

**MINOR PROJECT REPORT
On**

**TITLE OF PROJECT
PV Airlines (Airlines Ticket Booking System)**



**KANPUR INSTITUTE OF MANAGEMENT STUDIES
(AFFILIATED TO A.K.T.U. UNIVERSITY, LUCKNOW)**

MASTER OF COMPUTER APPLICATION (MCA)

**UNDER THE GUIDANCE
OF
Mr. Abhishek Kumar Verma**

**Submitted by:
Priya
Roll No: 2207140140024**



Kanpur Institute of Technology

**Address: 22 KM Milestone, Kanpur-Lucknow Highway,
P.O Singrausi, Gadan Khera, Uttar Pradesh 209801**

CERTIFICATE

This is to certify that the work embodied in this Project Report entitled “PV Airlines” (i.e. Airlines ticket booking system being submitted by “Priya”, 3rd Sem for partial fulfilment of the requirement for the degree of “ Master of computer Applications ” discipline in “Kanpur Institute of Management studies” during the academic session 2022-2024 is a record of piece of work, carried out by student under my supervision and guidance in the“ Department of Computer Applications ”, Kanpur Institute of technology.

Project Developer:

Priya
(2207140140024)

Project Supervisor:

Mr. Abhishek Kumar Verma
(Assistant professor)



Kanpur Institute of Technology

**Address: 22 KM Milestone, Kanpur-Lucknow Highway,
P.O Singrausi, Gadan Khera, Uttar Pradesh 209801**

ACKNOWLEDGEMENT

I want to convey my heartfelt appreciation and gratitude to my professors as well as my supervisors for providing me with the chance to work on this wonderful project “PV Airlines” (i.e. Airlines ticket booking System)”, which also aided me in conducting extensive study and learning about a bunch of new topics. I am quite grateful to them.

Second, I want to thank my family and friends for their assistance in completing this project in such a period of time. It was quite useful in terms of expanding my knowledge and abilities. Thank you to everyone who has helped completing the Project.

Contents

Abstract	ii
Certificate	iii
Acknowledgments	iv
1. Introduction	1
1.1 Motivation and Overview	1
1.3 Objective	2
2. Software Requirement Analysis	7
2.1 Define the problem	7
2.2 Define the modules and their functionalities (SRS)	7
3. Software Design	16
3.1	16
3.2	17
4. Testing	22
4.1	22
4.2	22
5. Implementation and User Interface	27
5.1	27
5.2	27
References/Bibliography	28
6. Appendices	

CHAPTER -1

INTRODUCTION

A Airlines Ticket Booking system project for local flights that allows users to book local flights tickets and get ticket receipt online. This Airlines Ticket project provides login rights for normal users and admin. A normal user may login and get a ticket online, print it and travel by flight. The ticketing process consists of a ticket booking form. The form allows the user to choose his source and destination. The source is the station from where the user will be boarding the flight. Destination is the station he needs to get down at. The system also consists of an option to select whether ticket is a single journey or a return ticket.

OVERVIEW & MOTIVATION

Airlines Ticket Booking is web-based application to provide information and accommodation more efficiently. Basically, it gives the information of ticket booking for a particular region, so here we provide the facility of booking-ticket and only those people can access that who are valid user. Nobody can see the information of others. Here we fill the registration form for registration so that any user can use that application. It is easy to use for any user. It saves the time in more efficient manner.

OBJECTIVE

- To develop an integrated system for Airlines Ticket Booking .
- Providing offline facility for passenger to book the ticket easily.
- Saves time as well.
- To view all information at one place like the booking-ticket.
- Passenger can see the status and many more.
- To increase the accuracy and efficiency of information delivery.
- To reduce time.

ORGANIZATION OF THE PROJECT

Sr. No.	PHASES	TIME DURATION
1.	Software Requirement Specification	2 weeks
2.	System Design	4 weeks
3.	Coding	3 weeks
4.	Implementation	4 weeks
5.	Testing	1 week

Chapter 2

SOFTWARE REQUIREMENT ANALYSIS

Software Requirement Specification (SRS) document usually contains a software vendor's understanding of a customer's software requirements. This document ensures that the software vendor and the customer are in agreement as to the features required in the software system being built. SRS is created after the initial requirement elicitation phase in which Software vendor interacts with the customer to understand the software needs.

An SRS is written in precise, clear and plain language so that it can be reviewed by a business analyst or customer representative with minimal technical expertise. However it also contains analytical models (entity relationship diagrams, data flow diagrams etc.) which can be used for the detailed design and the development of the software system. SRS is one of the most critical pieces of software development since it acts as the bridge between the software developers and business analysts. An incomplete or incorrect SRS can have disastrous effects on a software project.

The benefits of a good SRS are:-

1. A contract between the customer and the software vendor – A good SRS document specifies all the features required in the final system including technical requirements and interface requirements. SRS document is used by the user to determine whether the software vendor has provided all the features in the delivered software system. To the Software vendor it provides a solid foundation to fix the scope of the software system.
2. Enables costing and pricing of the project – A well-defined SRS enables software developers to accurately estimate the amount of effort required to build the software product.

3. Input for detailed design – A good SRS enables experienced developers to convert the requirements directly to a technical design. For example, a well-defined data dictionary can be easily converted to a database specification.
4. Management of user expectations – Since SRS precisely defines project scope; it ensures that user expectations don't change during software development. If they do, SRS can be modified and costing/pricing can be done again on the changes required.

REQUIREMENT ANALYSIS

HARDWARE REQUIREMENTS OF THE SYSTEM:

This phase of the software development process deals with a brief study of different hardware used in the computerized system. There is a list of hardware materials used during the making and also during the use of the proposed system. As the new system to be made into a computerized functional system, requirement of a computer is must. All the hardware needed here are generally the basic configuration of a typical office computer. A list of the hardware requirement used in the system is given below:

- **Minimum Configuration:**

To run the application software of the system in the computer the minimum configuration required is as below:

- ❖ 233 MHz Pentium® processor or other compatible
- ❖ Intel® Chipset Motherboard
- ❖ 512 MB SD-RAM
- ❖ 2 – 4 GB Hard Disk
- ❖ 4x Compact Disc drive or faster
- ❖ Monitor. [640 x 480 Display]
- ❖ Keyboard and Mouse.

Recommended Configuration:

The system can run on the above-mentioned system, however for perfectness and clearness it is recommended to use a computer system having the below listed configuration. The configuration listed below can give best and optimum result at instance during the working of the system.

- ❖ 1 GHz Pentium® 4 processor or other compatible
- ❖ Intel® 81x Chipset Motherboard
- ❖ 1024 MB SD-RAM
- ❖ 10-20 GB Hard Disk
- ❖ 52x Compact Disc drive or faster
- ❖ Color Monitor. [800 x 600 Display]
- ❖ Keyboard
- ❖ Mouse
- ❖ 132 Column Dot Matrix Printer/Laser Printers
- ❖ 500 V.A UPS. [Used in case of power failure]
- ❖ Mobile phone(4-G)

Note: The proposed system of local train ticketing can work on both the above mentioned computer configuration. It is just for knowledge that the system can also work with the old technology as well as the new technology.

SOFTWARE REQUIREMENTS OF THE SYSTEM:

Along with the hardware, used in the system it requires software to make a system as well as to run a system with the computer hardware. Collection of different types of hardware into a specific type can form a computer but it cannot execute different process on its own. For efficient and proper working of any system software must be installed. This software may be in the form of operating system or application software. To make the system one also needs to use software.

Software used in designing (code) of the system:

Operating System: LINUX kernel operating system

Application Software: Android

TOOLS

- **LINUX**

The LINUX is a monolithic Unix-like computer operating system Kernel. The LINUX operating system is based on it and deployed on both traditional computer systems such as PC and servers.

The Android OS for Tablet Computers, smart phones and smart watches is also based atop the LINUX kernel. Android makes the kernel, the most popular OS kernel in the world.

Android:

Android is developed by Andy Rubin and later bought by Google. It is a mobile operating system basically used for touch screen devices such as smartphones and tablets. Android's user interface is based on direct manipulation. In addition to touch screen Google has further developed Android TV, Android Auto, Android Wears. Android has the largest base of all OS of any kind.

FEASIBILITY ANALYSIS

Feasibility defined as the practical extent to which a project can be performed successfully. To evaluate feasibility, a feasibility study is performed, which determines whether the solution considered to accomplish the requirements is practical and workable in the software. Information such as resource availability, cost estimation for software development, benefits of the software to the organization after it is developed and cost to be incurred on its maintenance are considered during the feasibility study. The objective of the feasibility study is to establish the reasons for developing the software that is acceptable to users, adaptable to change and comfortable to established standards. Various other objectives of feasibility study are listed below.

1. To analyze whether the software will meet organizational requirements.
2. To determine whether the software can be implemented using the current technology and within the specified budget and schedule.
3. To determine whether the software can be integrated with other existing software.

TECHNICAL FEASIBILITY:-

Technical feasibility assesses the current resources (such as hardware and software) and technology, which are required to accomplish user requirements in the software within the allocated time and budget. For this, the software development team ascertains whether the current resources and technology can be upgraded or added in the software to accomplish specified user requirements. Technical feasibility also performs the following tasks.

1. Analyzes the technical skills and capabilities of the software development team members.
2. Determines whether the relevant technology is stable and established.
3. Ascertain that the technology chosen for software development has a large number of users so that they can be consulted when problems arise or improvements are required.

OPERATIONAL FEASIBILITY:-

Operational feasibility assesses the extent to which the required software performs a series of steps to solve business problems and user requirements. This feasibility is dependent on human resources (software development team) and involves visualizing whether the software will operate after it is developed and be operative once it is installed. Operational feasibility also performs the following tasks.

1. Determines whether the problems anticipated in user requirements are of high priority.
2. Determines whether the solution suggested by the software development team is acceptable.
3. Analyzes whether users will adapt to new software.
4. Determines whether the organization is satisfied by the alternative solutions proposed by the software development team.

1. To analyze whether the software will meet organizational requirements.
2. To determine whether the software can be implemented using the current technology and within the specified budget and schedule.
3. To determine whether the software can be integrated with other existing software.

TECHNICAL FEASIBILITY:-

Technical feasibility assesses the current resources (such as hardware and software) and technology, which are required to accomplish user requirements in the software within the allocated time and budget. For this, the software development team ascertains whether the current resources and technology can be upgraded or added in the software to accomplish specified user requirements. Technical feasibility also performs the following tasks.

1. Analyzes the technical skills and capabilities of the software development team members.
2. Determines whether the relevant technology is stable and established.
3. Ascertain that the technology chosen for software development has a large number of users so that they can be consulted when problems arise or improvements are required.

OPERATIONAL FEASIBILITY:-

Operational feasibility assesses the extent to which the required software performs a series of steps to solve business problems and user requirements. This feasibility is dependent on human resources (software development team) and involves visualizing whether the software will operate after it is developed and be operative once it is installed. Operational feasibility also performs the following tasks.

1. Determines whether the problems anticipated in user requirements are of high priority.
2. Determines whether the solution suggested by the software development team is acceptable.
3. Analyzes whether users will adapt to new software.

4. Determines whether the organization is satisfied by the alternative solutions proposed by the software development team.

ECONOMICAL FEASIBILITY: -

Economic feasibility determines whether the required software is capable of generating financial gains for an organization. It involves the cost incurred on the software development team, estimated cost of hardware and software, cost of performing feasibility study, and so on. For this, it is essential to consider expenses made on purchases (such as hardware purchase) and activities required to carry out software development. In addition, it is necessary to consider the benefits that can be achieved by developing the software. Software is said to be economically feasible if it focuses on the issues listed below.

1. Cost incurred on software development to produce long-term gains for an organization.
2. Cost required for conducting full software investigation (such as requirements elicitation and requirements analysis).
3. Cost of hardware, software, development team, and training.

MODULE Functionalities

CUSTOMER LOGIN

- He can view the stations.
- Passenger can book-ticket.
- He can also see the amount left and can also check the status.

ADMIN LOGIN

1. Admin can check the information of customer. Admin can also view the list of all approved information and list of those who have completed booking.
2. Admin can view all transactions.

Admin can view the list and can also view their individual profile

Chapter 3

SOFTWARE DESIGN

DATA FLOW DIAGRAM (DFD)

INTRODUCTION OF DATA FLOW DIAGRAM: -

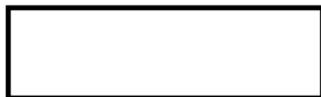
A Data Flow Diagram (DFD) is a graphical representation of the "flow" of data through an information system. A data flow diagram can also be used for the visualization of data processing. Data flow diagrams are commonly used during problem analysis. A DFD shows movement of data through the different transformations or processes in the system. DFD shows what kind of data will be input to the system and what data is received as the output.

Data Flow Diagrams (DFDs) are the method of choice over technical descriptions for three principal reasons-

1. DFDs are easier to understand by technical and nontechnical audiences.
2. DFDs can provide a high level system overview, complete with boundaries and connections to other systems.
3. DFDs can provide a detailed representation of system components.

ELEMENTS OF DATA FLOW DIAGRAM:-

- 1. External Entity-** An entity is the source or destination of data. The source in a DFD represents these entities that are outside the context of the Entities are often represented by rectangle



- 2. Process -** A process shows a transformation or manipulation of data flows within the system. A process is the manipulation or work that transforms data, performing computations, making decisions, or directing data flows based on business rules. In other words, a process receives input and generates some output.



- 3. Data Stores-** A data store is a holding place for information within the system. It is represented by an open ended narrow rectangle. A data store is where a process stores data between processes for later retrieval by that same process or another one.

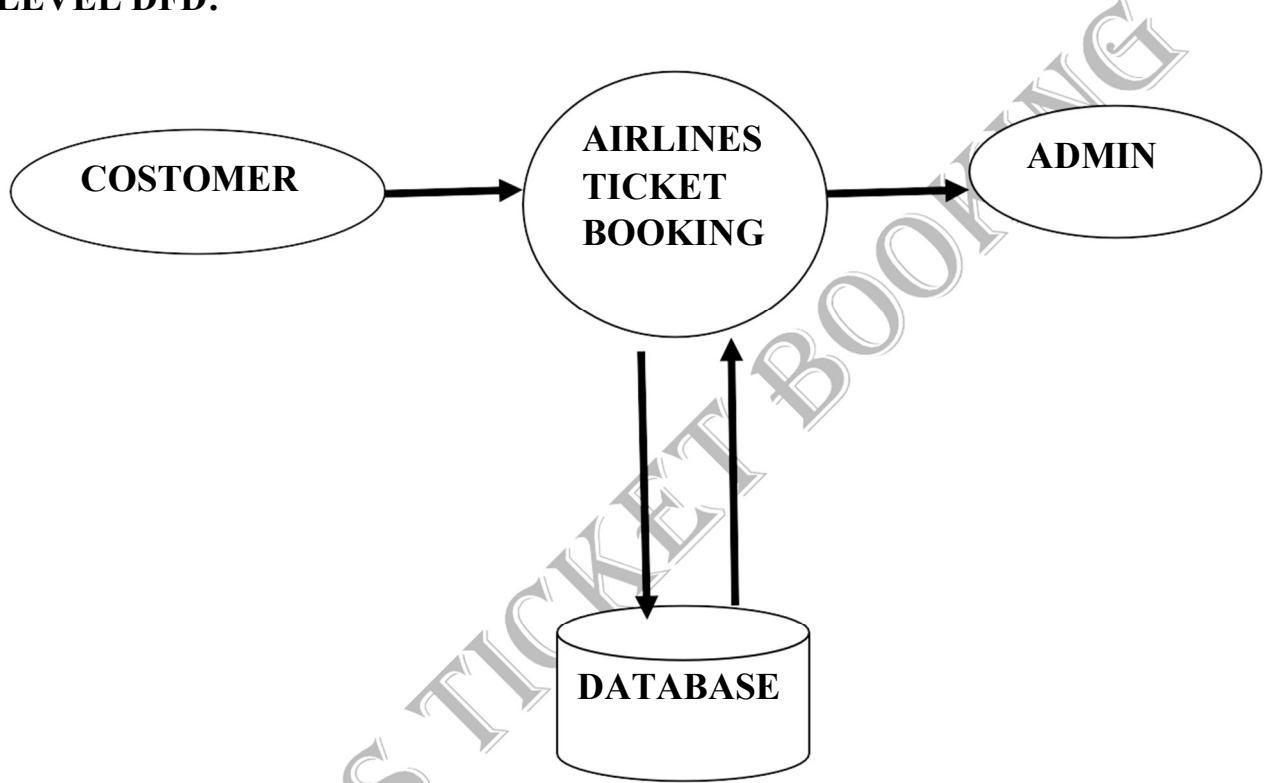


- 4. Data Flow-** A data flow shows the flow of information from its source to its destination. A data flow is represented by a line, with arrowheads showing the direction of flow. Each data flow may be referenced by the processes or data stores at its head and tail, or by a description of its contents.

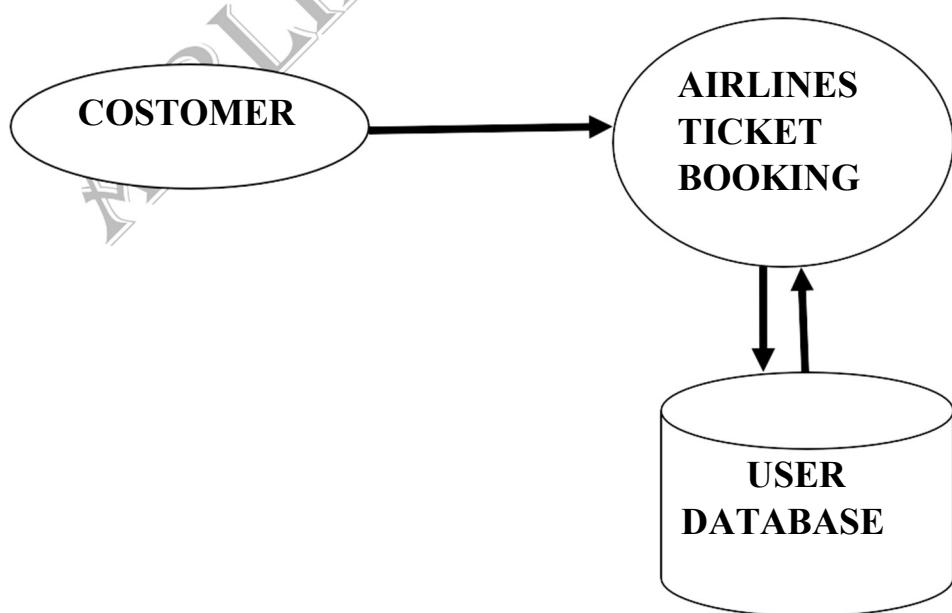


Level-wise Dataflow Diagram

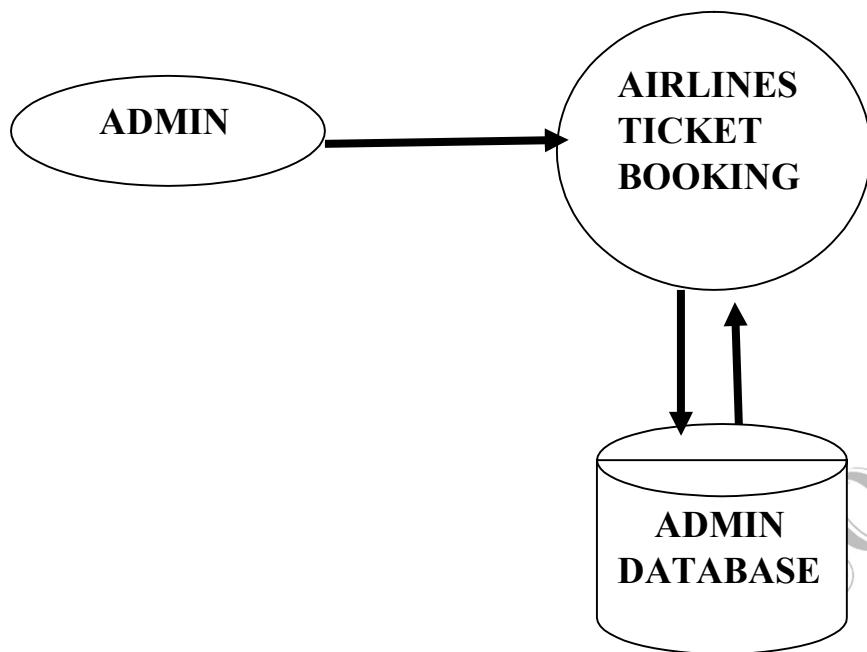
0 LEVEL DFD:



1 LEVEL DFD :



1 LEVEL DFD :



DATABASE DESIGN

Database design is the process of producing a detailed data model of a database. This logical data model contains all the needed logical and physical design choices and physical storage parameters needed to generate a design in a data definition language, which can then be used to create a database. The database design specifies how the data of the software is going to be stored. Databases are programs that allow for the storage and retrieval of large quantities of related data. Databases consist of tables that contain data. When creating a database you should think about what tables you are going to create and what relationships exist between the data in your tables. In other words, you have to think about the design of your database. A good database design will ensure the integrity and maintainability of your data.

DATABASE SCHEMA

- **LOGIN DATABASE**

ATTRIBUTE	TYPE	CONSTRAINT	DESCRIPTION
Name	varchar	Not Null	Passenger name
Username	varchar	Primary key	Id
Password	password	Not null	Security
Age	int	Not null	Passenger age
Mobile No.	int	Not null	Mobile number

Table 4.1

- **BOOKING DATABASE**

ATTRIBUTE	TYPE	CONSTRAINT	DESCRIPTION
Username	varchar	Primary key	Id
Source Station	varchar	Not null	Source station
Destination Station	varchar	Not null	Destination station
Type of Journey	varchar	Not null	Single or Double
Cost	int	Not null	Cost of Journey
Amount Left	int	Not null	Amount left in recharge

Chapter 4

IMPLEMENTATION & USER INTERFACE

User interface is the front-end application view to which user interacts in order to use the software. User can manipulate and control the software as well as hardware by means of user interface.

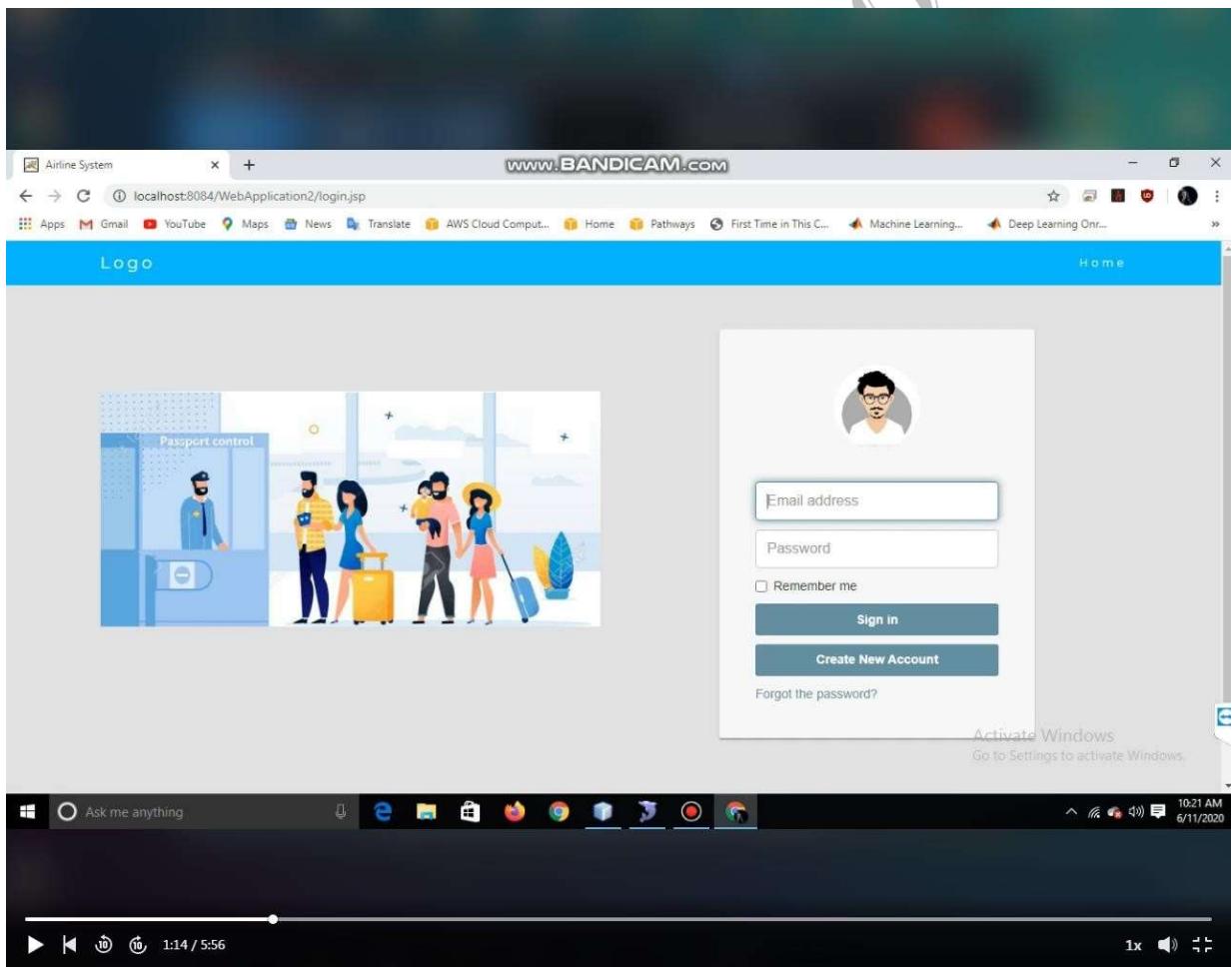
User interface is part of software and is designed such a way that it is expected to provide the user insight of the software. User Interface provides fundamental platform for human-computer interaction.

User Interface can be graphical, text-based, audio-video based, depending upon the underlying hardware and software combination. User Interface can be the software becomes more popular if its user interface is:

- Attractive
- Simple to use
- Responsive in short time
- Clear to understand
- Consistent on all interfacing screens

USER INTERFACE OF PROJECT

- User Login Page



- Search Flight

The screenshot shows a web browser window titled "Airline System" with the URL "localhost:8084/WebApplication2/index.jsp". The page has a blue header with the word "Airline" and navigation links for "ABOUT", "SERVICES", "TESTIMONIAL", "PRICING", "CONTACT", and "Login". Below the header is a large image of an airplane tail with the text "FLYING AIRWAYS". A red button labeled "Click Me" is visible on the image. The main content area is titled "SEARCH FLIGHT". It contains fields for "Source City" (Kanpur) and "Destination City" (Sri Nagar), a "Date" field showing "04/11/2020", and a "Search Flight" button. Below these fields is a table with columns: Flight Name, Company, Seats, Source, Destination, Dep. Time, Arr. Time, and a price column showing "5555". The table has one row with the text "kanpur to srinagar". At the bottom of the screen, there is a taskbar with various icons and a system tray showing the date and time as "10:21 AM 6/11/2020".

Passenger Detail :

The screenshot shows a web-based flight booking application. At the top, there is a search bar with the placeholder "Flight Id" and a "Search Flight" button. Below this is a table displaying flight information:

Flight Name	Company	Seats	Source	Destination	Dep. Time	Arr. Time	Price
kanpur to srinagar	Kingfisher	100	Kanpur	Sri Nagar	2020-06-11 11:44:00.0	2020-06-11 13:43:00.0	5555

Below the table, the heading "Passenger Detail" is displayed. There are four input fields for passenger information: "Passenger Name" (Jyoti Prajapati), "Passenger Email" (jyoti@gmail.com), "Passenger DOB" (03/11/1999), and "Passenger Gender" (Female). A "Book Now" button is located below these fields. The bottom of the screen shows a Windows taskbar with various icons and system status.

Final Ticket

The screenshot shows a web browser window with two tabs open. The active tab displays a ticket detail page for an Airline System. The page includes passenger information, flight details, and a summary section.

Passenger Detail:

- Name: Shubham Prajapati
- Email: shubhamprajapati@axiscolleges.in
- Date of Birth: (1999-03-11)
- Gender: Female
- Email: jyoti@gmail.com
- Phone: 9235647810

Ticket Detail:

- From: Kingfisher
- To: Kanpur
- Date: 2020-06-11 11:44:00.0
- From: Sri Nagar
- To: Kanpur
- Date: 2020-06-11 13:43:00.0

Summary:

- Export as PDF
- Print

The browser's address bar shows the URL: localhost:8084/WebApplication2/user/my_ticket.jsp?tid=13. The status bar at the bottom indicates the date as 6/11/2020 and the time as 10:24 AM.

User Home: View Ticket User Home: Ticket + www.BANDICAM.COM

localhost:8084/WebApplication2/user/my_ticket.jsp?tid=13

Apps Gmail YouTube Maps News Translate AWS Cloud Comput... Home Pathways First Time in This C... Machine Learning... Deep Learning Onr...

Kingfisher **Kanpur** **Sri Nagar**
Phone: (123) 456-7890 2020-06-11 11:44:00.0 2020-06-11 13:43:00.0
Fax: (123) 456-7890

Passenger Detail
Jyoti Prajapati
Date of Birth : (1999-03-11)
Gender : Female
Email : jyoti@gmail.com
Phone : 9235647810

Amount Detail

	TOTAL
Base Fare	5555.0 INR
Tax	0.00 INR
SUBTOTAL	5555.0 INR
PAYPAL FEE (0.0%)	0.00 INR
	TOTAL
	5555.0 INR

Windows Go to Settings to activate Windows

Ask me anything e 10:24 AM
6/11/2020

▶ ⏪ ⏹ 3:53 / 5:56 1x 🔊

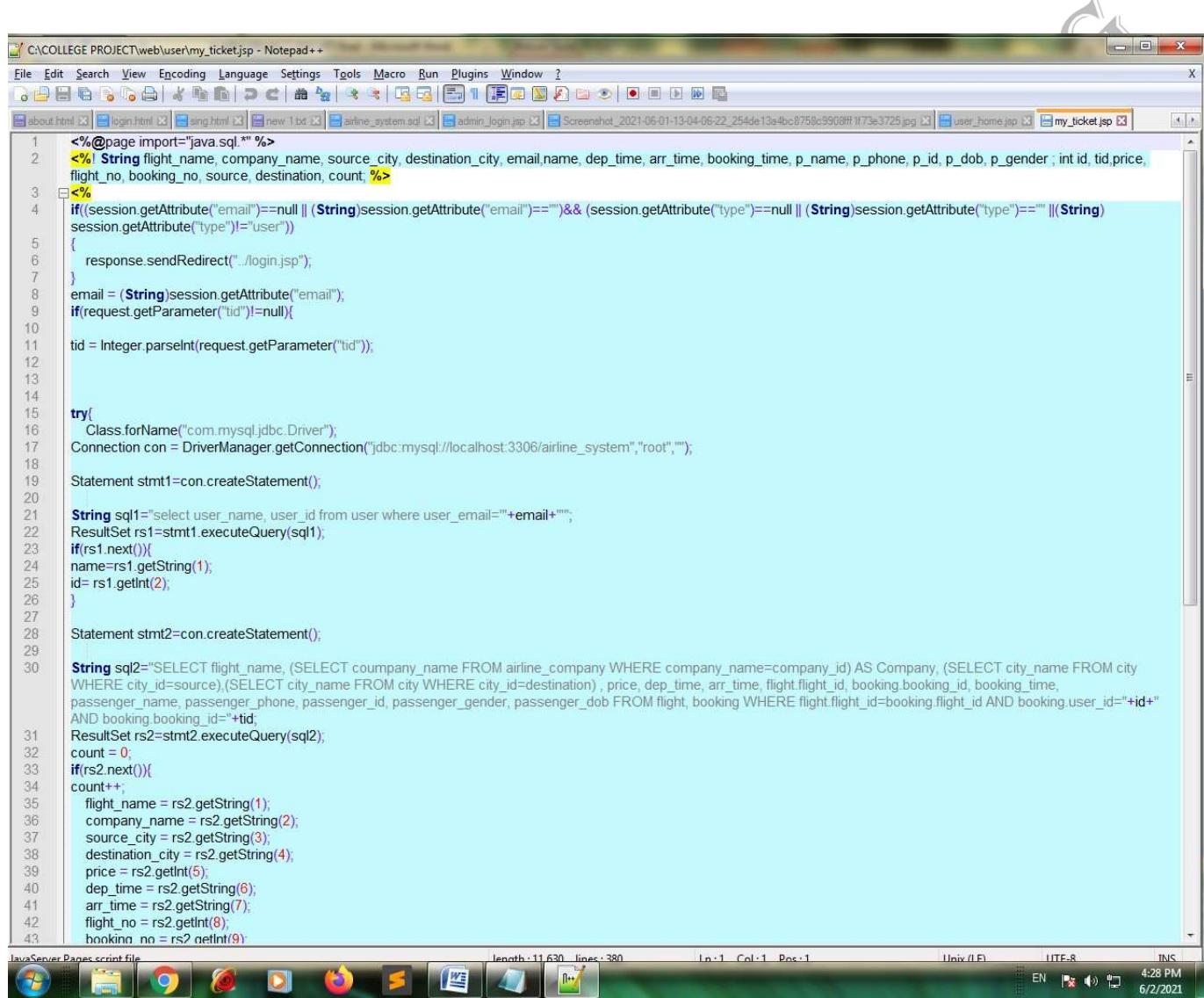
Chapter 5

CONCLUSION

Airlines Ticket Booking is an application which help the passenger to book the ticket. The best thing in this Web application is that it can run offline also. Now user does not have to stand in a queue to take a ticket or miss the train. Passenger can book the ticket not for one way only but for both way. With this user does not have to book the ticket again and again.

APPENDICES

1. User Login



The screenshot shows a Notepad++ window with Java code for a user login script. The code uses JDBC to interact with a MySQL database named 'airline_system'. It checks if session attributes for email and type are null or empty, and if the type is not 'user'. If so, it redirects to 'login.jsp'. It then retrieves the user's ID from the database using their email. Finally, it performs a JOIN query to get flight and booking details for the user.

```
1 <%@page import="java.sql.*" %>
2 <% String flight_name, company_name, source_city, destination_city, email.name, dep_time, arr_time, booking_time, p_name, p_phone, p_id, p_dob, p_gender ; int id, tid, price,
3 flight_no, booking_no, source, destination, count; %>
4 <%
5 if((session.getAttribute("email")==null || (String)session.getAttribute("email")=="")&& (session.getAttribute("type")==null || (String)session.getAttribute("type")=="") ||(String)
6 session.getAttribute("type")!="user")){
7     response.sendRedirect("../login.jsp");
8 }
9 email = (String)session.getAttribute("email");
10 if(request.getParameter("tid")!=null){
11     tid = Integer.parseInt(request.getParameter("tid"));
12 }
13
14 try{
15     Class.forName("com.mysql.jdbc.Driver");
16     Connection con = DriverManager.getConnection("jdbc:mysql://localhost:3306/airline_system","root","");
17
18     Statement stmt1=con.createStatement();
19
20     String sql1="select user_name, user_id from user where user_email='"+email+"';
21     ResultSet rs1=stmt1.executeQuery(sql1);
22     if(rs1.next()){
23         name=rs1.getString(1);
24         id= rs1.getInt(2);
25     }
26
27     Statement stmt2=con.createStatement();
28
29     String sql2="SELECT flight_name, (SELECT company_name FROM airline_company WHERE company_name=company_id) AS Company, (SELECT city_name FROM city
30     WHERE city_id=source),(SELECT city_name FROM city WHERE city_id=destination) , price, dep_time, arr_time, flight.flight_id, booking.booking_id, booking_time,
31     passenger_name, passenger_phone, passenger_id, passenger_gender, passenger_dob FROM flight, booking WHERE flight.flight_id=booking.flight_id AND booking.user_id="+id+
32     " AND booking.booking_id="+tid;
33     ResultSet rs2=stmt2.executeQuery(sql2);
34     count = 0;
35     if(rs2.next()){
36         count++;
37         flight_name = rs2.getString(1);
38         company_name = rs2.getString(2);
39         source_city = rs2.getString(3);
40         destination_city = rs2.getString(4);
41         price = rs2.getInt(5);
42         dep_time = rs2.getString(6);
43         arr_time = rs2.getString(7);
44         flight_no = rs2.getInt(8);
45         booking_no = rs2.getInt(9);
46     }
47 }
```

DATABASE CODE OF SQL

```
C:\COLLEGE PROJECT\airline_system.sql - Notepad++  
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?  
about.html about.html login.html sing.html new1.txt airline_system.sql admin_login.jsp user_home.jsp my_ticket.jsp  
16 CREATE DATABASE /*!32312 IF NOT EXISTS*/`airline_system` /*!40100 DEFAULT CHARACTER SET latin1 */;  
17  
18 USE `airline_system`;  
19  
20 /*Table structure for table 'admin' */  
21  
22 DROP TABLE IF EXISTS `admin`;  
23  
24 CREATE TABLE `admin` (  
25     `admin_id` int(11) NOT NULL AUTO_INCREMENT,  
26     `admin_name` varchar(50) NOT NULL,  
27     `admin_email` varchar(50) NOT NULL,  
28     `admin_password` varchar(50) NOT NULL,  
29     PRIMARY KEY (`admin_id`),  
30     UNIQUE KEY `UNIQUE` (`admin_email`)  
31 ) ENGINE=InnoDB AUTO_INCREMENT=2 DEFAULT CHARSET=latin1;  
32  
33 /*Data for the table 'admin' */  
34  
35 insert into `admin`(`admin_id`, `admin_name`, `admin_email`, `admin_password`) values (1,'Shubham','a@gmail.com','Admin123');  
36  
37 /*Table structure for table 'airline_company' */  
38  
39 DROP TABLE IF EXISTS `airline_company`;  
40  
41 CREATE TABLE `airline_company` (  
42     `company_id` int(11) NOT NULL AUTO_INCREMENT,  
43     `company_name` varchar(30) NOT NULL,  
44     PRIMARY KEY (`company_id`)  
45 ) ENGINE=InnoDB AUTO_INCREMENT=5 DEFAULT CHARSET=latin1;  
46  
47 /*Data for the table 'airline_company' */  
48  
49 insert into `airline_company`(`company_id`, `company_name`) values (1,'IndiGo'),(2,'Kingfisher'),(3,'AirIndia'),(4,'SpiceJet');  
50  
51 /*Table structure for table 'booking' */  
52  
53 DROP TABLE IF EXISTS `booking`;  
54  
55 CREATE TABLE `booking` (  
56     `booking_id` int(11) NOT NULL AUTO_INCREMENT,  
57     `user_id` int(11) NOT NULL,  
58     `flight_id` int(11) NOT NULL,  
59     `booking_time` datetime NOT NULL,  
60     `passenger_name` varchar(50) NOT NULL,  
61     `passenger_dob` date NOT NULL,  
62     `passenger_id` varchar(50) NOT NULL,  
63     `passenger_phone` varchar(15) NOT NULL,
```

References

<https://www.youtube.com/@CodeWithHarry>

www.w3schools.com