

**VISVESVARAYA TECHNOLOGICAL UNIVERSITY  
BELAGAVI, KARNATAKA**



*A Mini Project Report on*

**“AIRLINE RESERVATION SYSTEM”**

*Submitted in the partial fulfillment for the requirements for the DBMS Lab with Mini Project*

*in*

**INFORMATION SCIENCE AND ENGINEERING**

*By*

**Mr. Anupchandra Rao MC  
Mr. Gadilinga K**

**USN: 1BY15IS009  
USN: 1BY15IS020**

*Under the guidance of*

**Mrs. Veena N**  
Assistant Professor  
Department of ISE, BMSIT&M.



**DEPARTMENT OF INFORMATION SCIENCE AND ENGINEERING  
BMS INSTITUTE OF TECHNOLOGY & MANAGEMNT  
YELAHANKA, BENGALURU-560064**

**2017-2018**

**VISVESVARAYA TECHNOLOGICAL UNIVERSITY  
BELAGAVI, KARNATAKA**

**BMS INSTITUTE OF TECHNOLOGY & MANAGEMENT  
YELAHANKA, BENGALURU-560064**

**DEPARTMENT OF INFORMATION SCIENCE & ENGINEERING**



**CERTIFICATE**

This is to certify that the Project work entitled “**Airline Reservation System**” is a bonafide work carried out by **Mr. Anupchandra Rao MC (1BY15IS009)**, **Mr. Gadilinga K (1BY15IS020)** in partial fulfillment of DBMS Lab with Mini Project for the award of **Bachelor of Engineering Degree in Information Science and Engineering** of the Visvesvaraya Technological University, Belagavi during the year 2017-18. It is certified that all corrections/suggestions indicated for Internal Assessment have been incorporated in this report. The project report has been approved as it satisfies the academic requirements in respect of Mini Project work for the B.E Degree.

**Signature of the Guide**

Mrs. Veena N  
Assistant Professor  
Department of ISE

**Signature of the HOD**

Dr. Manjunath T.N  
Professor and Head  
Department of ISE

**Signature of the Principal**

Dr. Mohan Babu G.N  
Principal  
BMSIT&M

**EXTERNAL EXAMINERS**

Name of the Examiners

- 1.
- 2.

Signature with Date

# ACKNOWLEDGEMENT

We are happy to present this project after completing it successfully. This project would not have been possible without the guidance, assistance and suggestions of many individuals. We would like to express our deep sense of gratitude and indebtedness to each and every one who has helped us make this project a success.

We heartily thank our **Principal, Dr. Mohan Babu G.N, BMS Institute of Technology & Management** for his constant encouragement and inspiration in taking up this project.

We heartily thank our **Head of Department Dr. Manjunath T. N, Dept. of Information Science and Engineering, BMS Institute of Technology & Management** for his constant encouragement and inspiration in taking up this project.

We heartily thank our **Project Co-Ordinator Dr. Pushpa S.K, Dept. of Information Science and Engineering, BMS Institute of Technology & Management** for her constant encouragement and inspiration in taking up this project.

We gracefully thank our Project guide, **Mrs. Veena N, Asst. Professor, Dept. of Information Science and Engineering**, for her encouragement and advice throughout the course of the Project work.

Special thanks to all the staff members of Information Science Department for their help and kind co-operation.

We also thank our parents and friends for their unconditional love and encouragement and support given to us in order to finish this precious work.

Last but not the least we would like to thank God for giving us the strength and motivation through the course of this Project.

By,

Anupchandra Rao MC

Gadilinga K

# **ABSTRACT**

Airline reservation System is a computerized system used to store and retrieve information and conduct transactions related to air travel. The project is aimed at exposing the relevance and importance of Airline Reservation Systems. It is projected towards enhancing the relationship between customers and airline agencies through the use of ARSs, and thereby making it convenient for the customers to book the flights as when they require such that they can utilize this software to make reservations.

This software has two parts. First is user part and the administrator part. User part is used as a front end and administrator is the back end. Administrator is used by airline authority. It will allow the customers to access database and allow new customers to sign up for online access. The system allows the airline passenger to search for flights that are available based on the cities, and arrival dates. The system displays all the flight's details such as flight no, name,

After search the system display list of available flights and allows customer to choose a particular flight. Then the system checks for the availability of seats on the flight. If the seats are available then the system allows the passenger to book a seat. Otherwise it asks the user to choose another.

To book a flight the system asks the customer to enter his details such as name, address, age, gender and contact number. It then books the flight and update the airline database and user database. The system also allows the customer to cancel his/her reservation, if any problem occurs the system allows the user to modify the ticket details.

The main purpose of this software is to reduce the manual errors involved in the airline reservation process and make it convenient for the customers to book the flights as when they require such that they can utilize this software to make reservations, modify reservations or cancel a particular reservation.

# TABLE OF CONTENTS

• Chapter 1 Introduction	1
○ 1.1 Outline	1
○ 1.2 Motivation and Scope	1
○ 1.3 Problem Statement	2
○ 1.4 Limitations	2
• Chapter 2 Requirements Specification	3
○ 2.1 Functional Requirements	3
○ 2.2 Non-Functional Requirements	4
○ 2.3 Domain Constraints	5
• Chapter 3 Requirements and System Analysis	6
○ 3.1 Overall Process of the Project	6
○ 3.2 Components/Subsystem Design	7
• Chapter 4 System Design	10
○ 4.1 Entity Relationship Diagram	10
○ 4.2 Schema Diagram	10
• Chapter 5 Implementation	11
○ 5.1 Description of DBMS	11
○ 5.2 Description of Integrated Development Environment	11
• Chapter 6 Testing	13
○ 6.1 Component Tests	13
○ 6.2 System Tests	16
• Chapter 7 Interpretation of Results	17
Conclusion	22
References	23

## **List of Figures/Tables**

• 3.1.1: Overall System Design	6
• 3.2.1: Reservation Process Flow Diagram	7
• 3.2.2: Cancellation Process Flow Diagram	7
• 3.2.3: Reschedule Process Flow Diagram	8
• 3.2.4: Display Process Flow Diagram	8
• 4.1.1: ER Diagram	10
• 4.2.1: Schema Diagram	10
• 6.1.1 User Login Module Tests	13
• 6.1.2 Ticket Reservation Module Tests	13
• 6.1.3 Ticket Cancellation Module Tests	14
• 6.1.4 Reschedule Module Tests	14
• 6.1.5 Confirmed Passengers Module Tests	14
• 6.1.6 Waitlisted Passengers Module Tests	15
• 6.1.7 Ticket Status Module Tests	15
• 6.1.8 Flight Query Module Tests	15
• 6.2.1 System/Integration Tests	16
• 7.1: Login Window	17
• 7.2: Main Interface	17
• 7.3: Reservation Window	18
• 7.4: Cancellation Window	18
• 7.5: Reschedule Window	19
• 7.6: Confirmed Tickets Window	19
• 7.7: Waitlisted Passengers	20
• 7.8: Ticket Status Window	20
• 7.9: Flight Query Module	21

## **Chapter 1:**

### **Introduction**

#### **1.1 Outline**

The Airline Reservation System (ARS) is a software application to assist an airline with transactions related to making ticket reservations, which includes querying, rescheduling, reserving and cancelling tickets.

Minimize repetitive work done by the system administrator and reservation clerks.

Maintain consistency among different access modes. The users should be basically taken through the same steps by the system as they go through in conventional desk-reservation systems.

Maintain customer information in case of emergency, e.g. flight cancellation due to inclement weather. The profile can also be used by the airline company to track user preferences and travel patterns to serve them better, plan routes, for better marketing and efficient scheduling of flights.

Minimize the number of vacant seats on a flight and maximize flight capacity utilization.

#### **1.2 Motivation and Scope**

A survey conducted by airline companies shows that users of an existing reservation system would respond favorably to an ARS that satisfied or helped them satisfy the following objectives:

Reduce effort and frustration for travellers in scheduling a trip, especially by reducing the search effort for the flight they need to take.

Show all possible combinations and itineraries available for a pair of origin-destination cities.

Reduce redundancy in the information required from the customers in order for them to buy tickets, create user accounts etc.

Check the validity of input data and give a feedback to the user in case of errors or inconsistency. And Protect customers' privacy concerns.

Make it easy for travellers to check the ticket status or make changes to their trip.

### **1.3 Problem Statement**

Design and develop User Interface for Flights reservation system that facilitates the passengers to enquire about the Flights available on the basis of source and destination, booking and cancellations of tickets, enquiry about the status of booked tickets etc.

### **1.4 Limitations**

ARS requires the Java Virtual Machine to be installed without which it will not be able to function.

ARS does not follow Google's material design guidelines and uses fairly outdated Swings Architecture.

ARS does not provide information about travel packages and discounts available during festive season.

ARS cannot dynamically update to pricing and needs to be done by the Database Administrator.



## **Chapter 2**

### **Requirements Specification**

#### **2.1 Functional Requirements**

##### **Request for Login**

The system shall require a user to register, in order to carry out any transactions with it except for checking the availability of tickets. It will ask the user for the following information at the least – a user id, a password if correct the system allows the user to carry out flight transactions.

##### **Checking Availability**

Once the user is logged in the system offers the option of rescheduling the ticket and allows the user to enter the travel date once submitted the ticket is rescheduled to the new date.

##### **Making Reservations/Blocking/Confirmation**

Having taken the input from the user , the system shall now proceed to update the reservation database DB-reservation. It will decrement the number of available seats on the particular flight for the particular class by the number of travelers being represented by the user.

It simultaneously generates a confirmation number and displays it to the user for him to note down. The ticket has been reserved.

##### **Reschedule Ticket**

The system shall now ask the user to select new dates from the calendar-menu. In case, there are no available tickets for the dates entered, it displays a suitable message informing him that rescheduling to that date is not possible.

In case there are tickets available, the system proceeds to update the database.

The system accesses DB-reservation and decrements the number of available seats on the flight(s) by the number of members in the user's travel party.

##### **Cancellation**

The system shall also give the user an option to cancel a confirmed ticket or a blocked ticket.

The latter case is simpler and will be dealt with first – the system shall first log on the user and request the blocking number. Then it accesses DB-reservation and updates it by incrementing the number of available seats by the number of people in the user's travel party.

In the former case, i.e., for a confirmed ticket, it asks for the confirmation number and accesses DB-reservation and presents the details of the trip.

### **View Ticket Status**

The system shall allow a user to view all information about his trip. After logging him on, it asks for his blocking number or his confirmation number. It accesses DB-reservation and retrieves the details of the trip and presents them to the user in a convenient format.

### **Query Flight Details**

The system shall allow any user to access the details about the arrival and departure times of a flight by requesting the user to input the flight number and date. The system accesses DB-schedule and presents the time of arrival and departure.

## **2.2 Non-Functional Requirements**

### **Performance**

Response time of the Airline Reservation System should be less than 2 second most of the time. Response time refers to the waiting time while the system accesses, queries and retrieves the information from the databases (DB-user, DB-schedule etc) (A local copy of flight schedule database is maintained as DB-schedule to reduce this access time)

ARS shall show no visible deterioration in response time as the number of users or flight schedule data increases

### **Reliability**

ARS shall be available 24 hours a day, 7 days a week

ARS shall always provide real time information about flight availability information.

ARS shall be robust enough to have a high degree of fault tolerance. For example, if the user enters a negative number of passengers or a value too large, the system should not crash and shall identify the invalid input and produce a suitable error message.

ARS shall be able to recover from hardware failures, power failures and other natural catastrophes and rollback the databases to their most recent valid state.

### **Usability**

ARS shall provide a easy-to-use graphical interface similar to other existing reservation system so that the users do not have to learn a new style of interaction.

Any notification or error messages generated by ARS shall be clear, succinct, polite and free of jargon.

### **Integrity**

Only system adminster has the right to change system parameters, such as pricing policy etc. The system should be secure and must use encryption to protect the databases.

Users need to be authenticated before having access to any personal data.

### **Interoperability**

ARS shall minimize the effort required to couple it to another system, such as flight schedule database system.

## **2.3 Domain Constraints**

Regulatory policies: It is a mandatory that no text book must be left empty or contains insufficient data.

Hardware limitations: There must be a 64 MB on board memory

Control functions: The software must be very user-friendly and display appropriate error messages.

Interfaces to other applications: Not applicable.

Parallel operations: It must support many users simultaneously.

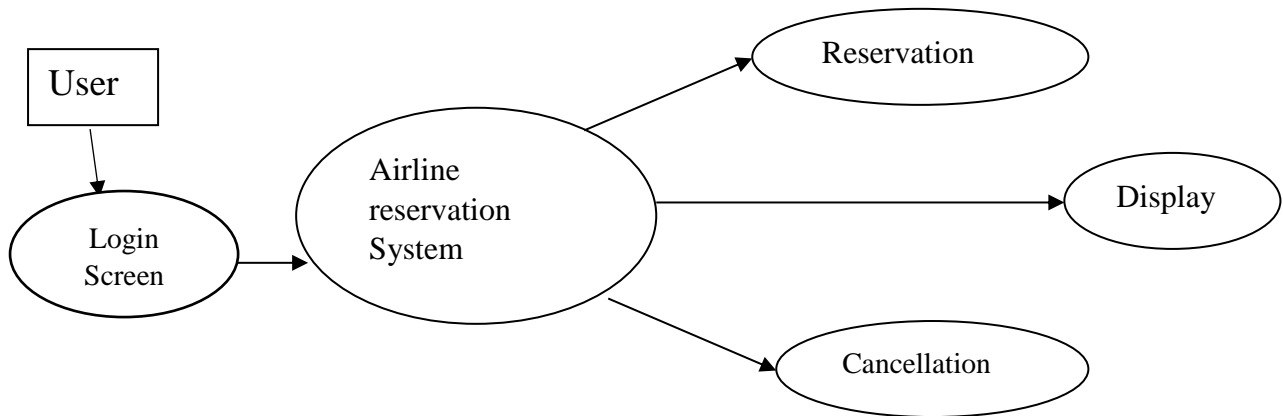
Safety/security considerations: The application must be exited always normally.

Higher order language requirements: VB

## Chapter 3

### System/Requirements Analysis

#### 3.1 Overall System Description



**Fig. 3.1.1: Overall System Design**

The overall description of the system is as follows:

The user is first presented with a login screen where he asked to enter his Username and Password. If the correct input is received from the user, he is taken to the Main Interface where he will be presented with an array of choices.

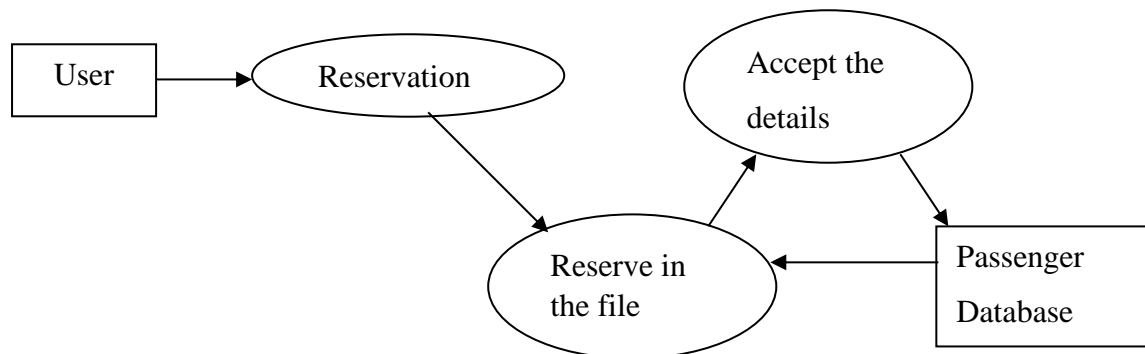
Inputting a wrong Username and Password displays a message to the user to check the details he has entered and to correct it.

Only the Database Administrator has the authority to add and remove user accounts.

In the Main Interface the user is allowed to reserve a ticket, cancel a ticket, reschedule a ticket, query information of all booked tickets, query information of all waitlisted tickets, check details of confirmed tickets, and check details of available flights.

## 3.2 Components/Subsystem Design

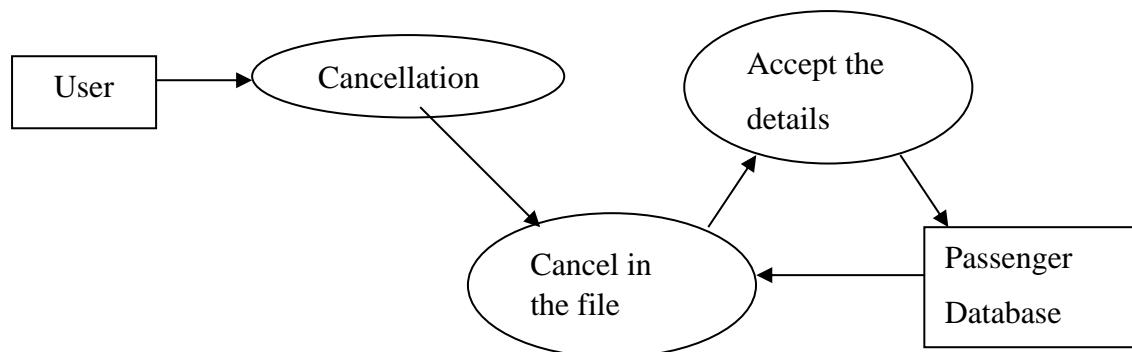
### Reservation Module



**Fig. 3.2.1: Reservation Process Flow Diagram**

In the reservation module the user is asked to first select the flight in which he/she wishes to travel. Next he is made to enter his travel date, full name, age, gender, phone number and address. If all information is in the correct format the system proceeds to check if seats are available in the flight for the corresponding date if available the system generates a ticket number which is displayed to the user and the ticket is confirmed.

### Cancellation Module

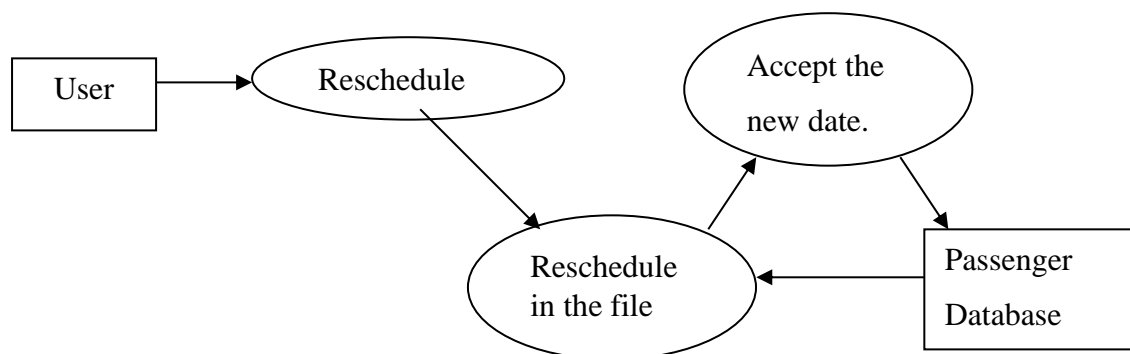


**Fig. 3.2.2: Cancellation Process Flow Diagram**

This module allows the user to cancel the ticket that he has previously booked. He is asked to enter the ticket number which was given to him while confirming the ticket using this number the ticket is cancelled.

Entering an incorrect ticket displays an alert to the user to enter correct ticket number.

### Reschedule Module

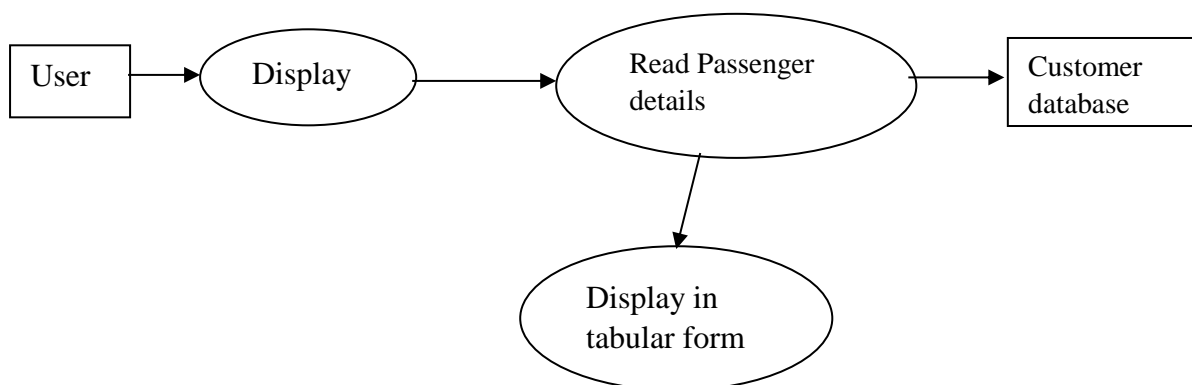


**Fig. 3.2.3: Reschedule Process Flow Diagram**

In case the changes his travel plans this module provides the functionality of rescheduling the ticket which he previously booked.

The user is asked to enter the ticket number, this number is checked by the system to see if it belongs to a valid ticket if it does then the ticket is updated to reflect the new travel date entered by the user.

### Display Module



**Fig. 3.2.4: Display Process Flow Diagram**

This module can be viewed as a collection of 4 modules:

>Confirmed Tickets Window

>Waitlisted Tickets Window

>Ticket Status Window

>Flight Query Window

Confirmed tickets window displays all the tickets that are confirmed for the specific flight and date as specified by the user.

If an invalid date is specified an error message is displayed.

Confirmed tickets window displays all the tickets that are put in the waitlist for the specific flight and date as specified by the user.

If an invalid date is specified an error message is displayed.

Here the user enters the Booking/Ticket number and is shown the corresponding ticket details in tabular format. Entering an invalid ticket number produces an error message.

The Flight is selected based on the boarding and destination point following which the date is entered. The system processes the input and returns the available seats in each class along with the ticket prices.

## Chapter 4

### System Design

#### 4.1 Entity Relationship Diagram

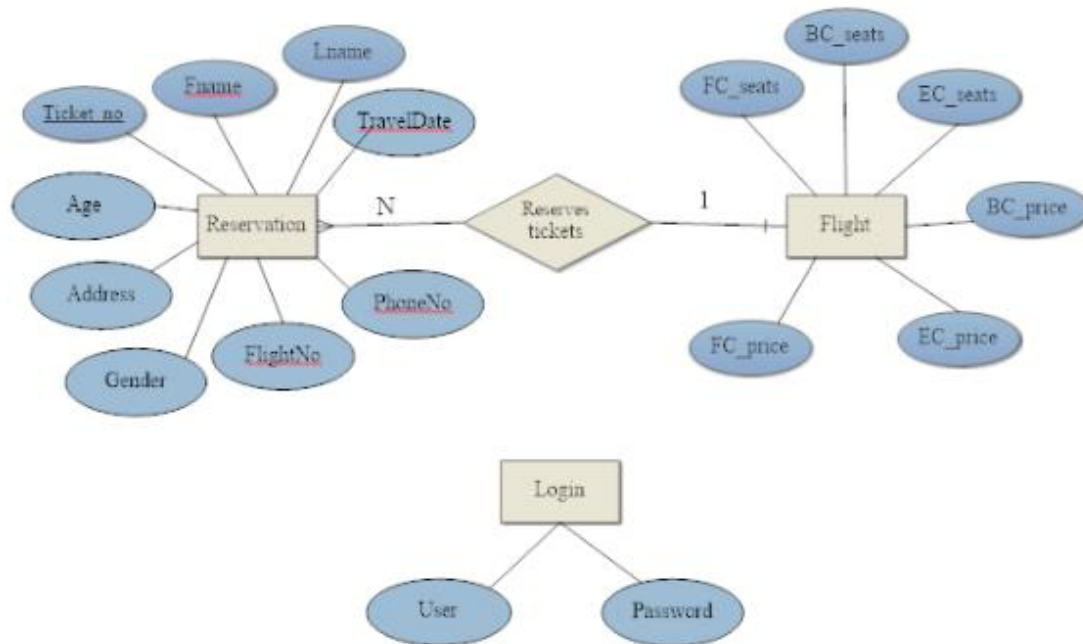


Fig. 4.1.1: ER Diagram

#### 4.2 Schema Diagram

Reservation

<u>TicketNo</u>	Fname	Lname	TravelDate	FlightNo	Age	Gender	PhoneNo
-----------------	-------	-------	------------	----------	-----	--------	---------

Flights

<u>FlightNo</u>	<u>TravelDate</u>	FC_seats	BC_seats	EC_seats	BC_price	EC_price	FC_price
-----------------	-------------------	----------	----------	----------	----------	----------	----------

Login

User	Password
------	----------

Fig. 4.2.1: Schema Diagram



## Chapter 5

### Implementation

#### 5.1 Description of Database Used

##### Oracle Database 11g Express Edition

Oracle Database (commonly referred to as Oracle RDBMS or simply as Oracle) is an object-relational database management system produced and marketed by Oracle Corporation.

The Oracle RDBMS stores data logically in the form of tablespaces and physically in the form of data files ("datafiles"). Tablespaces can contain various types of memory segments, such as Data Segments, Index Segments, etc. Segments in turn comprise one or more extents. Extents comprise groups of contiguous data blocks. Data blocks form the basic units of data storage.

A DBA can impose maximum quotas on storage per user within each tablespace.

Most Oracle database installations come with a default schema called SCOTT. After the installation process sets up sample tables, the user logs into the database with the username scott and the password tiger.

Other default schemas include:

- SYS (essential core database structures and utilities)
- SYSTEM (additional core database structures and utilities, and privileged account)

**Version used: 11g Express Edition**

#### 5.2 Description of Integrated Development Environment

##### Oracle SQL Workbench

Oracle SQL Developer is a free integrated development environment that simplifies the development and management of Oracle Database in both traditional and Cloud deployments. SQL Developer offers complete end-to-end development of your PL/SQL applications, a worksheet for running queries and scripts, a DBA console for managing the database, a reports interface, a complete data modelling solution, and a migration platform for moving your 3rd party databases to Oracle.

**Version Used: SQLcl 17.3**

## **Eclipse JEE Neon**

Eclipse is an integrated development environment (IDE) used in computer programming, and is the most widely used Java IDE.<sup>[6]</sup> It contains a base workspace and an extensible plug-in system for customizing the environment. Eclipse is written mostly in Java and its primary use is for developing Java applications, but it may also be used to develop applications in other programming languages via plug-ins.

**Version Used: Neon.3 Release (4.6.3)**

## **Eclipse Window Builder Plug-In**

WindowBuilder is composed of SWT Designer and Swing Designer and makes it very easy to create Java GUI applications without spending a lot of time writing code. Use the WYSIWYG visual designer and layout tools to create simple forms to complex windows; the Java code will be generated for you. Easily add controls using drag-and-drop, add event handlers to your controls, change various properties of controls using a property editor, internationalize your app and much more.

## Chapter 6

### Testing

#### 6.1 Component Test

##### Login Module

TEST UNIT	TEST CASE	RESULT
Login Screen	An invalid username or password is entered by the user	The system generates a message saying “invalid user id” or invalid password, whichever is the case.
Login Screen	An valid username or password is entered by the user	The system grants access to the user and takes him to the Main Interface

**Table 6.1.1 User Login Module Test**

##### Reservation Module

TEST UNIT	TEST CASE	RESULT
Reservation Window	Wrong format of information entered into the data fields of the flight booking page	The system generates an error message to the user indicating that the wrong format of data is entered.
Reservation Window	Correct data entered into the fields in the register page.	Check for availability of ticket if tickets are not available system generate message indicating no seats available.
Reservation Window	Correct data entered into the fields in Reservation Page.	Check for availability of ticket if tickets are available, the system generates a Booking/Ticket number and confirms the ticket.

**Table 6.1.2 Ticket Reservation Module Tests**

**Cancellation Module**

TEST UNIT	TEST CASE	RESULT
Cancellation Window	Enter incorrect ticket number	System Displays a message to enter correct ticket number.
Cancellation Window	Enter correct ticket number	Cancel the ticket having corresponding Ticket Number.

**Table 6.1.3 Ticket Cancellation Module Tests****Rescheduling Module**

TEST UNIT	TEST CASE	RESULT
Rescheduling Window	Enters incorrect ticket number	System Displays message to enter the correct ticket number.
Rescheduling Window	Enters correct ticket number	System Opens the page to reschedule ticket.

**Table 6.1.4 Reschedule Module Tests****Confirmed Tickets Module**

TEST UNIT	TEST CASE	RESULT
Confirmed Tickets Window	Wrong format of information entered into the data fields of the confirmed tickets window.	The system generates an error message to the user indicating that the wrong format of data is entered and prompts him to re-enter the data.
Confirmed Tickets Window	Correct data entered into the fields of the confirmed tickets window.	Check for availability of ticket if tickets are not available system generate message indicating no seats available
Confirmed Tickets Window	Correct data entered into the fields in the register page	Check for availability of ticket if tickets are available it logs him onto the system and displays the page containing all confirmed tickets.

**Table 6.1.5 Confirmed Passengers Module Tests**

**Waitlisted Tickets Module**

Test unit	Test case	Result
Waitlisted Tickets Window	Enter date	If tickets are there in waiting list system will display those ticket number
Waitlisted Tickets Window	Enter date	If tickets are found in waiting list system will display no ticket found

**Table 6.1.6 Waitlisted Passengers Module Tests**
**Displaying ticket information**

TEST UNIT	TEST CASE	RESULT
Display	Enter ticket number	If ticket number is valid system will display ticket information
Display	Enter ticket number	If ticket number is in valid then system will invalid ticket number

**Table 6.1.7 Ticket Status Module Tests**
**Flight Information Module**

TEST UNIT	TEST CASE	RESULT
Flight Information Module	Enter flight information and data	Displays flight information

**Table 6.1.8 Flight Query Module Tests**

## 6.2 System Testing

TEST UNIT	TEST CASE	RESULT
Ticket Reservation	Click on Ticket Reservation button	Opens Ticket Reservation Window
Cancellation of Ticket	Click on Cancel Ticket button	Opens Ticket Cancellation Window
Rescheduling	Click on Rescheduling button	Opens Ticket Rescheduling Window
Confirmed Ticket	Click on Confirmed Tickets button	Opens Confirmed Tickets Window
Waitlisted Tickets	Click on Waitlisted Tickets button	Opens Waitlisted Tickets Window
Tickets Status	Click on Ticket Status button	Opens Ticket Status Window
Flight information	Click on Flight Enquiry button	Opens Flight Information Window

**Table 6.2.1 System/Integration Tests**

## Chapter 7

### Interpretation of Results

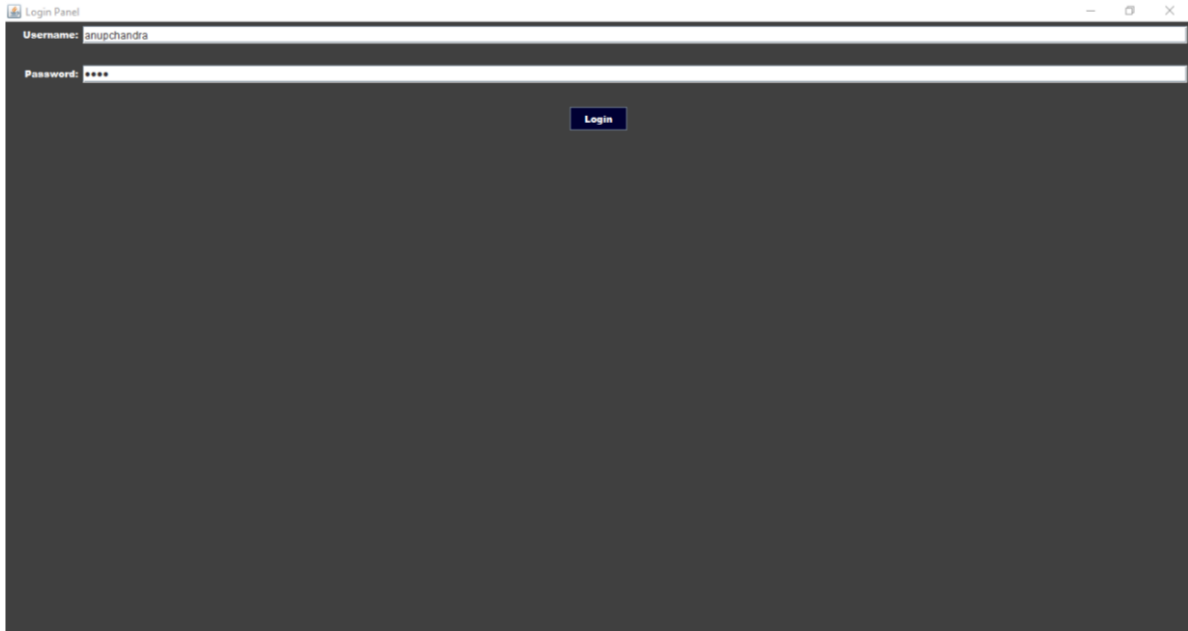


Fig. 7.1: Login Window

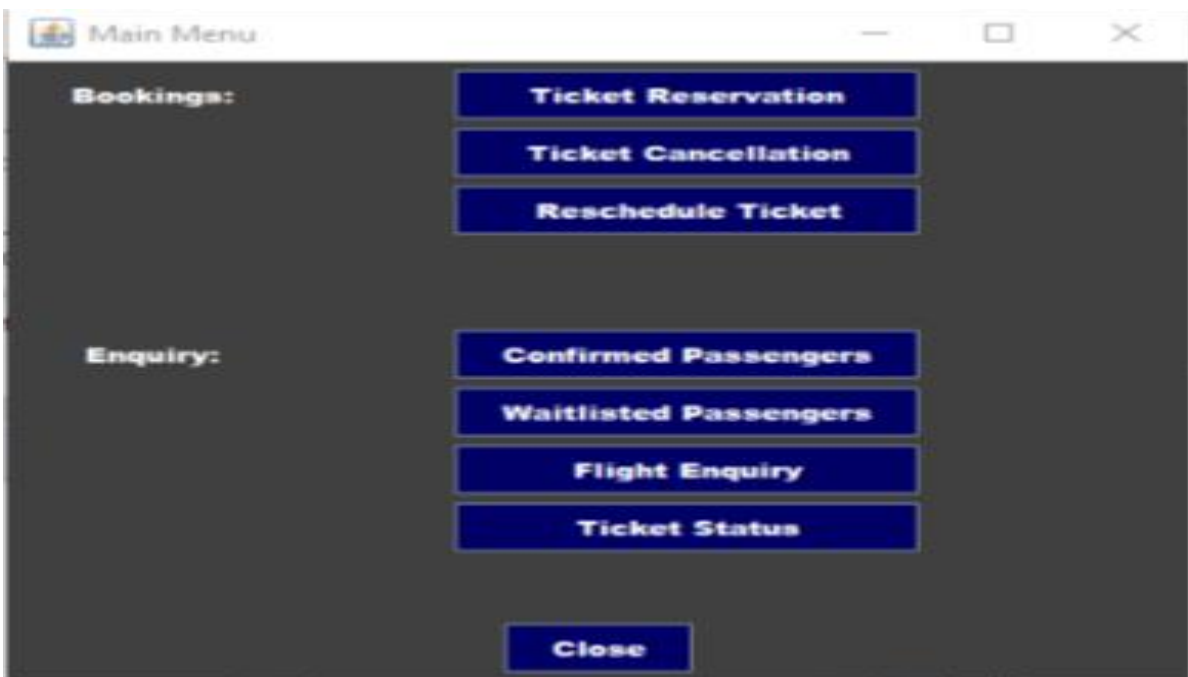
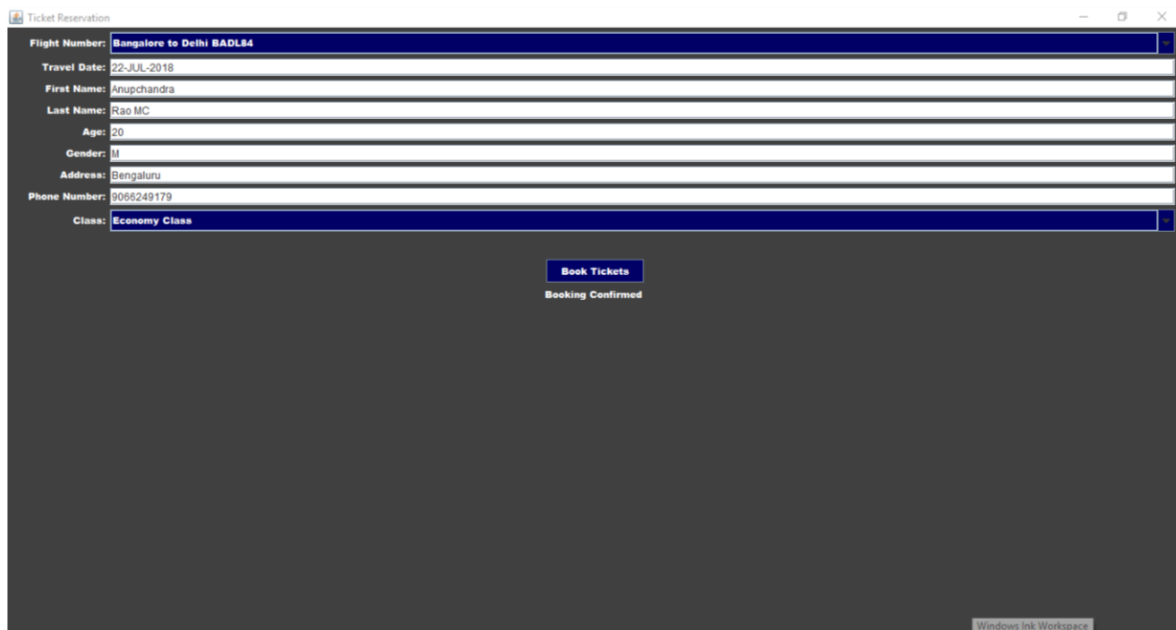


Fig. 7.2: Main Interface

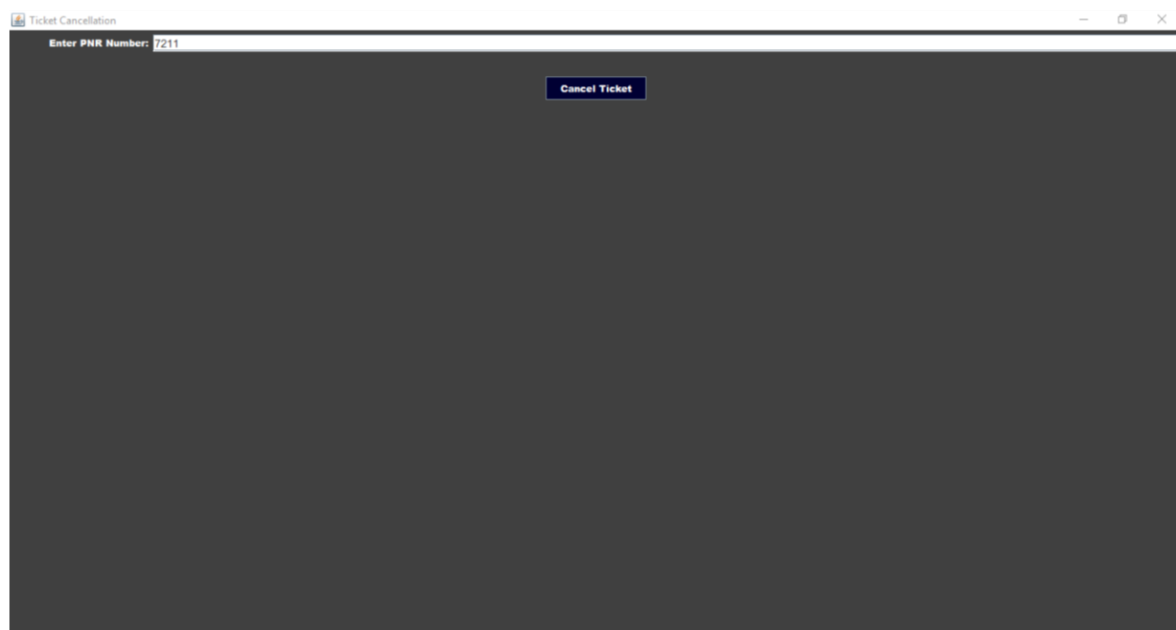


The screenshot shows a window titled "Ticket Reservation". It contains a form with the following fields and values:

Flight Number:	Bangalore to Delhi BADL84
Travel Date:	22-JUL-2018
First Name:	Anupchandra
Last Name:	Rao MC
Age:	20
Gender:	M
Address:	Bangalore
Phone Number:	9066249179
Class:	Economy Class

Below the form, there is a blue button labeled "Book Tickets". Below the button, the text "Booking Confirmed" is displayed. The window has a dark gray background and a title bar with standard window controls. A small "Windows Ink Workspace" watermark is visible in the bottom right corner.

**Fig. 7.3: Reservation Window**



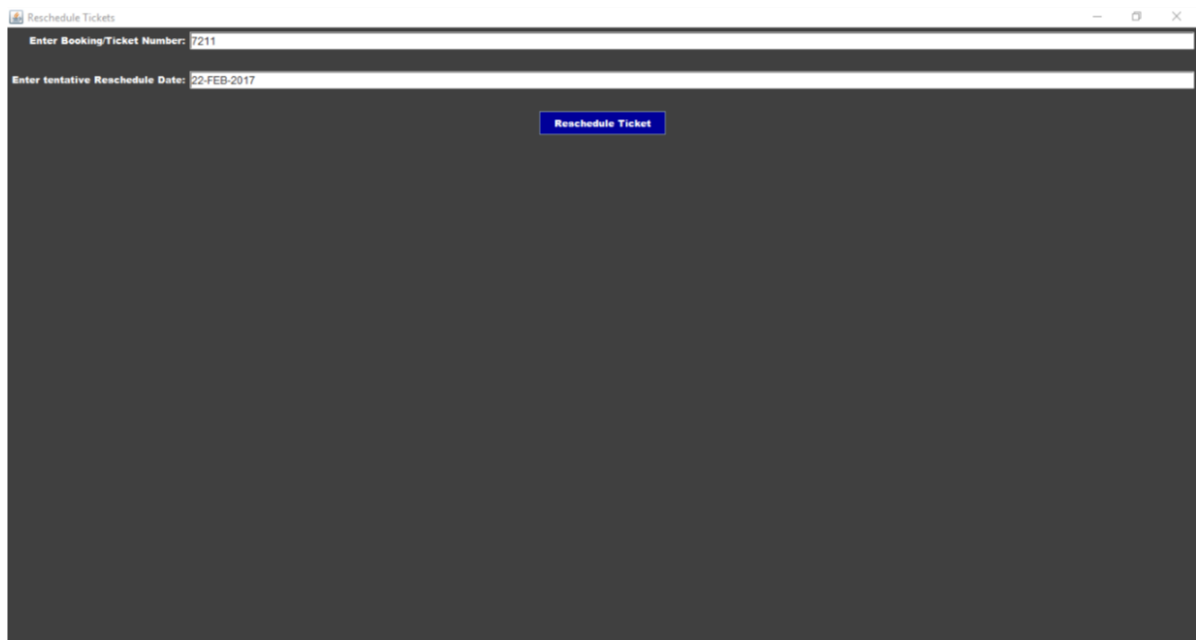
The screenshot shows a window titled "Ticket Cancellation". It contains a form with the following fields and values:

Enter PNR Number:	7211
-------------------	------

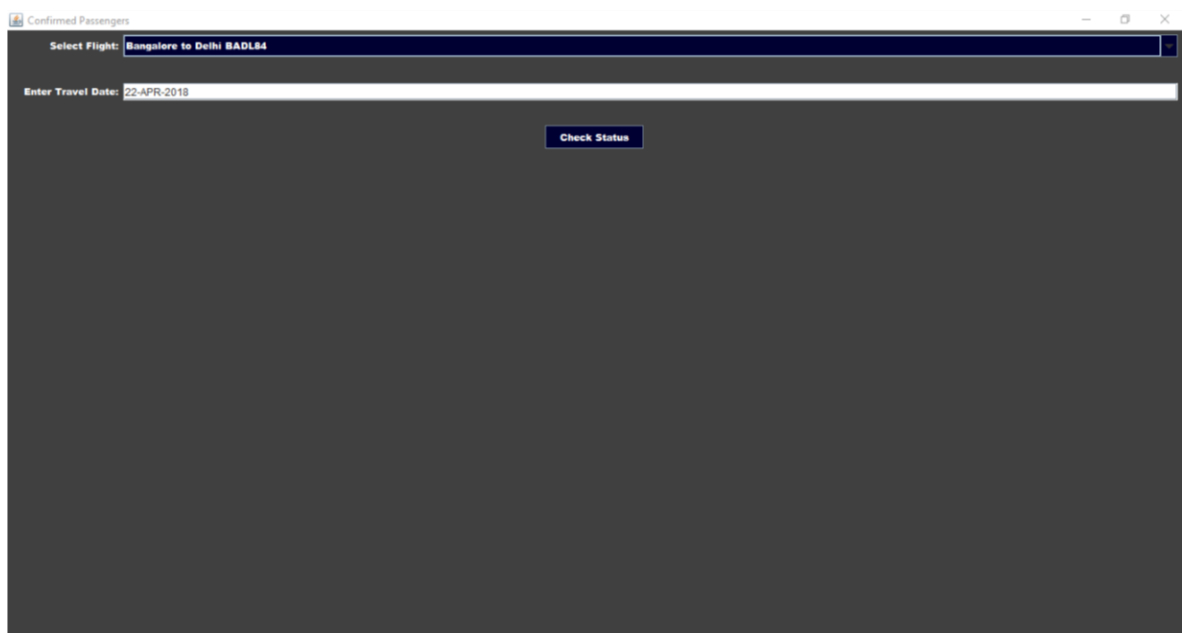
Below the form, there is a blue button labeled "Cancel Ticket". The window has a dark gray background and a title bar with standard window controls.

**Fig. 7.4: Cancellation Window**

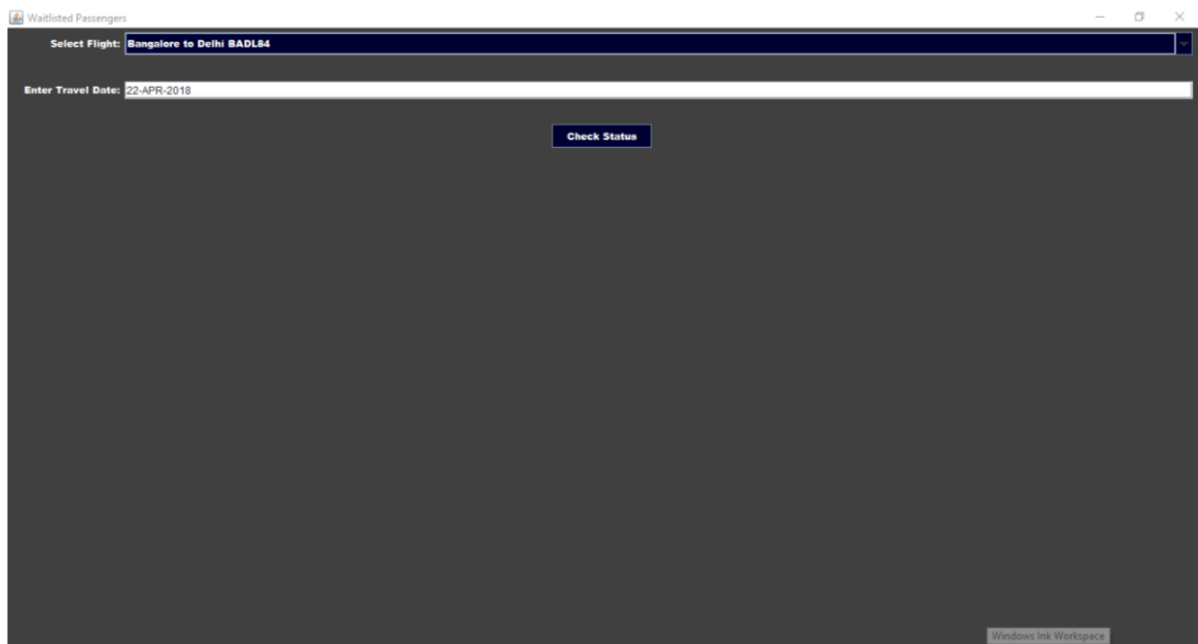




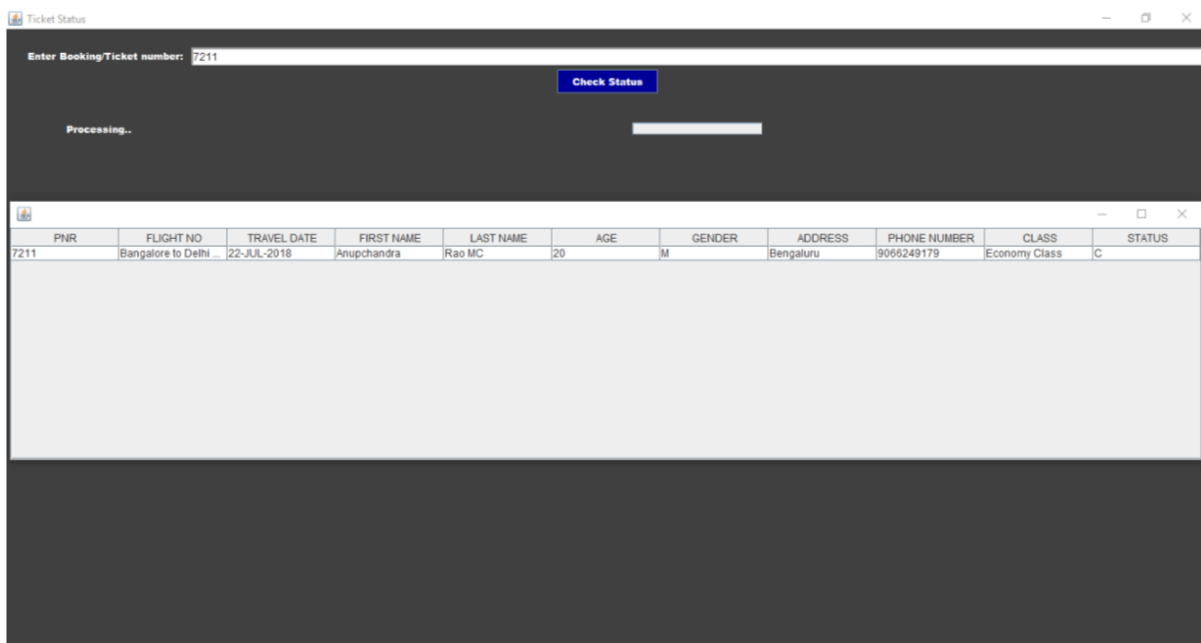
**Fig. 7.5: Reschedule Window**



**Fig. 7.6: Confirmed Tickets Window**



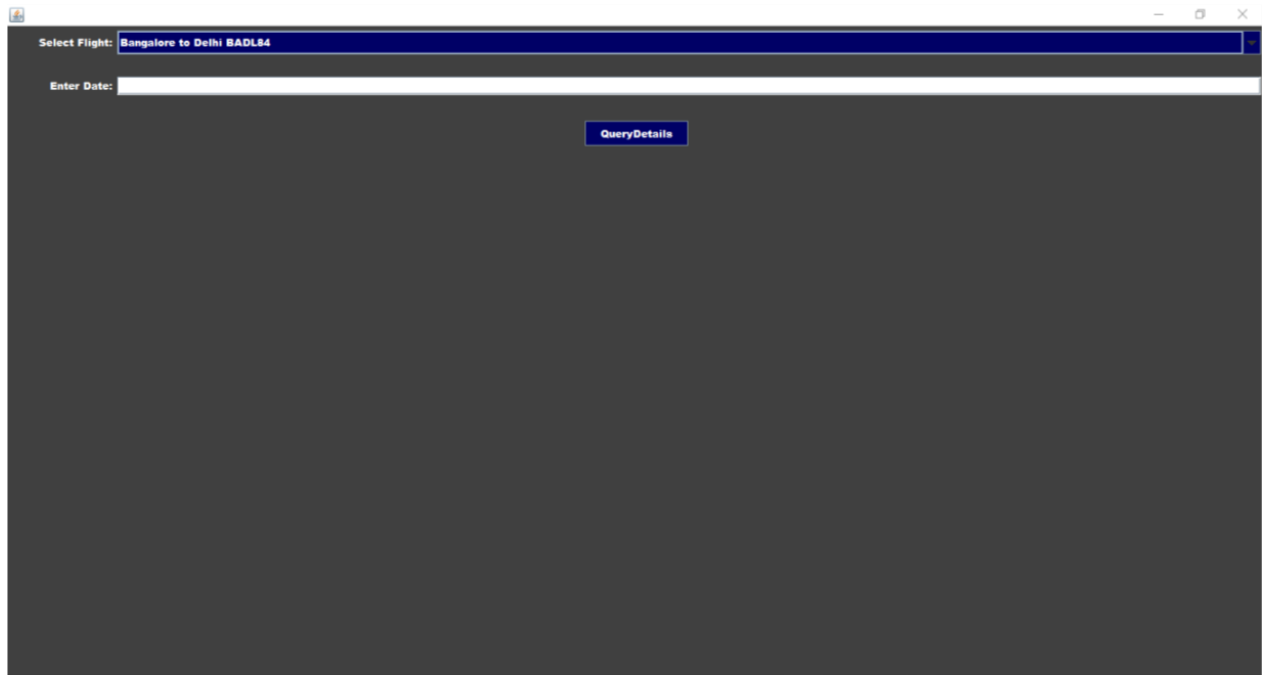
The 'Waitlisted Passengers' window features a dark-themed interface. At the top, a 'Select Flight:' dropdown menu is set to 'Bangalore to Delhi BA0184'. Below this, an 'Enter Travel Date:' field contains '22-APR-2018'. A blue 'Check Status' button is positioned in the center of the window. The bottom right corner displays 'Windows Ink Workspace'.

**Fig. 7.7: Waitlisted Passengers**

The 'Ticket Status' window has a dark background. It includes an 'Enter Booking/Ticket number:' field with '7211' and a blue 'Check Status' button. Below the button, the text 'Processing...' is shown next to a progress bar. A table with 12 columns is displayed below the progress bar. The table contains one data row for PNR 7211. The columns are: PNR, FLIGHT NO, TRAVEL DATE, FIRST NAME, LAST NAME, AGE, GENDER, ADDRESS, PHONE NUMBER, CLASS, and STATUS.

PNR	FLIGHT NO	TRAVEL DATE	FIRST NAME	LAST NAME	AGE	GENDER	ADDRESS	PHONE NUMBER	CLASS	STATUS
7211	Bangalore to Delhi ...	22-JUL-2018	Anupchandra	Rao MC	20	M	Bengaluru	9066249179	Economy Class	C

**Fig. 7.8: Ticket Status Window**



**Fig. 7.9: Flight Query Module**

## Conclusion

Once this project is completed it offers users the following functionalities:

Remotely reserve tickets for available flights on any date using the reservation module. Users can also reschedule their tickets by entering their new date of travel. Tickets can also be cancelled by entering the correct Ticket/Booking number.

The system can also display the list of Confirmed Tickets for a particular flight on a particular date. It can also display the listed of passengers who are put on a waitlist in case Tickets are not available.

The booked tickets details can be displayed by entering the valid Ticket number.

Finally, the number of seats available for a Particular Flight and date for each class can be provided along with their respective prices.

This system reduce redundancy in the information required from the customers in order for them to buy tickets, create user accounts etc.

## References

- Database systems Models, Languages, Design and Application Programming, RamezElmasri and Shamkant B. Navathe, 7th Edition, 2017
- Database management systems, Ramakrishnan, and Gehrke, 3rd Edition, 2014, McGraw Hill
- Herbert Schildt: JAVA the Complete Reference, 7th/9th Edition, Tata McGraw Hill, 2007.
- Jim Keogh: J2EE-TheCompleteReference, McGraw Hill, 2007.
- StackOverflow: [www.stackoverflow.com](http://www.stackoverflow.com)
- Codeproject: [www.codeproject.com](http://www.codeproject.com)