AIRLINE RESERVATION SYSTEM (ARS)

REPORT

SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENT FOR

Six Month Industrial Training

at

Naresh IT, Hyderabad

(from July-2019 to Nov-2019)

SUBMITTED BY

Name: Nitesh kumar

Branch: IT

C. Roll No: 1621107

U. Roll No: 1607126

Information Technology Department

GURU NANAK DEV ENGINEERING COLLEGE

LUDHIANA, INDIA

STUDENT'S DECLARATION

I hereby certify that the work which is being presented in the training report with the project entitled "**Airline Reservation System**" by **Nitesh Kumar**, University Roll No.- **1607126** in partial fulfilment of requirements for the award of degree of B.Tech. (Information Technology) submitted in the Department of Information Technology at **GURU NANAK DEV ENGINEERING COLLEGE, LUDHIANA under I.K. GUJRAL PUNJAB TECHNICAL UNIVERSITY** is an authentic record of my own work carried out under the supervision of **Mr. Hari Krishna(Java Developer)** and **Mr. Vijay Kumar(MySQL Expert)** of **Naresh IT**. The matter presented has not been submitted by me in any other University / Institute for the award of B.Tech. Degree.

Student Name: **Nitesh Kumar (URN: 1607126)**

(Signature of Student)

This is to certify that the above statement made by the candidate is correct to the best of my knowledge.

Signature of Internal Examiner

The External Viva-Voce Examination of the student has been held on 31.05.2019.

Signature of External Examiner

Signature of HOD

NARESH i technologies®

SOFTWARE TRAINING & DEVELOPMENT

An ISO 9001 : 2015 Certified Company

COURSE COMPLETION CERTIFICATE

This is to certify that Mr. / [redacted]. NITESH KUMAR

has completed a course on CORE JAVA. ADVANCED JAVA

successfully from JULY 2019

to NOVEMBER - 2019

CERTIFIED NARESH i TECHNOLOGIES ISO 9001 : 2015

Center Head

H.O.: Opp.Satyam Theatre, D.No.201-204, Durga Bhavani Plaza, Ameerpet, Hyderabad - 500 016
Phone: 040-23746666,23734842 info@nareshit.com www.nareshit.com nareshit nareshitech nareshit
Branch Office: CHENNAI Ph:95660 42345/95660 42845 | VIJAYAWADA Ph: 9293226789

GURU NANAK DEV ENGINEERING COLLEGE, LUDHIANA

TO WHOM IT MAY CONCERN

I hereby certify that "Nitesh kumar" Roll No:1607126 of Guru Nanak Dev Engineering College Ludhiana, has undergone six-month industrial training from July-2019 to Nov-2019 at our organization to fulfill the requirements for the award of degree of B. Tech Information Technology. He works on Airline Reservation system (ARS) project during the training under the supervision of **Hari Krishna**. During his tenure with us we found him sincere and hard working. Wishing him a great success in the future.

Signature of the Student

Signature of the SUPERVISOR (S)

(Seal of Organization)

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 between the two travel cities, namely the “Departure city” and “Arrival city” for a particular departure and arrival dates. The system displays all the flight’s details such as
flight no, name, price and duration of journey etc.

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 flight.

To book a flight the system asks the customer to enter his details such as name, address, city, state, and credit card number and contact number. Then it checks the validity of card and book 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 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.

ACKNOWLEDGEMENT

I am highly grateful to the **Dr. Sehijpal Singh**, Principal, Guru Nanak Dev Engineering College (GNDEC), Ludhiana, for providing this opportunity to carry out the six months industrial training at Naresh IT.

The constant guidance and encouragement received from **Prof. G.S SODHI** T&P, GNDEC Ludhiana has been of great help in carrying out the project work and is acknowledged with reverential thanks.

I would like to express a deep sense of gratitude and thanks profusely to **Naresh Nimmala** Director/CEO of Company, Without the wise counsel and able guidance, it would have been impossible to complete the report in this manner.

The help rendered by **Mr. Nataraz (Natraj)**, Java trainer for experimentation is greatly acknowledged.

I express gratitude to other faculty members of Information Technology department of GNDEC for their intellectual support throughout the course of this work.

Finally, WE are indebted to all whosoever have contributed in this report work.

Nitesh Kumar

LIST OF FIGURES

List of Tables

Table of Contents

1. Introduction to Organization

Naresh I Technologies (Pronounced: Naresh IT) is a leading software training institute providing Software Training, Project Guidance, IT Consulting and Technology Workshops.

Using our enhanced global software training delivery methodology, innovative software training approach and industry expertise, it provides high-value corporate training services that enable our clients to enhance business performance, accelerate time-to-market, increase productivity and improve customer service.

It serves Global 100 companies and the leading software vendors in Banking & Financial Services, Insurance, Telecommunications, Technology and Media, Information & Education industries. It designs and mentor human resources for our clients who create competitive advantage.

Services Offered by Company:

Naresh I Technologies Major Aspect of Concentration is IT Training at all the levels like Academic or Corporate Level. The mode of training and the course content for the training program varies depending upon the participants and their requirements. Hence, we have segregated our training division into the below segments so that maximum efficacy can be derived from the training. we are recognized to be the connoisseur in providing the best training program to the Candidates who are inclined to be Software Professionals.

- Classroom Training
- On-Campus Training
- Online Training
- Corporate Training

2. INTRODUCTION TO PROJECT

1.1 Overview

Air travel has become less and less cheap over the years, as a result the no. of passengers has increased considerably. Unlike a train or bus ticket where we have to stand in a queue besides a counter, air tickets are usually booked online.

The proposed system is a GUI portal where the user can book their air travel tickets between two cities on a particular date.

There will be a provision for searching the flights for a travel destination on an input date, the search result will comprise of the full details such as flight id, flight name, cost, arrival and departure time etc. A cancellation option is also given in case user wishes to abandon his travel for various reasons.

There are only four modules of project:

1. Flight Reservation module
2. Flight Cancellation module
3. Booking module
4. Add Customers detail

In this GUI application Customers with legal username and password can Sign In. A Customer can book, cancel the flight and see all flight details in between two airports.

Flight are booked through Flight reservation module in which all details regarding customer and flight are entered through the database if you know our customer id and flight id.

It is important to remember your customer id and ticket no in this whole process, if you forgot your customer id and ticket no than you can ask me in the help section with your query.

1.2 Project Implementation (Windows Based Application)

A. It is a GUI based project.
B. It is a part of Java Foundation Classes (JFC) that *is* used to create window-based applications.
C. It is built on the top of AWT (Abstract Windowing Toolkit) API and entirely written in java Swing.
D. Java Swing provides platform-independent and lightweight components.
E. Swing is platform independent and enhanced MVC (Model –View – Controller) framework for Java application.

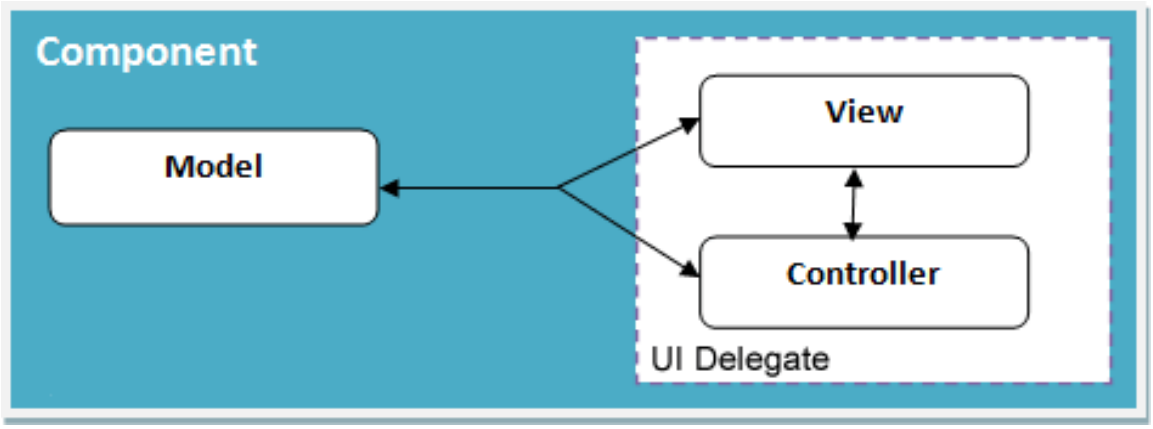


Fig No: 2.2.1

1.3 Objectives of Project

The main objective of the project is to develop the GUI application for airline reservation system. The project has the following objective:

1. Airline Reservation System contains the details about flight schedules, passenger reservation and ticket records.

2. To provide user convenience.

3. Inventory data is imported and maintained through a schedule distribution system over standardized interface.

4. In combination with the fares and booking conditions stored in the Fare Quote System the price for each sold seat is determined.

5. To cancel the booked ticket.

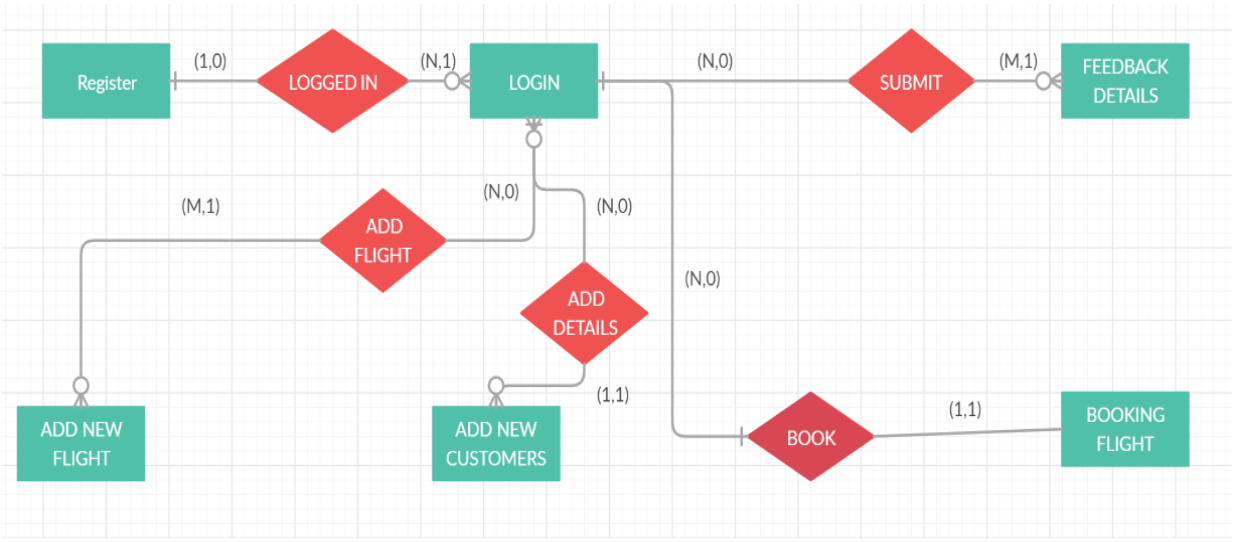
1.4 Problem Formulation

The GUI based "Airline reservation system (ARS)" project is an attempt to simulate the basic concepts of airline reservation system. The system enables the customer to do things such as search for airline flights for two cities on a specified date, choose a flight based on the details, reservation of flight and cancellation of reservation.

The system allows the airline passenger to search for flights that are available between the two cities, namely the "Departure city" and "Arrival city" for a particular departure and arrival times. The system displays all the flight's details such as flight no, name, flight cost, and duration of journey etc.

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

To book a flight the search checks the customer to enter his details such as ID, name, father's name, gender, date of birth, address and image. Then it checks the validity of card and book 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.



E-R Diagram of Project (**Fig No: 2.4.1**)

1.5 Identification/Reorganization of Need

1) The whole process of this project is easy and comfortable.

2) In this project Customer can do everything which is related to project.

3) If a user has a username and password than it can do the whole process of this project as like add customer detail, flight detail, see flight and book ticket.

4) In this project it is not compulsory to ticket will cut by manager or admin. Ticket will cut by the any customer which has own username and password of this whole system.

5) If you want to know my whole project than you can see the about part of the project.

6) If you have any query about project then you can ask me any query in the help and feedback department.

7) If you want to ask me any question than it is compulsory to give own customer id and flight id than you can give own mobile number than submit your query. Mobile helps for reply of your question.

8) If you want to see the all flight details and booking details than you can see all these in the Account section.

9) In the reservation process you can fetch all data from the database and you can fetch flight detail with flight id than you can select class, sets, amount than you can book your flight. In this section you can see your details and flight details in between two cities.

10) In the enquiry section you can see your ticket detail i.e. your ticket is booked or not. For this section you can enter your ticket no than all data fetch from database.

11) If you want to delete the ticket details than you can cancel the ticket in the Ticket window section.

If you want you can go to Facebook, Instagram and twitter and google services.

12) In the Contact section it is compulsory to give your country and give your departure city.

1.6 Existing System

The company follows a manual system for maintaining the Airline Reservation System.

1) Whenever any requisition comes to the counter to issue a ticket the firstly the ticket capacity checked whether the required tickets is present then issue a ticket and update the reservation information and an entry is done to the ticket issue register.

2) The day-to-day entries are made manually into the book that has gotten all the relevant entries.

3) Check the Ticket details regularly whether any ticket is not in, if so ticket issue process is cancelled. In the same way ticket cancellation is followed. If fare is introduced then entry on relevant book, that process applies to flight-schedule, Airbus, And Branch.

4) During analysis, data collected on the various files, decision points and transaction by the present system.

5) The effectiveness of the system depends on the way in which the data is organized in the existing system, much of the data is entered manually and it can be very time consuming. When records are accessed frequently, managing such records becomes difficult. Therefore, organizing data becomes difficult.

❖ **The major limitations are:**

1. Modification is complicated
2. Much time consuming
3. Security
4. Unauthorized access of data
5. Maintaining and managing data is very costly and time consuming, because there are many documents that have to be transferred to relative branches.
6. Records might get lost or be insufficient due to manual errors.
7. Complexity
8. Accuracy

1.7 Proposed System

The proposed system is better and more efficient than existing system by keeping in mind all the drawbacks of the present system to provide a permanent to them.

The primary aim of the new system is to speed up the transaction. User friendliness is another peculiarity of the proposed system. Message are displayed in message boxes to make the system user friendly.

The main advantages of the proposed system is the reduction in labor as it will be possible so search the details of various places. Every record checked for completeness and accuracy and them it is entered into the database.

The comments and valid message are provided to get away redundant data. Another important feature of the proposed system is the data security provided by the system.

The main objectives of the proposed system are:

1. Complex functions are done automatically.
2. Processing time can be minimized
3. Simple and easy to manage
4. Changes of errors reduced
5. Faster and more accurate than the existing system
6. Easy for handling report

The proposed system is complete software for Airline Reservation System, which is more efficient, Reliable, Faster and accurate for processing.

Feature of the proposed system:

1. The system has been developed under **NetBeans IDE 8.2** as a front-End and **MYSQL Database** as a back-End tool.
2. Increase security, speed, storing and accuracy.

Advantages of proposed system:

1. The proposed system due to computerized is much faster in reservation process, cancellation process and transaction.
2. Transfer of information from various branches would become easier and faster.

Limitation for proposed system:

1. Per ticket only one other service is included at a time.

2. Passenger can't transfer his/her reservation from one flight to another flight.

2.8 Unique Feature of system

A. 24/7 booking available for customer.

B. All the details monitored by the customer from anywhere.

C. Customer can select different city with respect to his/her choice.

D. It is compulsory to verify your email id to click on verify button.

E. Easy cancellation process.

F. Easy and flexible booking policies for travels.

G. Display terms and condition, details and flight to the travels.

H. Provide customer history with just one click.

I. Provides flight details with just one click.

3. **Requirement analysis**

i. The screen formats and organization. The interface should be easy to understand and user include such as

ii. The introductory screen will be the first displayed which will allow the user to choose either of 2 options.

iii. That view flight detail within booking a ticket.

iv. Windows format and organization when our customer choose some other option, then the information pertaining to that choice will be displayed in new window.

v. That ensure multiple windows to be visible on the screen.

vi. The customer can switch among them.

vii. The data format entered by the users will be alphanumeric.

viii. The end message, when there were some exceptions raising error like entering invalid details, then error message will be displayed prompting the users to re-enter the details.

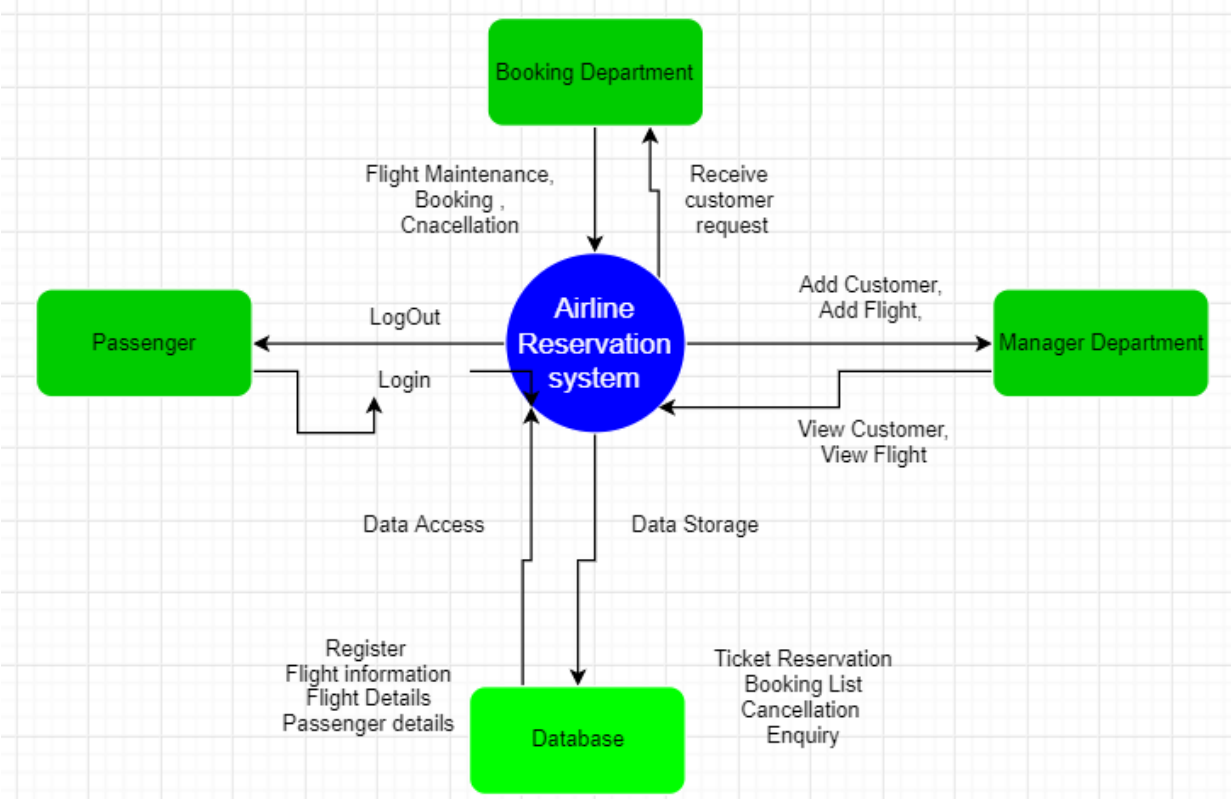


Fig No: 3.1

3.1 Feasibility Study

Feasibility study is most important and it concerns great to the passenger, for which new system is to be developed. It includes the study of success and level of satisfaction that the passenger and user will get from the system.

Feasibility means possibilities (to some extent), i.e. this study explores the possibility of implementation of new system as a substitute to old system removing all short comings and including all new requirements.

An initial determine in a proposal that whether an alternative system is feasible or not. To determine feasibility of candidate system in all respect I need to consider following feasibility factors:

i. Economic feasibility

ii. Technical feasibility

iii. Operational feasibility

A. Economic feasibility

Since the existing system is manual, the organization posses the computer and doesn't require to set the environment.

Cost involved in the software packages include the MYSQL 3306 for storage of data as a back end. Other required software is that which they already have like windows 10. This costs around 4190/- and front-end software is free of costs.

B. Technical feasibility

It is determined by the hardware and supporting software.

Hardware requirement: Stand-alone computer with enough main memory space & backup storage more than 2 GB to support DBMS and huge database.

Software requirement: MYSQL 3306

C. Operational feasibility

It includes training the user staff on the customer system. The data entry design is similar to other forms being used by the user staff. Data entry forms are very user friendly and data entry job has been kept minimum.

So, training of 3-4 will be sufficient. Even for generating reports the user has to supply 2-3 parameters only.

So, the management must take corrective actions prior in advance in order to start the further proceedings.

3.2 Software Requirement Specification

a) Hardware Interface

The system must basically support certain input and output devices. Their description are as follows:

Table No: 1

PROCESSOR	Intel(R) Core (TM) i5-7200U CPU @ 2.50GHz 2.70GHz
RAM	8 GB
MONITOR	15” COLOR
HARD DIEK	1000 GB
KEYBOARD	Hp keyboard
MOUSE	Hp Mouse
CD DRIVE	DVD/CD-ROM drives

b) Software Interface

Table No: 2

FRONT END	Java, Advanced Java (Swing)
BACK END	MYSQL DATABASE
OPERATING SYSTEM	WINDOWS 10

A. Data Requirement

The data requirement of the Airline Reservation System is divided among the customer of the application.

These data requirements can be explained in detail as follow:

a) Use Case name: User Registration

i. **Description:** This use case describes the scenario where the user registers own detail in the application. And Login to the application.

ii. **Actor:** User or the Customer

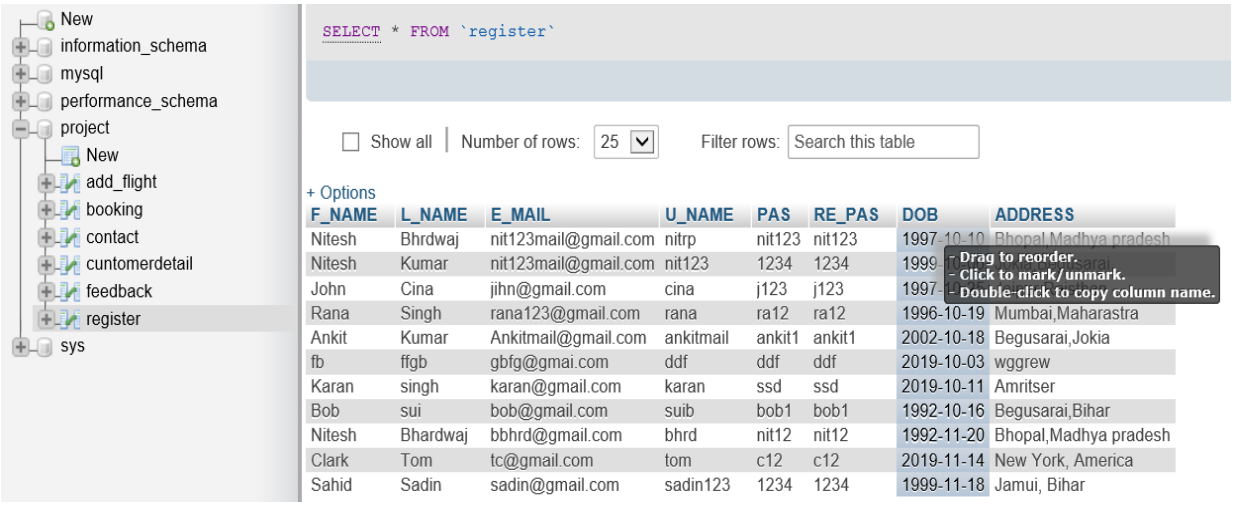
iii. **Input:** The user or the customer will have to provide all the necessary details present in the customer registration form of the application.

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

b) Use Case Name: User Login

i. **Description:** This use case describes the scenario where the user logs into the application, with the username and password he has provided while registering with the system.

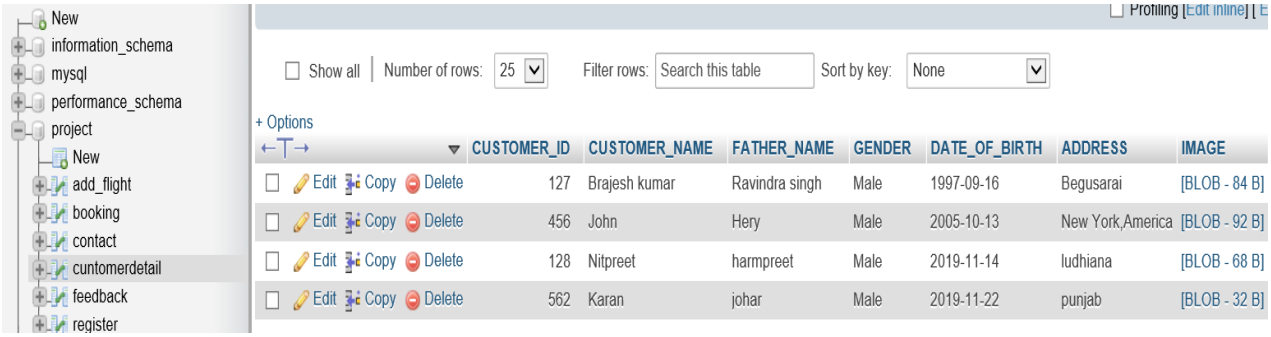
Table No: 3



c) Use Case Name: Add Customer Details

i. **Description:** This use case describes the all details of customer's information.

Table No: 4

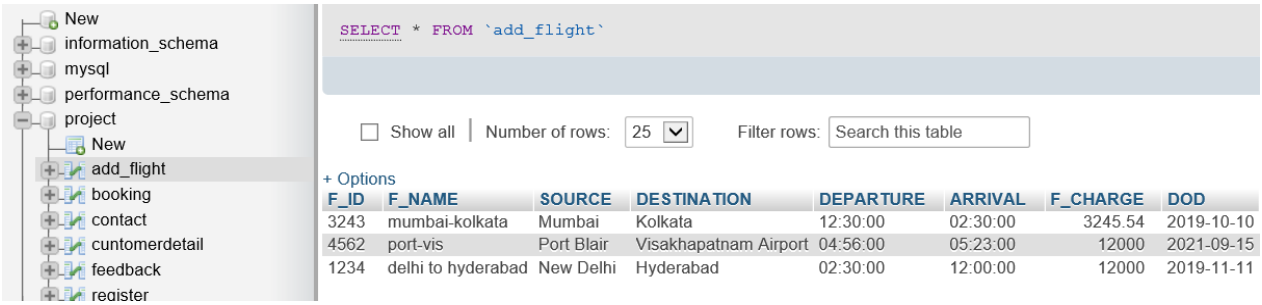


ii. **View Customer:** This form is uses to see own details in the page. Which is fetch the data from the table "customer detail".

d) Use Case Name: Add Flight

i. **Description:** This use case describes the scenario where the user adds flight and view flight.

Table No: 5



e) Use Case Name: Booking Instruction

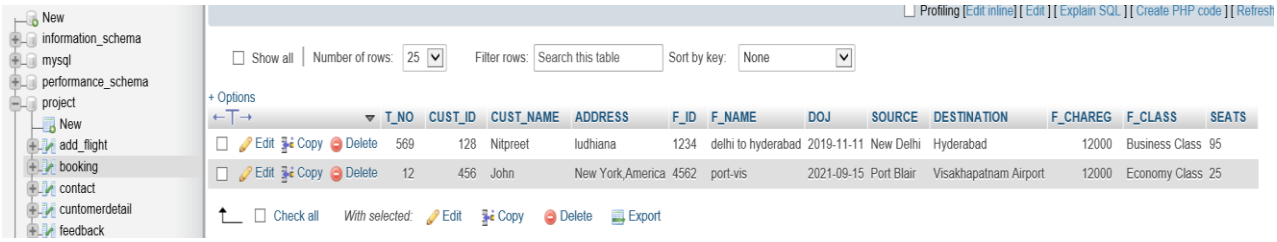
i. **Description:** This use case describes the scenario where the user views the instruction for booking flight, packages, or cancel the ticket.

ii. **Actor:** User or the Customer

f) Use Case Name: Account

i. **Description:** This use case describes the scenario where the customer saw the booking list and customer details and logs out after the work is done.

Table No: 6

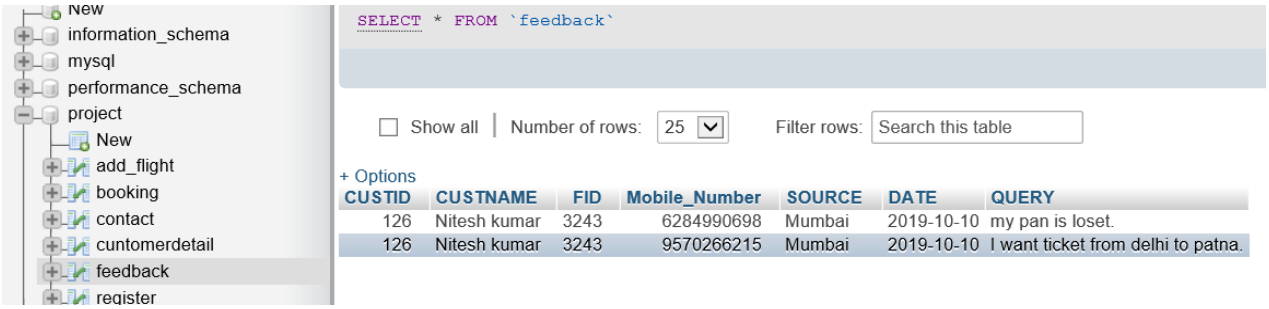


g) Use Case Name: Help and Feedback

i. **Description:** This use case describes the scenario where customer can turn to and give our feedback about the whole system.

ii. **Actor:** Customer

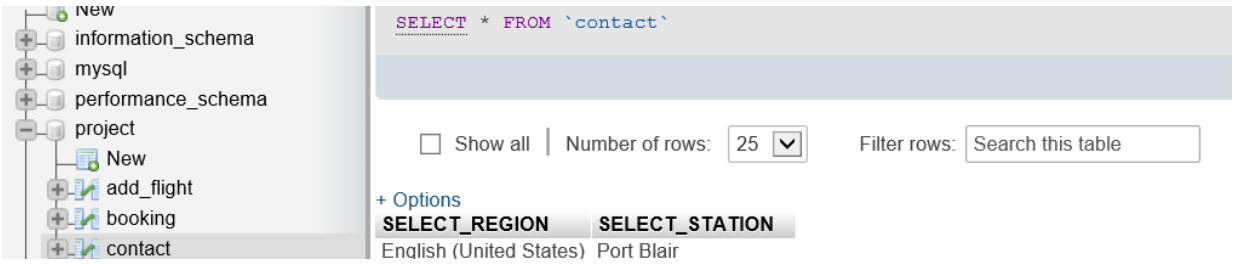
Table No: 7



h) Use Case Name: Contact

i. **Description:** This use case describes the scenario where the customer can contact to the administration department.

Table No: 8



B. Functional Requirement

The functional requirement of the airline Reservation System are divided among the customer and the administrator of the application.

These functional requirements can be explained in details as follows:

a. Use Case Name: User Registration

i. **Description:** This use case describes the scenario where the user registers with the application by providing all the necessary details, in order to make reservation or booking for flight.

ii. **Actor:** User or the admin

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

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

b. Use Case Name: User Login

i. **Description:** This use case describes the scenario where the user logs into the registering, with the username, password and user type he has provided while registering with the system.

ii. **Actor:** User or the Admin

iii. **Input:** The user or the admin creates a username, password and user type at the time of registering with the system. He then uses them to login to the system and make reservation or view any information.

iv. **Output:** The application then verifies the authenticity of the username, password and user type that the customer has provided and follows the user to view the information available on the system, if the username, password and user type are valid.

c. **Use Case Name: Contact the company**

i. **Description:** This use case describes the scenario where the user contacts the company for any information.

ii. **Actor:** User or the Admin

iii. **Input:** The customer can contact the airline company, requesting them for any information he needs.

iv. **Output:** The application verifies the authenticity of the username and password that the customer has provided and follows the admin to view the contact information for the company.

d. **Use Case Name: Booking Instruction**

i. **Description:** This use case describes the scenario where the user views the instruction for booking flights, see flight.

ii. **Actor:** User or the admin

iii. **Input:** After the customer logs onto the application with his username and password, he can look up the instruction posted on the website for booking flights, see flight details and customer information.

iv. **Output:** The application verifies the authenticity of the username and password and display the how to book instruction page.

e. **Use Case Name: Book Flight**

i. **Description:** This use case describes the scenario where the admin books airline tickets.

ii. **Actor:** User or the admin

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

iv. **Output:** The application verifies the authenticity of the username, password and user type then display information related to various flights to the customer.

f. Use Case Name: Login/Logout

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

ii. **Actor:** Administrator

iii. **Input:** The administrator of the website logs into the application with the username, password and user type provide to him.

iv. **Output:** The application verifies the authenticity and display the home page of the administrator.

g. Use Case Name: Add or Delete Customer information

i. **Description:** This use case describes the scenario where the administrator adds and delete customer information in the system database.

ii. **Actor:** Administrator

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

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

h. Use Case Name: Add or Delete flight information

i. **Description:** This use case describes the scenario where the administrator adds and delete flight information in the application database.

ii. **Actor:** Administrator

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

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

i. Use Case Name: Cancellation of reservation

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

ii. **Actor:** Administrator

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

iv. **Output:** The application authenticates the administrator and them displays the page where the administrator looks up the customer who has requested cancellation of reservation. After canceling the reservation, the administrator then books new flight.

j. Use Case Name: E-mail confirmation

i. **Description:** This use case describes the scenario where the administrator verifies the e-mail of customer of the application.

ii. **Actor:** Administrator

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

iv. **Output:** The application then authenticates the administrator and displays the page where the administrator can verify e-mail of the customer. Mails are verified at the registration time.

C. Nonfunctional Requirements

a. Performance Requirements

i. The system shall accommodate 250 users during the peak time window of 24 hour.

ii. All GUI page generated by the system cannot be downloaded.

iii. Responses to queries shall take no longer than 10 second to load onto the screen after the user submits the query.

b. Security Requirement

i. Users shall be required to log in to the system for their own reservation information and modification with e-mail address and password.

ii. The system shall permit only authorize member who are on the list of authorize menu managers to do administrator's task.

iii. The system shall permit customer to view only their own previously placed orders, not orders placed by other customers.

c. Software Quality Attribute

i. **Availability:** The system shall be available to users on the computer 99.9% of the time. And should be available to administrator all the time.

ii. **Robustness:** If the connection between the user and the system is broken prior to an order being either confirmed or canceled, the system shall enable the user to recover an incomplete reservation.

iii. **Consistency:** If you want you can book ticket any time and the consistency should also hold among user view.

3.3 Validation

I. Validation in customers Details:

a. In the customer form you will fill all the section because all the section is important for the admin for ticket booking.

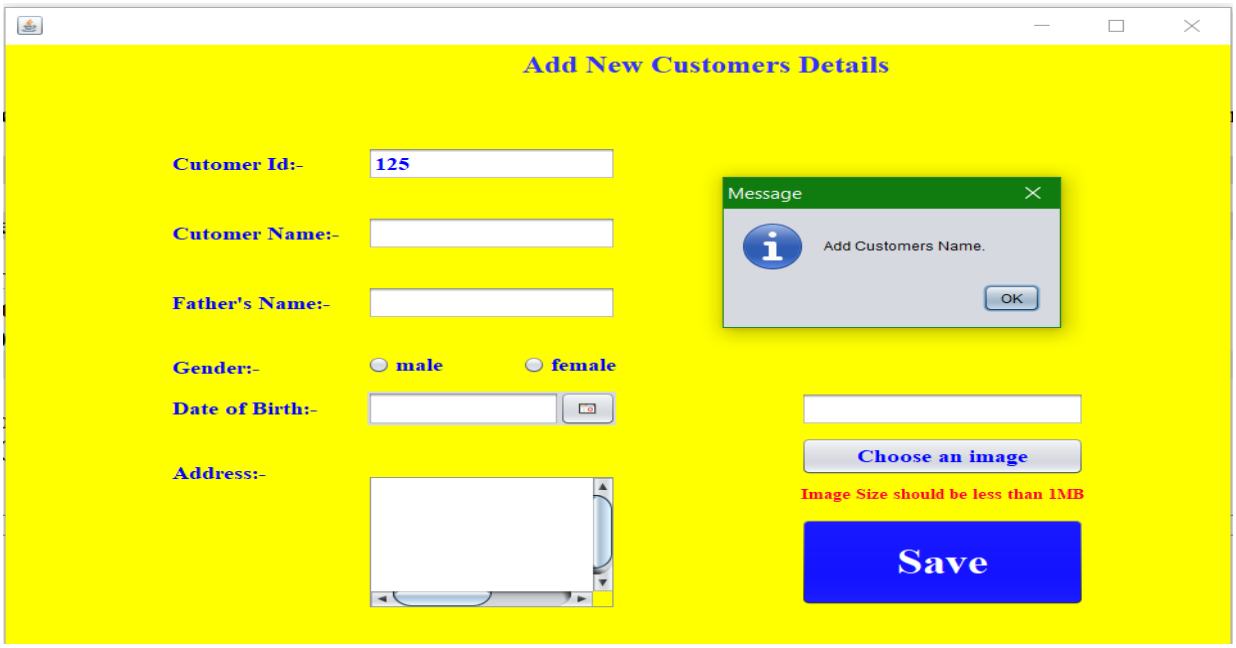


Fig No: 3.4.1

II. Validation in flights Details:

a. In the Flight details you will fill all the section because all the section important for flight if you give the wrong information than it is very effected for all Passenger.

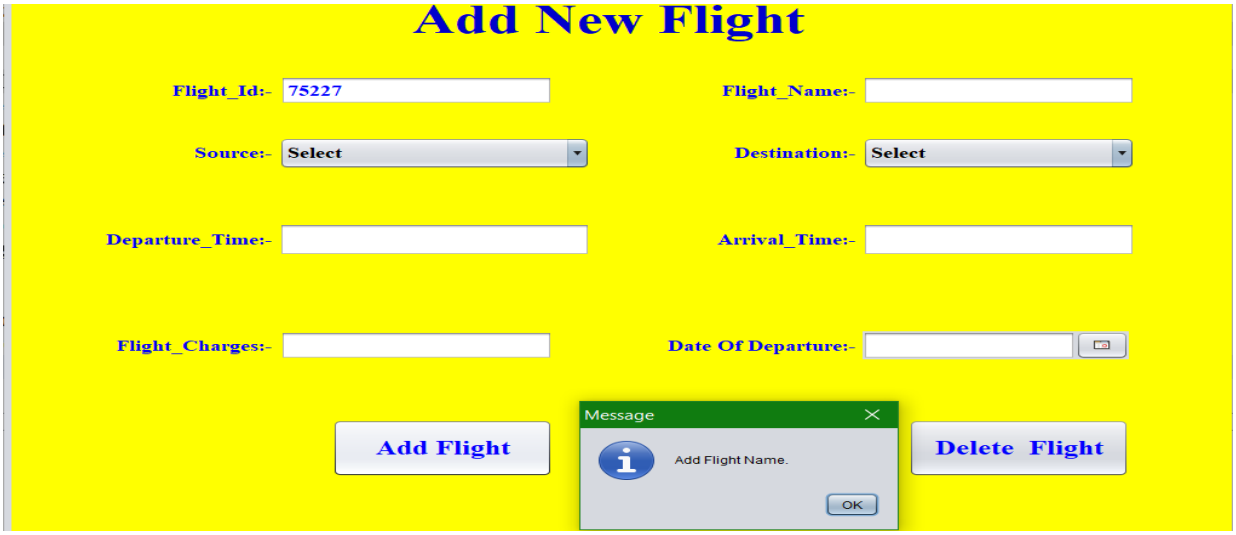


Fig No: 3.4.2

3.4 Expected Hurdles

i. We can't upload the GUI application on the server.

ii. It used in only one windows laptop in which these applications are made.

iii. It is very easy to understand all the details of application's pages and easy to fill the data i.e. mentioned in the application form.

3.5 SDLC Model

Software Development Life Cycle (SDLC) is a process used by the software industry to design, develop and test high quality software. The SDLC aims to produce a high-quality software that meets or exceeds customer expectations, reaches completion within times and cost estimates.

The life cycle defines a methodology for improving the quality of software and the overall development process.

i. SDLC is the acronym of Software Development Life Cycle.

ii. It is also called as Software Development Process.

iii. SDLC is a framework defining tasks performed at each step in the software development process.

iv. ISO/IEC 12207 is an international standard for software life-cycle processes.

This image is describing the Software Development Life Cycle of this project:

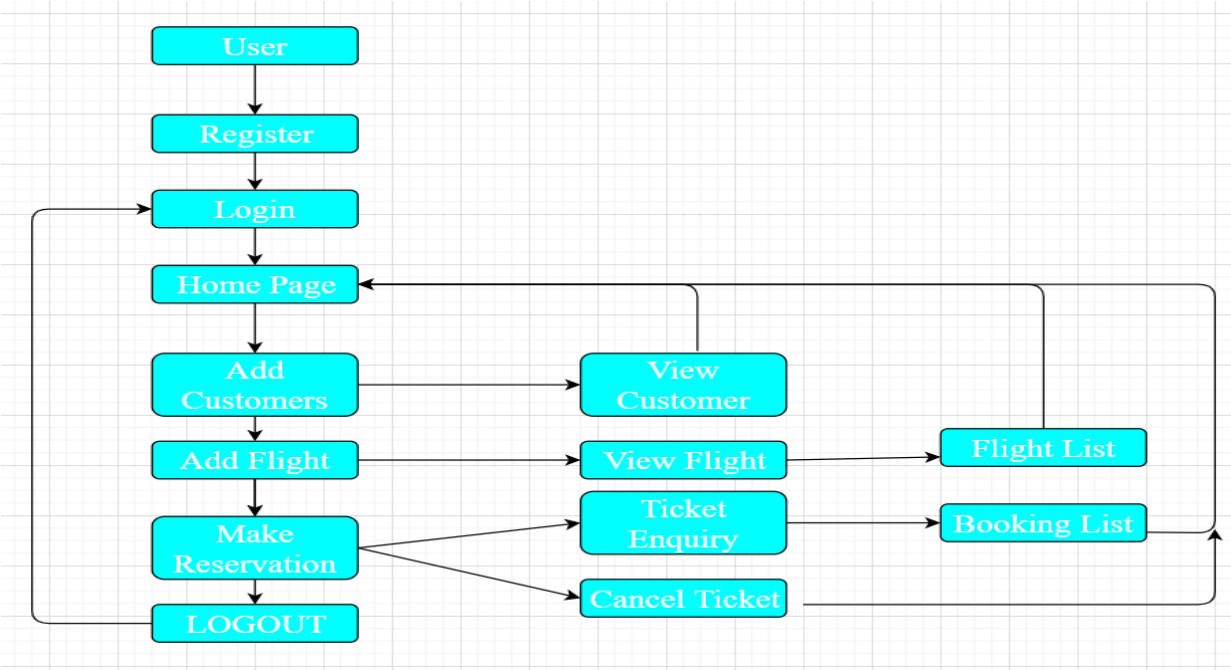


Fig No:3.6.1 SDLC Diagram OF BOOKING

4. System Design

4.1 User Requirement

The system entitled “Airline Reservation System” is GUI application, which aims is to make the reservation of the customer. Admin update the details of flights and customers. It is developed by using Java (Swing), MYSQL related database.

4.2 Flow Chart

A flowchart is a type of diagram that represents an algorithm, workflow or process, showing the steps as boxes of various kinds, and their order by connecting them with arrows. This diagrammatic representation illustrates a solution model to a given problem. Flowcharts are used in analyzing, designing, documenting or managing a process or program in various fields.

This flowchart explains the overall working of the Airline reservation system. The flow of application commences with user registration since every customer need to get himself registered for accessing the application.

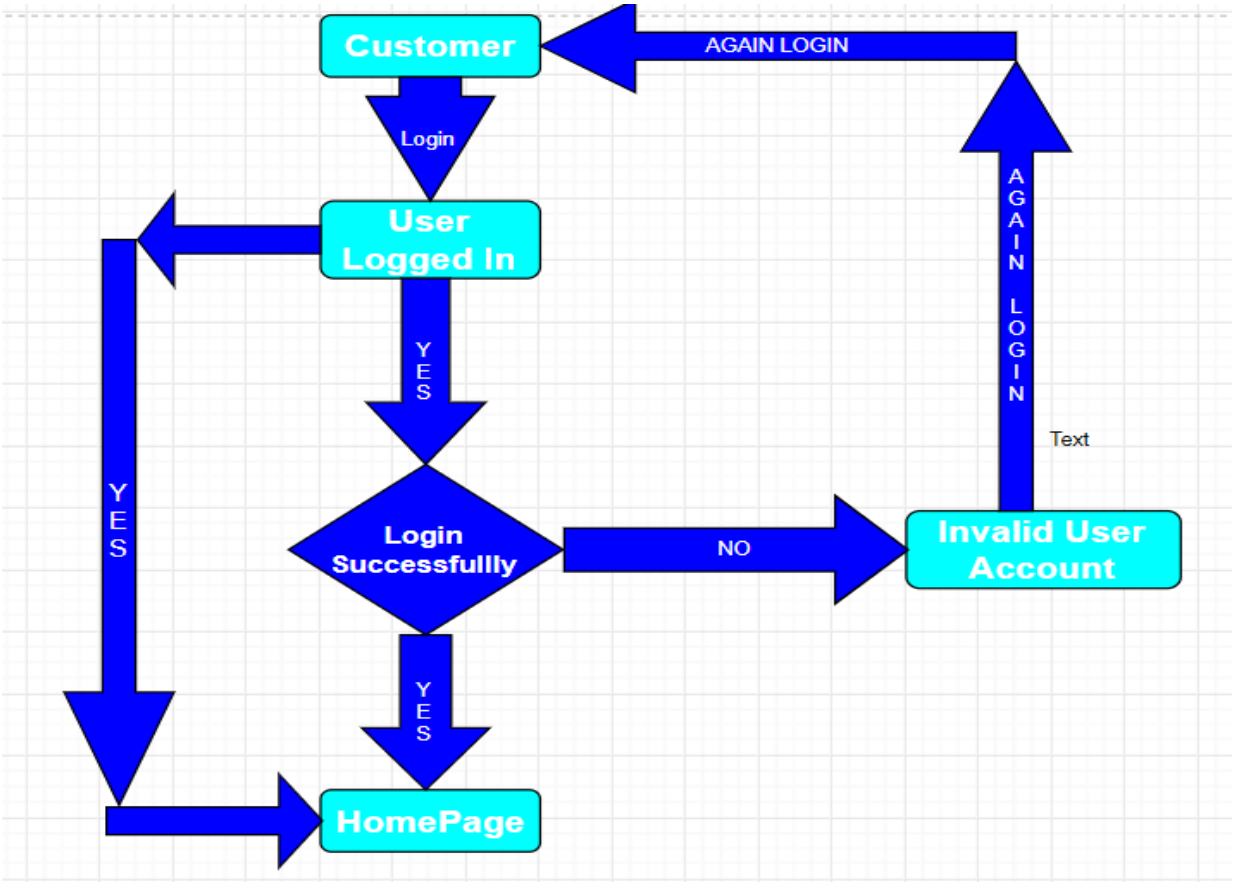


Fig No: 4.2.1 Login activity Flow Chart

Figure 4.2.1 shows the airline reservation system activities flow chart which explains first of all user have to login in application if user is authorized then he/she will be redirected to recent booking activity and then start the booking of flight if the user is not authorized then again login window will appear and the student have a login with the registered username, password and user type.

4.3 Data Flow Diagram

A data flow diagram (DFD) is a graphical representation of the "flow" of data through an information system, modelling its process aspects. A Data Flow Diagrams is a structured analysis and design tool that can be used for flowcharting in place of, or in association with, information-oriented and process-oriented systems flowcharts A DFD is often used as a preliminary step to create an overview of the system, which can later be elaborated. DFD's can also be used for the visualization of data processing (structured design).

A DFD shows what kind of information will be input to and output from the system, where the data will come from and go to, and where the data will be stored. It does not show information about the timing of process or information about whether processes will operate in sequence or in parallel.

We usually begin withdrawing a context diagram, a simple representation of the whole system. To elaborate further from that, we drill down to a level 0 diagram with additional information about the major functions of the system. This could continue to evolve to become a level 1 diagram when further analysis is required. Progression to level 2, 4 and so on is possible but anything beyond level 3 is not very common. Please bear in mind that the level of detail asked for depends on your process change plan. Following DFD of given application is:

Zero Level Data Flow Diagram of Airline Reservation System Application

This is the zero level DFD of the Airline Reservation System application, where elaborate the high-level process of Booking. it is a Basic overview of the of the whole booking process.

It should be easily understood by a wide audience including add customer, add flight, booking, cancel ticket and see all flight list in Zero level of the DFD of the chat application.

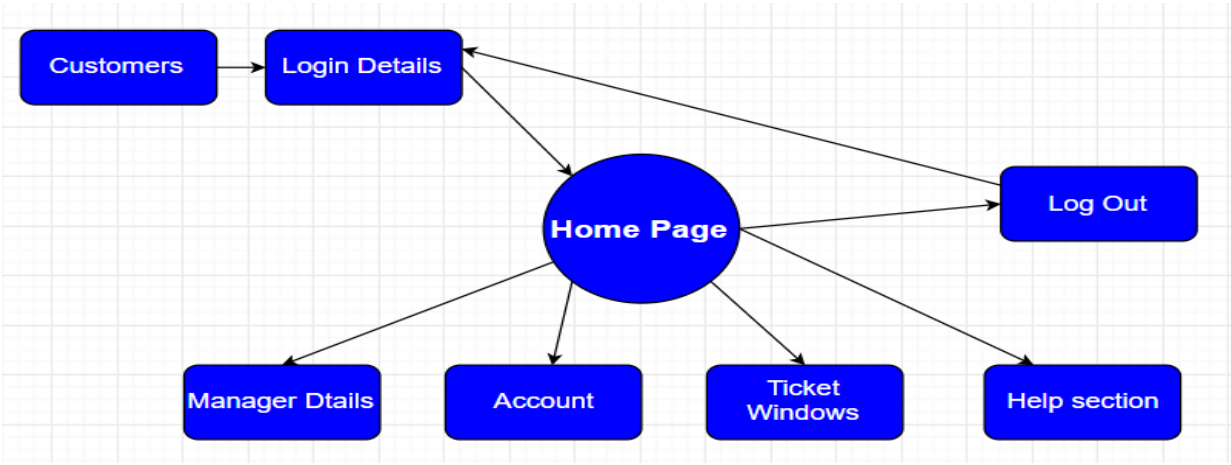


Fig No: 4.3.1 Zero Level DFD- Airline Reservation System

First Level Data Flow Diagram of Airline Reservation System Application

First Level DFD of the Airline Reservation System Application shows how the System is divided into subsystem, each of which deal with the one or more of the data flows to or from an External agent, and which together provide all of the functionality of the Airline Reservation System application.

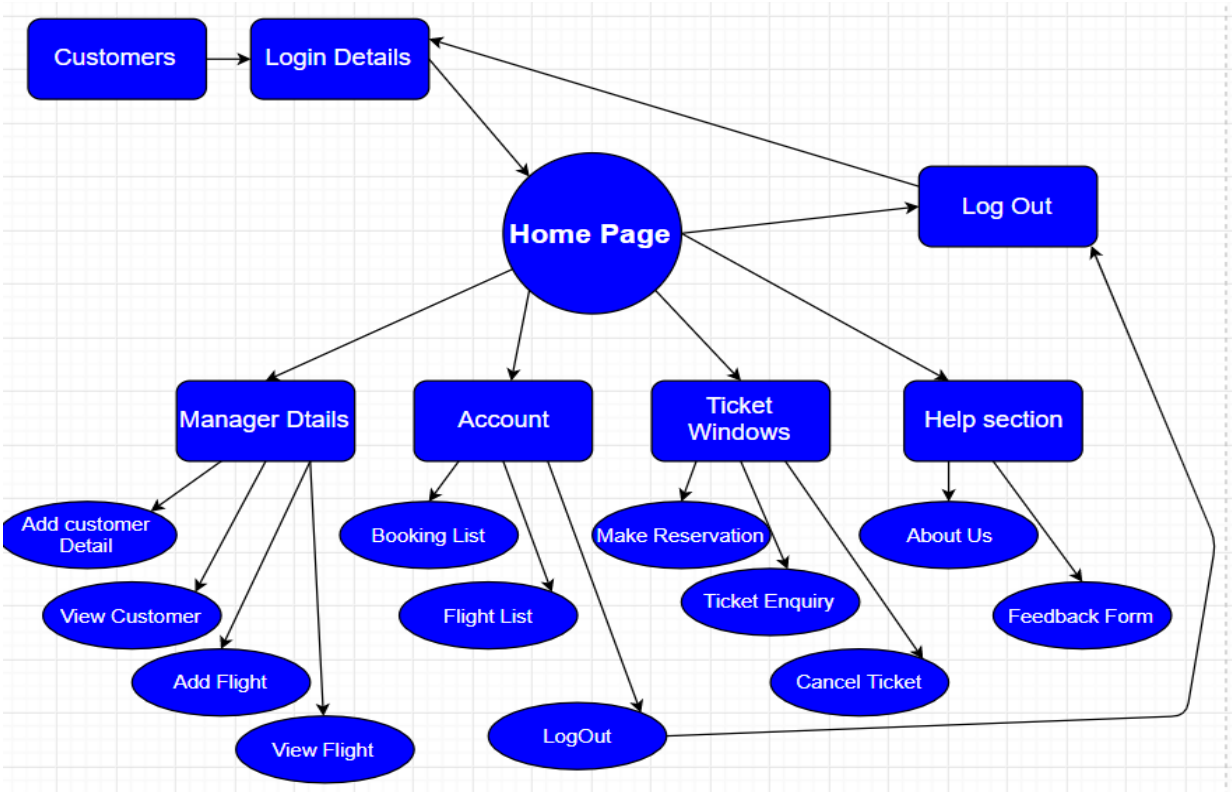


Fig No: 4.3.2 First Level DFD- Airline Reservation System

4.4 Database Design

A database is a separate application that stores a collection of data. Each database has one or more distinct APIs for creating, accessing, managing, searching, and replicating the data it holds. Other kinds of data stores can be used, such as files on the file system or large hash tables in memory but data fetching and writing would not be so fast and easy with those type of systems. So now a days we use relational database management systems (RDBMS) to store and manager huge volume of data.

This is called relational database because all the data is stored into different tables and relations are established using primary keys or other keys known as foreign keys.

A Relational Database Management System (RDBMS) is a software that:

i. Enables you to implement a database with tables, columns, and indexes.

ii. Guarantees the Referential Integrity between rows of various tables.

iii. Updates the indexes automatically.

iv. Interprets an SQL query and combines information from various tables.

a) RDBMS Terminology:

Before I proceed to explain MySQL Database system, let's revise few definitions related to database.

i. **Database -** A database is a collection of tables, with related data.

ii. **Table -** A table is a matrix with data. A table in a database looks like a simple spreadsheet.

iii. **Column -** One column (data element) contains data of one and the same kind, for example the column postcode.

iv. **Row -** A row (= tuple, entry or record) is a group of related data, for example the data of one subscription.

v. **Redundancy** - Storing data twice, redundantly to make the system faster.

vi. **Primary Key -** A primary key is unique. A key value cannot occur twice in one table. With a key you can find at most one row.

vii. **Foreign Key -** A foreign key is the linking pin between two tables.

4.5 TABLE STRUCTURE

Figure 4.5.1 shows the list of database tables. There is one database which name is “project” And database contain 6 tables as like:

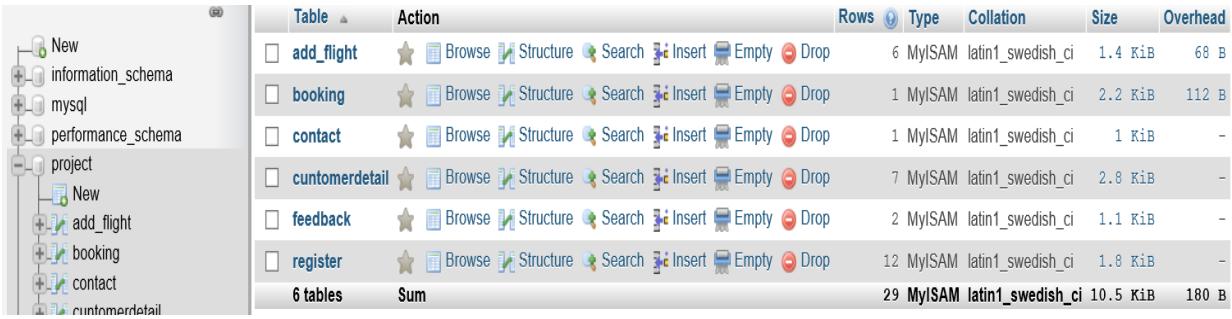


Table	Action	Rows	Type	Collation	Size	Overhead
add_flight	Browse Structure Search Insert Empty Drop	6	MyISAM	latin1_swedish_ci	1.4 KiB	68 B
booking	Browse Structure Search Insert Empty Drop	1	MyISAM	latin1_swedish_ci	2.2 KiB	112 B
contact	Browse Structure Search Insert Empty Drop	1	MyISAM	latin1_swedish_ci	1 KiB	-
cuntomerdetail	Browse Structure Search Insert Empty Drop	7	MyISAM	latin1_swedish_ci	2.8 KiB	-
feedback	Browse Structure Search Insert Empty Drop	2	MyISAM	latin1_swedish_ci	1.1 KiB	-
register	Browse Structure Search Insert Empty Drop	12	MyISAM	latin1_swedish_ci	1.8 KiB	-
6 tables	Sum	29	MyISAM	latin1_swedish_ci	10.5 KiB	180 B

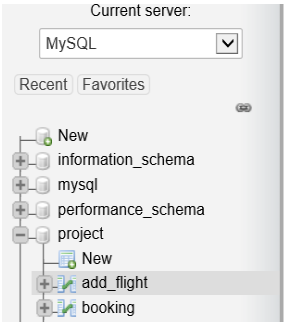
Figure 4.5.1 Database Design

Figure 4.5.2 shows the list of databases table. There is a table which name is “Register table” in which has 9 rows in this table.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	F_NAME	varchar(15)	latin1_swedish_ci		No	None			Change Drop More
2	L_NAME	varchar(15)	latin1_swedish_ci		No	None			Change Drop More
3	E_MAIL	varchar(50)	latin1_swedish_ci		No	None			Change Drop More
4	U_NAME	varchar(10)	latin1_swedish_ci		No	None			Change Drop More
5	PAS	varchar(10)	latin1_swedish_ci		No	None			Change Drop More
6	RE_PAS	varchar(10)	latin1_swedish_ci		No	None			Change Drop More
7	DOB	date			Yes	NULL			Change Drop More
8	ADDRESS	text	latin1_swedish_ci		No	None			Change Drop More
9	Usertype	varchar(25)	latin1_swedish_ci		No	None			Change Drop More

Figure 4.5.2 Rows of Registered Table

Figure 4.5.3 shows the list of databases table. There is a table which name is “Flight table” in which has 8 rows in this table.



#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	F_ID	int(10)			No	None			Change Drop More
2	F_NAME	varchar(150)	latin1_swedish_ci		No	None			Change Drop More
3	SOURCE	varchar(20)	latin1_swedish_ci		No	None			Change Drop More
4	DESTINATION	varchar(25)	latin1_swedish_ci		No	None			Change Drop More
5	DEPARTURE	time			No	None			Change Drop More
6	ARRIVAL	time			No	None			Change Drop More
7	F_CHARGE	double			No	None			Change Drop More
8	DOD	date			Yes	NULL			Change Drop More

Figure 4.5.3 Rows of flight Table

Figure 4.5.4 shows the list of databases table. There is a table which name is "Booking table" in which has 12 rows in this table.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	T_NO	int(10)			No	None			Change Drop More
2	CUST_ID	int(10)			No	None			Change Drop More
3	CUST_NAME	varchar(25)	latin1_swedish_ci		No	None			Change Drop More
4	ADDRESS	varchar(150)	latin1_swedish_ci		No	None			Change Drop More
5	F_ID	int(15)			No	None			Change Drop More
6	F_NAME	varchar(50)	latin1_swedish_ci		No	None			Change Drop More
7	DOJ	date			No	None			Change Drop More
8	SOURCE	varchar(25)	latin1_swedish_ci		No	None			Change Drop More
9	DESTINATION	varchar(25)	latin1_swedish_ci		No	None			Change Drop More
10	F_CHAREG	double			No	None			Change Drop More
11	F_CLASS	varchar(25)	latin1_swedish_ci		No	None			Change Drop More
12	SEATS	varchar(10)	latin1_swedish_ci		No	None			Change Drop More

Figure 4.5.4 Rows of booking Table

Figure 4.5.5 shows the list of databases table. There is a table which name is "Feedback table" in which has 7 rows in this table.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	CUSTID	int(15)			No	None			Change Drop More
2	CUSTNAME	varchar(50)	latin1_swedish_ci		No	None			Change Drop More
3	FID	int(15)			No	None			Change Drop More
4	Mobile_Number	bigint(10)			No	None			Change Drop More
5	SOURCE	varchar(50)	latin1_swedish_ci		No	None			Change Drop More
6	DATE	date			No	None			Change Drop More
7	QUERY	text	latin1_swedish_ci		No	None			Change Drop More

Figure 4.5.5 Rows of Feedback Table

Figure 4.5.6 shows the list of databases table. There is a table which name is "Contact table" in which has 2 rows in this table.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	SELECT_REGION	varchar(120)	latin1_swedish_ci		No	None			Change Drop More
2	SELECT_STATION	varchar(50)	latin1_swedish_ci		No	None			Change Drop More

Figure 4.5.6 Rows of Contact Table

5. Testing:

Testing a program consists of providing the program with a set of test inputs (or test cases) and observing if the program behaves as expected.

If the program fails to behave as expected, then the conditions under which failure occurs are noted for later debugging and correction.

Some commonly used terms associated with testing are:

i. **Failure:** This is a manifestation of an error (or defect or bug). But, the mere presence of an error may not necessarily lead to a failure.

ii. **Test case:** This is the triplet [I.S.O], where I is the data input to the system, S is the state of the system at which the data is input, and O is the expected output of the system.

iii. **Test suite:** This is the set of all test cases with which a given software product is to be tested.

In this project we followed unit testing. Each and every module is tested individually.

Table No:9 Test case 1: Register and Log-in

Input	Desire Output	Actual Output	Status
Enter the Register details.	Specific activity page should be displayed based on the accessibility criteria.	Specific recent message page is displayed based on the accessibility criteria.	Pass
Enter the Login details.	Specific activity page should be displayed based on the accessibility criteria.	Specific recent message Username, password give then next step is continuing.	Pass

Table No:10 Table case 2: All process of this application

Input	Desire Output	Actual Output	Status
In home Page go to and click on manager detail.	Add the customer with details.	View the customer details with its ID.	Pass
Click on Add flight Details.	Add the all Flight Details with his details.	View the customer details with its flight ID.	Pass
Click on Ticket Window and click on make reservation.	Make the reservation with all detail which fetch from the database.	Click the book and reservation is successful.	Pass
Click on the ticket enquiry.	Give the customer id of the customer which had given at the adding of customer detail.	All the data fetch from database with the help of id.	pass
Click on the cancel ticket.	Give the ticket number i.e. given at the booking time.	Click on search which is fetch all data of those ticket number and click on cancel.	Pass
Click On booking List.	Click on “see booking List”.	All the data fetch from database.	Pass
Click on Flight List.	Select the source and destination and click on show button.	All flight details fetch from database.	Pass
Click on about us and contact us.	If you click on contact us than you collect my number and click on about us than you will know about me.	All details about me.	Pass

If you click on Log Out button.	Click on log out.	Go to the Login page of this application.	Pass
Click on help Button.	After click on help button you will find a form.	Give the customer id and click on search and again give the flight id click on search write your query and click on send.	Pass
If you click on Facebook, Instagram, Twitter and Google.	After click you just leave the application page.	Web page is open on the Google.	Pass

6. Development and Implementation

6.1 Introduction to Languages

Java (Swing)

Swing, a part of Java Federation Classes (JFC) is the next generation GUI toolkit that allows us to develop large scale enterprise applications in Java. It is a set of classes which provides many powerful and flexible components for creating graphical user interface.

Earlier, the concept of Swing did not exist in Java and the user interfaces were built by using the Java's original GUI system, AWT. Because of the limitations of the AWT, Swing was introduced in 1997 by the Sun Microsystems. It provides new and improved components that enhance the look and functionality of GUIs.

With Java 1.1, Swing was used as a separate library. However, it was fully integrated into Java with the start of Java 1.2. So, user working with Java 1.2 can easily work with Swing.

Features of Java (Swing)

a) **Platform Independent:** It is platform independent, the swing components that are used to build the program are not platform specific. It can be used at any platform and anywhere.

b) **Lightweight:** Swing components are lightweight which helps in creating the UI lighter. Swings component allows it to plug into the operating system user interface framework that includes the mappings for screens or device and other user interactions like key press and mouse movements.

c) **Plugging:** It has a powerful component that can be extended to provide the support for the user interface that helps in good look and feel to the application. It refers to the highly modular-based architecture that allows it to plug into other customized implementations and framework for user interfaces. Its components are imported through a package called java. Swing.

d) **Manageable:** It is easy to manage and configure. Its mechanism and composition pattern allow changing the settings at run time as well. The uniform changes can be provided to the user interface without doing any changes to application code.

e) **MVC:** They mainly follows the concept of MVC that is Model View Controller. With the help of this, we can do the changes in one component without impacting or touching other components. It is known as loosely coupled architecture as well.

f) **Customizable:** Swing controls can be easily customized. It can be changed and the visual appearance of the swing component application is independent of its internal representation.

MySQL Database

MySQL is a fast, easy-to-use RDBMS being used for many small and big businesses. MySQLi Extension is developed, marketed, and supported by MySQL. MySQL is becoming so popular because of many good reasons −

i. MySQL is released under an open-source license. So, you have nothing to pay to use it.

ii. MySQL is a very powerful program in its own right. It handles a large subset of the functionality of the most expensive and powerful database packages.

iii. MySQL uses a standard form of the well-known SQL data language.

iv. MySQL works on many operating systems and with many languages including PHP, PERL, C, C++, JAVA, etc.

v. MySQL works very quickly and works well even with large data sets.

vi. MySQL is very friendly to PHP, the most appreciated language for web development.

vii. MySQL supports large databases, up to 50 million rows or more in a table. The default file size limit for a table is 4GB, but you can increase this (if your operating system can handle it) to a theoretical limit of 8 million terabytes (TB).

viii. MySQL is customizable. The open-source GPL license allows programmers to modify the MySQL software to fit their own specific environments.

WAMP

Stands for "Windows, Apache, MySQL, and PHP." WAMP is a variation of WAMP for Windows systems and is often installed as a software bundle (Apache, MySQL, and PHP). It is often used for web development and internal testing.

The most important part of the WAMP package is Apache (or "Apache HTTP Server") which is used run the web server within Windows. By running a local Apache web server on a Windows machine, a web developer can test webpages in a web browser without publishing them live on the Internet.

WAMP also includes MySQL and PHP, which are two of the most common technologies used for creating dynamic websites. MySQL is a high-speed database, while PHP is a scripting language that can be used to access data from the database. By installing these two components

locally, a developer can build and test a dynamic website before publishing it to a public web server.

While Apache, MySQL, and PHP are open source components that can be installed individually, they are usually installed together. One popular package is called "WampServer," which provides a user-friendly way to install and configure the "AMP" components on Windows.

NOTE: The "P" in WAMP can also stand for either Perl or Python, which are other scripting languages. The Mac version of WAMP is known as MAMP.
Components of wamp 3.7 for windows:

a) Java (Swing) 8

b) MySQL 5.6.20

c) Wamp 10.3.14

6.2 **Implementation with Screenshots**

Following are some snapshots of the project which shows the UI of the project and helps to understand the project and how it is working.

- **Figure 6.2.1** shows the Ui view of start page of the application which displayed user click on the application after that click on the “Sign In” or “Sign Up” Option then go to the register page and if you have already an account than you can click on thee Sign In option.

Fig No: 6.2.1 Start Page of the Application

❖ **Figure 6.2.2** shows the UI view of registration page of the application in which the user enters the user name, email id, password and then click on the register button for the registration.

Fig No: 6.2.2 Registration Page of the Application

❖ **Figure 6.2.3** shows the UI view of login page of the application in which the user enter username, email id, password and then click on the Login button for the Login.

Fig No: 6.2.3 Login Page of the Application

❖ **Figure 6.2.4** shows the UI view of user account information page of the application in which the activity Manager details, Ticket Enquiry, Account, about us, Contact Us and Help of the Home Page of the Project.

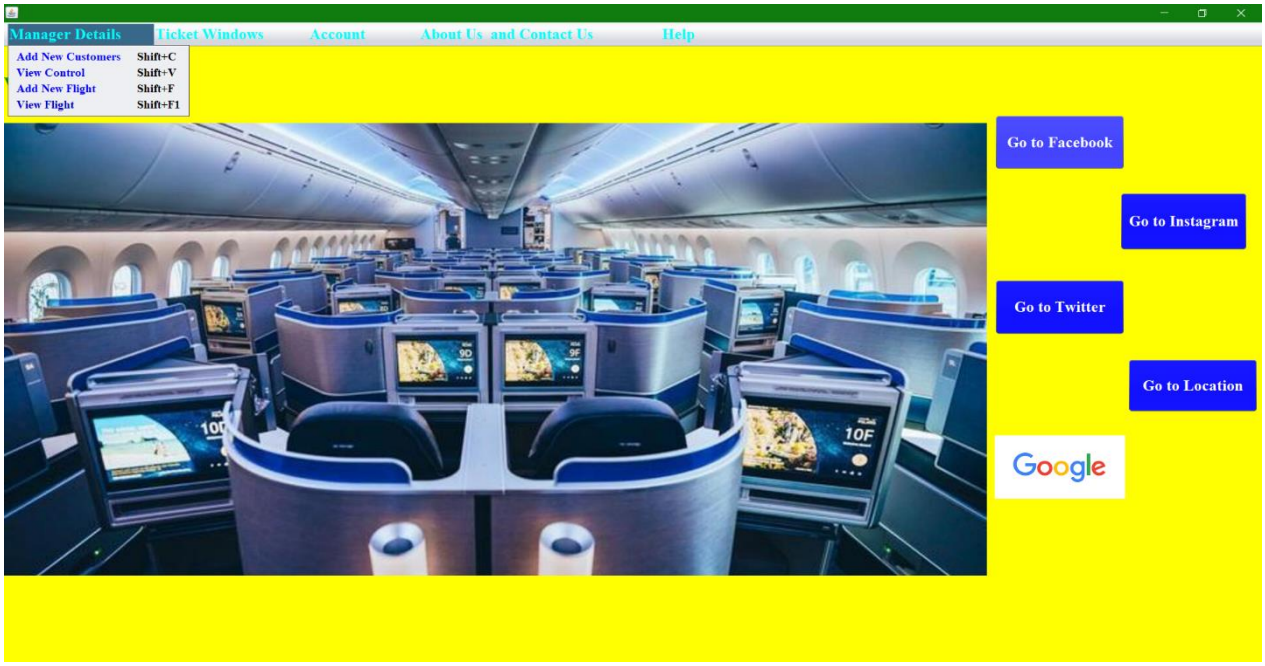


Fig No: 6.2.4 Home Page of the Application

❖ **Figure 6.2.5** shows the UI view of add new customer page of the Airline reservation system.

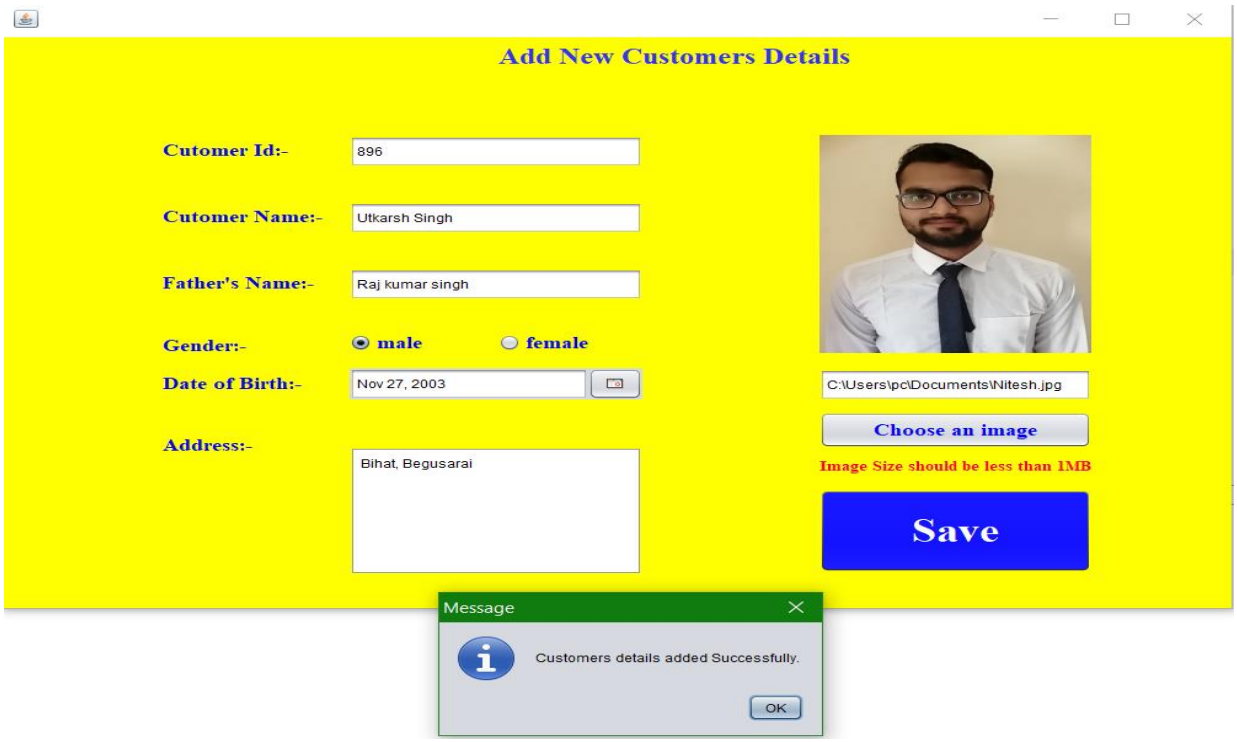


Fig No: 6.2.5 Add new Customer Page of the Application

❖ **Figure 6.2.6 shows** the UI view of the application page which shows the all customer details with the help of customer Id.

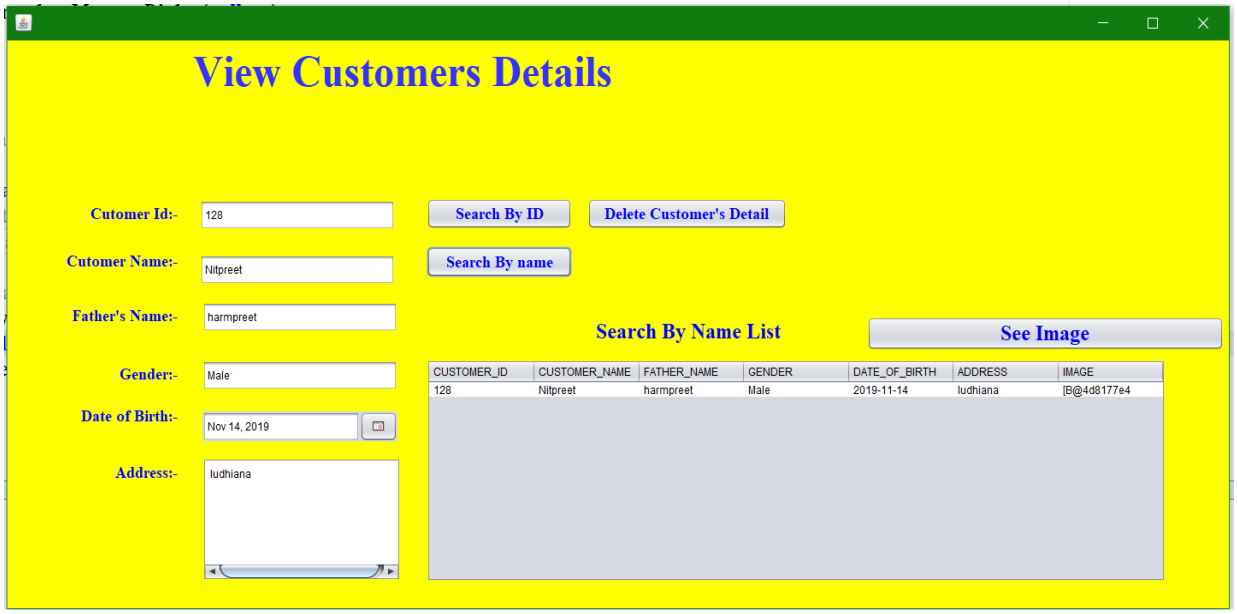


Fig No: 6.2.6 View Customer Page of the Application

❖ **Figure 6.2.7** shows the UI view of the application page in which you can add flight details with their id.

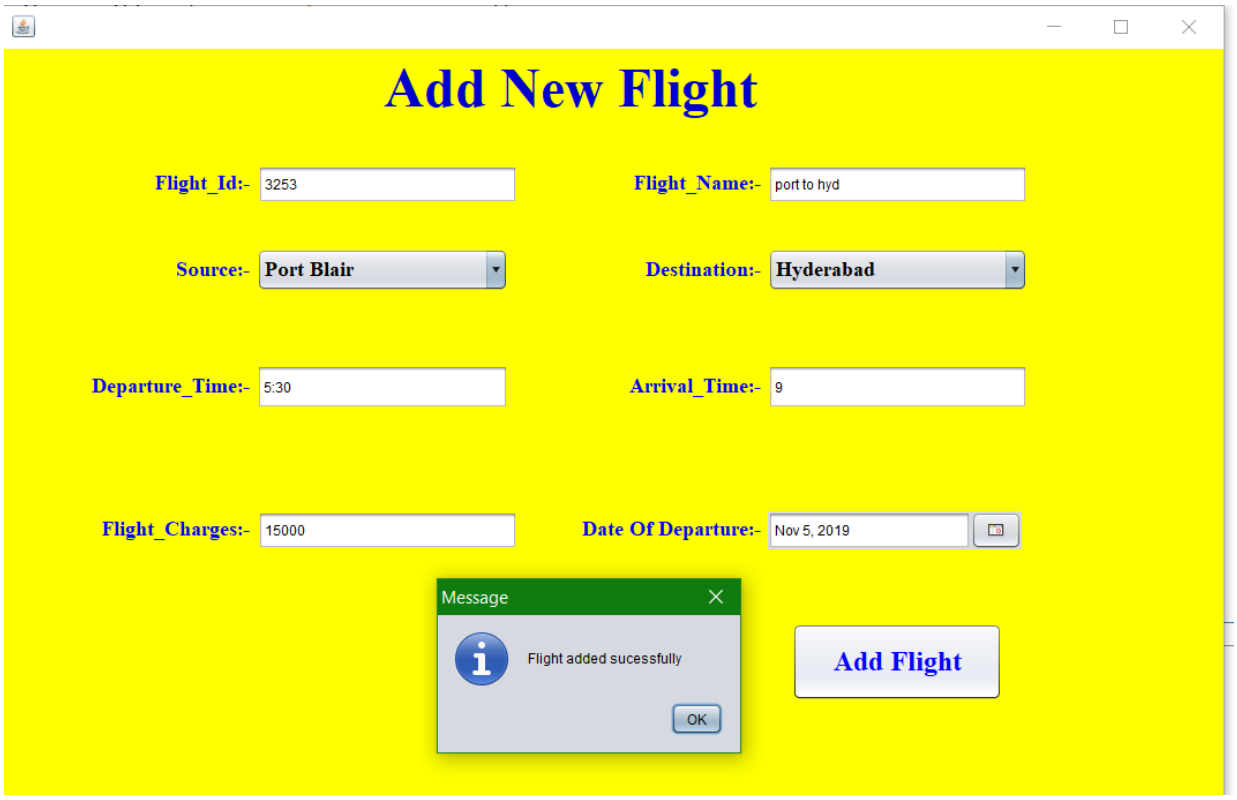


Fig No: 6.2.7 Add Flight Details Page of the Application

❖ **Figure 6.2.8 shows** the UI view of the application page which shows the all flight details with the help of flight Id.

Fig No: 6.2.8 View Flight Details Page of the Application

❖ **Figure 6.2.9** shows the UI view of the application page which shows to make reservation of flight with the help of ticket number, customer id and flight id.

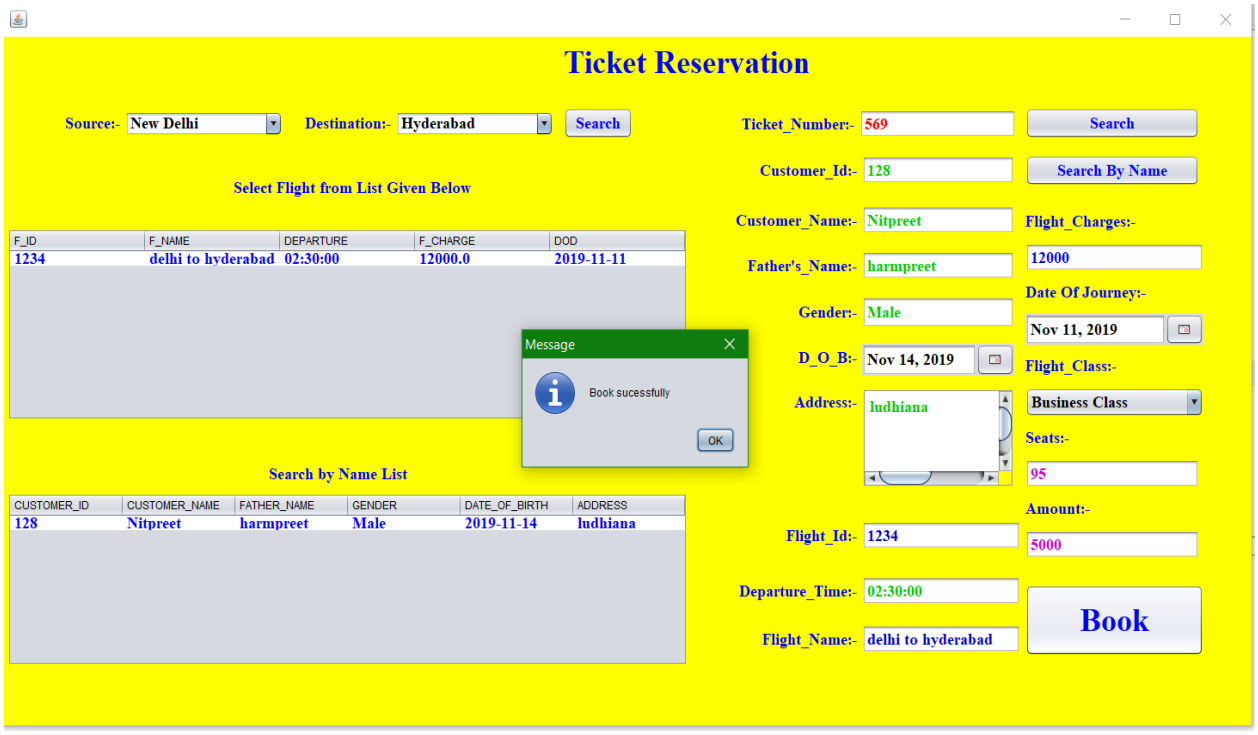


Fig No: 6.2.9 Make Reservation Page of the Application

❖ **Figure 6.2.10** shows the UI view of the application page in which you can enquiry the ticket with the help of ticket number.

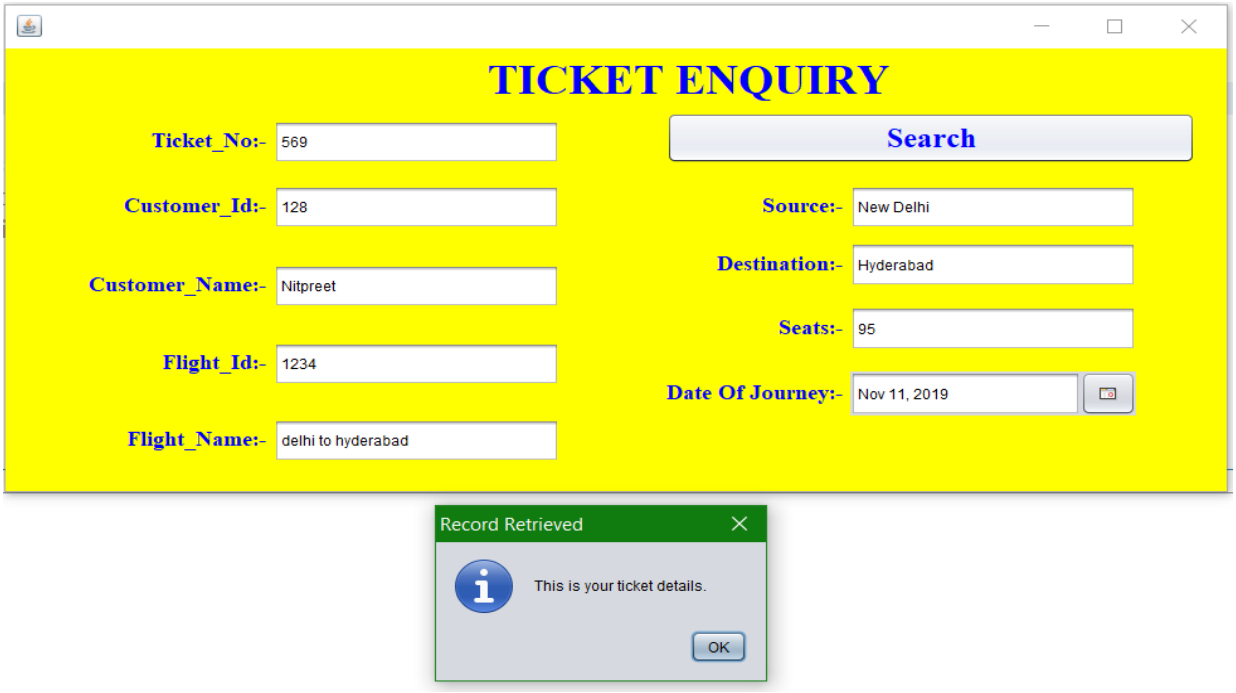


Fig No: 6.2.10 Ticket enquiry Page of the Application

❖ **Figure 6.2.11** shows the UI view of the application page in which you cancel the ticket with the help of customer ID.

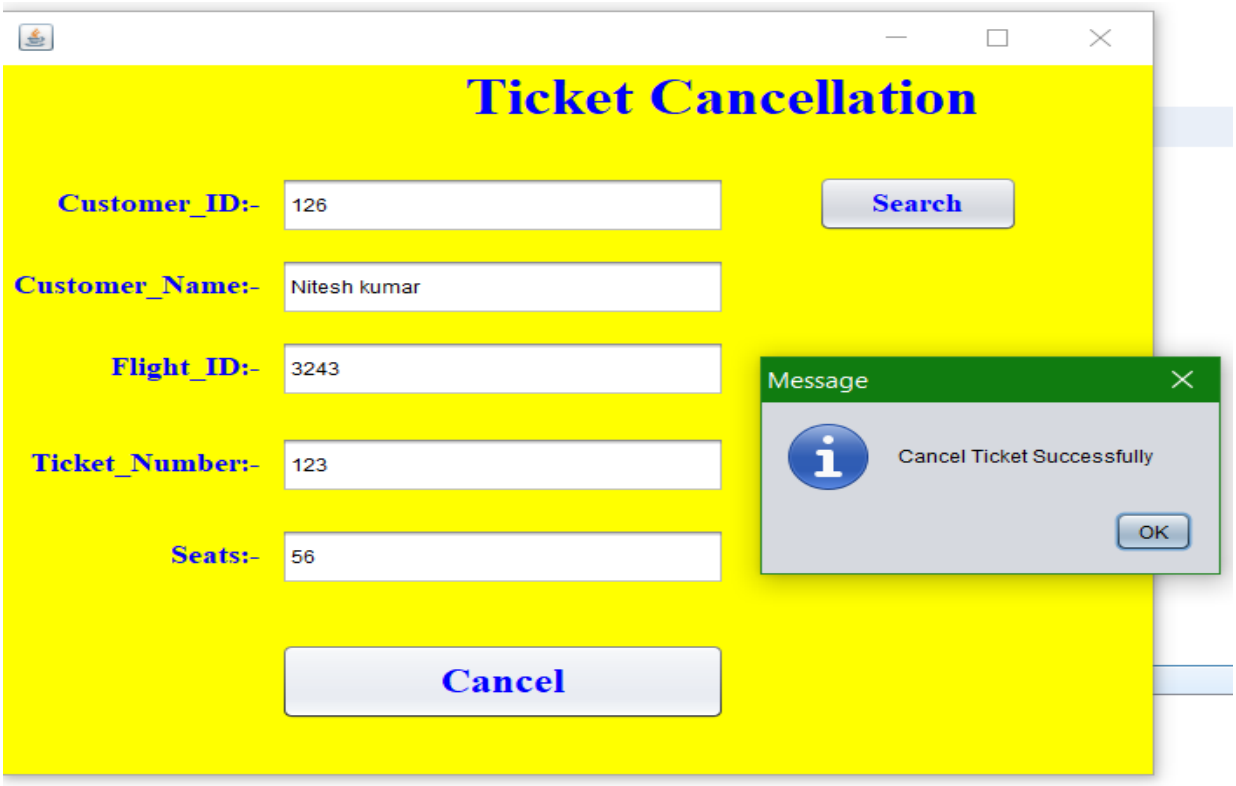


Fig No: 6.2.11 Cancel Reservation Page of the Application

❖ **Figure 6.2.12** shows the UI view of the application page in which booking list of the all reservation of flights.

Fig No: 6.2.12 Reservation Page of the Application

❖ **Figure 6.2.13** shows the UI view of the application page in which flight List are shown below.

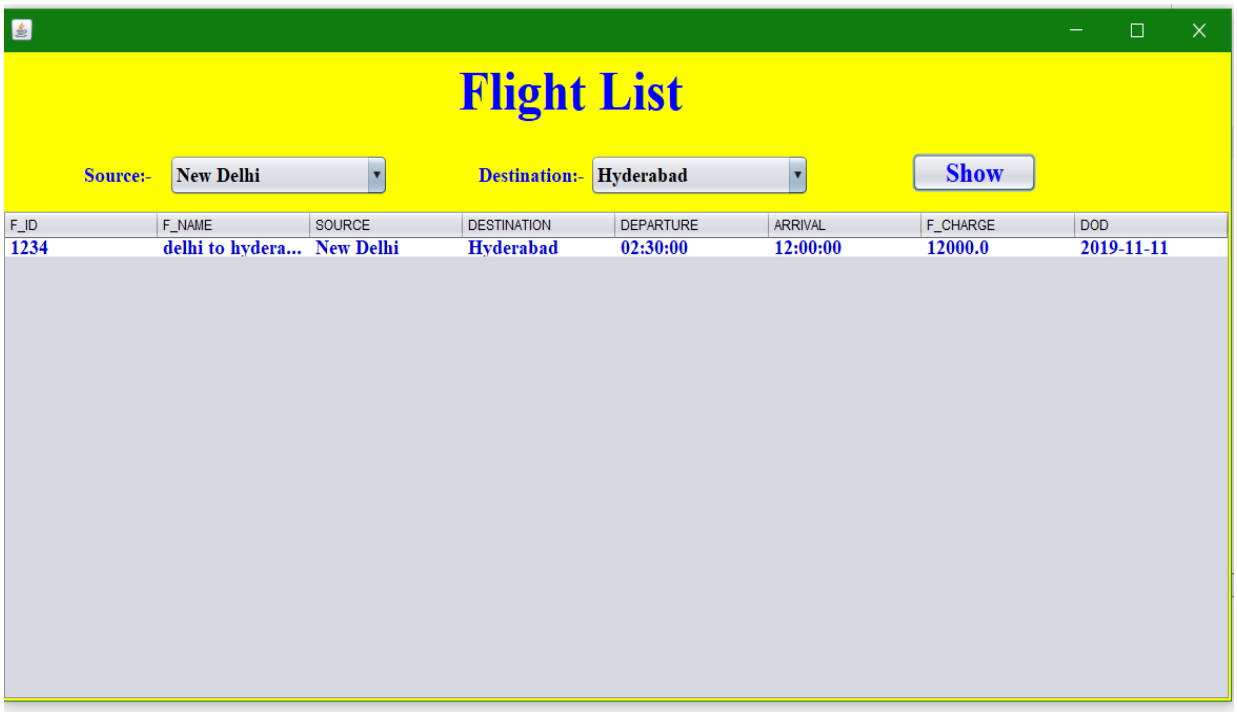


Fig No: 6.2.13 Flight List Page of the Application

❖ **Figure 6.2.14** shows the UI view of the application page in which about us of the project is shown below.

Fig No: 6.2.14 About page of the Application

❖ **Figure 6.2.15** shows the UI view of the application page in which contact us of the project is shown below.

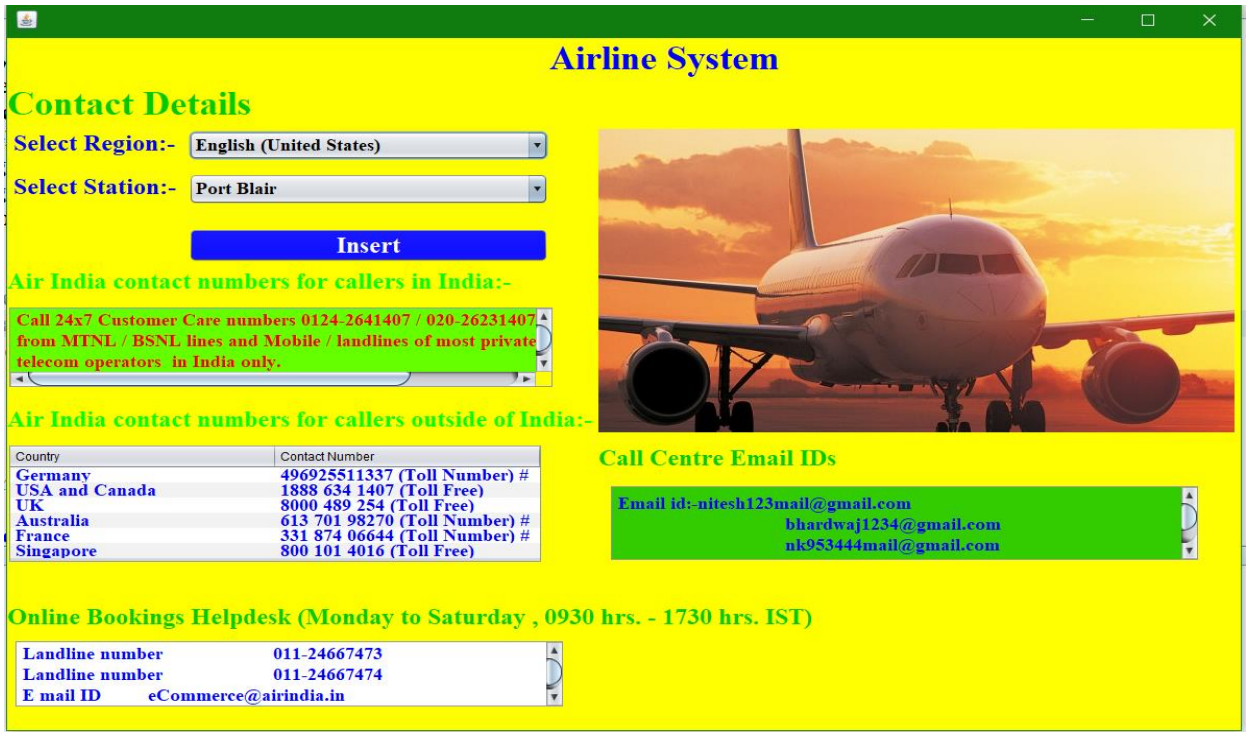


Fig No: 6.2.15 Contact Page of the Application

❖ **Figure 6.2.16** shows the UI view of the application page in which Feedback page of the application.

Fig No: 6.2.16 Feedback Page of the Application

❖ **Figure 6.2.17** shows the UI view of the application page in which Logout page of the application.

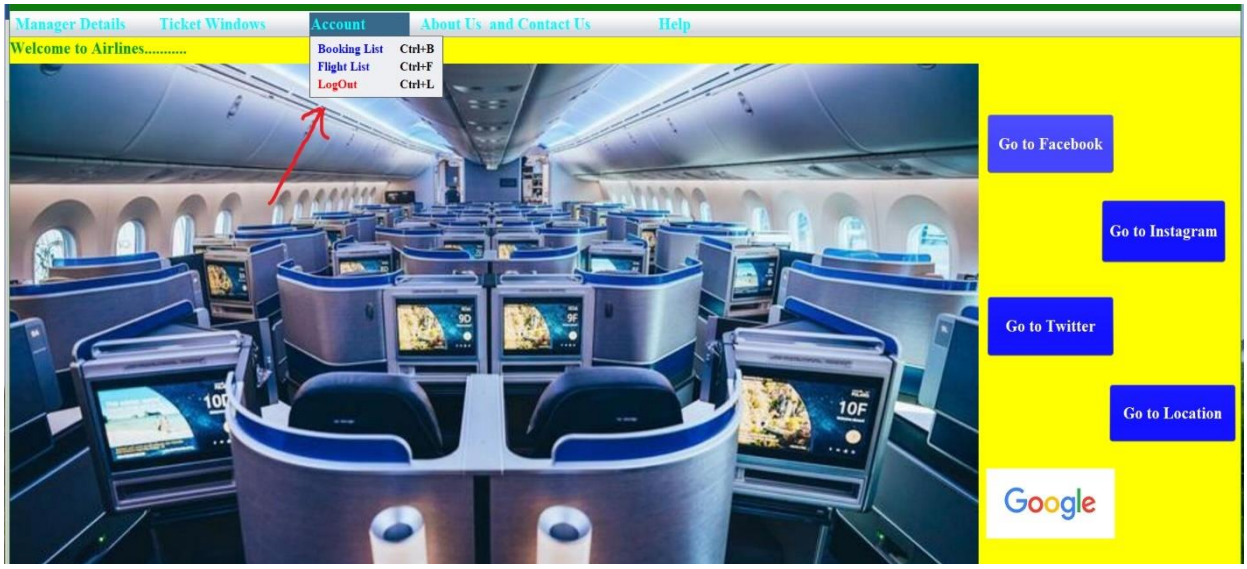


Fig No: 6.2.17 Log Out Page of the Application

7. Test Plan

- It is used to perfectly working application and to finding errors in programming.

1. Software testing:

- It is process of testing the functionality and correctness of software by running it.

A good test case is the one that has high probability of finding as yet undiscovered errors.

2. Black box testing:

- It is not a type of testing. It instead is a testing strategy which doesn't need any knowledge of internal design or code.

3. White box testing:

- It is a security testing method that can be used to validate whether code implementation follow indented design.

a) Levels of testing:

1. Unit testing:

- It verifies the functionality of a specific section of code, usually at the function level. The various testing modules are selecting type measuring mileage, invalid amount etc.

2. Component interface testing:

- It is used to check the handling of data passed b/w various units, or subsystem components.

3. System testing:

i. It tests a completely integrated system to verify that expenses are created successfully and reports are submitted.

ii. System Testing is the process of executing the program with the intention of finding out errors.

iii. During testing, the program to be tested is executed with a set of test cases and the output of the programs for the test case is evaluated to determine if the program is performing as it is expected to be.

iv. The success of testing in revealing errors in program depends critically on the test cases. In software system the use of testing is not limited to the testing phase. The results of testing are used later on during maintenance also.

v. During testing a test suite can be used to see that modification doesn't have any undesirable effect.

vi. The basic levels of testing are: Unit testing, Integration testing, System testing, Acceptance testing.

vii. These different levels of testing attempt to detect different types of faults.

Levels of Testing

i. **Unit Testing:** The level of testing is called unit testing. In this, different modules are tested against the specifications produced during design for the modules. Unit testing is essential for verification of the code produced during the coding phase, and hence the goal is set to test the internal logic of the modules.

ii. **Integration Testing**: The next level of testing is often called the integration testing. In this, many tested modules are combined into subsystems, which are then tested. The goal here is to see if the modules can be integrated properly, the emphasis being on testing interface between modules. This testing activity can be considered as testing design, and hence the emphasis on testing modules interactions.

iii. **Acceptance Testing**: It is sometimes performed with realistic data of the client to demonstrate that the software is working properly. Testing here focuses on the external behavior of the system.

IMPLEMENTATION

Steps of implementation are:

a) First load NetBeans and wamp in system.

b) Make a software. In this s/w The Airline Reservation system is stored.

c) First make all form.

d) Make main menu. Join every form with Main menu.

e) Main menu open. It shows all forms heading.

f) Now choose what form will be open then click.

g) If Reservation form is open then it shows Id and ticket number.

h) After put various value we click save bottom.

i) It automatically goes to report.

CONCLUSION

The Airline reservation system has been a way of minimizing the clerical work, which is almost a routine and consumes the most precious time.

This AIRLINE RESERVATION SYSTEM has been an attempt to help the user to minimizing his workload along with minimizing the paper works and saving of time.

The system has been developed in a way to make it very user friendly. It provides an on-line message and error detection and error messages every time the user needs. Any person having a little bit of window based can run this system without any pain.

Almost all the difficulties of manual reservation have been removed by this system. We welcome all the suggestion and other improvements, Which the system needs so that it covers all needs if the user in user way.

REFERENCES

The following references have been used by me, during all the phases of the "Airline Reservation System" project:

i. https://www.javatpoint.com/java-swing

Java Swing tutorial is a part of Java Foundation Classes (JFC) that is *used to create window-based applications*. It is built on the top of AWT (Abstract Windowing Toolkit) API and entirely written in java.

ii. Localhost Link:

http://localhost/phpmyadmin/sql.php?server=1&db=project&table=add_flight&pos=0

Install and start the **WAMP server**. Make sure that your **wamp** icon is green, if it's not green then it's not active. Open any browser and type **localhost** or 127.0. 0.1 on the address bar and you will see your **WAMP server** dashboard page.

iii. MySQL Link:

a. https://www.mysql.com/downloads/

MySQL is an open-source relational database management system. Its name is a combination of "My", the name of co-founder Michael Widenius's daughter, and "SQL", the abbreviation for Structured Query Language.

iv. https://netbeans.org/downloads/8.0.1/

NetBeans is an integrated development environment for Java. NetBeans allows applications to be developed from a set of modular software components called modules. NetBeans runs on Windows, macOS, Linux and Solaris.

v. Wamp server download:

https://sourceforge.net/projects/wampserver/

WampServer refers to a software stack for the Microsoft Windows operating system, created by Romain Bourdon and consisting of the Apache web server, OpenSSL for SSL support, MySQL database and PHP programming language.

vi. You tube Link:

a. https://www.youtube.com/watch?v=H28jJYk0Xds