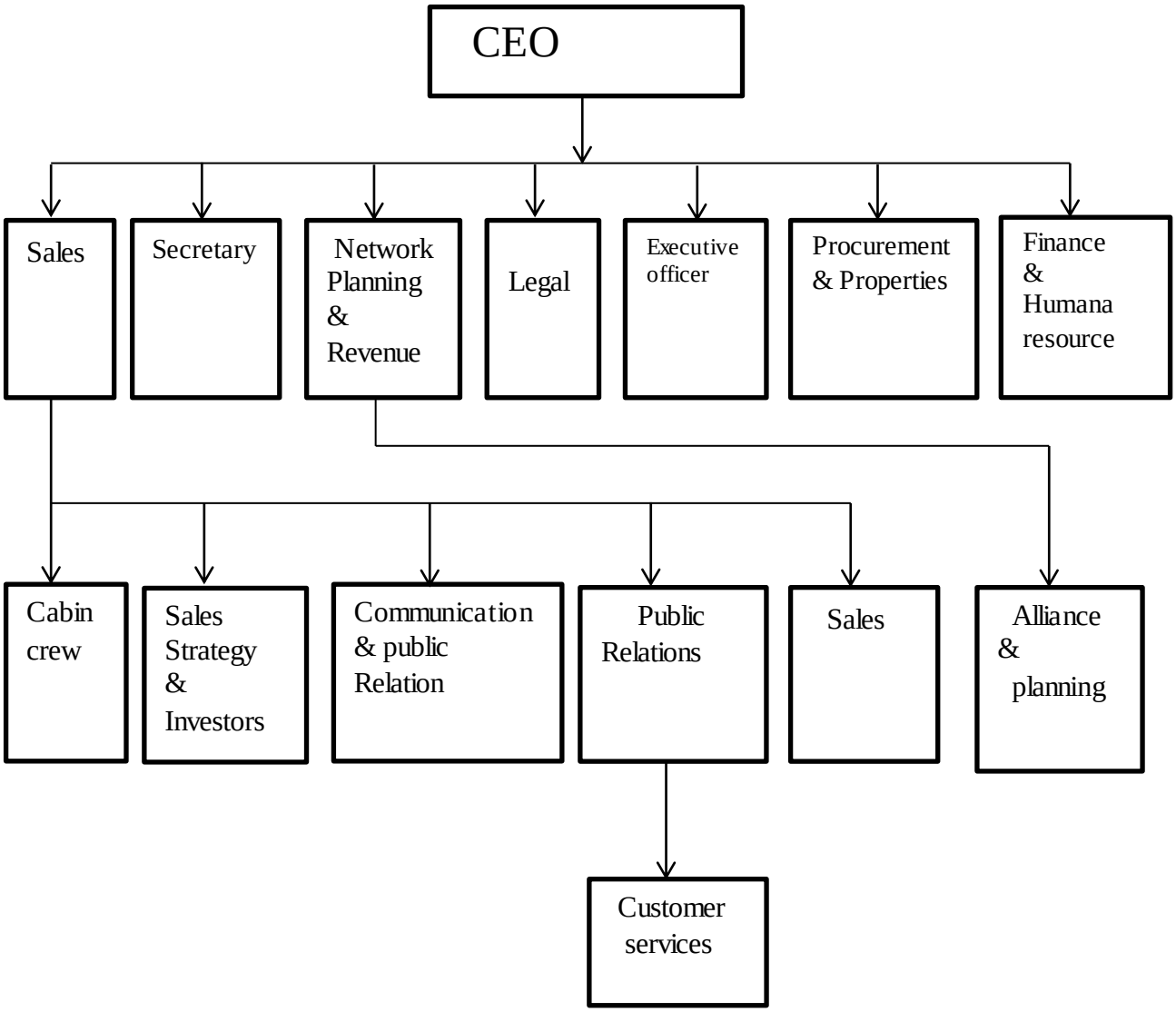


Diagram of Organizational Structure

1.5 Key Result Areas

a. **On Ground Services**: It is the process of booking ticket or checking in for flight, It ensures that every need on the ground is met.

b. **Check-in options** : Jet Airways offer multiple check-in options.

c. **Airport Lounges** : Jet Privilege Silver, Gold or Platinum card member or a Club Premiere passenger, can relax and enjoy complimentary snacks and beverages in jet Airways' plush airport lounges.

d. **Coach Services** : Airport Authority of India (A. A. I.) operates shuttle coaches for transit passengers from domestic to international airport and vice-versa at Mumbai and Delhi airports.

e. **Complimentary Chauffeur Drive :** A service specially for PREMIERE passengers traveling between Mumbai/Delhi and London (Heathrow).

f. **In-Flight Services** : Jet Airways continually endeavours to better its services, both on the ground and in the air. From crew, whose priority is passengers' comfort to the safety standards enforced to ensure that one is free of worr

1.6 Functions

a. **Online ticketing Reservation at jetairways.com: It Book**, pay and print your tickets instantly. Also, book multiple sectors for domestic and international flight at jetairways.com.

b. **IVR Interactive Voice Response: It** Book and pay for your tickets through our 24x7 call center over a secure IVR, and get your tickets via e-mail.

c. **Mobile Ticketing with Jet Wallet:** Book, pay and generate your eTicket. SMS 'Jet Wallet' to 56388 to download Jet Wallet on your GPRS phone.

d. **Pay Online service:** Book your ticket at Jet Airways reservation office and pay online at jetairways.com.

e. **Web Check-in:** Select your seat, print your boarding pass and proceed directly for security check.

f. **SMS Check-in**: Check-in anytime, anywhere and avail of a confirmed seat number on your mobile phone.

g. **Kiosk Check-in:** New age Kiosks at select airports in India help you select your seat and print your boarding pass at the touch of a screen.

h. Bus service: To make travel simpler, Jet Airways now provides a bus service to transport guests travelling to Dammam via Bahrain and vice-versa through King Fahad Causeway

i. **Secure Flight Passenger Data:** The United States Transportation Security Administration (TSA) has introduced a new 'Secure Flight Passenger Data' system for all guests travelling to / from the United States Of America.

The 'Secure Flight Passenger Data' system is also applicable to all guests travelling to / from / within Canada and flying over continental U.S.

Transportation Security Administration (TSA) requires you to provide the below information:

- Full Name (Your name as it appears on your passport)
- Date of Birth
- Gender
- Redress Number (if available)

j. **IVR Ticketing :Interactive Voice Response (IVR) based Payment and Ticketing**

Jet Airways' IVR based payment and ticketing service is the latest booking facility through our 24*7 call centre. Now book and pay for your eTickets over an exclusively customized and secure Interactive Voice Response (IVR) system.

Our IVR service allows you to complete your reservation, pay using credit cards through a secure gateway and instantly receive your eTickets via e-mail.

Once you confirm that you want to pay through the contact centre, our contact centre executive will transfer you to a secure IVR system that will request you to enter your credit card details. On a successful authorization of the credit card, you will get an automated response and your

2 Software Requirement Specifications

2.1 Introduction

A software requirements specification (SRS) is a complete description of the behavior of the system to be developed. This Software Requirement Specification is written in accordance with the IEEE STD 830-1998 model.

2.1.1 SRS Used in the project

The following subsections of Software Requirement Specifications Document should facilitate in providing the entire overview of the Information system "Airlines Reservation System" under development. This document aims at defining the overall software requirements for Passengers . Efforts have been made to define the requirements of the Information system exhaustively and accurately.

2.2 Purpose

The main purpose of Software Requirement Specifications Document is to describe in a precise manner all the capabilities that will be provided by the Software Application "Airlines Reservation System". It also states the various constraints which the system will be abide to. This document further leads to clear vision of the software requirements, specifications and capabilities. These are to be exposed to the development, testing team and end users of the software

2.3 Scope

- **a.** Airline Reservation System make the life of passengers very easy as they don't need to stand in queues for getting their seats reserved and they can easily make reservations on any airline just from our single system.
- **b.** It will also remove an extra burden from the Airline Department as most of the passengers and travel agencies use this service instead of making reservations from the counters.
- **c.** With the help of our system, customers can view all the different flight's availability with different timings for a particular date and it also allows them to reserve a seat, cancel a reservation or modify it.
- **d.** It can helps the customers and on the other, it will 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 fight due to some reason, the passengers are promptly informed.

2.4 Definition, acronyms, abbreviations

DBA : Database Administrator
ARS : Airlines Reservation System
IEEE : The Institute of Electrical and Electronics Engineers
SRS: Software Requirements Specification

2.5 References

1. jet airways
2. yatra.com

2.6 Overview

The rest of this SRS document describes the various system requirements, interfaces, features and functionality in detail.

2.7 Overall description of proposed system

This section contains the details about all the processes that are performed in the software system and also tells us about the input and output identification i.e. what is the input being given and what is the desired output.

1. Booking

Input: Departure city , Arrival city, Departure Date , class
Process: Operator will enter the above details and check for availability
Output: If operator will found the availability the output is flight Ticket

2. Cancellation

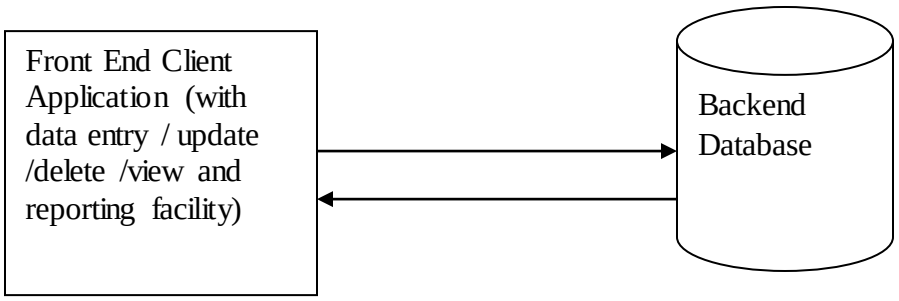
Input: Pnr No
Process: Operator will enter the Pnr no and check in the database
Output: If the process is successful the result is cancelling of ticket

3. Print Ticket

Input: Pnr No
Process: Operator will enter the Pnr no and check in the database
Output: If the process is successful the result is print of ticket it is used in case when Ticket is lost

2.8 Product Perspective

The application will be windows-based, self-contained and independent software product.



2.9 System Interfaces

None

2.10 Interfaces

2.10.1 Hardware Interfaces

- Screen resolution of at least 800*600 pixels- requires for proper and complete viewing of screens.
- Standalone systems or network based – not a concern, as it will be possible to run the application on any of these.
- Minimum processor requirements Pentium 4, 1.2 GHz
- printer

2.10.2Software Interfaces

- Visual Basic 6.0 is used as Frontend
- Oracle 10g is used as Backend.

2.10.3Communication Interfaces

None

2.11 Memory Constraints

At least 64 MB RAM and 2 GB space on hard disk will be required for running the application.

2.12 Operations

This product will not cover any automated housekeeping aspects of database. The DBA at client site will be manually deleting old/ non required data.

Database backup and recovery will also have to be handled by DBA.

2.13 Site Adaptation Requirement

The terminals at client side will have to support the hardware and software interfaces specified.

2.14 Product functions

The system will allow access only to authorized users with specific roles (Administrator, Operator). Depending upon the user's role, he/she will be able to access only specific modules of the system.
A summary of the major functions that the software will perform:
(i) A Login facility for enabling only authorized access to the system.
(ii) Users (with role operator) will add/update/delete the stored information and so on

2.15 User Characteristics

2.15.1 Educational Level: At least graduate and should be comfortable with English Language.
2.15.2 Technical Expertise: Should be a high or middle level employee of the Organization comfortable with using general purpose applications on a computer

2.16 Constraints

GUI is only in English.

2.17 Assumptions and Dependencies

None

2.18 Apportioning Requirement

Not Required

2.19 Specific Requirements

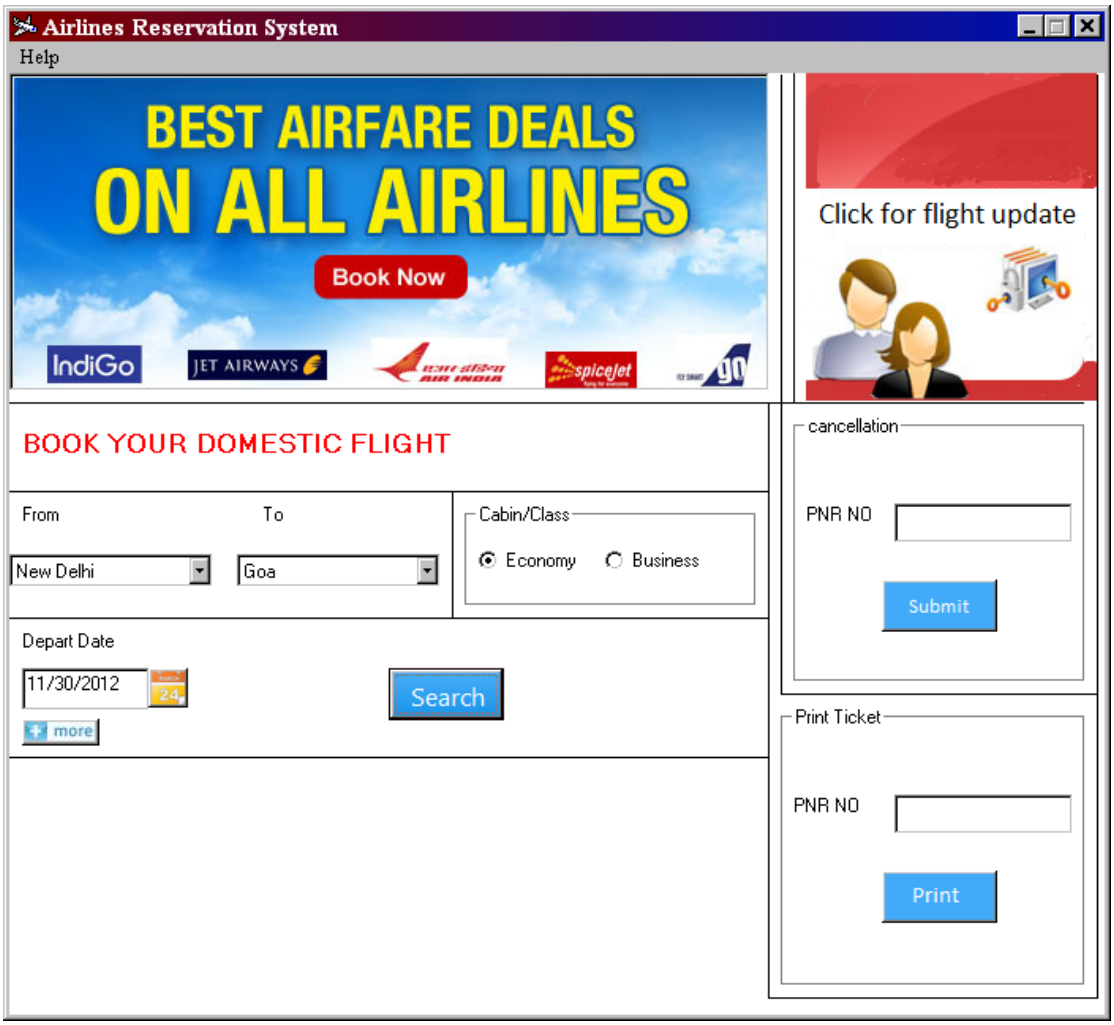
This section contains the software requirements to a level of detail sufficient to enable designers to design the system, and testers to test the system.

2.20 External Interfaces

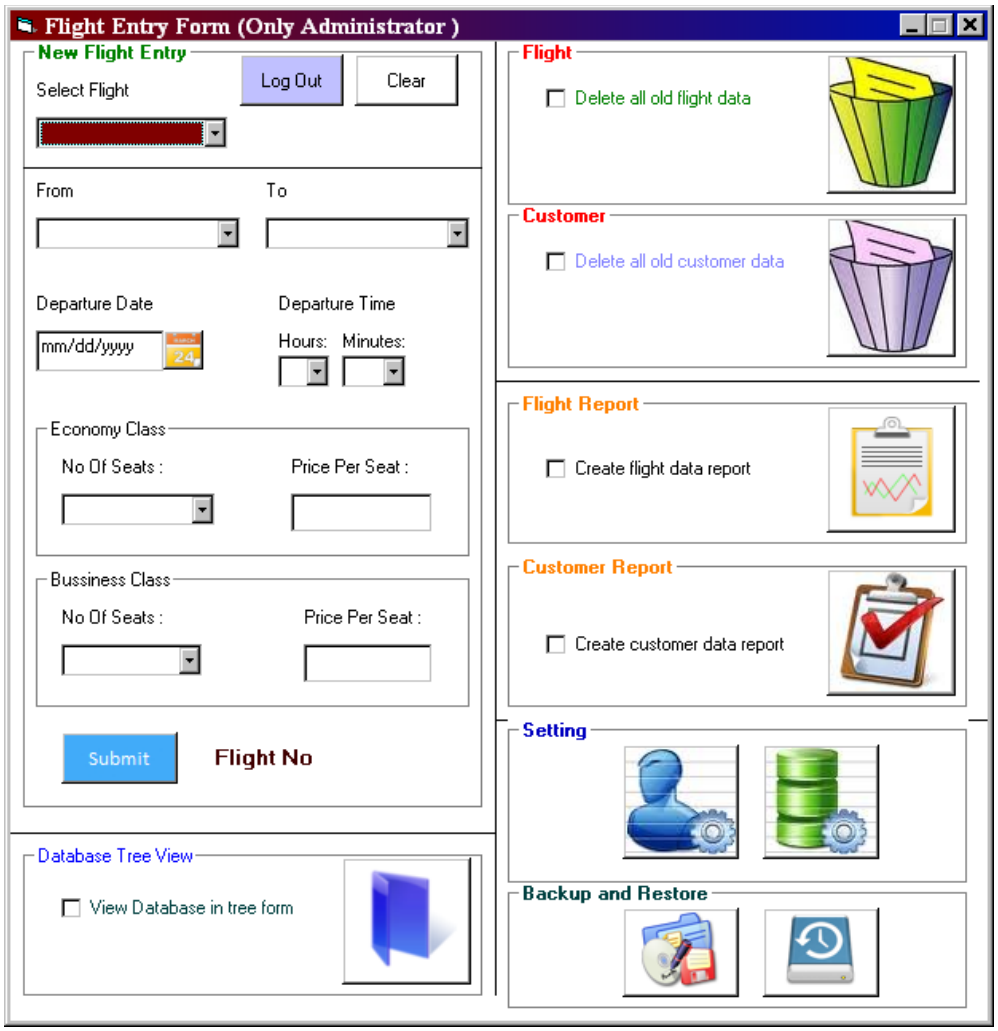
This interface will be the actual interface through which the administrator will communicate with the application and perform the desired tasks. The following screens will be provided:

2.21 User Interfaces

Menu Screen



Administrator Screen



2.22 Hardware Interfaces

Hardware Tools

Hard Disk	1GB
Ram	64MB
Processor	1.8Ghz or above

2.23 Software interfaces

Software Tools

Operating System	Windows XP
Front-End	Visual Basics
Back-End	Oracle10g

2.24 Communication Interfaces

None

2.25 System Features

a. Reservation of Ticket

Description:- In this module the User details will be entered by the data entry operater, the details like:- name of customer, age, sex, no of seats etc will be enter so that the data the next procedure can go on.

Validity checks

- Only data entry operator will be authorized to access this module.
- User name cannot be left blank.
- user's gender cannot be left blank.
- User contact details cannot be left empty.
- Date of departure cannot be left blank.
- Destination to Source station cannot be left blank.

Error Handling / Response to abnormal situations

If any of the above validations/sequencing flow does not hold true, appropriate error messages will be prompted to the user for doing the needful.

b. Printing of Ticket

Description:- The module provides the facility to generate tickets, the inputs to this module being the ticket details which we have mention above and discount (if any) and total amount , the output being the ticket which is formed by the system.

Validity checks

- The total amount can be never negative.
- Fully printed ticket with all the values

Error Handling / Response to abnormal situations

If any of the above validations/sequencing flow does not hold true, appropriate error messages will be prompted to the user for doing the needful.

c. Cancellation of Ticket

Description :- In this module if a passenger want to cancel the ticket as in case he/she don't want to travel, then the Pnr no which the passenger will get on his/her ticket have to enter that & make the cancellation**.**

Validity checks

- Pnr_no cannot be left blank.

Error Handling / Response to abnormal situations

If any of the above validations/sequencing flow does not hold true, appropriate error messages will be prompted to the user for doing the needful.

2.26 Performance Requirements

None

2.27 Logical Database Requirements

The proposed information system contains the following data tables in its Database collection.

2.27.1 Flight Database Table

FLIGHT DATABASE			
Field Name	**Field Type**	**Field Size(No of Character)**	**Description**
Flight_Company	Varchar	15	The name of the flight company it belongs to.
Depart_City	Varchar	10	The city from where the flight will take off
Arrival_City	Varchar	10	The city where the flight will land
Depart_Date	Varchar	10	Date of flight take off
Depart_Hour	Number	3	Time in hour of flight's take off time
Depart_Minute	Number	3	Time in minutes of flight's take off time
Eco_No_of_Seat	Number	4	Total economy seats in flight
Eco_Price	Varchar	10	Price of one economy seat
Bust_No_of_Seat	Number	4	Total business seats in flight
Bust_Price	Varchar	10	Price of one business seat
Flight_No	Varchar	15	Unique flight no system genrated

2.27.2 Customer & Reservation Database Table

CUSTOMER & RESERVATION DATABASE			
Field Name	**Field Type**	**Field Size(No of Character)**	**Description**
Title	Varchar	4	Title of customer
First_Name	Varchar	20	First Name of the Ticket for whom its booked
Last_Name	Varchar	20	Last Name of the Ticket for whom its booked
Date_of_Birth_Day	Number	2	Age of passenger according to day
Date_of_Birth_Month	Char	4	Age of passenger according to month
Date_of_Birth_Year	Number	4	Age of passenger according to year
Mobile_No	Char	15	Contact number/ mobile number of passenger
Address_Line_One	Varchar	50	Address of passenger
Address_Line_Two	Varchar	50	Sub address of passenger
City	Varchar	10	City where passenger living
State	Varchar	15	State where passenger living
Country	Varchar	15	Country where passenger living
Pincode	Char	8	Area code of the passengers residential
Pnr_No	Varchar	20	Pnr no of the ticket registered/ reserved (system generated)
Flight_No	Varchar	15	Unique flight no of plane
Cabin	Varchar	10	Section in which the seat reserved

2.27.3 Login Database Table

Login Database Table			
Field Name	**Field Type**	**Field Size(No of Character)**	**Description**
USERNAME	VARCHAR	10	Take the Username
PASSWORD	VARCHAR	20	For password

2.28 Design Constraints

None

2.28 Software System Attributes

- Reliability
  This application is a reliable product that produces fast and Verified output of all its processes.

- Availability
  This application will be available to use for our end users and help them to carry out their operations conveniently.

- Security
  The application will be password protected. User will have to enter correct username, password and role in order to access the application.

- Maintainability
  The application will be designed in a maintainable manner. It will be easy to incorporate new requirements in the individual modules.

- Portability
  The application will be easily portable on any windows-based system that has oracle installed.

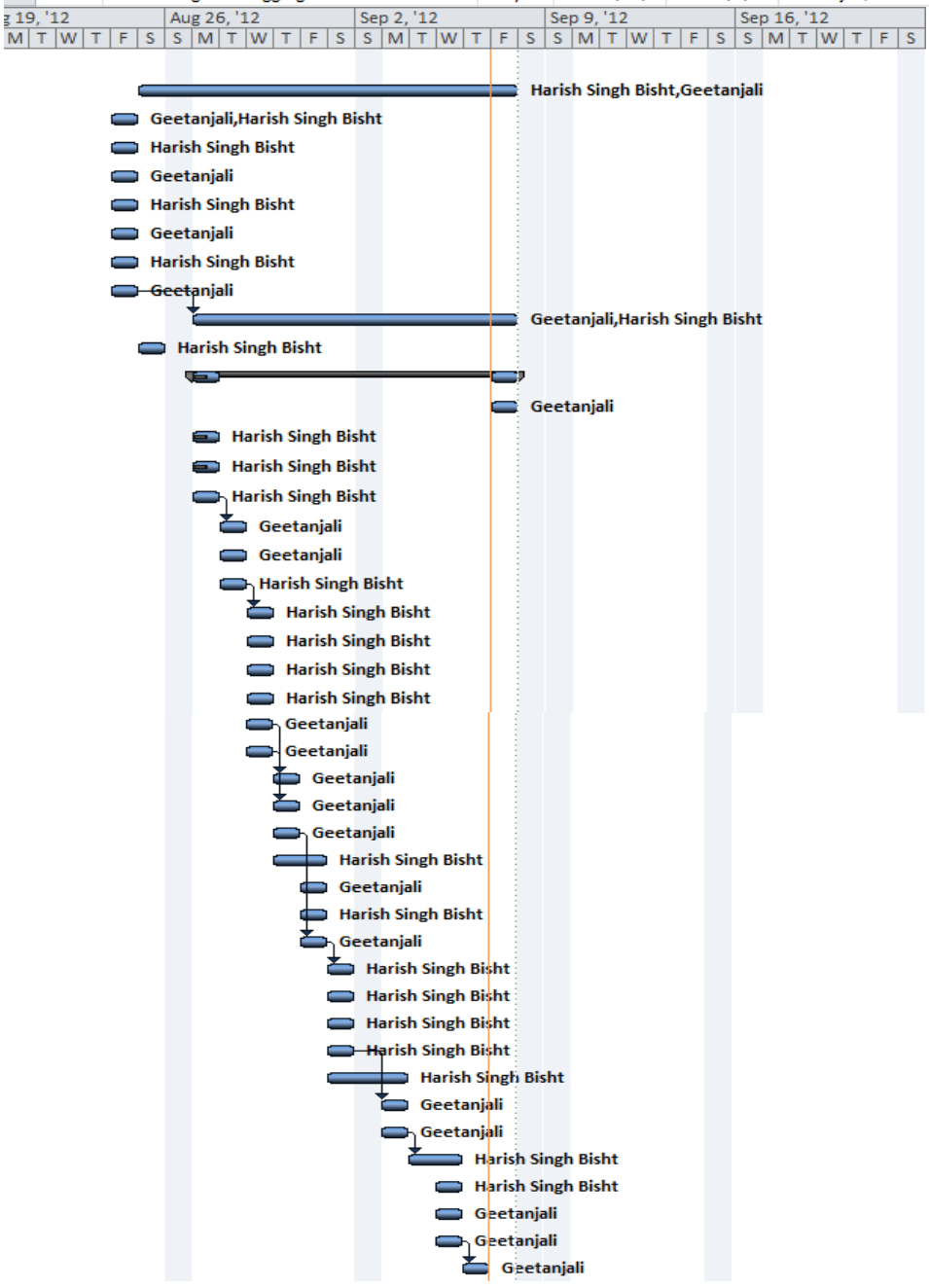
2.30 Other Requirements

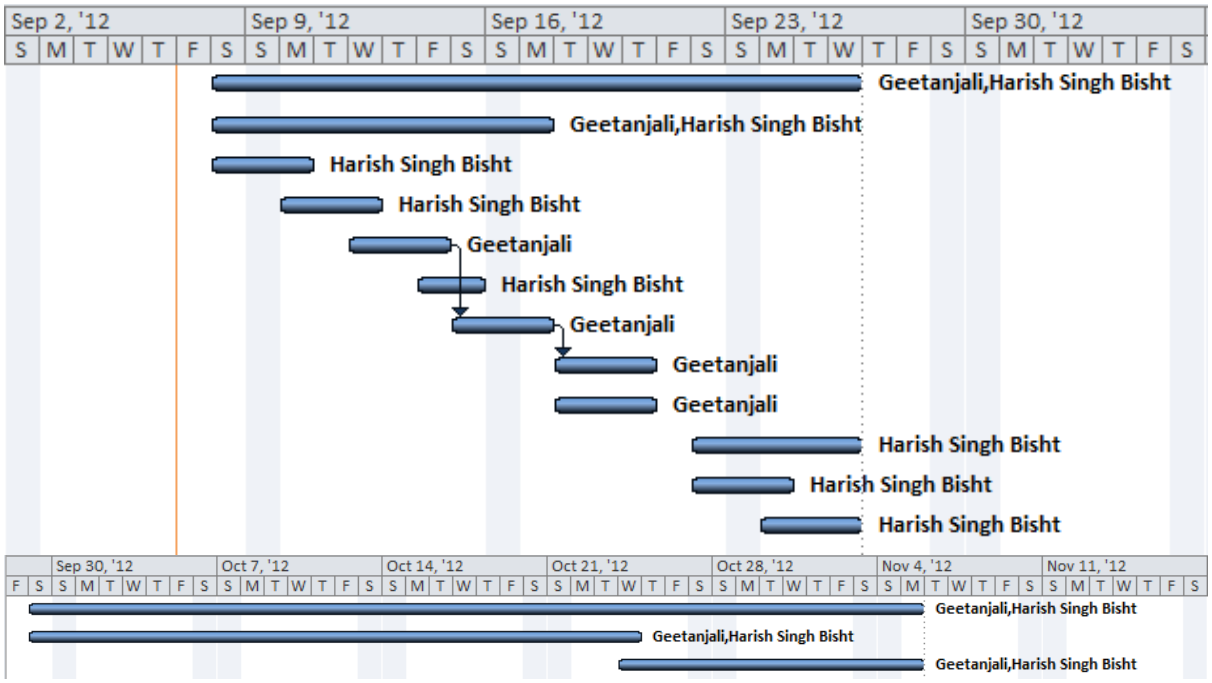
None

3 Gantt chart

	Task Mode	Task Name	Duratic	Start	Finish	Resource Names
1		Airlines Reservation System (Minor Projec				
2		Chapter-1 Introduction	12 days	Sat 8/25/12	Fri 9/7/12	Harish Singh Bisht,Geetanjali
3		1. Description Of Organization	1 day	Fri 8/24/12	Fri 8/24/12	Geetanjali,Harish Singh Bisht
4		1.1 Introduction	1 day	Fri 8/24/12	Fri 8/24/12	Harish Singh Bisht
5		1.2 History of Organization	1 day	Fri 8/24/12	Fri 8/24/12	Geetanjali
6		1.3 Objective of Organization	1 day	Fri 8/24/12	Fri 8/24/12	Harish Singh Bisht
7		1.4 Organizational Structure	1 day	Fri 8/24/12	Fri 8/24/12	Geetanjali
8		1.5 Key Result Areas	1 day	Fri 8/24/12	Fri 8/24/12	Harish Singh Bisht
9		1.6 Function	1 day	Fri 8/24/12	Fri 8/24/12	Geetanjali
10		2. System Requirement Specifications	11 days	Mon 8/27/1	Fri 9/7/12	Geetanjali,Harish Singh Bisht
11		2.1 Introduction	1 day	Sat 8/25/12	Sat 8/25/12	Harish Singh Bisht
12		2.2 Purpose	**11 days**	Mon 8/27/1	Fri 9/7/12	**Geetanjali**
13		2.3 Scope	1 day	Fri 9/7/12	Fri 9/7/12	Geetanjali
14		2.4 Definition, acronyms, abbreviations	1 day	Mon 8/27/1	Mon 8/27/12	Harish Singh Bisht
15		2.5 References	1 day	Mon 8/27/1	Mon 8/27/12	Harish Singh Bisht
16		2.6 Overview	1 day	Mon 8/27/1	Mon 8/27/12	Harish Singh Bisht
17		2.7 Overall description of proposed syste	1 day	Tue 8/28/1	Tue 8/28/12	Geetanjali
18		2.8 Product Perspective	1 day	Tue 8/28/1	Tue 8/28/12	Geetanjali
19		2.9 System Interfaces	1 day	Tue 8/28/1	Tue 8/28/12	Harish Singh Bisht
20		2.10 Interfaces	1 day	Wed 8/29/	Wed 8/29/1	Harish Singh Bisht
21		2.10.1 Hardware Interfaces	1 day	Wed 8/29/	Wed 8/29/1	Harish Singh Bisht
22		2.10.2 Software Interfaces	1 day	Wed 8/29/	Wed 8/29/1	Harish Singh Bisht
23		2.10.3 Communication Interface	1 day	Wed 8/29/	Wed 8/29/1	Harish Singh Bisht
24		2.11 Memory Constraints	1 day	Wed 8/29/	Wed 8/29/1	Geetanjali
25		2.12 Operations	1 day	Wed 8/29/	Wed 8/29/1	Geetanjali
26		2.13 Site Adaptation Requirement	1 day	Thu 8/30/1	Thu 8/30/12	Geetanjali
27		2.14 Product functions	1 day	Thu 8/30/1	Thu 8/30/12	Geetanjali
28		2.15 User Characteristics	1 day	Thu 8/30/1	Thu 8/30/12	Geetanjali
29		2.16 Constraints	2 days	Thu 8/30/1	Fri 8/31/12	Harish Singh Bisht
30		2.17 Assumptions and Dependencies	1 day	Fri 8/31/12	Fri 8/31/12	Geetanjali
31		2.18 Apportioning Requirement	1 day	Fri 8/31/12	Fri 8/31/12	Harish Singh Bisht
32		2.19 Specific Requirements	1 day	Fri 8/31/12	Fri 8/31/12	Geetanjali
33		2.20 External Interfaces	1 day	Sat 9/1/12	Sat 9/1/12	Harish Singh Bisht
34		2.21 User Interfaces	1 day	Sat 9/1/12	Sat 9/1/12	Harish Singh Bisht
35		2.22 Hardware Interfaces	1 day	Sat 9/1/12	Sat 9/1/12	Harish Singh Bisht
36		2.23 Software interfaces	1 day	Sat 9/1/12	Sat 9/1/12	Harish Singh Bisht
37		2.24 Communication Interfaces	2 days	Sat 9/1/12	Mon 9/3/12	Harish Singh Bisht
38		2.25 System Features	1 day	Mon 9/3/12	Mon 9/3/12	Geetanjali
39		2.26 Performance Requirements	1 day	Mon 9/3/12	Mon 9/3/12	Geetanjali
40		2.27 Logical Database Requirements	2 days	Tue 9/4/12	Wed 9/5/12	Harish Singh Bisht
41		2.28 Design Constraints	1 day	Wed 9/5/12	Wed 9/5/12	Harish Singh Bisht
42		2.29 Software System Attributes	1 day	Wed 9/5/12	Wed 9/5/12	Geetanjali
43		2.30 Other Requirements	1 day	Wed 9/5/12	Wed 9/5/12	Geetanjali
44		3. Gantt Chart	1 day	Thu 9/6/12	Thu 9/6/12	Geetanjali
45		Chapter-2 System Design	16 days	Sat 9/8/12	Wed 9/26/12	Geetanjali,Harish Singh Bisht
46		1. Physical Design	8 days	Sat 9/8/12	Mon 9/17/12	Geetanjali,Harish Singh Bisht
47		1.1 Block Diagram	2 days	Sat 9/8/12	Mon 9/10/12	Harish Singh Bisht
48		1.2 Processes	3 days	Mon 9/10/1	Wed 9/12/12	Harish Singh Bisht
49		1.3 Use Case Diagram	3 days	Wed 9/12/	Fri 9/14/12	Geetanjali
50		1.4 Data Flow Diagram	2 days	Fri 9/14/12	Sat 9/15/12	Harish Singh Bisht
51		1.5 Entity Relationship Diagram	2 days	Sat 9/15/12	Mon 9/17/12	Geetanjali
52		2. Database Design	3 days	Tue 9/18/1	Thu 9/20/12	Geetanjali
53		2.1 Data Dictionary	3 days	Tue 9/18/1	Thu 9/20/12	Geetanjali
54		3. Interface Design	4 days	Sat 9/22/12	Wed 9/26/12	Harish Singh Bisht
55		3.1 Input Design	2 days	Sat 9/22/12	Mon 9/24/12	Harish Singh Bisht
56		3.2 Output Design	3 days	Mon 9/24/1	Wed 9/26/12	Harish Singh Bisht

57	Chapater- 3 System Developmet & Impleme	27 days	Sat 9/29/12	Mon 11/5/12	Geetanjali,Harish Singh Bisht
58	1. Program Develppoment	19 days	Sat 9/29/12	Wed 10/24/1	Geetanjali,Harish Singh Bisht
59	2. Testing & debugging	9 days	Wed 10/24/1	Mon 11/5/12	Geetanjali,Harish Singh Bisht




Sep 2, '12
Sep 9, '12
Sep 16, '12
Sep 23, '12
Sep 30, '12
S M T W T F S S M T W T F S S M T W T F S S M T W T F S S M T W T F S
Geetanjali,Harish Singh Bisht
Geetanjali,Harish Singh Bisht
Harish Singh Bisht
Harish Singh Bisht
Geetanjali
Harish Singh Bisht
Geetanjali
Geetanjali
Geetanjali
Harish Singh Bisht
Harish Singh Bisht
Harish Singh Bisht
Sep 30, '12
Oct 7, '12
Oct 14, '12
Oct 21, '12
Oct 28, '12
Nov 4, '12
Nov 11, '12
F S S M T W T F S S M T W T F S S M T W T F S S M T W T F S S M T W T F S S M T W T F S S M T W T F S
Geetanjali,Harish Singh Bisht
Geetanjali,Harish Singh Bisht
Geetanjali,Harish Singh Bisht

Chapter 2: System Design

2.1 Physical Design

2.1.1 Block Diagram

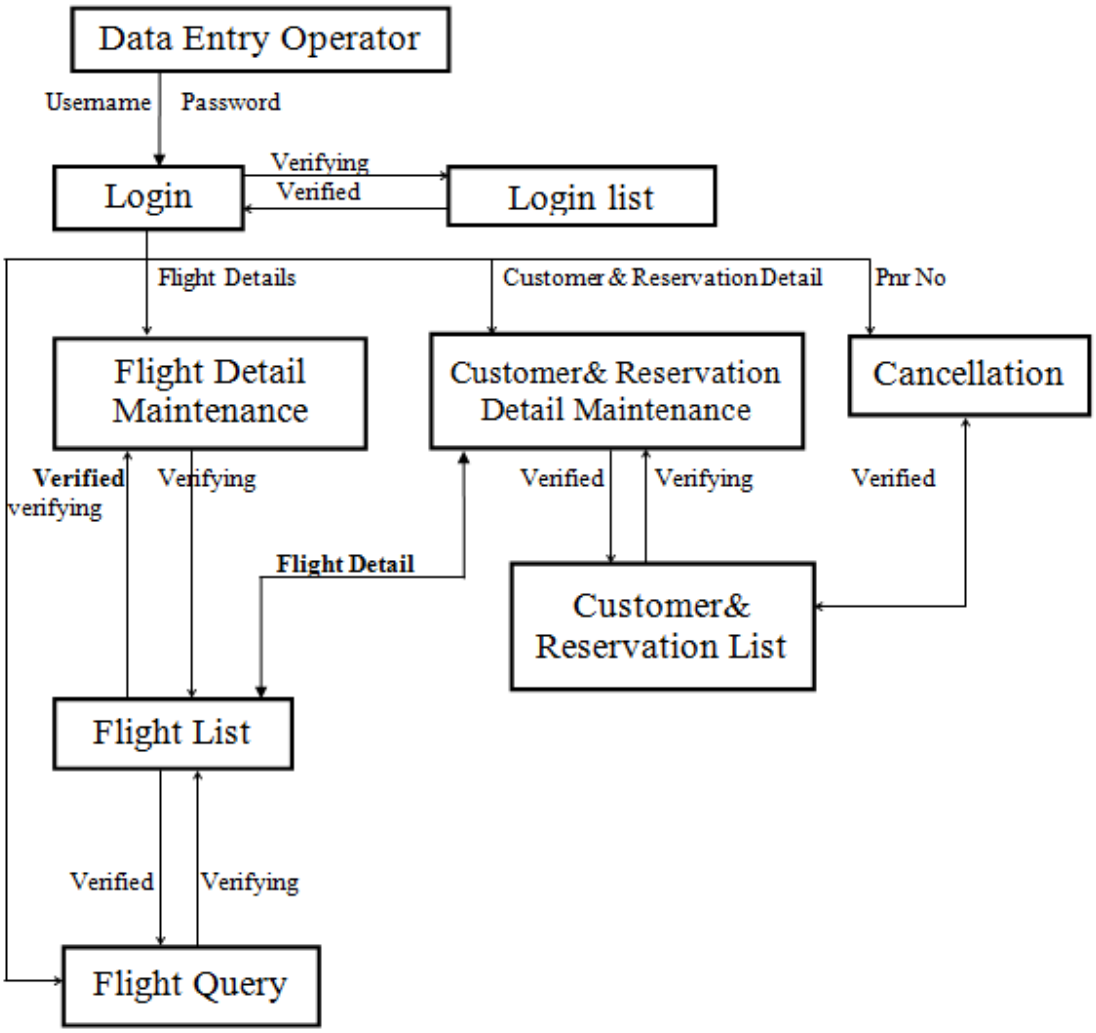


Figure No-1 Block Diagram

2.1.2 Use case diagram

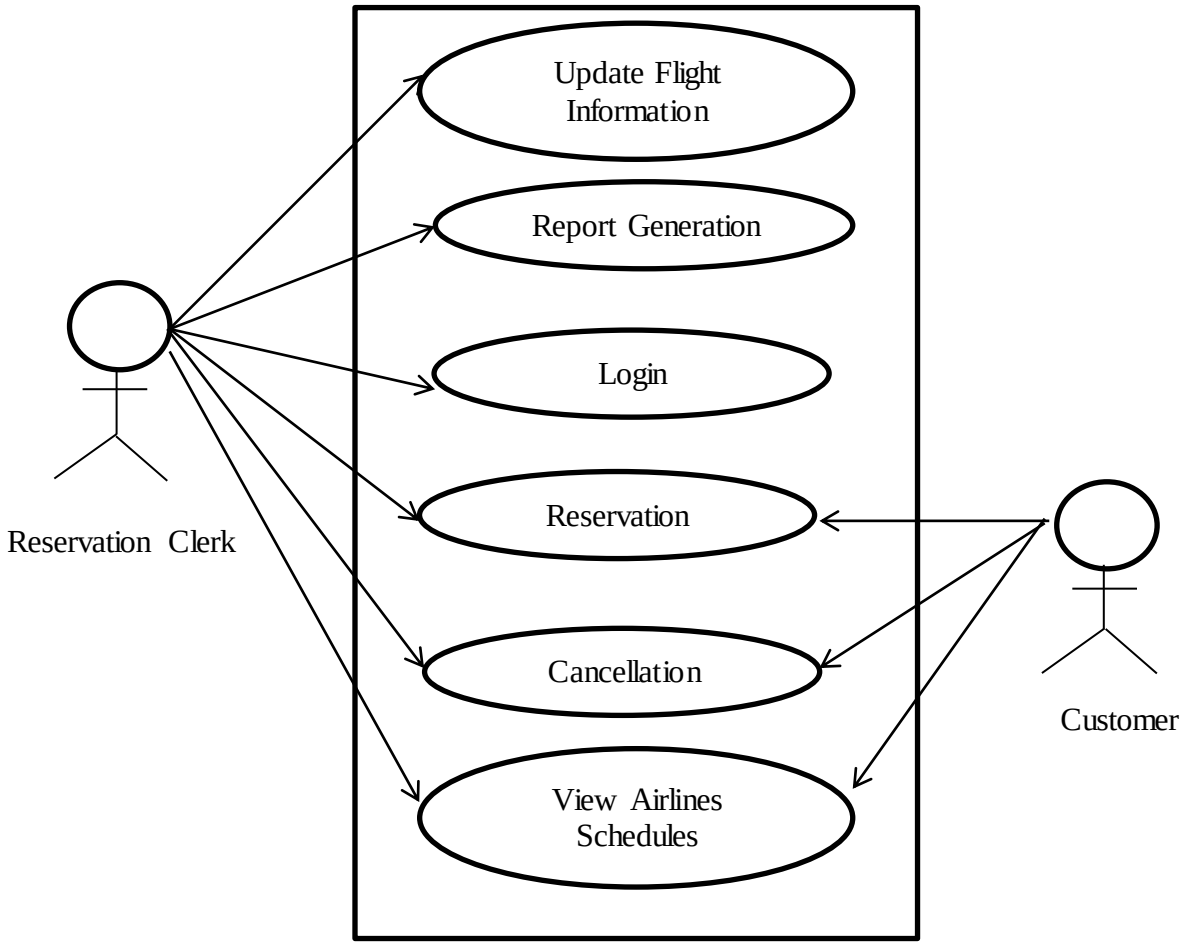


Figure No-2 Use case diagram for Airlines Reservation System

2.1.3 Data Flow Diagram

2.1.3.1 DFD 0 Level

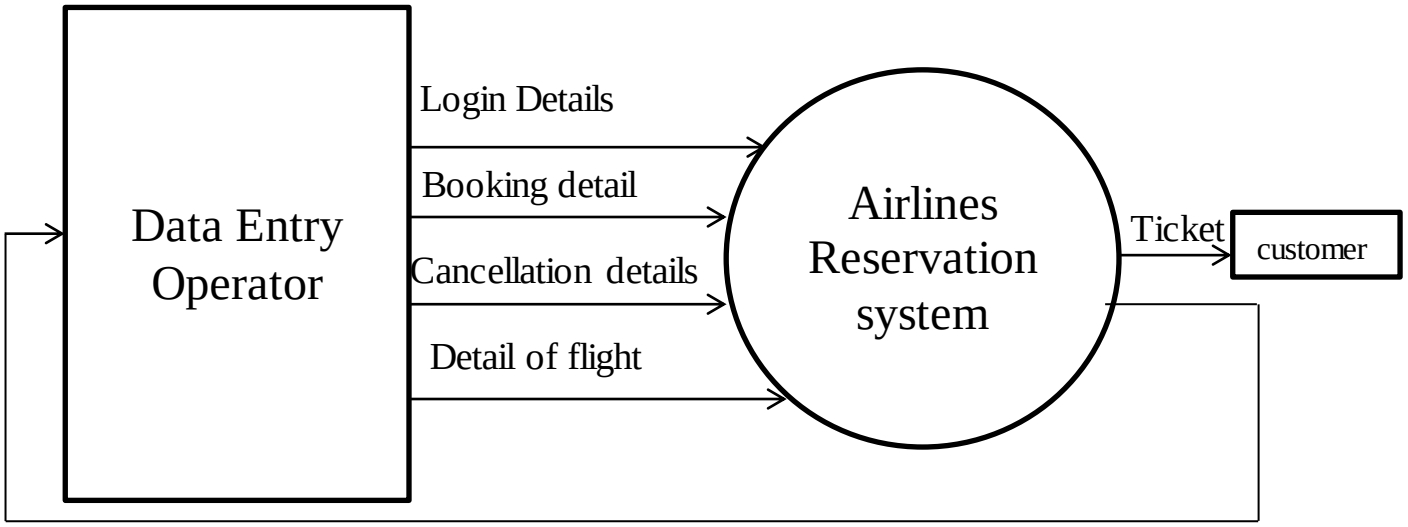


Figure No-3 Data Flow Diagram of 0 level

2.1.3.1 DFD 1 Level

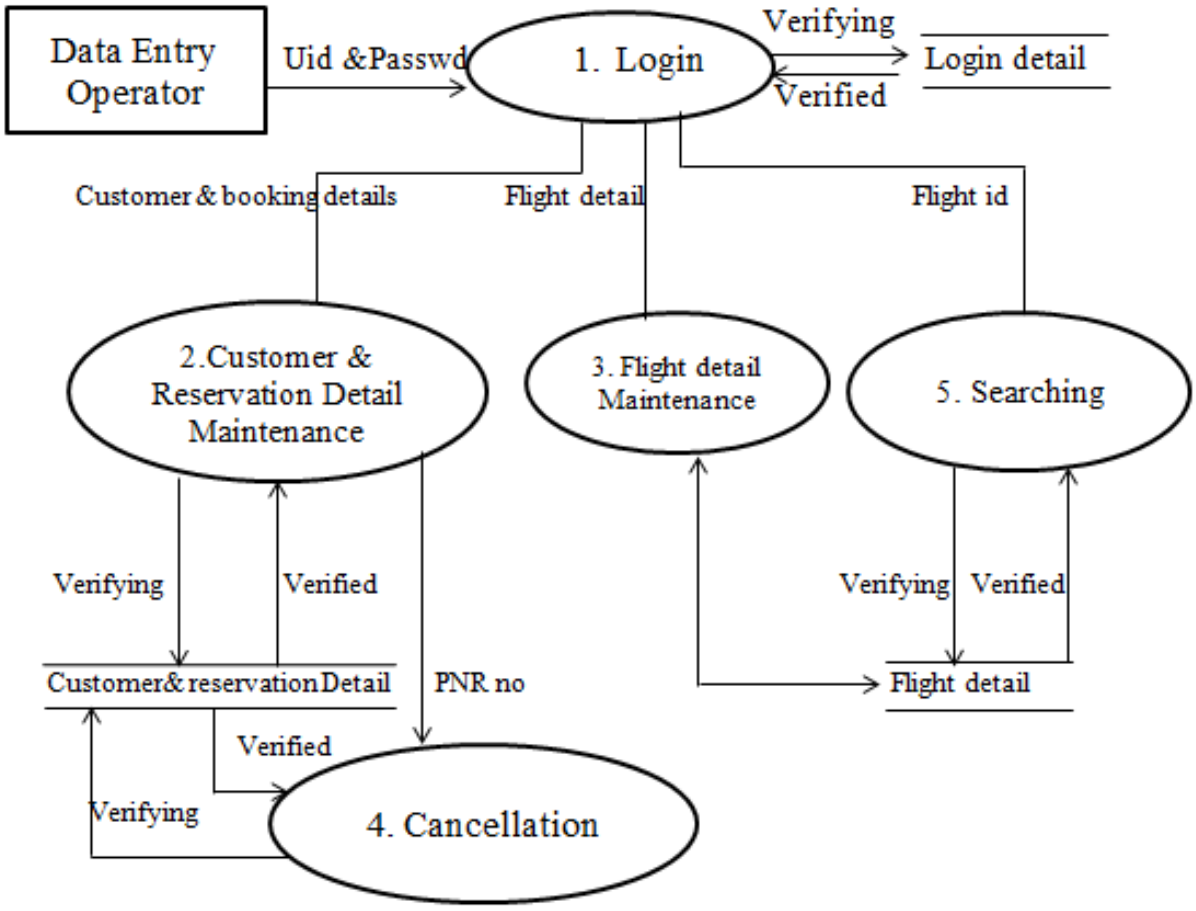


Figure No-4 Data Flow Diagram of 1 level

2.1.4 Entity Relationship Diagram

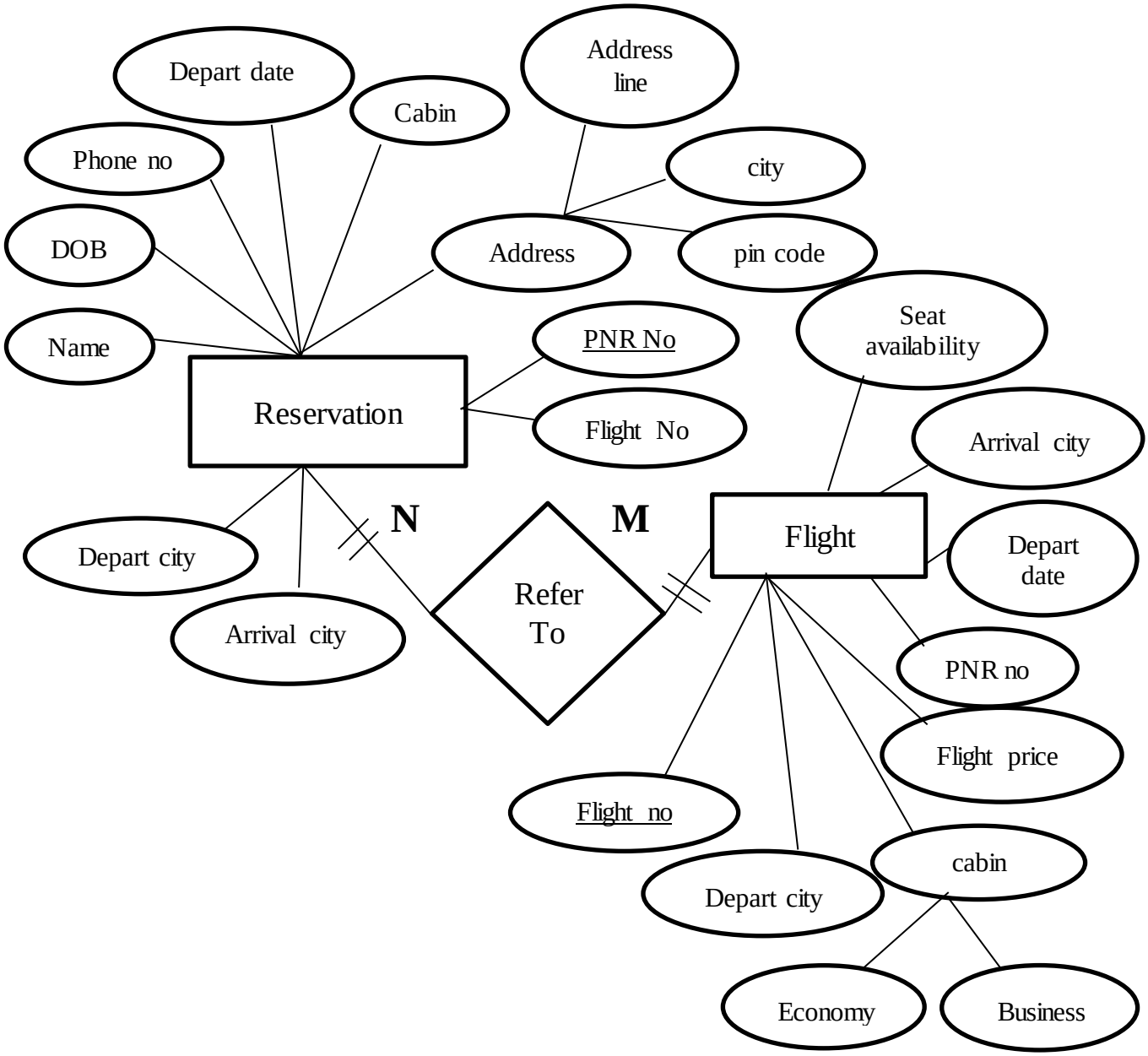
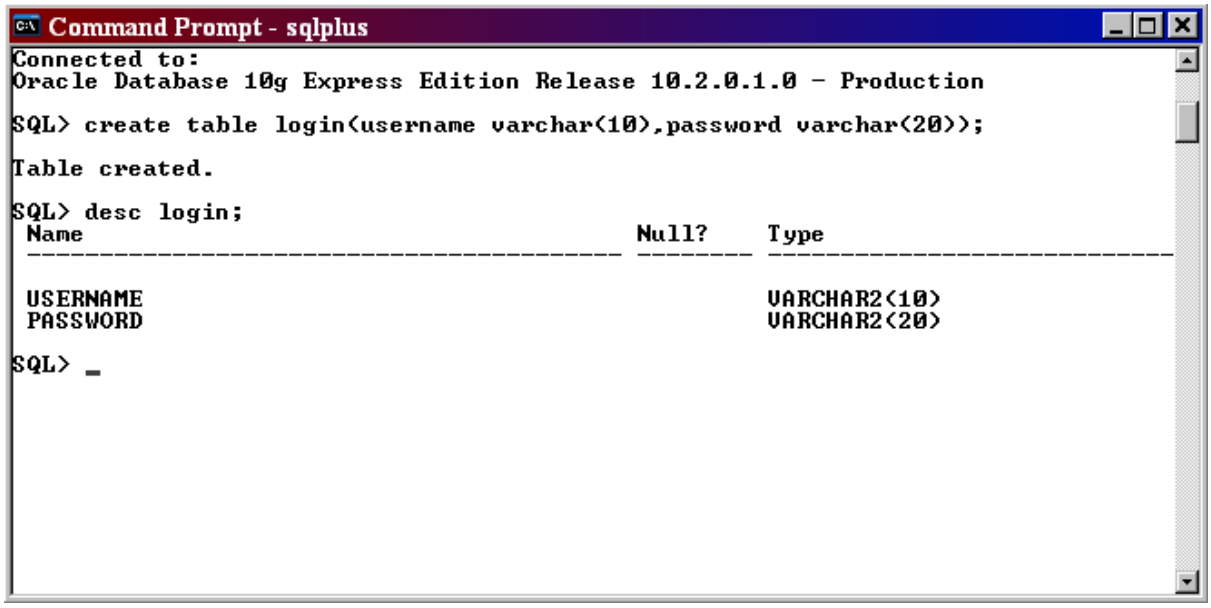


Figure No-5 ER Diagram of Airlines Reservation System

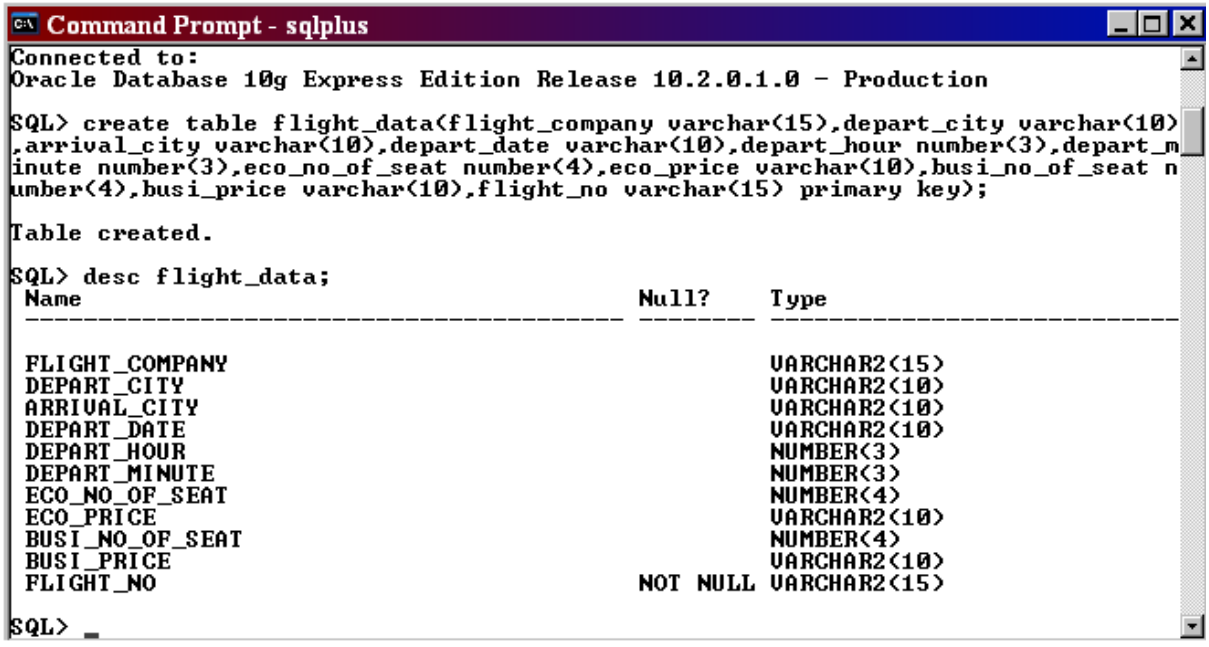
2.2 Database Design

The information system of "Airlines Reservation System" performs its function with the help of the data store in certain repositories called Databases of the system. Detailed descriptions of the various databases included in the information systems are tabulated as follows:

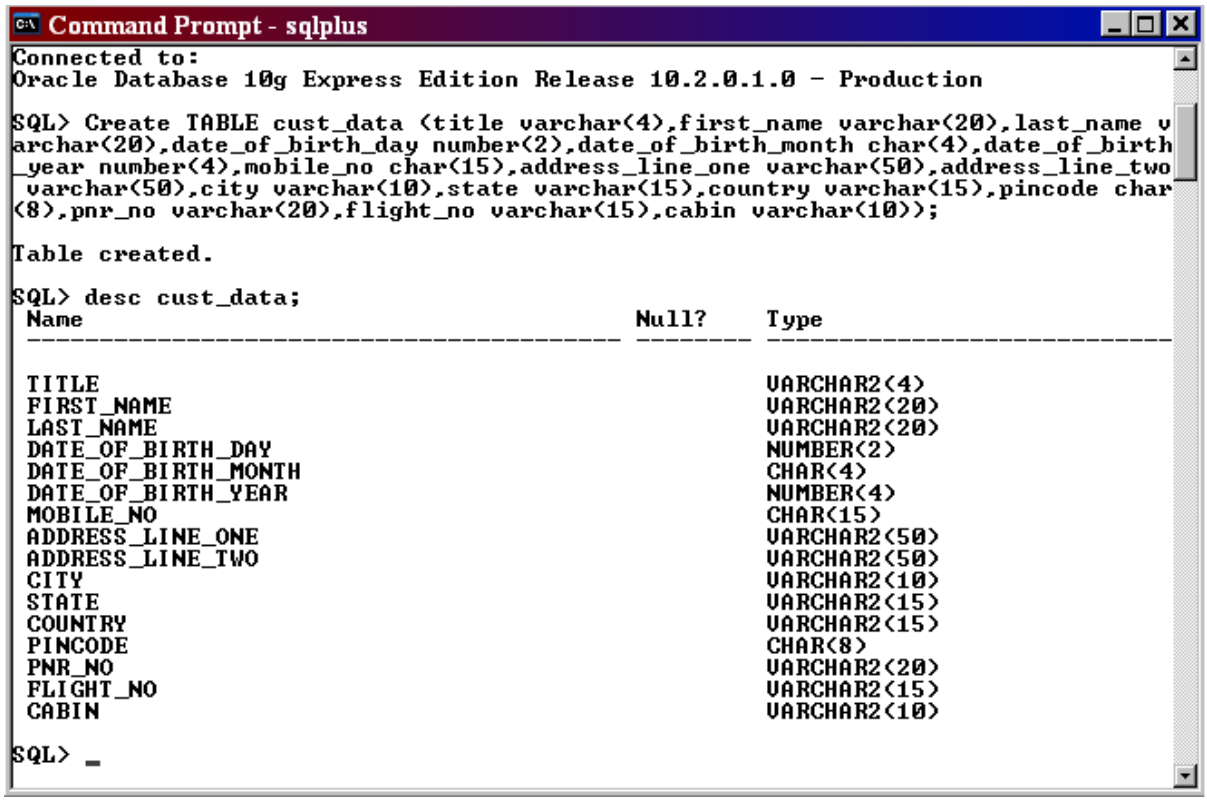
2.2.1 Login Database



2.2.2 Flight Database



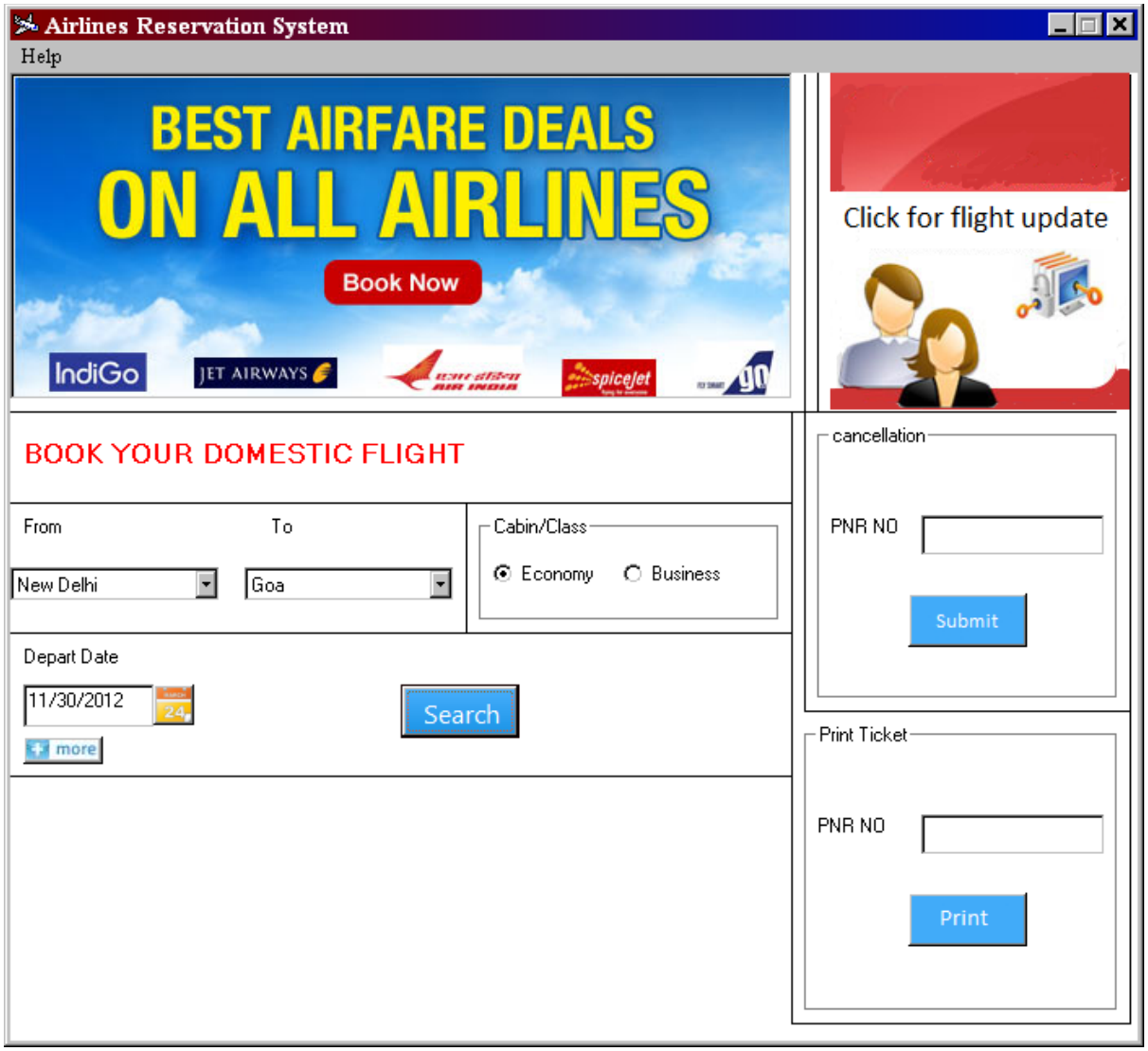
2.2.3 Customer & Reservation Database

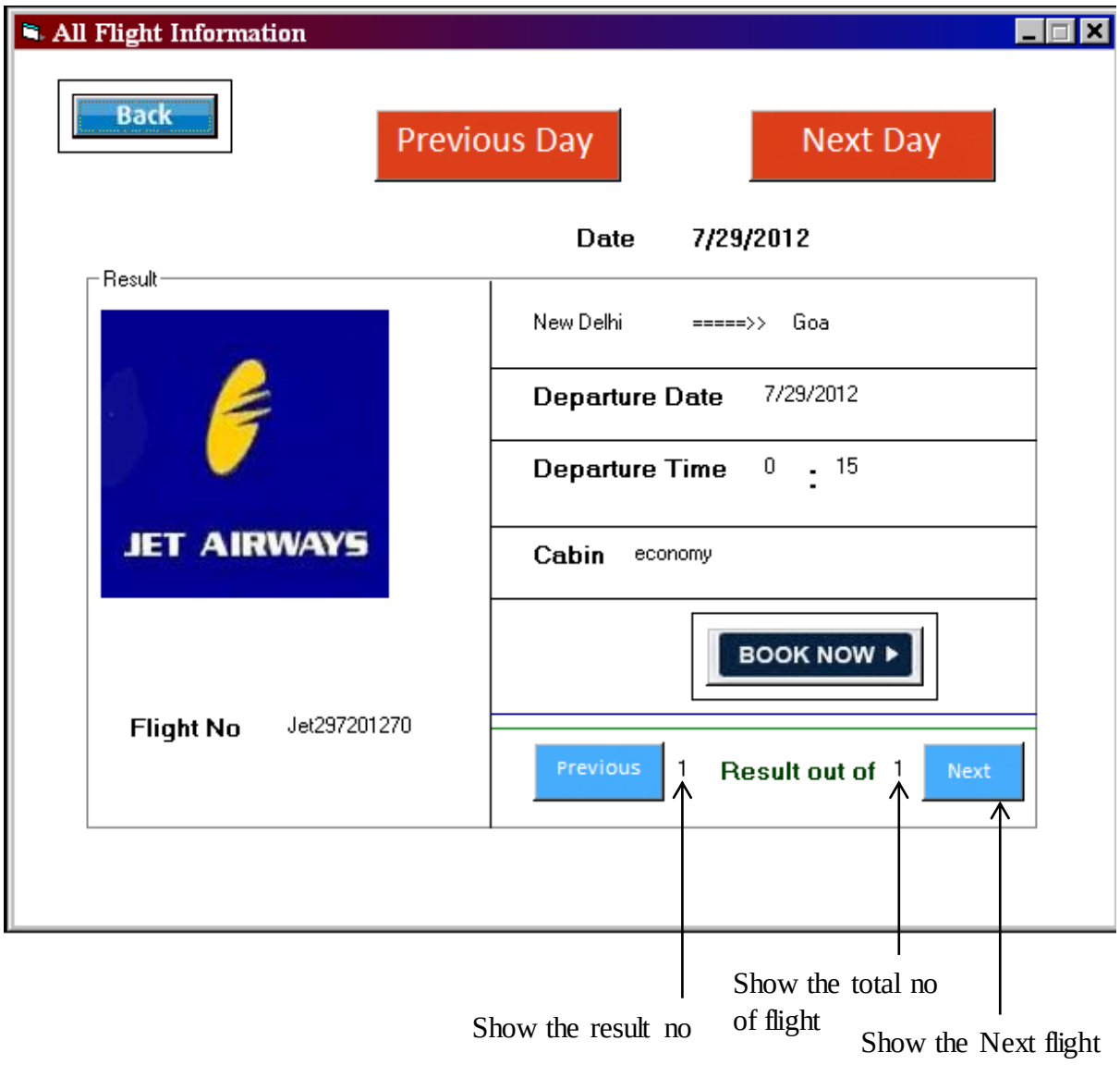


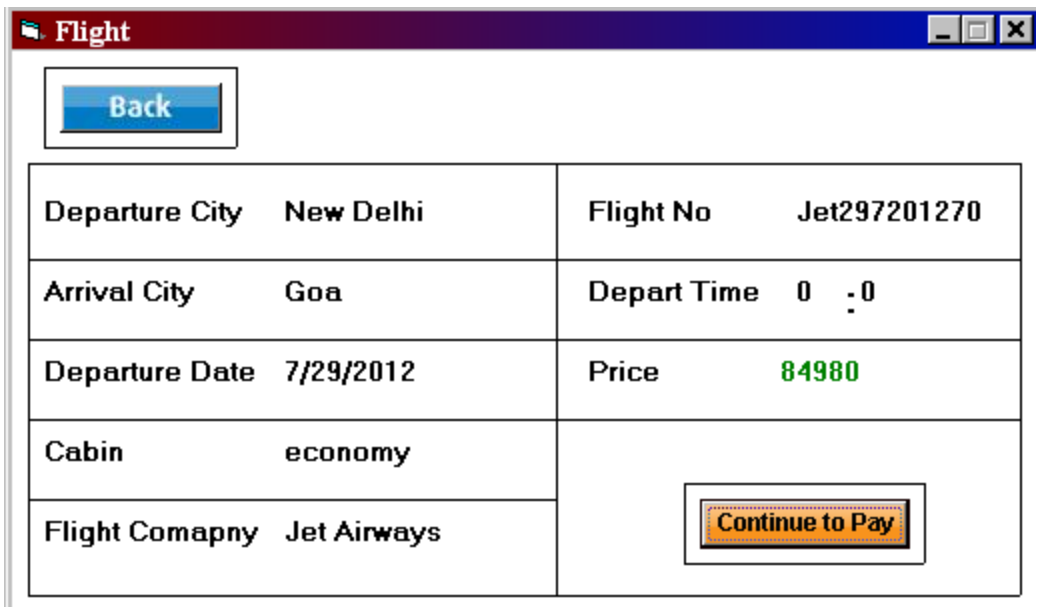
2.3 Interface Design

The interface design consists of the input and output source layouts. i.e. the input forms and screens and the report layouts that form as a source of outcome and income in the design and implementation of the information system under study

2.3.1 Main Form

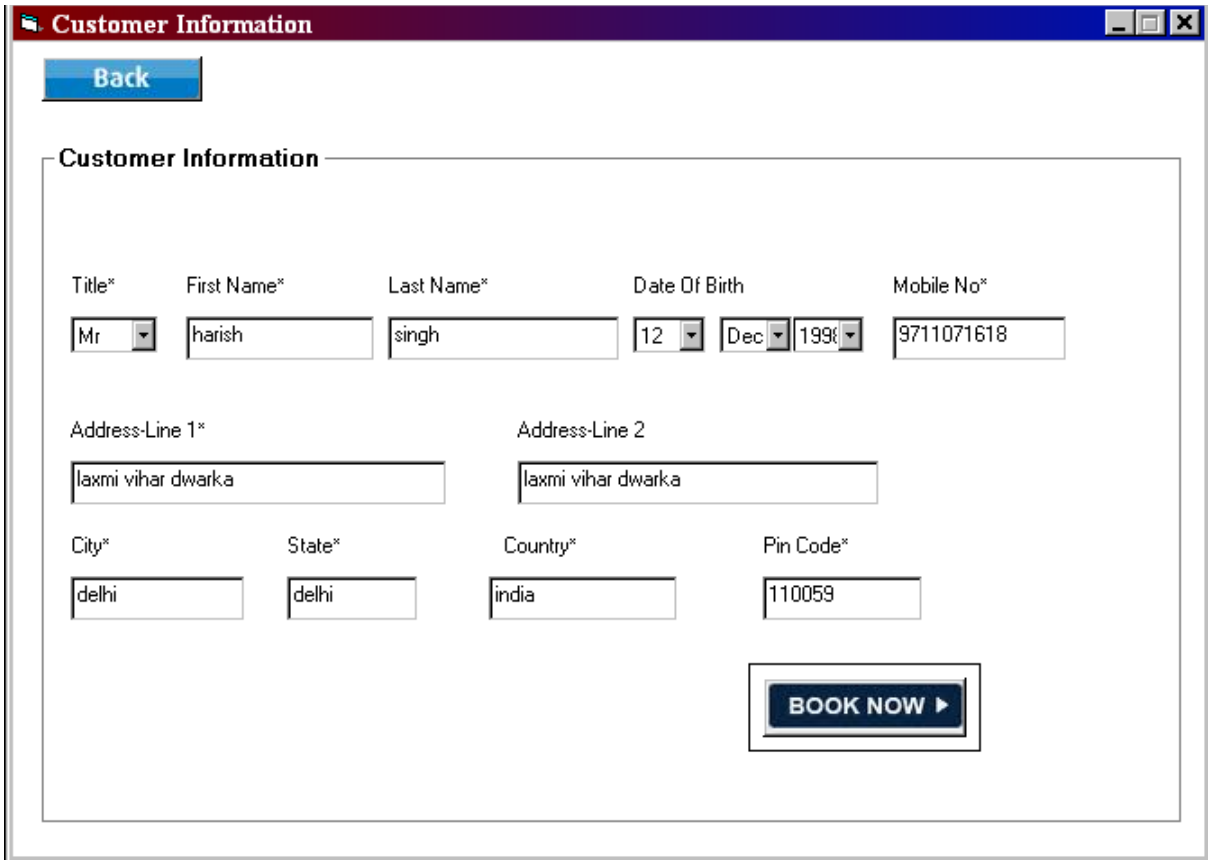



All Flight Information
Back
Previous Day
Next Day
Date 7/29/2012
Result
JET AIRWAYS
New Delhi =====>> Goa
Departure Date 7/29/2012
Departure Time 0 : 15
Cabin economy
BOOK NOW
Flight No Jet297201270
Previous 1 Result out of 1 Next
Show the result no
Show the total no of flight
Show the Next flight

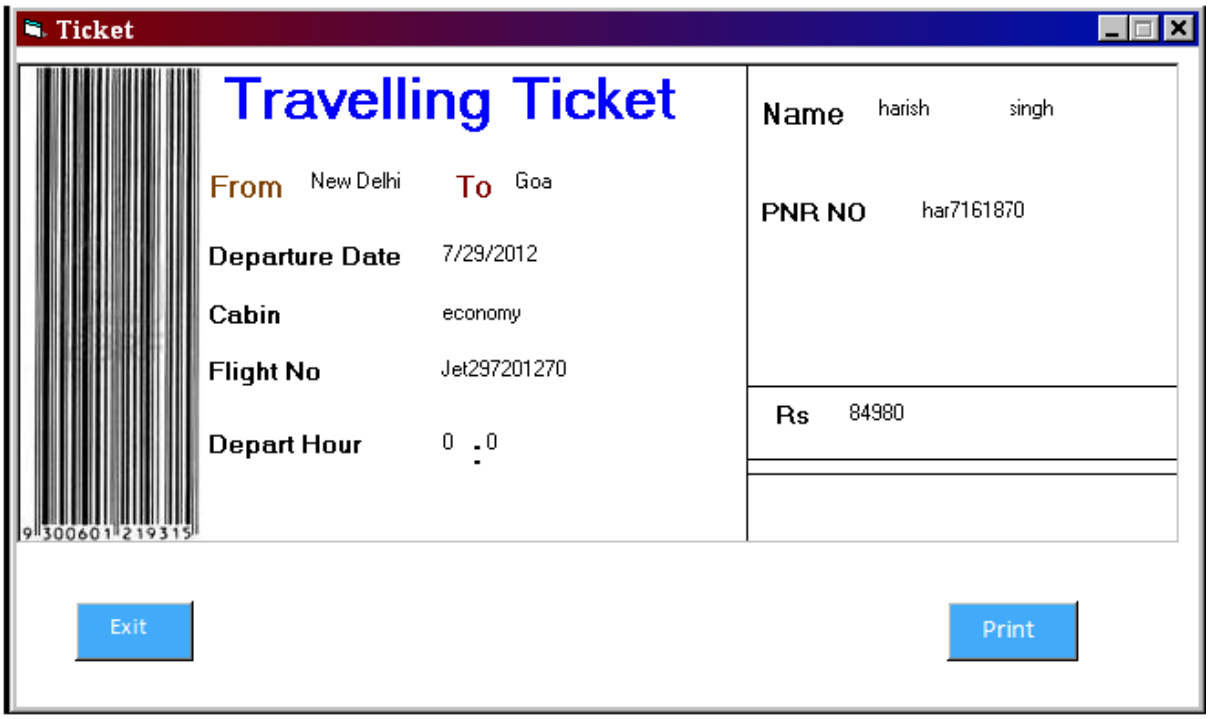


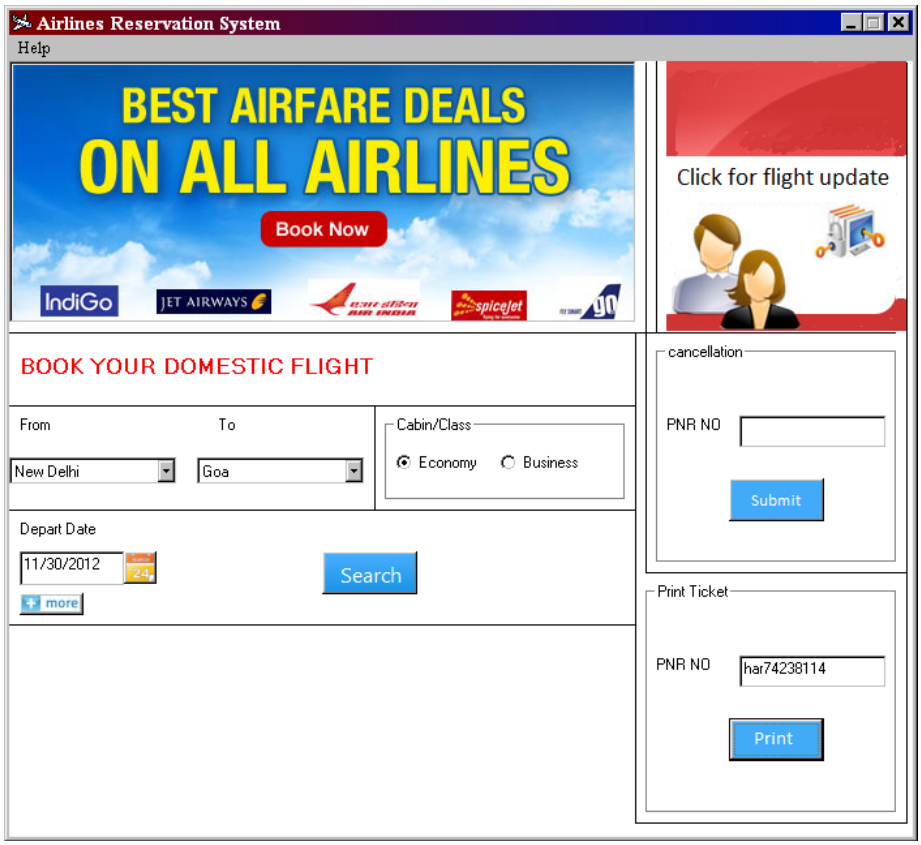
Show the complete information of selected flight

2.3.2 Customer information form

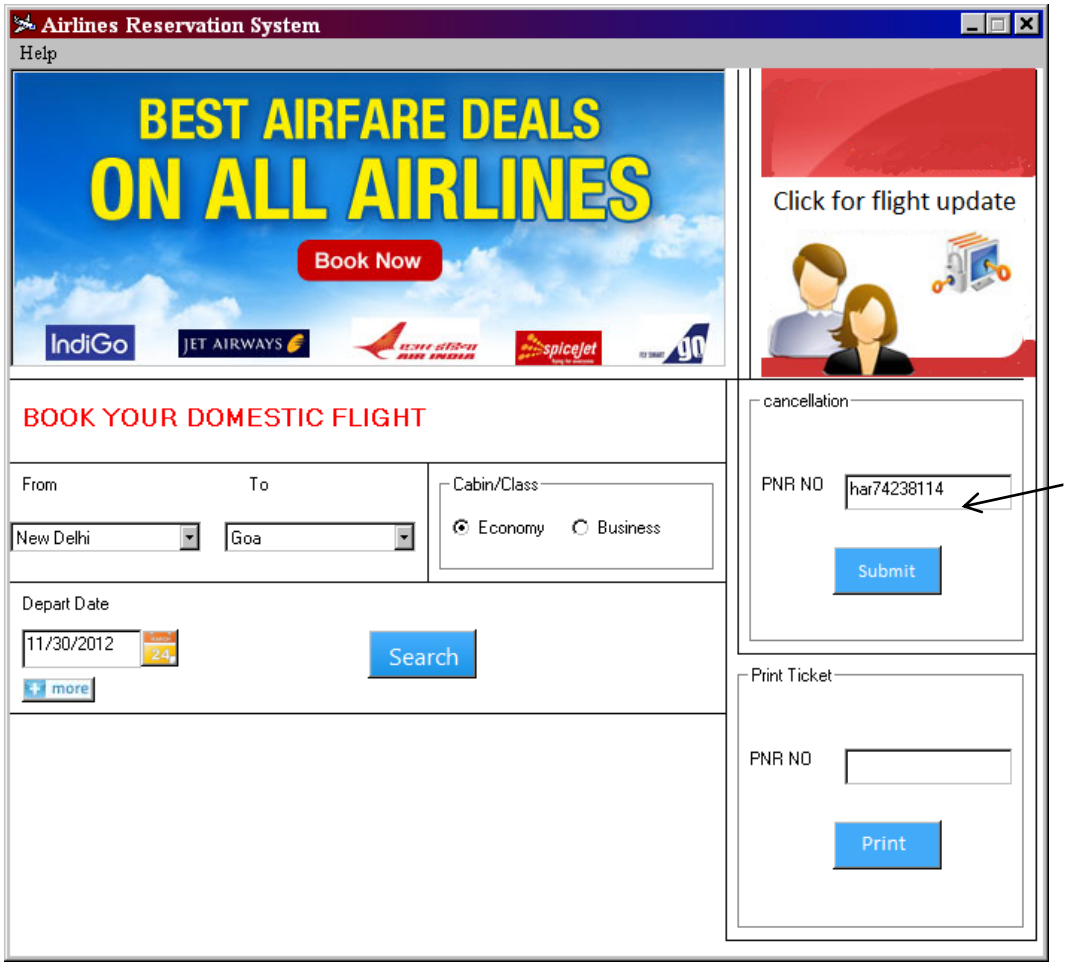


2.3.3 Ticket Form



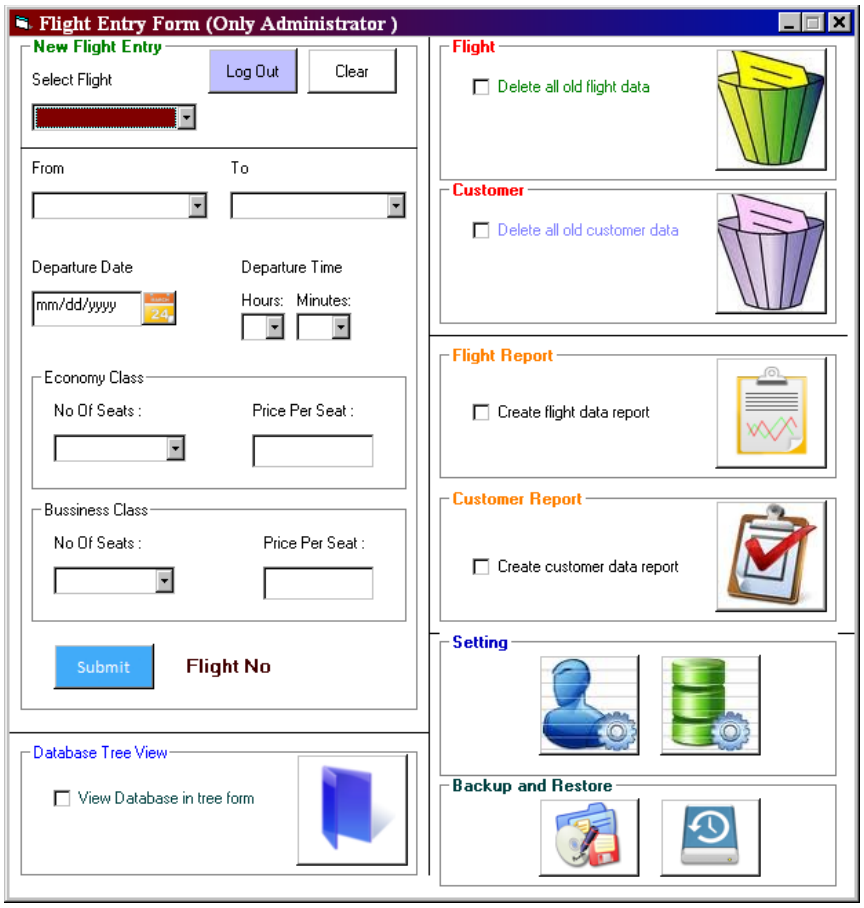


In case lose of ticket enter the PNR No and click on print



In case of ticket cancellation. enter the PNR No and click on submit

2.3.4 Flight Entry Form



Chapter-3: Systems Development & Implementation

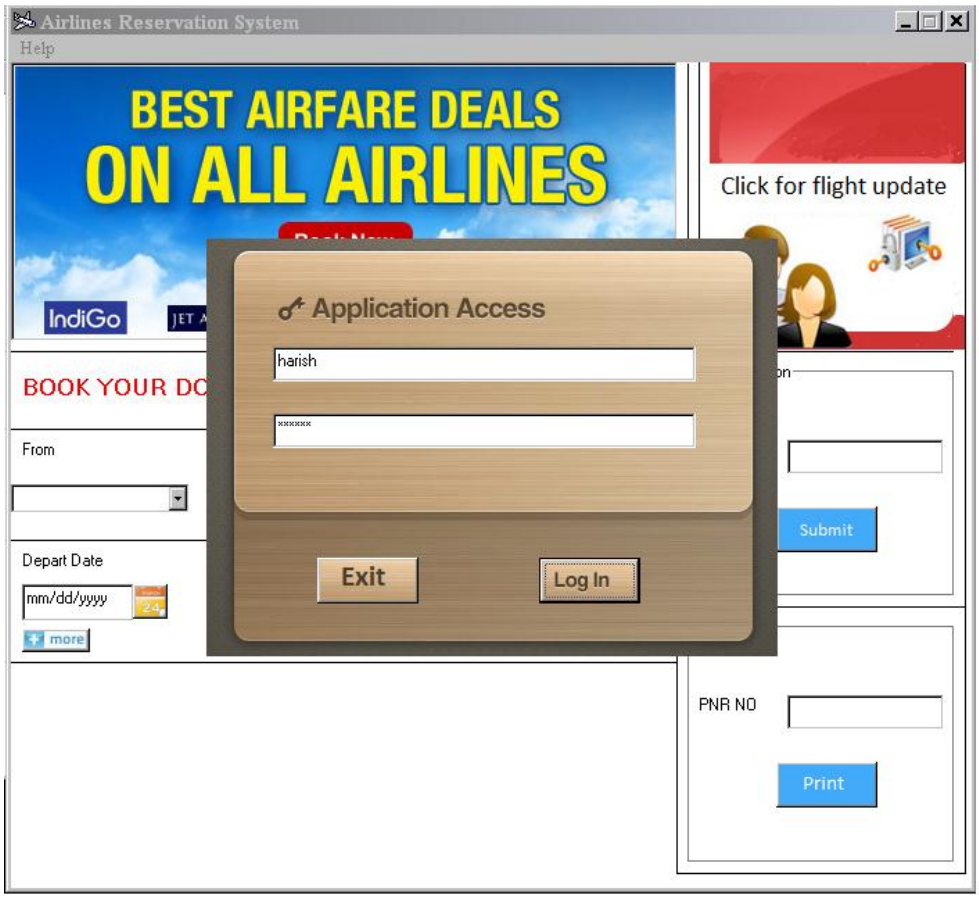
The basic purpose of designing & implementing the proposed system is to automate the process of day-to-day activities like searching the customer details, ticket reservation, ticket cancellation, generate ticket. The manual handling of the record is time consuming & highly prone to errors.

The complete set of rules & procedures related to sports club management for day to day activities is the basis for our system. Our project gives a brief idea regarding automated airline reservation system.

3.1 Program Development

3.1.1 LOGIN FORM

Output:



Code:

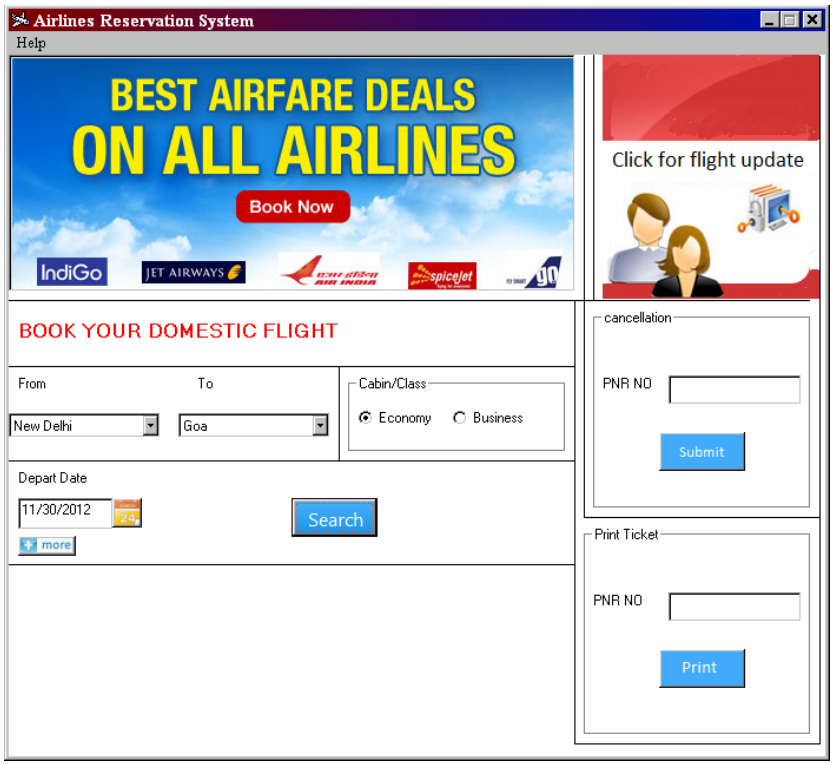
```
Dim db As New ADODB.Connection
Dim rs As New ADODB.Recordset

Private Sub back_Click()
End
End Sub

Private Sub Command1_Click()
Dim i As Integer
i = 0
On Error GoTo err_h:

If rs.BOF Then
MsgBox "currently no password is added in database .. use master password",
vbInformation
Exit Sub
End If

If Text1.Text = "" Then
Image1.Picture = ImageList.ListImages(4).Picture
Image1.Visible = True
End If

If Text2.Text = "" Then
Image2.Picture = ImageList.ListImages(4).Picture
Image2.Visible = True
End If
rs.MoveFirst
Do While Not rs.EOF
If Text1.Text = rs.Fields(0).Value Then
    Image1.Picture = ImageList.ListImages(3).Picture
    Image1.Visible = True
     i = 1
  If Text2.Text = rs.Fields(1).Value Then
     login_uname = rs.Fields(0).Value
     login_passwd = rs.Fields(1).Value
    Form1.Visible = True
    Text1.Text = ""
```

```
    Text2.Text = ""
    Form5.Visible = False
    Exit Sub
  Else
    Image2.Picture = ImageList.ListImages(4).Picture
     Image2.Visible = True
 End If
Else
   Image1.Picture = ImageList.ListImages(4).Picture
    Image1.Visible = True
End If
rs.MoveNext
Loop
If i = 1 Then
Image1.Picture = ImageList.ListImages(3).Picture
    Image1.Visible = True
    End If
Exit Sub
err_h:
MsgBox "contect to your admin or not connected to database"
End Sub

Private Sub Form_Load()
On Error GoTo err_h:
Open "database_connectivity.dat" For Binary As #1
Get #1, , user_name
Get #1, , pass_word
Close #1
Text1.TabIndex = 0
Picture1.Picture = ImageList.ListImages(1).Picture
Command1.Picture = ImageList.ListImages(2).Picture
Image1.Picture = ImageList.ListImages(3).Picture
Image2.Picture = ImageList.ListImages(4).Picture
back.Picture = ImageList.ListImages(5).Picture
db.ConnectionString = "dsn=airlines_data;uid=" & user_name & ";pwd=" & pass_word
& " "
db.Open
rs.Open "select * from login", db, adOpenDynamic, adLockOptimistic, adCmdText
Exit Sub
err_h:
```

```
MsgBox "currently  no password is added in database ", vbInformation
End Sub

Private Sub Text1_GotFocus()
Image1.Visible  = False
End Sub

Private Sub Text2_GotFocus()
Image2.Visible  = False
End Sub
```

3.1.2 MAIN FORM

Output:



Code:

```
Dim strSQL As String
Dim i As Integer
Dim time As Integer

Private Sub about_Click()
frmAbout.Show
End Sub

Private Sub business_cabin_Click()
Image6.Visible = False
cabin = "business"
End Sub

Private Sub Calendar1_Click()
Dim temp As Integer
temp = 0
If Calendar1.year < DatePart("yyyy", Now) Then
MsgBox "Wrong Date", vbInformation
    temp = 1
```

```
Else
    If Calendar1.month < DatePart("m", Now) And Calendar1.year <= DatePart("yyyy",
Now) Then
            MsgBox "Wrong Date", vbInformation
        temp = 1
        Exit Sub
    End If
       If Calendar1.day < DatePart("d", Now) And Calendar1.month <= DatePart("m",
Now) And
          Calendar1.year <= DatePart("yyyy", Now) Then
        MsgBox "Wrong Date", vbInformation
        temp = 1
      End If
End If
If temp = 0 Then
text1.Text = ""
text1.Text = Calendar1.Value
Calendar1.Visible = False
Else
text1.Text = "mm/dd/yyyy"
End If
End Sub

Private Sub Command1_Click()
If Calendar1.Visible = True Then
  Calendar1.Visible = False
Else
Calendar1.Visible = True
End If
End Sub

Private Sub Command1_GotFocus()
Image7.Visible = False
End Sub

Private Sub Command2_Click()
On Error GoTo err_h:
If Text2.Text = "" Then
 Image2.Visible = True
 Exit Sub
End If
pnr_cancel = Text2.Text
 Load Form4
If pnr_cancel = "notfound" Then
Image3.Visible = True
Unload Form4
```

```
Exit Sub
End If
Exit Sub
err_h:
MsgBox "error occureed while connecting to the database", vbExclamation
End Sub

Private Sub Command3_Click()
On Error GoTo err_h:
If Text3.Text = "" Then
Image9.Visible = True
Exit Sub
End If
Dim db As New ADODB.Connection
Dim rs As New ADODB.Recordset
Dim rs1 As New ADODB.Recordset
db.ConnectionString = "dsn=airlines_data;uid=" & user_name & ";pwd=" & pass_word
& " "
db.Open
rs.Open "select * from cust_data where pnr_no = '" & Text3.Text & "' ", db,
adOpenDynamic, adLockOptimistic, adCmdText
If rs.EOF Then
Image8.Visible = True
Else
rs1.Open "select * from flight_data where flight_no = '" & rs.Fields("flight_no") & "' ",
db, adOpenDynamic, adLockOptimistic, adCmdText

going_from = rs1.Fields("depart_city")

 going_to = rs1.Fields("arrival_city")
depart_date = rs1.Fields("depart_Date")
 cabin = rs.Fields("cabin")
flight_no = rs.Fields("flight_no")
first_name = rs.Fields("first_name")
last_name = rs.Fields("last_name")
 pnr_no = rs.Fields("pnr_no")
 If (0 = StrComp(cabin, "economy")) Then
   price = rs1.Fields("eco_price")
   Else
   price = rs1.Fields("busi_price")
 End If
depart_hour = rs1.Fields("depart_hour")
depart_minute = rs1.Fields("depart_minute")
rs1.Close
rs.Close
db.Close
```

```
dup_ticket = "true"
Load Form7
 Form7.Visible = True
End If
Exit Sub
err_h:
MsgBox "error occureed while connecting to the database", vbExclamation
End Sub

Private Sub Command4_Click()
If Frame4.Visible = True Then
Frame4.Visible = False
Command4.Picture = ImageList.ListImages(12).Picture
Exit Sub
End If
Image10.Visible = True
Option1.Value = True
Command4.Picture = ImageList.ListImages(13).Picture
Timer2.Enabled = True
Option2.Value = True
End Sub

Private Sub Command5_Click()
Load Form9
End Sub

Private Sub economy_cabin_Click()
Image6.Visible = False
cabin = "economy"
End Sub

Private Sub exit_Click()
Unload Form1
Unload Form2
Unload Form3
Unload Form4
Unload Form5
Unload Form6
Unload Form7
Unload Form8
Unload frmAbout
Unload frmSplash
End Sub

Private Sub Form_Click()
Calendar1.Visible = False
```

```
End Sub
Private Sub Form_Load()
Image2.Picture = ImageList.ListImages(7).Picture
Image4.Picture = ImageList.ListImages(7).Picture
Image5.Picture = ImageList.ListImages(7).Picture
Image6.Picture = ImageList.ListImages(7).Picture
Image7.Picture = ImageList.ListImages(7).Picture
Image9.Picture = ImageList.ListImages(7).Picture
Image3.Picture = ImageList.ListImages(8).Picture
Image8.Picture = ImageList.ListImages(8).Picture
Command1.Picture = ImageList.ListImages(9).Picture
show_flight_button.Picture = ImageList.ListImages(10).Picture
Image10.Picture = ImageList.ListImages(11).Picture
Image11.Picture = ImageList.ListImages(11).Picture
Command4.Picture = ImageList.ListImages(12).Picture
Command2.Picture = ImageList.ListImages(14).Picture
Command3.Picture = ImageList.ListImages(15).Picture
Image1.Picture = ImageList.ListImages(16).Picture
Image12.Picture = ImageList.ListImages(17).Picture
Calendar1.Visible = False
time = 10
Picturebox.Picture = ImageList.ListImages(2).Picture
Open "database_connectivity.dat" For Binary As #1
Get #1, , user_name
Get #1, , pass_word
Close #1
Calendar1.Value = Date
End Sub

Private Sub from_box_GotFocus()
Image4.Visible = False
Image12.Visible = False
End Sub

Private Sub helpbackup_Click()
Shell "winhlp32.exe -i backup help1.hlp"
End Sub

Private Sub helpbook_Click()
Shell "winhlp32.exe -i booking help1.hlp"
End Sub

Private Sub helpcancel_Click()
Shell "winhlp32.exe -i cancel help1.hlp"
End Sub
```

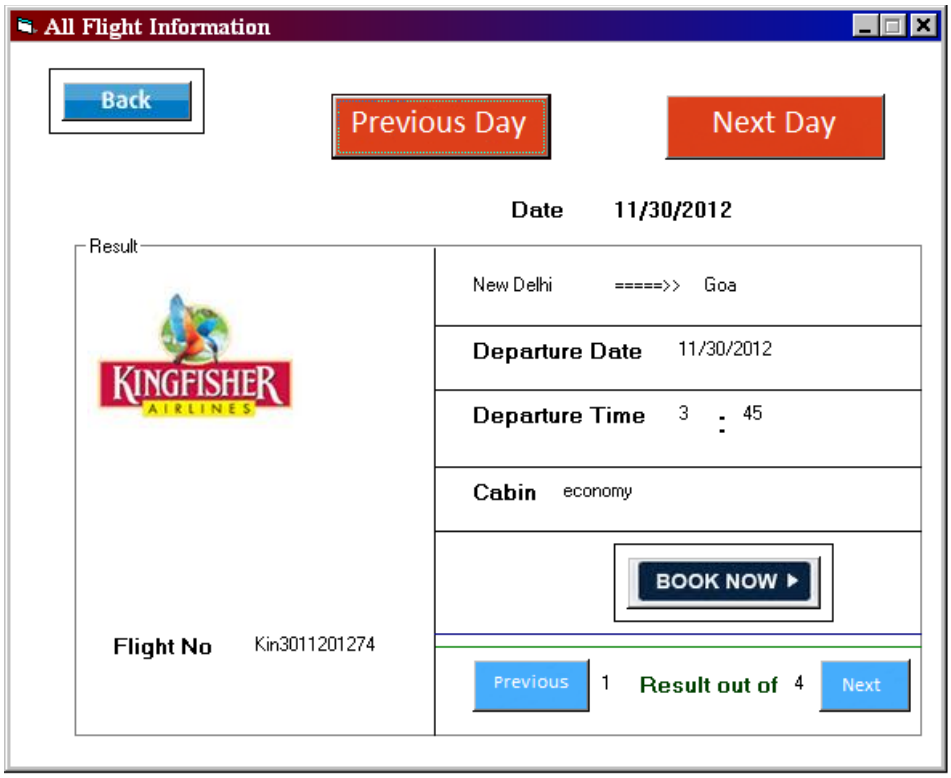
```
Private Sub helpchange_Click()
Shell "winhlp32.exe -i uidpassword help1.hlp"
End Sub

Private Sub helpconnection_Click()
Shell "winhlp32.exe -i connect help1.hlp"
End Sub

Private Sub helpdatabase_Click()
Shell "winhlp32.exe -i deletion help1.hlp"
End Sub

Private Sub helphelp_Click()
Shell "winhlp32.exe  help1.hlp"
End Sub

Private Sub helpmaster_Click()
Shell "winhlp32.exe -i masterpassword help1.hlp"
End Sub

Private Sub helpreprint_Click()
Shell "winhlp32.exe -i reprint help1.hlp"
End Sub

Private Sub helptreeview_Click()
Shell "winhlp32.exe -i treeview help1.hlp"
End Sub

Private Sub helpupdate_Click()
Shell "winhlp32.exe -i update help1.hlp"
End Sub

Private Sub Image1_Click()
Form6.Visible = True
End Sub

Private Sub Option1_GotFocus()
strSQL = "SELECT * FROM flight_Data"
Call show_database
End Sub
Private Sub Option2_Click()
strSQL = "SELECT * FROM cust_data"
Call show_database
End Sub
```

```
Private Sub show_flight_button_Click()
i = 0
Call check_all_fill
If i = 0 Then
going_from  = from_box.Text
going_to  = to_box.Text
depart_date = text1.Text
Image11.Visible  = True
Timer3.Enabled  = True
Exit Sub
End If
End Sub

Private Sub Text2_Click()
Image2.Visible  = False
Image3.Visible  = False
End Sub

Private Sub Text3_Click()
Image8.Visible  = False
Image9.Visible  = False
End Sub

Public Function check_all_fill()
  If from_box.Text  = "" Then
  i = 1
  Image4.Visible  = True
   End If
 If to_box.Text = "" Then
 i = 1
 Image5.Visible  = True
 End If
 If text1.Text  = "mm/dd/yyyy"  Then
  i = 1
  Image7.Visible  = True
  End If
If economy_cabin.Value  = False And business_cabin.Value  = False Then
i = 1
Image6.Visible  = True
End If
If from_box.Text  = to_box.Text  And from_box.Text  <> "" Then
i = 1
Image12.Visible  = True
End If
End Function
```

```
Private Sub Timer2_Timer()
Frame4.Visible = True
Image10.Visible = False
Timer2.Enabled = False
End Sub

Private Sub Timer3_Timer()
On Error GoTo err_h:
Form1.Visible = False
Form2.Visible = True
Image11.Visible = False
Timer3.Enabled = False
Exit Sub
err_h:
Form1.Visible = True
Image11.Visible = False
Timer3.Enabled = False
MsgBox "error occureed while connecting to the database", vbExclamation
End Sub

Private Sub to_box_GotFocus()
Image5.Visible = False
Image12.Visible = False
End Sub

Public Function show_database()
Dim oconn As New ADODB.Connection
Dim rs As New ADODB.Recordset
On Error GoTo err_h:
oconn.ConnectionString = "dsn=airlines_data;uid=" & user_name & ";pwd=" &
pass_word & " "
oconn.Open
rs.CursorType = adOpenStatic
rs.CursorLocation = adUseClient
rs.LockType = adLockOptimistic
rs.Open strSQL, oconn, , , adCmdText
Set DataGrid1.DataSource = rs
Exit Function
err_h:
MsgBox "error is occured while connecting to database", vbExclamation
End Function
```

3.1.3 All FLIGHT INFORMATION FORM

Output:



Code:

```
Dim count_no As Integer
Dim val As Integer
Dim db As New ADODB.Connection
Dim rs As New ADODB.Recordset

Private Sub book_now_Click()
flight_company = rs("flight_company").Value
flight_no = rs("flight_no").Value
depart_hour = rs("depart_hour").Value
depart_minute = rs("depart_minute").Value
db.Close
Form3.Visible = True
Unload Me
End Sub

Private Sub Command1_Click()
rs.Close
db.Close
```

```
Form1.Visible  = True
Unload  Me
End  Sub

Private  Sub Command2_Click()
If Command3.Enabled  = False  Then
   Command3.Enabled  = True
End  If
If val + 1 = count_no  Then
 Command2.Enabled  = False
End  If
Form2.Refresh
val = val + 1
Label7.Caption  = val
rs.MoveNext
Call  show_data
End  Sub

Private  Sub Command3_Click()
If Command2.Enabled  = False  Then
Command2.Enabled  = True
End  If
rs.MoveFirst
val = 1
Label7.Caption  = val
Call  show_data
Command3.Enabled  = False
End  Sub

Private  Sub Form_Load()
Command1.Picture  = ImageList.ListImages(1).Picture
previous_day.Picture  = ImageList.ListImages(2).Picture
next_day.Picture  = ImageList.ListImages(3).Picture
Command3.Picture  = ImageList.ListImages(4).Picture
Command2.Picture  = ImageList.ListImages(5).Picture
book_now.Picture  = ImageList.ListImages(6).Picture
On Error GoTo err_h:
db.ConnectionString  = "dsn=airlines_data;uid="  & user_name  & ";pwd="  & pass_word
& " "
db.Open
rs.Open "select  * from flight_data",  db, adOpenDynamic,  adLockOptimistic,  adCmdText
db.Properties.Refresh
rs.Properties.Refresh
 Call  start_searching
Exit  Sub
err_h:
```

```
 Unload Form2
End Sub
Public Function check_flight()
If 0 = StrComp(going_from, rs("depart_city").Value) Then
      If 0 = StrComp(going_to, rs("arrival_city").Value) Then
          If 0 = StrComp(depart_date, rs("depart_date").Value) Then
             If (0 = StrComp(cabin, "economy")) And rs("eco_no_of_seat").Value >= 1
Then
                 count_no = count_no + 1
                 rs.MoveNext
                 Exit Function
              Else
                 If (0 = StrComp(cabin, "business")) And (rs("busi_no_of_seat").Value >=
1) Then
                     count_no = count_no + 1
                  rs.MoveNext
                  Exit Function
                  End If
                  rs.MoveNext
                 Exit Function
               End If
           Else
              rs.MoveNext
                 Exit Function
           End If
      Else
          rs.MoveNext
           Exit Function
      End If
 Else
  rs.MoveNext
  Exit Function
 End If
End Function

Public Function show_data()
Do While (Not rs.EOF)
If 0 = StrComp(going_from, rs("depart_city").Value) Then
      If 0 = StrComp(going_to, rs("arrival_city").Value) Then
          If 0 = StrComp(depart_date, rs("depart_date").Value) Then
            If (0 = StrComp(cabin, "economy")) And (rs("eco_no_of_seat").Value >= 1)
Then
              Label1.Caption = rs("flight_no").Value
              Label2.Caption = going_from
              Label4.Caption = going_to
              Label5.Caption = depart_date
```

```
            Label6.Caption  = rs("depart_hour").Value
            Label11.Caption  = rs("depart_minute").Value
            Label16.Caption  = cabin
            price = rs("eco_price").Value
            Call show_images
            Exit Function
          Else
              If (0 = StrComp(cabin,  "business"))  And (rs("busi_no_of_seat").Value  >=
1) Then
               Label1.Caption  = rs("flight_no").Value
                 Label2.Caption  = going_from
                  Label4.Caption  = going_to
                 Label5.Caption  = depart_date
                 Label6.Caption  = rs("depart_hour").Value
                 Label11.Caption  = rs("depart_minute").Value
                 Label16.Caption  = cabin
                 price = rs("busi_price").Value
                 Call show_images
                 Exit Function
               Else
                 rs.MoveNext
               End If
            End If
         Else
            rs.MoveNext
         End If
      Else
         rs.MoveNext
       End If
Else
 rs.MoveNext
End If
Loop
End Function

Public Function show_images()
Select Case rs("flight_company").Value
Case "Jet Airways"
Image1.Picture  = ImageList.ListImages(7).Picture
Case "Indigo"
Image1.Picture  = ImageList.ListImages(8).Picture
Case "KingFisher"
Image1.Picture  = ImageList.ListImages(9).Picture
Case "SpiceJet"
Image1.Picture  = ImageList.ListImages(10).Picture
Case "AirIndia"
```

```
Image1.Picture  = ImageList.ListImages(11).Picture
End Select
End Function

Private Sub next_day_Click()
depart_date = DateAdd("d",  1, depart_date)
rs.MoveFirst
 Call start_searching
End Sub

Private Sub previous_day_Click()
depart_date = DateAdd("d",  -1, depart_date)
rs.MoveFirst
Call start_searching
End Sub

Public Function start_searching()
rs.MoveFirst
Command2.Enabled  = True
Command3.Enabled  = True
Label19.Caption  = depart_date
count_no  = 0
Do While  (Not rs.EOF)
Call check_flight
Loop
Label7.Caption  = 1
Label9.Caption  = count_no
val = 1
If (count_no  = 1) Then
Command2.Enabled  = False
Command3.Enabled  = False
End If
rs.MoveFirst
If count_no  >= 1 Then
Frame1.Visible  = True
Call show_data
Else
Frame1.Visible  = False
Label17.Visible  = True
End If
End Function
```

3.1.4 FLIGHT FORM

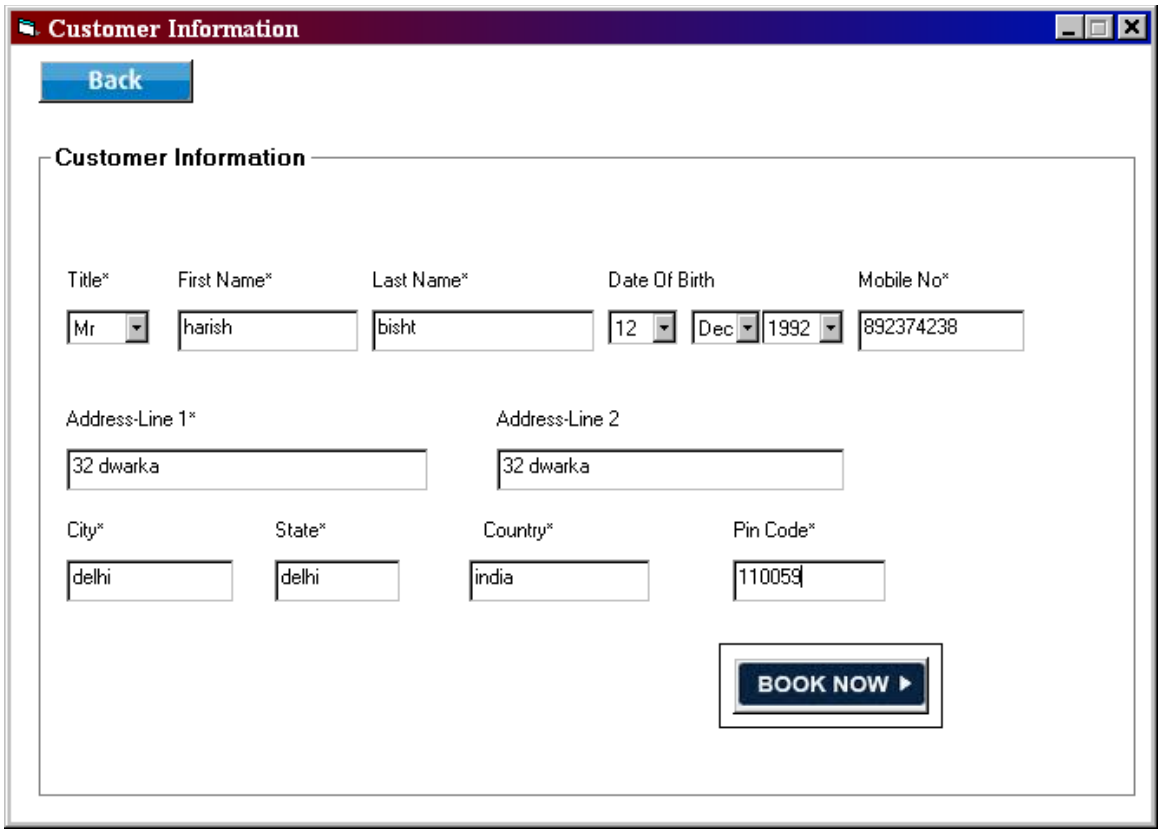
Output:

Code:

```
Private Sub Command1_Click()
Form3.Visible = False
Form8.Visible = True
End Sub
Private Sub Command2_Click()
Form2.Visible = True
Unload Me
End Sub

Private Sub Form_Load()
Command2.Picture = ImageList.ListImages(1).Picture
Command1.Picture = ImageList.ListImages(2).Picture
Label2.Caption = going_from
Label4.Caption = going_to
Label6.Caption = depart_date
Label8.Caption = cabin
Label10.Caption = flight_company
Label12.Caption = flight_no
Label14.Caption = depart_hour
Label16.Caption = depart_minute
Label18.Caption = price
End Sub
```

3.1.5 CUSTOMER INFORMATION FORM

Output:



Code:

```
Dim pnrno As String
Dim i As Integer
Dim db As New ADODB.Connection
Dim rs As New ADODB.Recordset

Private Sub add1_box_Change()
Image5.Visible = False
End Sub

Private Sub city_box_Change()
Image6.Visible = False
End Sub

Private Sub Command1_Click()
Form3.Visible = True
db.Close
Unload Me
```

```
End Sub

Private Sub continue_booking_button_Click()
On Error GoTo err_h:
i = 0
Call check_all_fill
If i = 0 Then
pnrno = Left(fname_box.Text, 3) & Right(mobile_box.Text, 5) & CInt(Int(Rnd() *
Int(Rnd() * 199)))
rs.Open "insert into cust_data values('" & title_box.Text & "','" & fname_box.Text & "','"
& lname_box.Text & "','" & birth_date.Text & "','" & birth_month.Text & "','" &
birth_year.Text & "','" & mobile_box.Text & "','" & add1_box.Text & "','" &
add2_box.Text & "','" & city_box.Text & "','" & state_box.Text & "','" &
country_box.Text & "','" & pin_box.Text & "','" & pnrno & "','" & flight_no & "','" &
cabin & "')", db, adOpenDynamic, adLockOptimistic, adCmdText
first_name = fname_box.Text
last_name = lname_box.Text
pnr_no = pnrno
continue_booking_button.Picture = ImageList.ListImages(5).Picture
dup_ticket = "false"
Timer1.Enabled = True
End If
Exit Sub
err_h:
MsgBox "fill all the checkbox correctly", vbExclamation
End Sub

Private Sub country_box_Change()
Image8.Visible = False
End Sub

Private Sub fname_box_Change()
Image2.Visible = False
End Sub

Public Function check_all_fill()
If title_box.Text = "" Then
Image1.Visible = True
i = 1
End If
If fname_box.Text = "" Then
  Image2.Visible = True
i = 1
  End If
If lname_box.Text = "" Then
Image3.Visible = True
```

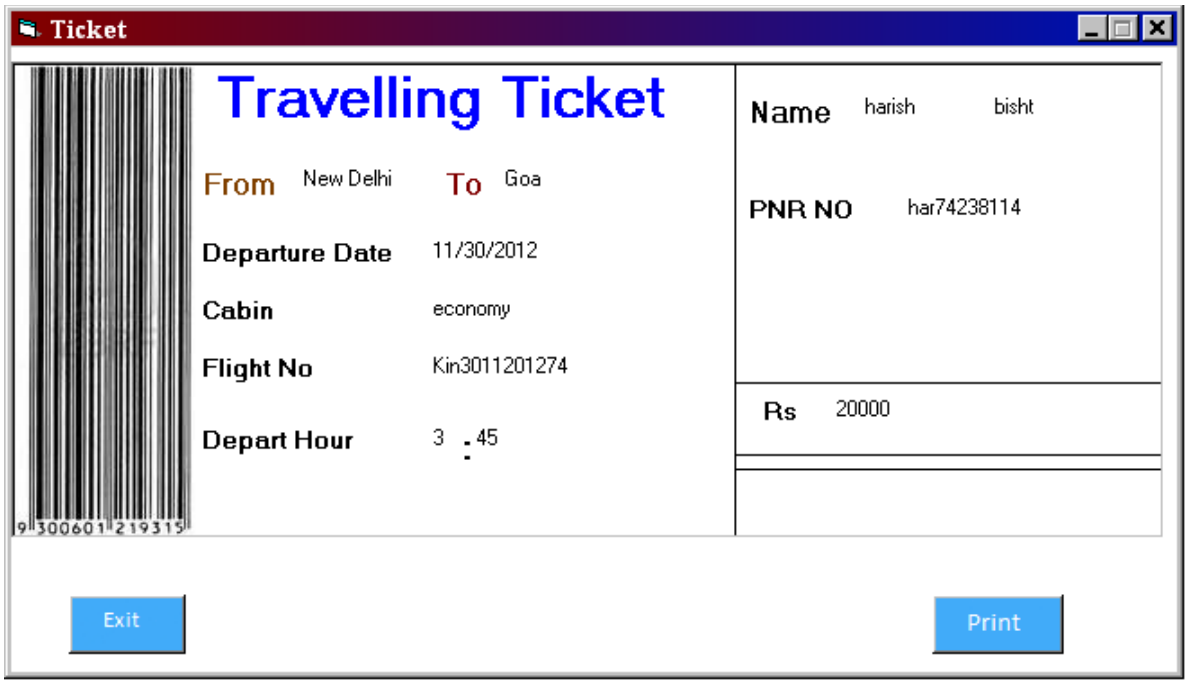
```
i = 1
End If
If mobile_box.Text  = "" Then
 Image4.Visible  = True
 i = 1
 End If
If add1_box.Text = "" Then
 Image5.Visible  = True
 i = 1
End If
If city_box.Text  = "" Then
 Image6.Visible  = True
 i = 1
End If
If state_box.Text  = "" Then
  Image7.Visible  = True
  i = 1
End If
If country_box.Text  = "" Then
  Image8.Visible  = True
 i = 1
End If
If pin_box.Text  = "" Then
  Image9.Visible  = True
  i = 1
End If
If Image10.Visible  = True Then
 i = 1
End If
End Function

Private Sub Form_Load()
Image1.Picture  = ImageList.ListImages(1).Picture
Image2.Picture  = ImageList.ListImages(1).Picture
Image3.Picture  = ImageList.ListImages(1).Picture
Image4.Picture  = ImageList.ListImages(1).Picture
Image5.Picture  = ImageList.ListImages(1).Picture
Image6.Picture  = ImageList.ListImages(1).Picture
Image7.Picture  = ImageList.ListImages(1).Picture
Image8.Picture  = ImageList.ListImages(1).Picture
Image9.Picture  = ImageList.ListImages(1).Picture
Image10.Picture  = ImageList.ListImages(2).Picture
continue_booking_button.Picture  = ImageList.ListImages(4).Picture
Command1.Picture  = ImageList.ListImages(3).Picture
db.ConnectionString  = "dsn=airlines_data;uid="  & user_name  & ";pwd="  & pass_word
& " "
```

```
db.Open
End Sub

Private Sub lname_box_Change()
Image3.Visible = False
End Sub

Private Sub mobile_box_Change()
Image4.Visible = False
Image10.Visible = False
End Sub

Private Sub mobile_box_KeyPress(KeyAscii As Integer)
If KeyAscii = vbKeyBack Then
Exit Sub
End If
If KeyAscii < Asc("0") Or KeyAscii > Asc("9") Then
KeyAscii = 0
End If
End Sub

Private Sub mobile_box_LostFocus()
If Len(mobile_box.Text) <= 5 Then
Image10.Visible = True
i = 1
Else
i = 0
End If
End Sub

Private Sub pin_box_Change()
Image9.Visible = False
End Sub

Private Sub pin_box_KeyPress(KeyAscii As Integer)
If KeyAscii < Asc("0") Or KeyAscii > Asc("9") Then
KeyAscii = 0
End If
End Sub

Private Sub state_box_Change()
Image7.Visible = False
End Sub

Private Sub Timer1_Timer()
continue_booking_button.Picture = ImageList.ListImages(4).Picture
```

```
db.Close
Form7.Visible  = True
Form8.Visible  = False
Timer1.Enabled  = False
End Sub

Private Sub title_box_Change()
Image1.Visible  = False
End Sub
Private Sub title_box_Click()
Image1.Visible  = False
End Sub
```

3.1.6 TICKET FORM

Output:



Code:

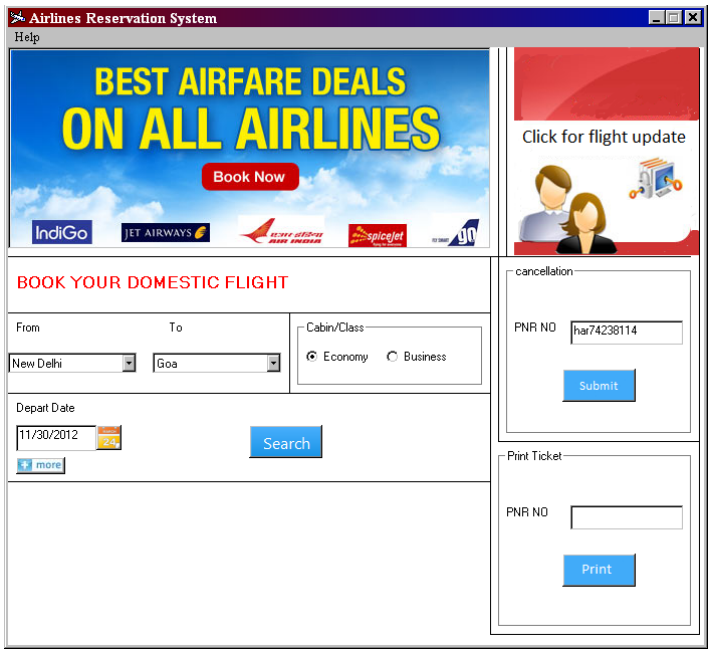
```
Dim temp As Integer
Dim db As New ADODB.Connection
Dim rs As New ADODB.Recordset

Private Sub Command1_Click()
CommonDialog1.PrinterDefault = True
CommonDialog1.ShowPrinter
Print Picture1
End Sub

Private Sub Command2_Click()
db.Close
Unload Form2
Unload Form3
Unload Form8
Form1.Visible = True
Unload Form7
End Sub

Private Sub Form_Load()
```

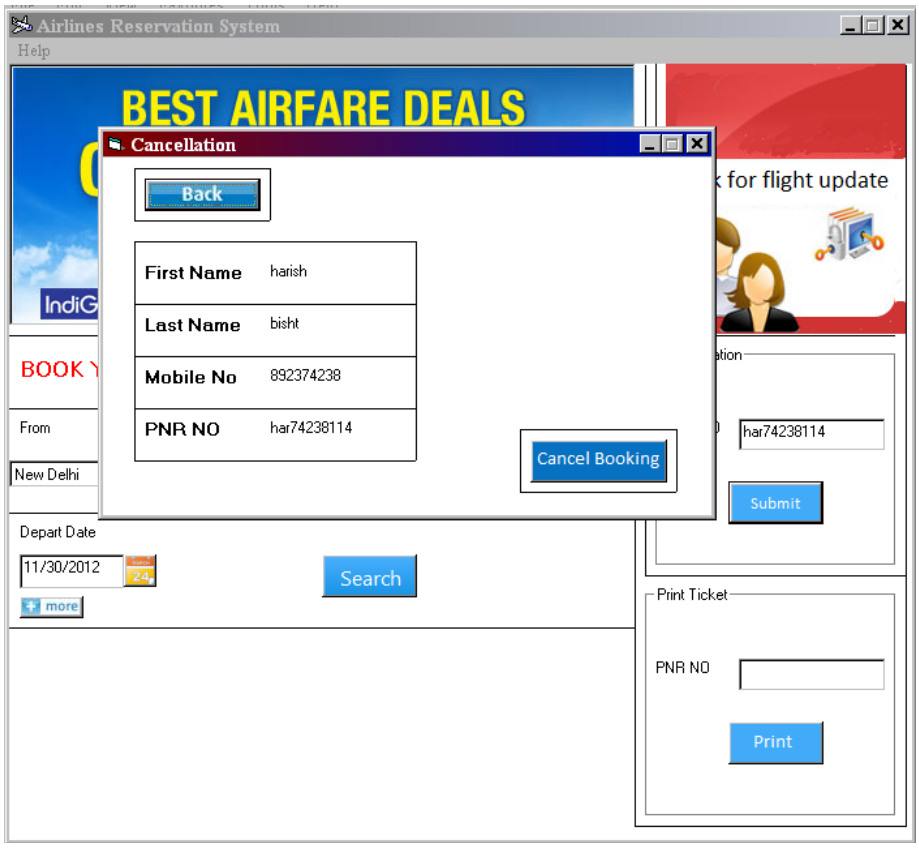
```
db.ConnectionString = "dsn=airlines_data;uid=" & user_name & ";pwd=" & pass_word
& " "
db.Open
Image1.Picture = ImageList.ListImages(1).Picture
Command1.Picture = ImageList.ListImages(3).Picture
Command2.Picture = ImageList.ListImages(2).Picture
Label3.Caption = going_from
Label5.Caption = going_to
Label7.Caption = depart_date
Label9.Caption = cabin
Label11.Caption = flight_no
Label13.Caption = first_name
Label14.Caption = last_name
Label18.Caption = pnr_no
Label16.Caption = price
Label20.Caption = depart_hour
Label22.Caption = depart_minute

If dup_ticket = "true" Then
dup_ticket = "false"
Exit Sub
End If
'code for decrement the seat from database
rs.Open "select * from flight_data", db, adOpenDynamic, adLockOptimistic, adCmdText
rs.MoveFirst
Do While (Not rs.EOF)
If 0 = StrComp(flight_company, rs("flight_company").Value) Then
 If 0 = StrComp(going_from, rs("depart_city").Value) Then
       If 0 = StrComp(going_to, rs("arrival_city").Value) Then
            If 0 = StrComp(depart_date, rs("depart_date").Value) Then
              If 0 = StrComp(flight_no, rs("flight_no").Value) Then
                 If (0 = StrComp(cabin, "economy")) And (rs("eco_no_of_seat").Value >= 1)
Then
                     temp = rs("eco_no_of_seat").Value
                     temp = temp - 1
                     rs.Fields("eco_no_of_seat") = temp
                     rs.Update
                      Exit Sub
                 Else
                   If (0 = StrComp(cabin, "business")) And (rs("busi_no_of_seat").Value >=
1) Then
                     temp = rs("busi_no_of_seat").Value
                     temp = temp - 1
                     rs.Fields("busi_no_of_seat") = temp
                     rs.Update
                     Exit Sub
```

```
                End If
              End If
            Else
            rs.MoveNext
            End If
          Else
         rs.MoveNext
          End If
      Else
         rs.MoveNext
       End If
Else
 rs.MoveNext
End If
Else
rs.MoveNext
End If
Loop
End Sub
```

3.1.7 MAIN FORM (FOR CANCELLATION)

Output:





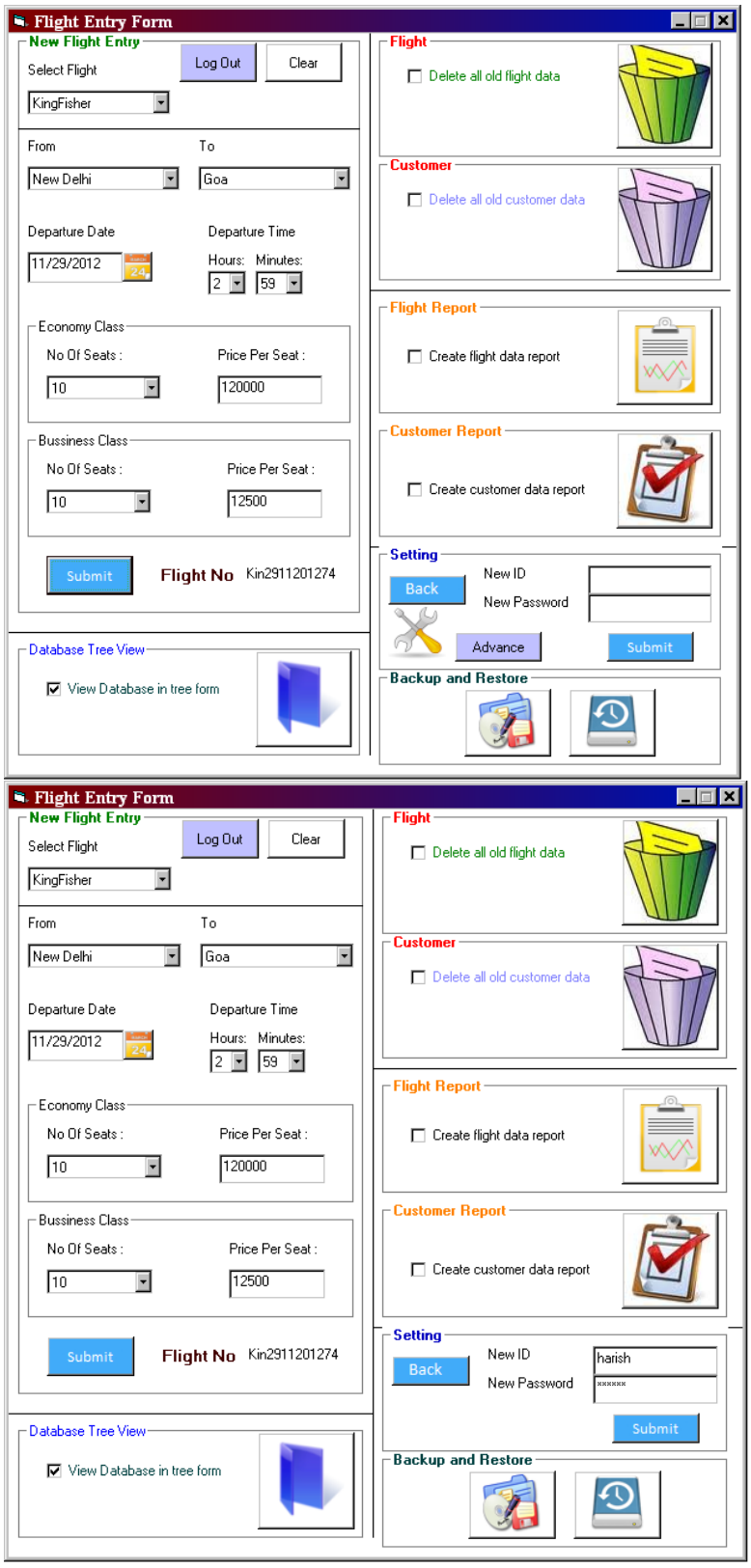
Code:

```
Dim temp As Integer
Dim db As New ADODB.Connection
Dim rs1 As New ADODB.Recordset
Dim rs As New ADODB.Recordset

Private Sub Command1_Click()
db.Close
Unload Me
End Sub

Private Sub Command2_Click()
 cabin = rs.Fields("cabin")
 flight_no = rs.Fields("flight_no")
 Call increment_seat
rs.Delete
db.Close
MsgBox " booking cancelled sucessful", vbInformation
Unload Me
End Sub

Private Sub Form_Load()
Command1.Picture = ImageList.ListImages(1).Picture
Command2.Picture = ImageList.ListImages(2).Picture
On Error GoTo err_h:
db.ConnectionString = "dsn=airlines_data;uid=" & user_name & ";pwd=" & pass_word
& " "
db.Open
rs.Open "select * from cust_data where pnr_no = '" & pnr_cancel & "' ", db,
adOpenDynamic, adLockOptimistic, adCmdText
If (Not rs.EOF) Then
Form4.Visible = True
Form4.Refresh
Label5.Caption = rs.Fields("first_name")
Label6.Caption = rs.Fields("last_name")
Label7.Caption = rs.Fields("mobile_no")
Label8.Caption = rs.Fields("pnr_no")
Else
pnr_cancel = "notfound"
db.Close
End If
Exit Sub
err_h:
Unload Form4
End Sub
```

```
Public Function increment_seat()
rs1.Open "select * from flight_data where flight_no = '" & flight_no & "' ", db,
adOpenDynamic, adLockOptimistic, adCmdText
If cabin = "economy" Then
 temp = rs1("eco_no_of_seat").Value
  temp = temp + 1
   rs1.Fields("eco_no_of_seat") = temp
        rs1.Update
Else
 temp = rs1("busi_no_of_seat").Value
  temp = temp + 1
   rs1.Fields("busi_no_of_seat") = temp
   rs1.Update
End If
End Function
```

3.1.8 FLIGHT ENTRY FORM

Output:



Code:

```
Dim setting  As Integer
Dim db As New ADODB.Connection
Dim rs As New ADODB.Recordset
Dim rs1 As New ADODB.Recordset
Dim flight_no  As String
Dim val As Integer
Private  Sub admin_setting_Click()
admin_setting.Visible  = False
database_setting.Visible  = False
back.Visible  = True
Label8.Visible  = True
Label9.Visible  = True
Text1.Visible  = True
Text2.Visible  = True
submit.Visible  = True
setting  = 0
Text1.SetFocus
Text2.PasswordChar  = "*"
End  Sub

Private  Sub advance_setting_Click()
Label8.Visible  = False
Label9.Visible  = False
Label10.Visible  = True
Text1.Visible  = False
Text2.Visible  = False
advance_setting.Visible  = False
submit.Visible  = False
Create_database.Visible  = True
Check5.Visible  = True
driver.Visible  = True
Combo1.Visible  = True
End  Sub

Private  Sub back_Click()
admin_setting.Visible  = True
database_setting.Visible  = True
back.Visible  = False
Label8.Visible  = False
Label9.Visible  = False
Text1.Visible  = False
Text2.Visible  = False
submit.Visible  = False
Image1.Visible  = False
advance_setting.Visible  = False
```

```
Create_database.Visible  = False
Check5.Visible  = False
driver.Visible  = False
Label10.Visible  = False
Combo1.Visible  = False
End Sub

Private Sub Calendar1_Click()
Dim temp As Integer
temp = 0
If Calendar1.year < DatePart("yyyy", Now) Then
MsgBox "Wrong Date", vbInformation
   temp = 1
Else
    If Calendar1.month < DatePart("m", Now) And Calendar1.year <= DatePart("yyyy",
Now) Then
       MsgBox "Wrong Date", vbInformation
        temp = 1
        Exit Sub
    End If
       If Calendar1.day < DatePart("d", Now) And Calendar1.month <= DatePart("m",
Now) And Calendar1.year <= DatePart("yyyy", Now) Then
        MsgBox "Wrong Date", vbInformation
        temp = 1
      End If
End If
If temp = 0 Then
date_box.Text = ""
date_box.Text = Calendar1.Value
Calendar1.Visible  = False
Else
date_box.Text = "mm/dd/yyyy"
End If
End Sub

Private Sub clear_button_Click()
date_box = "mm/dd/yyyy"
price_economy.Text = ""
price_business.Text = ""
Label12.Caption = ""
End Sub

Private Sub Command1_Click()
If Calendar1.Visible = True Then
 Calendar1.Visible = False
Else
```

```
Calendar1.Visible  = True
End If
End Sub

Private Sub Command2_Click()
On Error GoTo err_h:
db.Close
Unload Me
Exit Sub
err_h:
Unload Me
End Sub

Private Sub Command3_Click()
If Check3.Value  = 1 Then
flight_data_report.Show
Else
MsgBox "Please tick the check box", vbExclamation
End If
End Sub

Private Sub Command4_Click()
If Check4.Value  = 1 Then
cust_data_report.Show
Else
MsgBox "Please tick the check box", vbExclamation
End If
End Sub

Private Sub Command5_Click()
On Error Resume Next:
If Check6.Value  = 1 Then
Form10.Visible  = True
Else
MsgBox "Please tick the check box", vbExclamation
End If
End Sub

Private Sub create_backup_Click()
Dim day As Variant, month As Variant, year As Variant, add As Variant
On Error Resume Next
CommonDialog1.Filter  = "Backup file (*.backup)|*.backup"
CommonDialog1.FileName  = Date$
CommonDialog1.DialogTitle  = "select the location for Backup file"
CommonDialog1.Flags  = 0
CommonDialog1.ShowSave
```

```
Kill CommonDialog1.FileName 'check for existing files fo backup
 If CommonDialog1.Flags <> 0 Then
  rs.Open "select * from flight_data", db, adOpenDynamic, adLockOptimistic,
adCmdText
  rs1.Open "select * from cust_data", db, adOpenDynamic, adLockOptimistic,
adCmdText
  rs.MoveFirst
  Open CommonDialog1.FileName For Append As #1
  Do While (Not rs.EOF)
  Write #1, rs.Fields(0), rs.Fields(1), rs.Fields(2), rs.Fields(3), rs.Fields(4), rs.Fields(5),
rs.Fields(6), rs.Fields(7), rs.Fields(8), rs.Fields(9), rs.Fields(10)
  rs.MoveNext
     If (rs.EOF) Then
     Write #1, "***", "***", "***", "***", "***", "***", "***", "***", "***", "***",
"***"
     rs1.MoveFirst
     Do While (Not rs1.EOF)
     If IsNull(rs1.Fields(3)) Then day = "blank"
     If IsNull(rs1.Fields(4)) Then month = "blank"
     If IsNull(rs1.Fields(5)) Then year = "blank"
     If IsNull(rs1.Fields(8)) Then add = "blank"
      Write #1, rs1.Fields(0), rs1.Fields(1), rs1.Fields(2), day, month, year, rs1.Fields(6),
rs1.Fields(7), add, rs1.Fields(9), rs1.Fields(10), rs1.Fields(11), rs1.Fields(12),
rs1.Fields(13), rs1.Fields(14), rs1.Fields(15)
     rs1.MoveNext
     Loop
    End If
   Loop
  MsgBox "Backup creation sucessful", vbInformation
  Close #1
  rs.Close
  rs1.Close
Else
Exit Sub
End If
End Sub

Private Sub customer_delete_Click()
On Error GoTo err_h:
Dim tempi As Integer
Dim counter As Integer
counter = 0
tempi = 0
If Check2.Value = 1 Then
 rs1.Open "select * from cust_data ", db, adOpenDynamic, adLockOptimistic,
adCmdText
```

```
If (Not rs1.BOF) Then
rs1.MoveFirst
Else
MsgBox "Customer DataBase is empty", vbInformation
rs1.Close
Exit Sub
End If
rs.Open "select * from flight_data", db, adOpenDynamic, adLockOptimistic, adCmdText
If (rs.EOF) Then
  Do While (Not rs1.EOF)
  rs1.Delete
  rs1.MoveNext
  counter = counter + 1
  Loop
MsgBox counter & " customer data deleted"
rs.Close
rs1.Close
Exit Sub
Else
rs.MoveFirst
End If
Do While (Not rs1.EOF)
  Do While (Not rs.EOF)
    If 0 = StrComp(rs("flight_no").Value, rs1("flight_no").Value) Then
    tempi = 1
   rs.MoveNext
   Else
     rs.MoveNext
   End If
   If rs.EOF And tempi = 0 Then
     counter = counter + 1
     rs1.Delete
      tempi = 0
   End If
   Loop
rs.MoveFirst
rs1.MoveNext
Loop
MsgBox counter & " customer data deleted", vbInformation
rs.Close
rs1.Close
Else
MsgBox "Please tick the check box", vbExclamation
End If
Exit Sub
err_h:
```

```
MsgBox "error occureed while connecting to the database", vbInformation
End Sub

Private Sub database_setting_Click()
admin_setting.Visible = False
database_setting.Visible = False
back.Visible = True
Label8.Visible = True
Label9.Visible = True
Text1.Visible = True
Text2.Visible = True
submit.Visible = True
Image1.Visible = True
advance_setting.Visible = True
Text1.SetFocus
setting = 1
Text2.PasswordChar = ""
End Sub

Private Sub flight_delete_Click()
On Error GoTo err_h:
Dim tempdate As Date
Dim tempi As Integer
tempi = 0
tempdate = Date
tempdate = DateAdd("d", -1, tempdate)
If Check1.Value = 1 Then ' means checked
 rs.Open "select * from flight_data", db, adOpenDynamic, adLockOptimistic,
adCmdText
 If (Not rs.BOF) Then
  rs.MoveFirst
 Else
 MsgBox "Flight DataBase is empty", vbInformation
 rs.Close
 Exit Sub
 End If
 Do While (Not rs.EOF)
 If rs("depart_date").Value < tempdate Then
 rs.Delete
 tempi = tempi + 1
 End If
 rs.MoveNext
 Loop
 MsgBox tempi & " flight deleted", vbInformation
rs.Close
Else
```

```
MsgBox "Please Tick the Check box", vbExclamation
End If
Exit Sub
err_h:
MsgBox "error occureed while connecting to the database", vbInformation
End Sub

Private Sub Form_Click()
Calendar1.Visible = False
End Sub

Private Sub Form_Load()
Calendar1.Value = Date
Command1.Picture = ImageList.ListImages(1).Picture
submit_button.Picture = ImageList.ListImages(2).Picture
flight_delete.Picture = ImageList.ListImages(3).Picture
customer_delete.Picture = ImageList.ListImages(4).Picture
Command3.Picture = ImageList.ListImages(5).Picture
Command4.Picture = ImageList.ListImages(6).Picture
submit.Picture = ImageList.ListImages(2).Picture
back.Picture = ImageList.ListImages(7).Picture
admin_setting.Picture = ImageList.ListImages(8).Picture
database_setting.Picture = ImageList.ListImages(10).Picture
Image1.Picture = ImageList.ListImages(11).Picture
Command5.Picture = ImageList.ListImages(12).Picture
create_backup.Picture = ImageList.ListImages(13).Picture
restore_backup.Picture = ImageList.ListImages(14).Picture
On Error GoTo err_h:
db.ConnectionString = "dsn=airlines_data;uid=" & user_name & ";pwd=" & pass_word
& " "
db.Open
Calendar1.Visible = False
Exit Sub
err_h:
Calendar1.Visible = False
MsgBox "u have to need change database password and user name or fix the connectivity
problem", vbInformation
Timer1.Enabled = True
End Sub

Public Function check_all_are_fill()
val = 0
If select_flight.Text = "" Then
val = 1
End If
If from_box.Text = "" Then
```

```
val = 1
End If
If to_box.Text = "" Then
val = 1
End If
If date_box.Text = "mm/dd/yyyy" Then
val = 1
End If
If hour_box.Text = "" Or minutes_box.Text = "" Then
val = 1
End If
If seat_economy.Text = "" Or price_economy.Text = "" Or seat_business.Text = "" Or
price_business.Text = "" Then
val = 1
End If
If from_box.Text = to_box.Text Then
MsgBox "going to same city not possible"
val = 1
End If
End Function

Private Sub price_business_KeyPress(KeyAscii As Integer)
If KeyAscii = vbKeyBack Then
Exit Sub
End If
If KeyAscii < Asc("0") Or KeyAscii > Asc("9") Then
KeyAscii = 0
End If
End Sub

Private Sub price_economy_KeyPress(KeyAscii As Integer)
If KeyAscii = vbKeyBack Then
Exit Sub
End If
If KeyAscii < Asc("0") Or KeyAscii > Asc("9") Then
KeyAscii = 0
End If
End Sub

Private Sub restore_backup_Click()
Dim a As String, b As String, c As String, d As String, e As Variant, f As Variant, g As
Variant, h As Variant, i As Variant, j As Variant, k As Variant
Dim aa As String, ab As String, ac As String, ad As Variant, ae As String, af As Variant,
ag As Variant, ah As Variant, ai As Variant, aj As Variant, ak As Variant, al As Variant,
am As Variant, an As Variant, ao As Variant, ap As Variant
CommonDialog1.Filter = "Backup file (*.backup)|*.backup"
```

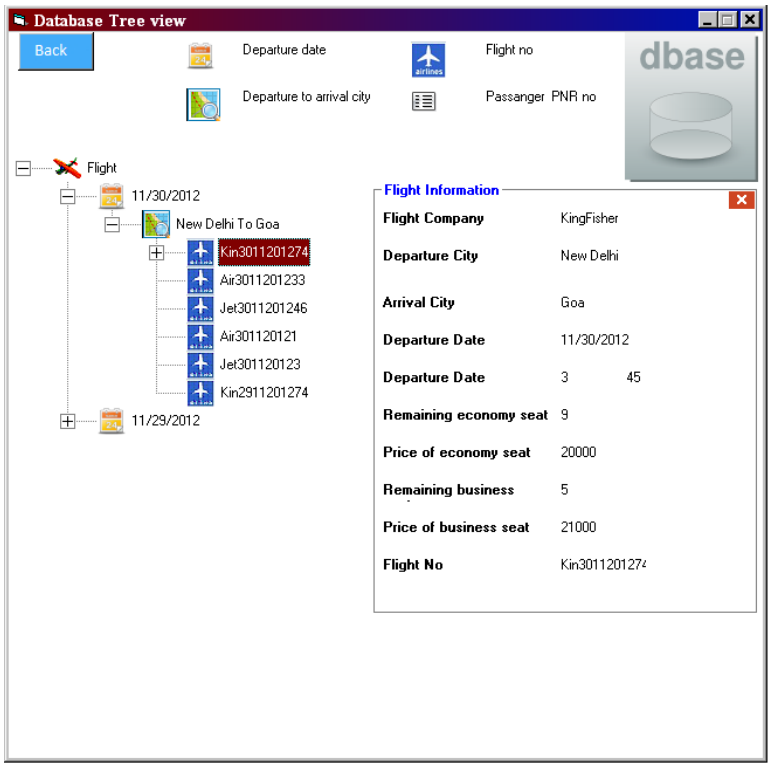
```
CommonDialog1.DialogTitle = "Select the airlines data backup file"
CommonDialog1.Flags = 0
CommonDialog1.ShowOpen
If CommonDialog1.Flags <> 0 Then
On Error GoTo err_h:
 Open CommonDialog1.FileName For Input As #1
 db.Execute "delete from flight_data"
 Do While (1)
 Input #1, a, b, c, d, e, f, g, h, i, j, k
 If a = "***" Then
   db.Execute "delete from cust_data"
    Do While (1)
        Input #1, aa, ab, ac, ad, ae, af, ag, ah, ai, aj, ak, al, am, an, ao, ap

          If ad = "blank" Then ad = ""
          If ae = "blank" Then ae = ""
          If af = "blank" Then af = ""
          If ai = "blank" Then ai = ""
          rs1.Open "insert into cust_data values('" & aa & "','" & ab & "','" & ac & "','" & ad
& "','" & ae & "','" & af & "','" & ag & "','" & ah & "','" & ai & "','" & aj & "','" & ak &
"','" & al & "','" & am & "','" & an & "','" & ao & "','" & ap & "')", db, adOpenDynamic,
adLockOptimistic, adCmdText
     Loop
    End If
  rs.Open "insert into flight_data values('" & a & "','" & b & "','" & c & "','" & d & "','" &
e & "','" & f & "','" & g & "','" & h & "','" & i & "','" & j & "','" & k & "')", db,
adOpenDynamic, adLockOptimistic, adCmdText
 Loop
 Close #1
Else
Exit Sub
End If
Exit Sub
err_h:
MsgBox "backup restore sucessful", vbInformation
Close #1
End Sub

Private Sub submit_button_Click()
On Error GoTo err_h:
Call check_all_are_fill
If val = 1 Then
  MsgBox "please fill all the entry", vbExclamation
  Exit Sub
Else
flight_no = Left(select_flight.Text, 3) & Calendar1.day & Calendar1.month &
```

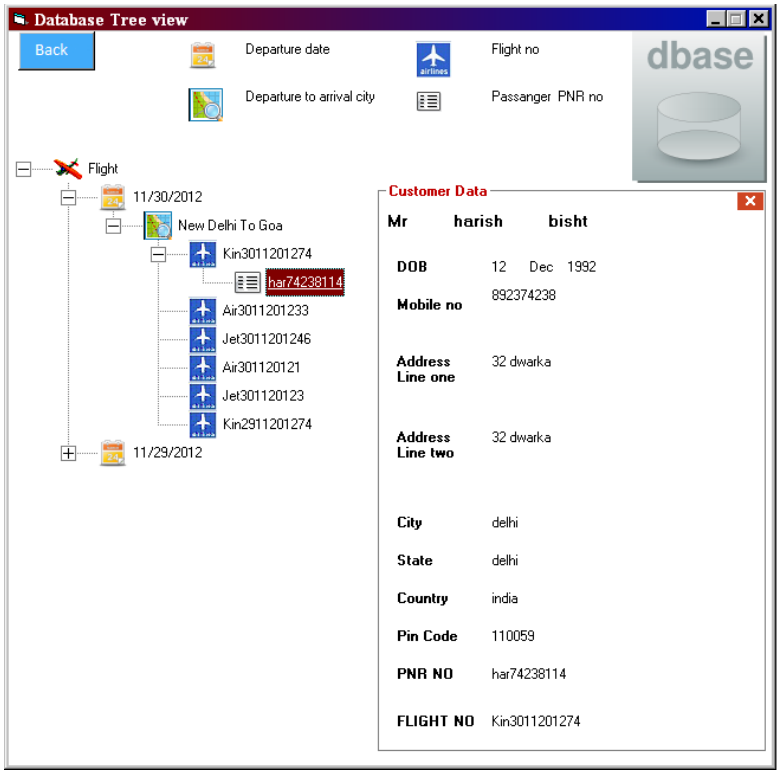
```
Calendar1.year & CInt(Int(Rnd() * Int(Rnd() * 199)))
rs.Open "insert into flight_data values('" & select_flight.Text & "','" & from_box.Text &
"','" & to_box.Text & "','" & date_box.Text & "','" & hour_box.Text & "','" &
minutes_box.Text & "','" & seat_economy.Text & "','" & price_economy.Text & "','" &
seat_business.Text & "','" & price_business.Text & "','" & flight_no & "')", db,
adOpenDynamic, adLockOptimistic, adCmdText
Label12.Caption = flight_no
End If
Exit Sub
err_h:
MsgBox "error occureed while connecting to the database", vbInformation
End Sub


Private Sub submit_Click()
On Error GoTo err_h:
Dim valdata As Variant
Dim valdata1 As Variant
If Text1.Text = "" Or Text2.Text = "" Then
MsgBox "Fill both of box", vbInformation
Exit Sub
End If
If setting = 0 Then
Call change_password
Else
Open "database_connectivity.dat" For Binary As #1
valdata = Text1.Text
valdata1 = Text2.Text
Put #1, , valdata
Put #1, , valdata1
Close #1
MsgBox "Changes Sucessful", vbInformation
Open "database_connectivity.dat" For Binary As #1
Get #1, , user_name
Get #1, , pass_word
Close #1
If db.State Then
db.Close
End If
db.ConnectionString = "dsn=airlines_data;uid=" & user_name & ";pwd=" & pass_word
& " "
db.Open
End If
valdata = ""
valdata1 = ""
admin_setting.Visible = True
```

```
database_setting.Visible = True
back.Visible = False
Label8.Visible = False
Label9.Visible = False
Text1.Visible = False
Text2.Visible = False
submit.Visible = False
Text1.Text = ""
Text2.Text = ""
Timer1.Enabled = False
database_setting.Picture = ImageList.ListImages(10).Picture
Image1.Visible = False
advance_setting.Visible = False
Exit Sub
err_h:
MsgBox "error occureed while reconnecting to the database", vbInformation
Timer1.Enabled = True
End Sub

Private Sub Timer1_Timer()
If (database_setting.Picture = ImageList.ListImages(9).Picture) Then
database_setting.Picture = ImageList.ListImages(10).Picture
Else
database_setting.Picture = ImageList.ListImages(9).Picture
End If
End Sub

Public Sub change_password()
On Error GoTo err_h:

Dim valdata As Variant
Dim valdata1 As Variant
valdata = Text1.Text
valdata1 = Text2.Text
db.Execute "delete from Login where username = '" & login_uname & "' and password=
'" & login_passwd & "' "
db.Execute "insert into login values('" & valdata & "','" & valdata1 & "')"
login_uname = Text1.Text
login_passwd = Text2.Text
MsgBox "username and password sucessfully changed", vbInformation
Exit Sub
err_h:
MsgBox "User name and password not found", vbInformation
End Sub
```

3.1.9 DATABASE TREE VIEW FORM

Output:





Code:

```
Dim db As New ADODB.Connection
Dim rs2 As New ADODB.Recordset
Dim rs1 As New ADODB.Recordset
Dim rs As New ADODB.Recordset

Private Sub Command1_Click()
db.Close
Unload Form10
End Sub

Private Sub Form_Click()
Image7.Visible = False
End Sub

Private Sub Form_Load()
On Error GoTo err_h:
db.ConnectionString = "dsn=airlines_data;uid=" & user_name & ";pwd=" & pass_word
& " "
db.Open
rs.Open "select * from flight_data", db, adOpenDynamic, adLockOptimistic, adCmdText
rs1.Open "select * from flight_data", db, adOpenDynamic, adLockOptimistic,
adCmdText
rs2.Open "select * from cust_data", db, adOpenDynamic, adLockOptimistic, adCmdText
Command1.Picture = ImageList1.ListImages(7).Picture
Image1.Picture = ImageList1.ListImages(1).Picture
Image2.Picture = ImageList1.ListImages(3).Picture
Image3.Picture = ImageList1.ListImages(4).Picture
Image4.Picture = ImageList1.ListImages(6).Picture
Image5.Picture = ImageList1.ListImages(8).Picture
Image6.Picture = ImageList1.ListImages(8).Picture
Image7.Picture = ImageList1.ListImages(9).Picture
search.Picture = ImageList1.ListImages(10).Picture
Image8.Picture = ImageList1.ListImages(11).Picture
Call treeview_data
Exit Sub
err_h:
MsgBox "error occureed while connecting to the database", vbInformation
Unload Form10
End Sub
Public Function treeview_data()
On Error GoTo err_h:
Dim i As Integer, j As Integer, k As Integer, l As Integer, m As Integer
Dim Data As Node
Set Data = TreeView1.Nodes.add
Data.Image = 2
```

```
TreeView1.Nodes(1).Text  = "Flight"
i = 0
Do While  (Not rs.EOF)
i = i + 1
rs.MoveNext
Loop
rs.MoveFirst
Dim node1() As Node
ReDim node1(i) As Node
Dim temparray()  As String
ReDim temparray(i)  As String
m = 0
For j = 1 To i
   For k = 0 To m
     If (temparray(k)  = rs.Fields(3))  Then
     l = 1
     End If
    Next k
     If l = 0 Then
     Set node1(j)  = TreeView1.Nodes.add(Data,  tvwChild,  , rs.Fields(3))
     node1(j).Image  = 1
     temparray(m)  = rs.Fields(3)
     m = m + 1
     End If
   rs.MoveNext
   l = 0
Next j
TreeView1.Nodes.Item(2).Sorted  = True
rs.MoveFirst
Dim children1  As Integer
children1  = TreeView1.Nodes(1).Children
Dim node2() As Node
ReDim node2(children1)  As Node
Dim node3 As Node
Dim node4 As Node
i = 0
Do While  (Not rs.EOF)
i = i + 1
rs.MoveNext
Loop
Dim z As Integer
rs.MoveFirst
For z = 1 To children1
rs.MoveFirst
ReDim temparray(i)  As String
 m = 0
```

```
 l = 0
 For j = 1 To i
   For k = 0 To m
    If (temparray(k)  = (rs.Fields(1)  + " " + "To"  + " " + rs.Fields(2)))  Then
    l = 1
    End If
   Next k
   If l = 0 And (rs.Fields(3)  = TreeView1.Nodes.Item(z  + 1)) Then
    Set node2(z)  = TreeView1.Nodes.add(TreeView1.Nodes.Item(z  + 1), tvwChild,  ,
rs.Fields(1)  + " " + "To"  + " " + rs.Fields(2))
     node2(z).Image  = 3
    temparray(m)  = rs.Fields(1)  + " " + "To"  + " " + rs.Fields(2)
    m = m + 1
       rs1.MoveFirst
       Do While  (Not rs1.EOF)
       If 0 <> InStr(Left(node2(z).Text,  (InStr(node2(z).Text,  "To")  - 2)), rs1.Fields(1))
And 0 <> InStr(Right(node2(z).Text,  (Len(node2(z).Text)  - InStr(node2(z).Text,  "To")  -
2)), rs1.Fields(2))  Then
         Set node3 = TreeView1.Nodes.add(node2(z),  tvwChild,  , rs1.Fields(10))
          node3.Image  = 4
           If rs2.BOF = False Then
         rs2.MoveFirst
         End If
            Do While  (Not rs2.EOF)
             If (rs1.Fields(10)  = rs2.Fields(14))  Then
              Set node4 = TreeView1.Nodes.add(node3,  tvwChild,  , rs2.Fields(13))
               node4.Image  = 6
             End If
             rs2.MoveNext
            Loop
      End If
       rs1.MoveNext
     Loop
   End If
   l = 0
   rs.MoveNext
Next j
Next z
'for expend the tree
TreeView1.Nodes(1).Expanded  = True
Exit  Function
err_h:
MsgBox " error while  connecting  to database"
End Function

Private  Sub Image5_Click()
```

```
Frame1.Visible  = False
Frame2.Visible  = False
End Sub

Private Sub Image6_Click()
Frame1.Visible  = False
Frame2.Visible  = False
End Sub

Private Sub search_Click()
On Error GoTo err_h:
TreeView1.Refresh
TreeView1.Nodes(1).Expanded  = False
Dim v As Integer, ul As Integer
v = 0
ul = 0
If Option1.Value  = True Then
  For v = 1 To TreeView1.Nodes.Count
  If (Text1.Text  = TreeView1.Nodes.Item(v)) Then
  TreeView1.Nodes(1).Expanded  = True
  TreeView1.Nodes.Item(v).FirstSibling.Parent.Expanded  = True
  TreeView1.Nodes.Item(v).FirstSibling.Parent.FirstSibling.Parent.Expanded  = True
  TreeView1.Nodes.Item(v).BackColor  = RGB(9, 115, 103)
  TreeView1.Nodes.Item(v).ForeColor  = RGB(250, 250, 250)
  ul = 1
  Exit Sub
  End If
  Next v
If (ul = 0) Then Image7.Visible  = True
End If
If Option2.Value  = True Then
 For v = 1 To TreeView1.Nodes.Count
  If (Text1.Text  = TreeView1.Nodes.Item(v)) Then
  TreeView1.Nodes(1).Expanded  = True
  TreeView1.Nodes.Item(v).FirstSibling.Parent.Expanded  = True
  TreeView1.Nodes.Item(v).FirstSibling.Parent.FirstSibling.Parent.Expanded  = True
TreeView1.Nodes.Item(v).FirstSibling.Parent.FirstSibling.Parent.FirstSibling.Parent.Exp
anded = True
  TreeView1.Nodes.Item(v).BackColor  = RGB(9, 115, 103)
  TreeView1.Nodes.Item(v).ForeColor  = RGB(250, 250, 250)
  ul = 1
  Exit Sub
  End If
  Next v
If (ul = 0) Then Image7.Visible  = True
End If
```

```
Exit Sub
err_h:
MsgBox "error occureed while connecting to the database", vbInformation
End Sub
Private Sub Text1_Click()
Image7.Visible = False
End Sub

Private Sub TreeView1_NodeClick(ByVal Node As MSComctlLib.Node)
On Error Resume Next
If Node.Image = 4 Then
Frame1.Visible = True
Frame2.Visible = False
rs.MoveFirst
Do While (Not rs.EOF)
If rs.Fields(10) = Node.Text Then
 Label2.Caption = rs.Fields(0)
 Label4.Caption = rs.Fields(1)
 Label6.Caption = rs.Fields(2)
 Label8.Caption = rs.Fields(3)
 Label10.Caption = rs.Fields(4)
 Label11.Caption = rs.Fields(5)
 Label13.Caption = rs.Fields(6)
 Label15.Caption = rs.Fields(7)
 Label17.Caption = rs.Fields(8)
 Label19.Caption = rs.Fields(9)
 Label21.Caption = rs.Fields(10)
 Exit Sub
 End If
rs.MoveNext
Loop
End If
Label26.Caption = ""
Label27.Caption = ""
Label28.Caption = ""
Label34.Caption = ""
If Node.Image = 6 Then
Frame1.Visible = False
Frame2.Visible = True
rs2.MoveFirst
Do While (Not rs2.EOF)
If rs2.Fields(13) = Node.Text Then
Label22.Caption = rs2.Fields(0)
Label23.Caption = rs2.Fields(1)
Label24.Caption = rs2.Fields(2)
Label26.Caption = rs2.Fields(3)
```

```
Label27.Caption  = rs2.Fields(4)
Label28.Caption  = rs2.Fields(5)
Label30.Caption  = rs2.Fields(6)
Label32.Caption  = rs2.Fields(7)
Label34.Caption  = rs2.Fields(8)
Label36.Caption  = rs2.Fields(9)
Label38.Caption  = rs2.Fields(10)
Label40.Caption  = rs2.Fields(11)
Label42.Caption  = rs2.Fields(12)
Label44.Caption  = rs2.Fields(13)
Label46.Caption  = rs2.Fields(14)
 Exit  Sub
 End  If
rs2.MoveNext
Loop
End  If
End  Sub
```

3.2 Testing and Debugging

Software Testing is an empirical investigation conducted to provide stakeholders with information about the quality of the product or service under test, with respect to the context in which it is intended to operate. Software Testing also provides an objective, independent view of the software to allow the business to appreciate and understand the risks at implementation of the software. Test techniques include, but are not limited to, the process of executing a program or application with the intent of finding software bugs.

Software Testing can also be stated as the process of validating and verifying that a software program/application/product:

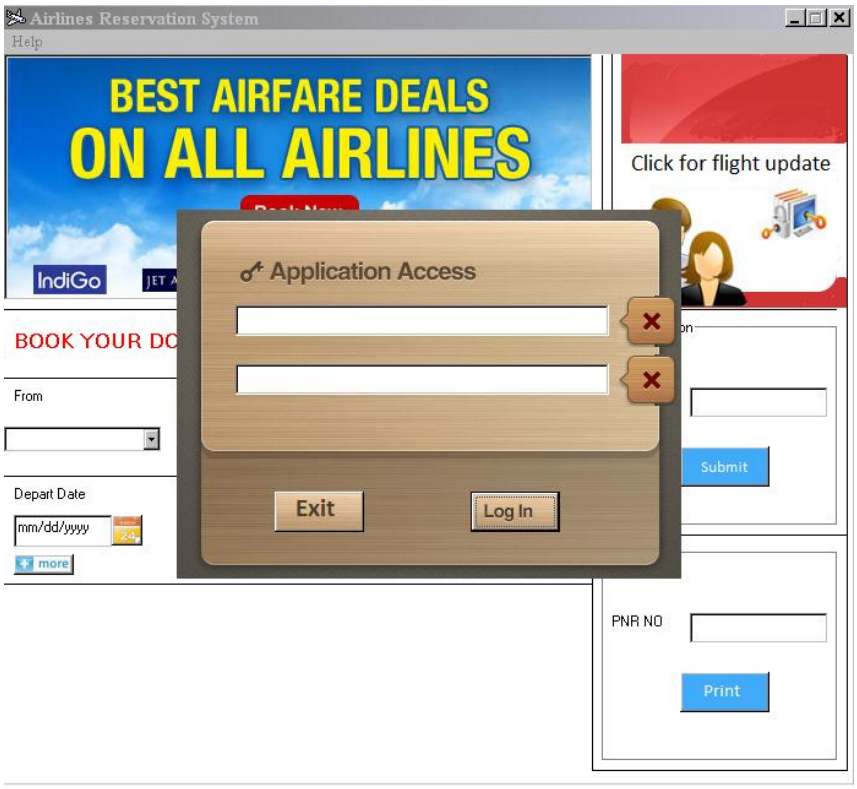
- meets the business and technical requirements that guided its design and development;
- works as expected; and
- can be implemented with the same characteristics

Software Testing, depending on the testing method employed can be implemented at any time in the development process. However, most of the test effort occurs after the requirements have been defined and the coding process has been completed. Different software development models will focus the test effort at different points in the development process. In a more traditional model, most of the test effort occurs after the requirements have been defined and the coding process has been completed. Newer development models, such as Agile or XP, often employ test driven development and place an increased portion of the testing up front in the development process, in the hands of the developer.

3.2.1 Login Form

Test Case: Value of the Login id and Password should be correct and cannot be empty

Output :



Permissible Input: The values entered should be correct, valid and no fields must be left blank

3.2.2 Main Form

Test Case: Value of the source city, destination city, cabin, departure date should be correct and cannot be empty

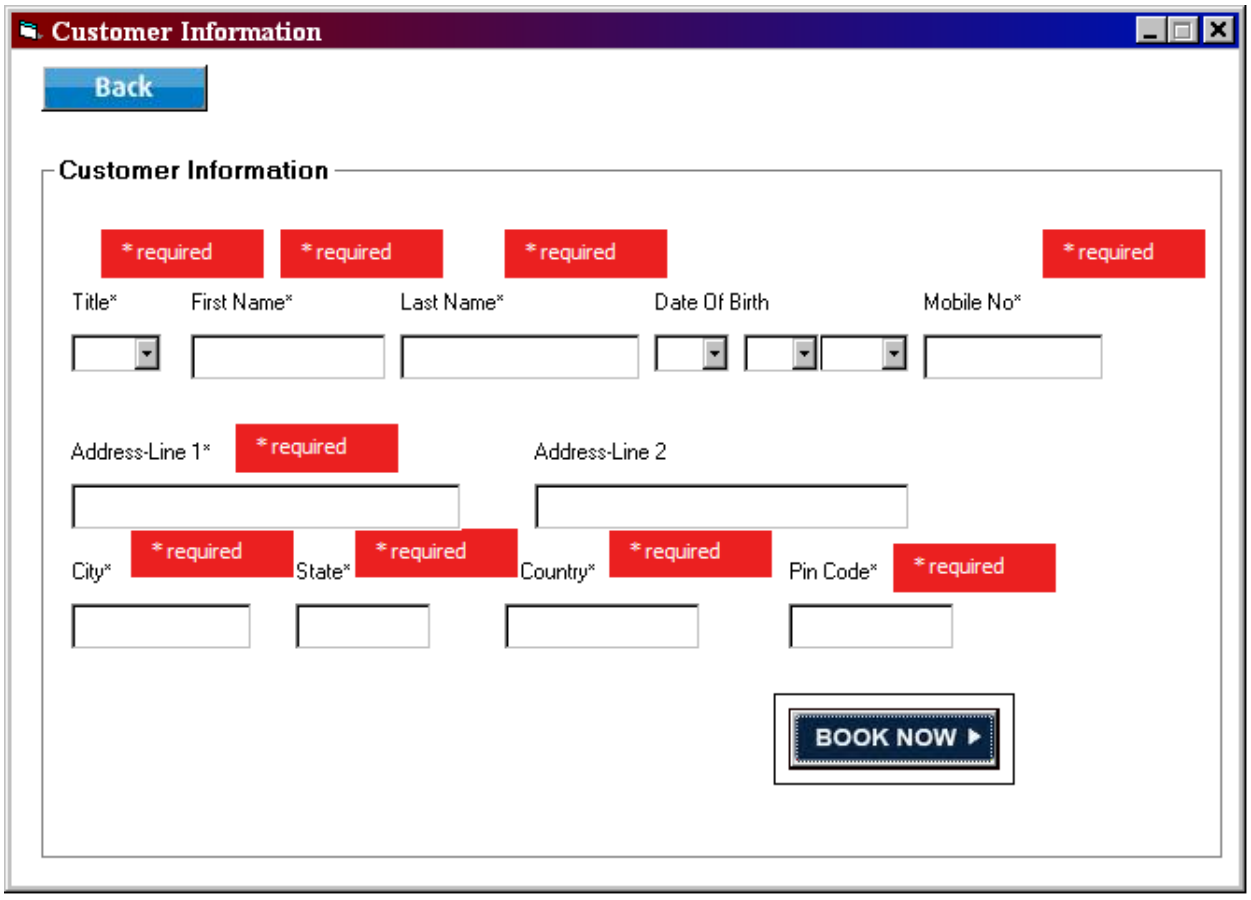
Output :

Permissible Input: The values entered should be correct, valid and no fields must be Left blank

3.2.3 Customer Information Form

Test Case: All the values in this form should be correct and cannot be empty

Output :



Permissible Input: The values entered should be correct, valid and no fields must be left blank

Chapter-4: Scope of Improvement, Summary and Conclusions

4.1 Objectives of the project

The various objectives of AIRLINE RESERVATION SYSTEM that have been achieved through this project are as follows:

- Simple to use, easy to understand
- Allows for easy search for the flight, Cancellation of ticket.
- Customer information is kept safe.
- Less time, efforts and resource consuming.
- To handle the most complex and challenging tasks.
- Database Treeview make easier search
- Reprint Ticket Option at the time of loss/theft

4.2 Scope of Improvement

There is always room for improvements, and the system we created can also be improved to some extent. There are some additional functionalities or tasks that can be added to our system as an improvement. These functionalities can be:

- System can be altered to support the searching of indirect flights
- The system can be further enhanced with a seat reservation available. It is to fulfill passengers request to sit where they prefer. They are allowed to choose their seat whether near to window's seat or in the middle.
- System can be altered to support booking ticket for more the one passenger
- System can be altered to support concession benefits or discounts allowed to customers.

4.3 Limitations

- One of the limitations of the software is the system only supports for domestic flight. Domestic flight means it only handles local flights (within India). This system is unable to support huge international flights for all countries.
- The system does not provide any feature to for passenger to change their reservation directly. Sometimes passengers might be frustrated to cancel their reservation and make a new one. Therefore, passengers are also advised to confirm their reservations before making any flights reservation

Appendix

References

i. www.makemytrip.com

ii. http://en.wikipedia.org/wiki/Airline_reservations_system

iii. "Computerized Reservation System". Retrieved on January 14, 2010 from http://en.wikipedia.org/wiki/Computer_reservations_system.htm

iv. www.yatra.com

v. C. Winston, S. Morrison(1995): 'The Evolution of the Airline Industry', BrookingsInstitution Press, South Dakota, Cf. p. 61-62, Computer Reservation Systems

Bibliography

i. Black book visual basic 6

ii. Software Engineering By KK Aggarwal