VISVESVARAYA TECHNOLOGICAL UNIVERSITY

"Jnana Sangama", Belgaum-590 014, Karnataka.

A
Mini Project Report
On

"AIRLINE RESERVATION SYSTEM"

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

BACHELOR OF ENGINEERING
IN
INFORMATION SCIENCE AND ENGINEERING

Submitted by

PRAVEEN R 1EW20IS056
PRAJWAL L H 1EW20IS055

Under the Guidance of

Ms. Likhitha S
Asst. Prof
Dept. of ISE.

DEPARTMENT OF INFORMATION SCIENCE AND ENGINEERING

EAST WEST INSTITUTE OF TECHNOLOGY

BANGALORE - 560 091
2022-2023

EAST WEST INSTITUTE OF TECHNOLOGY

Sy. No.63,Off. Magadi Road, Vishwaneedam Post, Bangalore - 560 091
(Affiliated To Visvesvaraya Technological University, Belgaum)

CERTIFICATE

This is to certify that the mini project work entitled "Airline Reservation System" presented by **PRAVEEN R(1EW20IS056) and PRAJWAL L H(1EW20IS055),** bonafied students of **EAST WEST INSTITUTE OF TECHNOLOGY**, Bangalore in partial fulfillment for the award of **Bachelor of Engineering** in **Information Science and Engineering** of the **Vishweshwaraiah Technological University**, Belgaum during the year **2022-2023**. It is certified all corrections/suggestions indicated have been in corporated in the report. The mini Project has been approved as it satisfies the academic requirements in respect of mini projectPrescribed for the said degree.

Signature of Guide	Signature of HOD	Signature of Principal
Ms.Likhitha S	**Dr Suresh M B**	**Dr. K Channakeshavalu**
Asst.Prof, Dept. of ISE	Prof & Head, Dept. of ISE	Principal
EWIT, Bangalore	EWIT, Bangalore	EWIT, Bangalore

External Viva

Name of the Examiners	**Signature with Date**
1. _____	**1.** _____
2. _____	**2.** _____

ACKNOWLEDGEMENT

Any achievement, be it scholastic or otherwise does not depend solely on the individual efforts but on the guidance, encouragement and cooperation of intellectuals, elders and friends. A number of Personalities , in their own capacities have helped me in carrying out this mini project work. We Would like to take this opportunity to thank all.

First and foremost we would like to thank **Dr. K Channakeshavalu**, Principal, EWIT, Bangalore,for his moral support towards completing our mini project work.

We would like to thank, **Dr. Suresh M B,** Professor and Head of Department of ISE, EWIT, Bangalore, for his valuable suggestions and expert advice.

We deeply express our sincere gratitude to our guide **Ms.Likhitha S** .Asst.Prof , Department of ISE EWIT, Bangalore for her able guidance throughout the mini project work and guiding us to organize the report in a systematic manner. We thank my Parents, and all the Faculty members of Department of Information science & Engineering for theirconstant support and encouragement.

Last, but not the least, we would like to thank our peers and friends who provided us with valuablesuggestions to improve our mini project.

PRAVEEN R [1EW20IS056]

PRAJWAL L H [1EW20IS055]

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 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, 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, allow new customers to sign up for online access.

The system allows the airline passenger to search for flights that available between the two travel cities, namely the “Departure city” and “Arrival city For a” particular departure and arrival dates. The system displays all 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.

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

1.1 Introduction to File Structures

File Structures is the Organization of Data in Secondary Storage Device in Such a way that minimizes the access time and the storage space.

File Structure: It is a combination of

- Representation for data in files and
- The operations for accessing the data

It allows applications to read write and modify data search .An improvement in file Structure design make an application hundred times faster .To access the data faster From the storage disk we use file structure .Disks are slow which are used to pack Thousands of megabytes of data.

Records and its types:

Are collection of fields, possibly of different Data types, typically in fixed number of sequence. The fields are also be called members. For example, a date could be sorted as a record containing a numeric year field, a Month field represented as a string, and a numeric day of month field. A record are distinguished from arrays by the fact that the number of fields is typically fixed, each field has a name, and that each field may have a different types. A record type is a data type that describes such values and variables. The Definition includes specifying the data type of each field and an identifiers by which it can be accessed.

Why we need File Structure??

As we know without proper structure of organizing is ordinary system, it will generate some types of problems. So avoiding this kind of problems we go for File Structure. The File structure means its tell about how the system will stored and access the record from memory and also its tell how the disk are performed its tasks and also how to speed up the execution of transferring data. Data processing from a computer science prospective:

1.1.1 Field Structures:

Fixed Length Fields: The method to organize fields is by limiting the maximum size of each field. The advantage in this method is that since the size of each field is fixed, the entire field can be read at once.

i) Length Indicator Fields: The length of each field is specified as a prefix to actualdata

Delimited Fields: Any special character which is not a part of actual data can be used as separator.

Field Count: The number of the fields in each record is fixed.

1) Index Structure for Records: An index is a collection of key field and reference field
2) Key Field: Key Field is a member of record which can uniquely identify the record.
3) Reference Field: Reference Field contains the value that points to the address of the corresponding record in the file.

In an airline reservation system, there are various data structures used to organize and store information. Two important types of structures are field structures and record structures. Let's explore each of them in the context of an airline reservation system.

- Reservation ID: A unique identifier assigned to each reservation.
- Passenger Name: The name of the passenger who made the reservation.
- Flight Number: The unique identifier for a specific flight.
- Departure City: The city from which the flight departs.
- Arrival City: The city to which the flight arrives.
- Departure Date: The date of the flight's departure.
- Arrival Date: The date of the flight's arrival.
- Seat Number: The assigned seat for the passenger.
- Ticket Price: The cost of the ticket for the reservation.

These fields can vary depending on the complexity and requirements of the airline reservation system.

1.1.2 Record Structures:

A record structure in an airline reservation system refers to the way information is organized and stored for each individual reservation. It represents a complete set of data related to a single reservation made by a passenger. The record structure groups relevant fieldstogether to provide a cohesive representation of the reservation.

For example, a typical record structure in an airline reservation system may include fields suchas reservation ID, passenger name, flight number, departure city, arrival city, departure date, arrival date, seat number, ticket price, passenger contact information, payment details, and special requests. Each field within the record structure holds specific information related to the reservation, such as identifying details, flight information, pricing, and passenger preferences.

These record structures are typically stored in a database or file system, allowing for efficient retrieval, manipulation, and management of reservation data. The record structure plays a crucial role in maintaining the integrity and accuracy of reservation information, enabling airline staff to manage bookings, modify reservations, process payments, and provide necessary services to passengers.

By organizing the data within a reservation into a structured record structure, the airline reservation system can effectively store and manage the vast amount of information associated with each reservation, facilitating smooth operations and ensuring a seamless experience for both passengers and airline personnel.

CHAPTER 2

Introduction of Airline Reservation System

The project aims to design an Airline Reservation System application which enables the customers to search and book flight . Airline Reservations Systems contain fare tariffs, passenger reservations and ticket records.

2.1 Overview of the project

The Airline Reservation System project is an implementation of a general Airline Ticketing website like Orbitz, which helps the customers to search the availability of flights, book and cancel the flight tickets. This project also covers adding, deleting or modifying the customer details and flights. In general, this website would be designed to perform like any other airline ticketing website available online.

An airline reservation system is a software solution that enables airlines to manage flight bookings, reservations, and related operations. It provides a platform for passengers to search for flights, make reservations, and complete secure payment transactions. Additionally, the system streamlines administrative tasks for airline staff, such as managing flight schedules,seat availability, and passenger information.

The key components and functionalities of an airline reservation system typically include:

1. Flight Search and Booking: Passengers can search for flights based on various criteria such as departure city, arrival city, dates, and passenger count. The system displays available flights,along with their schedules, fares, and seat availability. Passengers can select flights and complete the booking process by providing passenger details and making secure payments.

2. Seat Selection: The system allows passengers to choose their seats during the booking process. It provides visual seat maps indicating available and occupied seats, allowing passengers to select their preferred seats or upgrade for additional comfort.

3. Reservation Management: The system handles the management of reservations, including storing passenger information, flight details, and payment records. It allows passengers to view and modify their reservations, change flight dates, upgrade seats, or cancel bookings.

Payment Processing: Integration with secure payment gateways enables passengers to make online payments for their reservations. The system securely processes payment transactions, verifies payment details, and generates payment confirmation records.

4. Flight Status and Notifications: The system provides real-time flight status updates, informing passengers about any changes or delays in their flights. It sends notifications to passengers via email, SMS, or mobile apps, keeping them informed about their travel arrangements.

Administrative Tools: The system includes administrative interfaces for airline staff to manage flights, flight schedules, seat availability,

2.2 Purpose of this project:

The purpose of this project is to implement or to design a database for an airline reservation system to check the flight details,book and cancel flight tickets. It makes the process of booking and cancelling flight tickets simple and easy for the passengers.

The purpose of an airline reservation system project is to develop a robust and efficient software solution that enables airlines to manage flight bookings, reservations, and related operations effectively. The project aims to provide a reliable and user-friendly platform for passengers to search for flights, make reservations, and facilitate secure payment transactions. Additionally, the project seeks to streamline administrative tasks for airline staff, such as managing flight schedules, seat availability, and passenger information.

The primary purposes of an airline reservation system project can be summarized as follows:

1. Improve Customer Experience: The project aims to enhance the customer experience by providing a user-friendly and intuitive platform for searching and booking flights. Passengers should be able to easily access flight information, view available seats, select preferred options,and complete reservations with confidence.

2. Simplify Reservation Management: The project aims to simplify the process of managing reservations for airline staff. This includes functionalities such as seat allocation, updating flight schedules, handling modifications or cancellations, and ensuring accurate record-keeping of passenger details and payment information.

Enable Secure Payment Processing: The project focuses on integrating secure payment gateways to ensure safe and reliable payment processing for ticket purchases. This involves implementing industry-standard security measures and encryption protocols to protect sensitive payment information and prevent fraudulent activities.

3. Enhance Operational Efficiency: The project seeks to improve the overall operational efficiency of airlines by automating processes, reducing manual effort, and minimizing errors.

This includes features such as automated seat assignment, real-time flight status updates, integration with external systems (e.g., payment gateways, flight tracking services), and generating comprehensive reports for analysis and decision-making.

4. Support Business Growth and Scalability: The project is designed to support the growth of an airline's business by providing a scalable and flexible reservation system. As the airline expands its operations and passenger base, the system should be capable of handling increased user load, managing larger databases, and accommodating new features or integrations.

By achieving these purposes, an airline reservation system project can contribute to a streamlined and efficient airline operation, improved customer satisfaction, and better overall business performance.

2.3 Problem definition:

Normally a person wants to reserve his ticket and he has to contact at nearest Overseas Travels branch. The Airline Reservation System provides an interface to schedule flights and reservations for an airline through internet. Its responsibility is to keep track of system users, customers, Airbus information, flight information and cancellation.

During the course of an airline reservation system project, various challenges and problems may arise. Here are some common problems that can occur:

1. Requirements Changes: Changes in requirements or scope can occur throughout the project,leading to additional work, delays, and potential conflicts with the original plan.

Poorlymanaged requirements changes can result in scope creep and impact project timelines

and resources.

2. Technical Challenges: Complex technical requirements, integration with external systems, or scalability issues can pose significant challenges during the project. Technical difficulties may arise from compatibility issues, performance bottlenecks, or limitations of the chosen

3. Data Quality and Integration: Integrating data from different sources, such as flight schedules, passenger details, and payment information, can be complex. Inconsistent or incomplete data, data migration challenges, and data synchronization issues between systems

4. Security and Privacy Concerns: Ensuring the security and privacy of passenger data,p aymentinformation, and system infrastructure is crucial. Addressing potential vulnerabilities, complying with data protection regulations, and implementing robust security measures require careful planning and thorough testing.

5. System Performance and Scalability: As the number of users and transactions increase, the system's performance and scalability become critical. Inadequate system architecture, inefficient database design, or suboptimal coding practices can result in slow response times, system crashes, or limited scalability, leading to a poor user experience.

6. Stakeholder Management: Involving multiple stakeholders with different expectations and priorities can pose challenges in decision-making, communication, and coordination. Balancing the needs of passengers, airline staff, management, and regulatory authorities requires effective stakeholder management throughout the project.

7. Testing and Quality Assurance: Insufficient testing efforts or inadequate test coverage can lead to undiscovered bugs, functional issues, or poor system reliability. Testing all possible scenarios, including edge cases and exceptional conditions, is essential to ensure a robust and stable reservation system.

8. Time and Resource Constraints: Project timelines and resource availability can impose constraints on the project. Insufficient time for development, testing, or project management tasks, coupled with limited resources, can lead to compromised quality, rushed

9. User Adoption and Training: Introducing a new reservation system requires proper user adoption and training. Resistance to change, inadequate user training, or difficulties in transitioning from legacy systems can impact the successful implementation and utilization of

10. the new system. Regulatory and Compliance Requirements: Airline reservation systems must comply with various regulations, including data protection laws, privacy regulations, and industry standards.

Keeping up with evolving legal requirements and ensuring compliance throughout the project can be challenging.

It is crucial to proactively identify and address these problems during the project to mitigate risks and ensure successful delivery of the airline reservation system. Effective project management, regular communication with stakeholders, continuous monitoring, and risk mitigation strategies can help overcome these challenges.

2.4 Project plan and scope of the project:

Airline Reservation System is one the modifications that were carried out in the Passenger Service System so that the working and availability of Service area can be broadened. On onehand, it helps the customers and on the other, it also makes the life of the airline service companies easier by keeping all the records of the passengers and if there is any change in thefight due to some reason, the passengers are promptly informed. This system is also used by companies to keep track of user preferences of regular travelers so that they can provide better service and give offers to customers.

Planning and scoping an airline reservation system project involves defining the objectives, deliverables, and key activities required to develop and implement the system. Here's an outline of the plan and scope for an airline reservation system project:

1. Project Objectives:

- Develop a user-friendly and efficient airline reservation system.

- Enable passengers to search and book flights.

- Facilitate secure payment processing for ticket purchases.

- Provide administrative tools for managing flights, reservations, and passenger information.

- Ensure system scalability and reliability to handle high volumes of transactions.

- Enhance customer experience by offering additional features such as seat selection and itinerary management.

2. Project Deliverables:

- Functional airline reservation system with a user interface for passengers and administrators.

- Database structure and management system for storing flight, reservation, and passengerinformation.

- Payment gateway integration for secure online transactions.

- Reporting and analytics capabilities to track reservations, revenue, and other relevantmetrics.

The scope of an airline reservation system is quite extensive, encompassing various aspects of managing and facilitating the reservation and booking processes for airlines. Here are some paragraphs describing the scope of an airline reservation system:

1. Reservation Management: One of the primary functions of an airline reservation system is to manage and handle flight reservations. It includes allowing passengers to search for available flights based on their preferences such as destination, date, and class, and enabling them to reserve seats on those flights. The system should also handle seat availability, waitinglists, and seat assignments to ensure efficient management of reservations

Ticketing and Fare Management: The airline reservation system should incorporate ticketing and fare management functionalities. It includes generating e-tickets or boarding

passes for passengers upon successful reservation. The system should also handle fare calculations, apply pricing rules based on factors such as passenger type, seasonality, and promotional offers, and enable seamless integration with payment gateways for ticket purchases.

Flight Scheduling and Inventory Management: An efficient reservation system needs to integrate with flight scheduling and inventory management systems. It should have real-time access to flight schedules, seat availability, and flight information to provide accurate and up-to-date information to passengers during the reservation process. This integration allows for

2. better management of flight capacities, optimizes seat utilization, and ensures smooth operations.

3. Customer Relationship Management (CRM): The scope of an airline reservation system includes incorporating CRM functionalities to enhance customer satisfaction and loyalty. It should store customer profiles, manage frequent flyer programs, and provide personalized services to passengers. This could include maintaining passenger preferences, managing special requests (e.g., meal preferences, wheelchair assistance), and offering loyalty rewards to frequent travelers.

4. Integration with Ancillary Services: The reservation system can be extended to integrate with ancillary services such as baggage handling, in-flight services, and special accommodations. Passengers should have the option to add ancillary services to their reservations, such as extra baggage, seat upgrades, and meal selections. The system should facilitate seamless coordination between various departments to ensure a smooth travel experience for passengers.

5. Reporting and Analytics: An airline reservation system should provide comprehensive reporting and analytics capabilities. It should generate reports on reservation trends, passenger demographics, revenue analysis, and flight performance metrics. These insights can assist airlines in making data-driven decisions, optimizing their operations, and improving overall efficiency.

6. Integration with External Systems: The scope of an airline reservation system may also

involve integration with external systems such as global distribution systems (GDS) and online travel agencies (OTA). This allows for broader distribution of flight inventory and availability through multiple channels, reaching a wider customer base.

In summary, the scope of an airline reservation system covers various aspects of reservation management, ticketing and fare management, flight scheduling and inventory management, customer relationship management, integration with ancillary services, reporting and analytics, and integration with external systems. A comprehensive and well-designed reservation system enhances the efficiency of airline operations, improves customer experience, and supports the overall growth and success of the airline.

CHAPTER 3

IMPLEMENTATION

3.1 Existing system

The existing system is that the passenger must fill up the data manually and must submit it to the reservation counter. It may take a lot of time to process it and to book the flight. Therefore there is wastage of time. Since the data is entered manually, the probability of error or mistakes is high**.**

System analysis for the existing system is a crucial step in understanding its structure, functionalities, and performance. It involves a comprehensive evaluation of the system's components, processes, and interactions to identify strengths, weaknesses, and areas for improvement. This analysis encompasses various aspects, including studying the system's architecture, data flow, user interface, business logic, security measures, and integration capabilities. Additionally, it involves gathering requirements from stakeholders, conducting interviews, and analyzing user feedback to gain insights into user expectations and needs. The system's scalability, reliability, performance, and maintenance requirements are also assessed during the analysis phase. By conducting a thorough system analysis, a solid foundation is established for making informed decisions and recommendations for system enhancements, optimizing efficiency, addressing shortcomings, and aligning the system with business goals and user requirements.

3.2 Drawbacks of existing system

Entering the record:

Entry of each record is done manually each time the record is done manual each time the record is maintained on paper and it maximizes the maintenance of additional files.

Searching the record:

Due to absence of unique identification of a flight, the searching of record takes much time

and increases the time wastage.

Deleting the record:

In the current system the concept of deleting record is tedious

Modifications of record:

If any modification is required it is done directly on the documents being preserved in correspondence to account information.

3.3 Proposed System

To avoid the limitation of current system it's necessary to design and develop a new system which have the following benefit and the existing system.

1. Everything is automated which reduce the risk factor.
2. Flexibility in generating of information
3. Quick retrieved and maintenance of data.
4. Highly accurate
5. User satisfaction

3.4 Advantages of proposed system

The proposed system due to computerized is much faster in reservation process, cancellation process and transactions

Transfer of information from various branches would become easier and faster.

Managing and maintaining data becomes easier and cost effective due to very high amount and reliability of storage space available in the proposed system

Customer services can not only be satisfied but also enhanced to the extent that one can obtain or cancel a reservation from any given time

The hardware and software requirements of an airline reservation system can vary depending on factors such as the scale of operations, user load, and specific technological choices made by the airline. However, here are the general hardware and software requirements for an airline reservation system:

3.5 Hardware Requirements:

The hardware requirements for an airline reservation system can vary depending on the scale and complexity of the system. Here are some paragraphs describing the hardware requirements for an airline reservation system:

1. Servers and Data Storage: An airline reservation system requires robust servers and data storage infrastructure to handle the high volume of reservation requests, store passenger and flight data, and ensure reliable system performance. This includes powerful server machines with sufficient processing power, memory, and storage capacity to handle concurrent user connections and manage the system's databases efficiently. The choice of servers can vary from dedicated on-premises servers to cloud-based solutions, depending on the specific needs and scalability requirements of the airline.

2. Networking Equipment: A reliable and secure network infrastructure is essential for an airline reservation system to handle the communication between various components of the system, such as user interfaces, databases, payment gateways, and external systems. This includes routers, switches, firewalls, and load balancers to ensure seamless connectivity, network security, and efficient data transfer between different parts of the system.

3. Workstations and User Interfaces: The reservation system requires workstations or terminals for airline staff to access and manage the system. These workstations should have sufficient processing power, memory, and network connectivity to handle the reservation system's user interfaces and related applications. Depending on the scale of operations, these workstations can be desktop computers, laptops, or dedicated terminals located at airline ticket counters, customer service desks, and reservation centers.

4. Backup and Redundancy Systems: To ensure data integrity and minimize downtime, an airline reservation system should have backup and redundancy systems in place. This includes redundant servers, data storage, and networking equipment to provide failover and backup capabilities. Regular data backups should be performed to safeguard critical passenger and reservation information in case of system failures or data corruption.

5. Security Infrastructure: Security is of paramount importance for an airline reservation system to protect passenger data, prevent unauthorized access, and ensure secure transactions.

This involves the implementation of robust security measures such as firewalls, intrusion detection systems, encryption mechanisms, and secure socket layer (SSL) certificates to safeguard the system from external threats and cyber-attacks.

6. Integration and Interface Components: Depending on the integration requirements of the airline reservation system with external systems such as global distribution systems (GDS) and payment gateways, additional hardware components may be needed. This can include dedicated hardware interfaces, connectivity modules, and specialized equipment to establish secure and reliable connections with external systems.

It's important to note that the hardware requirements can vary significantly based on factors such as the scale of operations, the number of concurrent users, the geographical distribution of users, and the specific functionalities and integrations required by the airline. Therefore, a thorough analysis of the airline's requirements, scalability considerations, and performance expectations should be conducted to determine the precise hardware requirements for an airline reservation system.

3.6 Software Requirements:

The software requirements for an airline reservation system encompass various components and technologies needed to develop and operate the system effectively. Here are some paragraphs describing the software requirements for an airline reservation system:

1. Operating System: The choice of an operating system depends on the technology stack and preferences of the development team. Common options include Windows, Linux, or Unix-based operating systems. The selected operating system should be stable, secure, and capable of handling the system's expected workload.

2. Web Development Framework: To build the user interfaces and web-based components of the airline reservation system, a suitable web development framework is necessary. Frameworks like Ruby on Rails, Django, or Laravel provide a foundation for creating interactive and dynamic web applications. These frameworks offer features such as database integration, session management, and user authentication to facilitate efficient development.

3. Database Management System: A robust and scalable database management system

(DBMS) is crucial for storing and retrieving passenger data, flight schedules, and other relevant information. Commonly used DBMS options include MySQL, PostgreSQL, Oracle, or Microsoft SQL Server. The selected DBMS should handle high traffic volumes, ensure data integrity, and support efficient querying and indexing.

4. Application Programming Languages: The choice of programming languages depends on the preferences and expertise of the development team. Common languages used for developing airline reservation systems include Java, C#, Python, or PHP. These languages provide the necessary flexibility, performance, and integration capabilities required for building the various system components.

5. Payment Gateway Integration: An essential aspect of an airline reservation system is the integration of secure and reliable payment gateways. This enables passengers to make online payments for their reservations. Integration with payment gateways such as PayPal, Stripe, or Braintree requires adherence to their APIs and software development kits (SDKs).

6. Security Measures: Security is paramount in an airline reservation system to protect passenger data, prevent unauthorized access, and ensure secure transactions. Implementing security measures such as encryption, secure socket layer (SSL) certificates, and secure coding practices is essential. Regular security audits and vulnerability assessments should also be conducted to identify and address potential security risks.

7. Reporting and Analytics: To monitor system performance, track reservations, and generate insightful reports, the airline reservation system may require reporting and analytics tools. These tools can provide data visualization, generate performance metrics, and offer insights into passenger booking patterns and trends. Popular reporting tools include Tableau, Power BI, or custom-built reporting modules.

8. Integration with External Systems: Depending on the airline's requirements, integration with external systems such as global distribution systems (GDS), flight inventory systems, or customer relationship management (CRM) platforms may be necessary. Integration APIs, web services, or messaging protocols should be employed to facilitate seamless data exchange and synchronization.

CHAPTER 4

DESCRIPTION

4.1 What is Airline Reservation System

As the name suggests Airline Reservation System is software that handles the entire booking data of the Flights. It is fully based on the concept of reserving flight tickets for various destinations. Previously the task of handling the tickets at a time was very difficult, so there was a need for software that can handle all Airline Reservation System.

Therefore the Airline Reservation System was designed. After the release of this system, the stress and workload of the employee were absolutely finished. It was also time-wasting for the travelers to book a Ticket previously. But now it hardly takes 10 to 15 minutes to book a ticket wherever the passenger is.

4.2 Features of Airline Reservation System Project in C++

An airline reservation system typically encompasses a range of features to facilitate flight bookings, manage reservations, and provide a seamless experience for passengers and administrators. Here are some common features of an airline reservation system:

An airline reservation system typically encompasses a range of features to facilitate flight bookings, manage reservations, and provide a seamless experience for passengers and administrators. Here are some common features of an airline reservation system:

An airline reservation system offers a range of features designed to streamline and enhance the booking process for passengers and ensure efficient operations for airlines. Here are some paragraphs describing the key features of an airline reservation system:

1. Flight Search and Booking: The system provides a user-friendly interface for passengers to search for flights based on their preferred dates, destinations, and other criteria. It displays available flight options, including flight timings, fares, and seat availability. Passengers can select their desired flights and proceed with the booking process, including seat selection, meal preferences, and any additional services.

2. Online Payment Integration: To facilitate secure and convenient transactions, the system integrates with various payment gateways, enabling passengers to make online payments using credit cards, debit cards, or other preferred payment methods. The integration ensures that payment transactions are encrypted and compliant with industry security standards, providing a seamless and reliable payment experience for passengers.

3. Seat Selection and Cabin Class Options: Passengers can view the seating layout of the aircraft and choose their preferred seats during the booking process. The system provides real-time seat availability and allows passengers to select seats based on their preferences, such as window seats, aisle seats, or seats with extra legroom. Additionally, passengers can select their desired cabin class, such as economy, business, or first class.

4. Baggage Management: The system incorporates features to manage baggage information, including baggage allowances, restrictions, and fees. Passengers can review and select appropriate baggage options based on their travel needs. The system also generates baggage tags and provides guidelines to ensure compliance with airline policies and regulations.

5. Loyalty Program Integration: Airlines often offer loyalty programs to reward frequent flyers. The reservation system integrates with these programs, allowing passengers to accumulate and redeem points for future bookings, upgrades, or other benefits. Passengers can view their loyalty program status, track points, and access exclusive offers through the reservation system.

6. Flight Status and Notifications: Passengers can stay updated on their flight status, including real-time departure and arrival information, gate changes, and delays. The system sends automated notifications to passengers via email or SMS, keeping them informed about any changes or disruptions to their travel plans. This feature helps passengers plan their journeys more efficiently and minimizes inconvenience caused by unexpected events.

7. Itinerary Management: Once the booking is confirmed, the system generates an itinerary for passengers, providing a comprehensive summary of their travel details, including flight timings, layovers, seat numbers, and other relevant information. Passengers can access and manage their itineraries through the system, making changes or requesting additional services as needed.

8. Check-In and Boarding Pass Generation: The reservation system enables passengers to check-in online before their flights. Passengers can enter their required travel information, select their preferred check-in method (e.g., web check-in or mobile check-in), and generate boarding passes. This feature expedites the check-in process at the airport, saving time and improving the overall travel experience.

9. Ancillary Service Management: Airlines offer various ancillary services, such as pre-paid meals, extra baggage, in-flight entertainment, or special assistance. The reservation system allows passengers to add or customize these services during the booking process, ensuring a personalized travel experience tailored to their preferences.

10. Reporting and Analytics: The reservation system provides airlines with comprehensive reporting and analytics capabilities. It generates reports on booking trends, revenue analysis, passenger demographics, and other key performance indicators. These insights help airlines optimize their operations, make informed business decisions, and enhance the overall customer experience.

These features collectively contribute to a seamless and efficient booking experience for passengers while empowering airlines to manage their operations effectively. The specific features and functionalities may vary depending on the airline's requirements, business model, and integration with other systems or third-party services.

4.3 Modules of Airline Reservation System Project in C++

Make Reservation.

- Cancel Reservation.
- Search Passenger.
- Change Reservation.
- Display Passengers.
- Quit.

CHAPTER 5

Data Flow diagram :

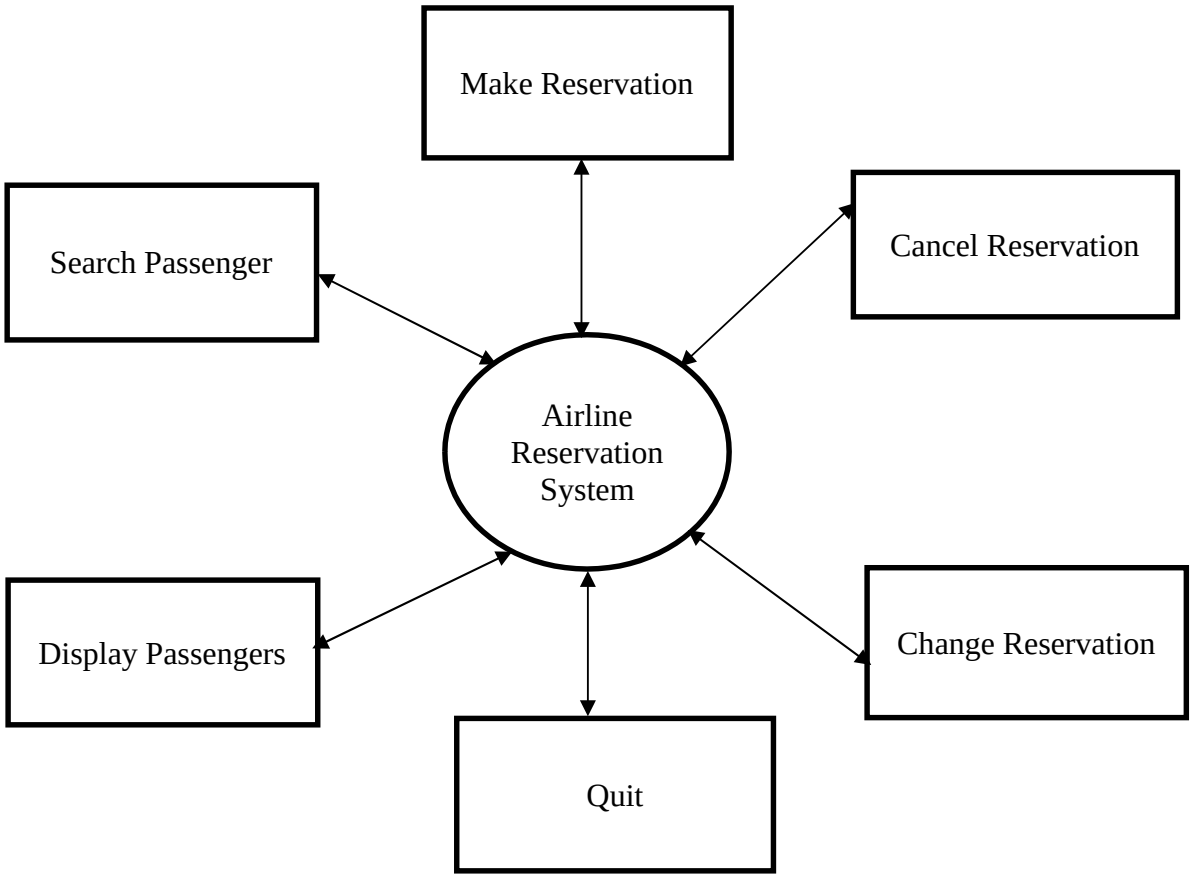


Figure 5.1 Data Flow Diagram

As information moves through software, it is modified by a series of transformations. A *Data Flow Diagram* (DFD) is a graphical technique that depicts information flow and the transformations that are applied as data move from input to output. The data flow diagram is know as a data flow graph or a bubble chart.

The *Data Flow Diagram* may be used to representation a system or software at any level of abstraction. In fact, DFDs may be used partitioned into levels that represent increasing information flow and functional detail. Therefore, the DFD provides a mechanism for functional modeling as information flow modeling.

The Data Flow Diagram (DFD) serves two purposes:

To provide an indication of how data are transformed as they move through the system and
To depict the functions that transforms the data flow.
Integration testing in an airline reservation system aims to verify the interactions between different system components and ensure their proper integration. Here are some test cases for integration testing in an airline reservation system:

5.2 Test Cases For Integration Testing :

Integration testing is an essential phase in software development where individual components or modules of a system are combined and tested together to ensure proper functioning and interaction between them. When it comes to an airline reservation system, integration testing plays a crucial role in validating the integration of various system components. Here are some example test cases for integration testing in an airline reservation system:

1. User Registration and Login Integration: This test case verifies the seamless integration between the user registration module and the login module. It ensures that user credentials are properly stored during registration and can be successfully used to authenticate and log in to the system.

2. Flight Search and Booking Integration: This test case focuses on the integration between the flight search module and the booking module. It checks if the selected flight details are correctly passed from the search results to the booking module, allowing users to proceed with the reservation process without any data loss or inconsistencies.

3. Payment Gateway Integration: This test case validates the integration between the booking module and the payment gateway. It ensures that payment transactions are securely processed, and the appropriate payment status is updated in the system after successful or failed transactions. It also verifies the handling of any errors or exceptions that may occur during the payment process.

4. Seat Selection and Availability Integration: This test case tests the integration between the seat selection module and the seat availability module. It verifies that the selected seat is

correctly assigned to the passenger during the booking process and that the seat availability is updated in real-time to prevent double bookings or conflicts.

5. Flight Status and Notification Integration: This test case checks the integration between the flight status module and the notification module. It verifies if the system can accurately retrieve and display real-time flight status information and send appropriate notifications to passengers regarding any changes or updates to their flights.

6. Loyalty Program Integration: This test case focuses on the integration between the loyalty program module and the booking module. It ensures that passenger loyalty points or rewards are correctly tracked and applied during the booking process, allowing passengers to redeem their points or receive any eligible benefits.

7. Ancillary Service Integration: This test case validates the integration between the ancillary service module and the booking module. It verifies if passengers can successfully add or customize ancillary services, such as pre-paid meals or extra baggage, during the booking process, and if the selected services are correctly associated with the passenger's reservation.

8. Reporting and Analytics Integration: This test case tests the integration between the reporting and analytics module and other system components. It verifies if the system can accurately collect and analyze relevant data from different modules, generating comprehensive reports and analytics that provide valuable insights into the system's performance and usage.

These test cases cover critical integration points within an airline reservation system and ensure that the different components work seamlessly together, providing a reliable and user-friendly experience for passengers while supporting efficient operations for the airline. Additional test cases can be designed based on the specific functionalities and integration requirements of the system.

When testing the input and output functionalities of an airline reservation system, you need to ensure that data entered by users is accurately processed and that the system produces the expected outputs. Here are some testcases for input and output testing in an airline reservation system:

5.3 Test Cases For Input and output Testing:

Input and output test cases are designed to validate the correctness and accuracy of the input and output

functionalities of a software system. In the context of an airline reservation system, these test cases focus on ensuring that the system correctly handles various inputs and produces the expected outputs. Here are some examples of input and output test cases for an airline reservation system:

1. Input Validation Test Case: This test case checks the system's behavior when invalid or incorrect inputs are provided. For example, entering a non-numeric value in a field that expects a numeric input or entering an invalid date format. The test case verifies that the system properly detects and handles such inputs, providing appropriate error messages to the user.

2. Flight Search Test Case: This test case verifies the accuracy of the flight search functionality. It involves entering specific search criteria, such as the origin and destination airports, date of travel, and passenger count, andcheckingifthesystemreturnsthecorrectlistofavailableflightsthatmatchthecriteria.

3. Booking Confirmation Test Case: This test case focuses on the booking confirmation process. It involves simulating a booking transaction by selecting a flight, entering passenger details, and confirming the booking. The test case checks if the system generates a booking confirmation message or ticket with the correct flight details, passenger information, and reservation number.

4. Seat Selection Test Case: This test case ensures the proper functioning of the seat selection feature. It involves selecting a seat during the booking process and checking if the system correctly reserves the chosen seat, updates the seat availability, and displays the selected seat in the booking confirmation or ticket.

5. Cancellation and Refund Test Case: This test case focuses on the cancellation and refund process. It involves canceling a booked reservation and verifying if the system correctly updates the reservation status, releases theseat for future bookings, and processes any applicable refunds or cancellation fees.

6. Flight Status Update Test Case: This test case checks the system's ability to provide real-time updates on flight statuses. It involves simulating changes in flight status, such as delays or cancellations, and verifying if the system promptly reflects these updates and notifies affected passengers accordingly.

CHAPTER 6
PSEUDOCODE

```
#include <iostream>
#include <fstream>
#include <string>
#include <vector>
using namespace std;

struct Passenger {
    string firstName;
    string lastName;
    string ID;
    string phoneNumber;
    int seatNumber;
};

bool isIDExists(const string& searchID) {
    ifstream infile("passengers.txt");
    if (!infile) {
        cout << "Error opening file!" << endl;
        return false;
    }

    Passenger passenger;

    while (infile >> passenger.firstName
                  >> passenger.lastName
                  >> passenger.ID
                  >> passenger.phoneNumber
                  >> passenger.seatNumber) {
        if (passenger.ID == searchID) {
            infile.close();
            return true;
        }
    }

    infile.close();
    return false;
}

bool isSeatBooked(int seatNumber) {
    ifstream infile("passengers.txt");
    if (!infile) {
        cout << "Error opening file!" << endl;
        return false;
    }

    Passenger passenger;
```

```
    while (infile >> passenger.firstName
                  >> passenger.lastName
                  >> passenger.ID
                  >> passenger.phoneNumber
                  >> passenger.seatNumber) {
        if (passenger.seatNumber == seatNumber) {
            infile.close();
            return true;
        }
    }

    infile.close();
    return false;
}

void makeReservation() {
    ofstream outfile("passengers.txt", ios::app);
    if (!outfile) {
        cout << "Error opening file!" << endl;
        return;
    }

    Passenger passenger;

    cout << "Enter first name: ";
    cin >> passenger.firstName;
    cout << "Enter last name: ";
    cin >> passenger.lastName;

    // Validate and get a unique ID
    while (true) {
        cout << "Enter ID \n(ID FROM 1 TO 100) :";
        cin >> passenger.ID;

        if (isIDExists(passenger.ID)) {
            cout << "Error: ID already exists! Please enter a unique ID." << endl;
        } else {
            break;
        }
    }

    cout << "Enter phone number: ";
    cin >> passenger.phoneNumber;

    // Validate and get a unique seat number
    while (true) {
        cout << "Enter seat number: ";
        cin >> passenger.seatNumber;

        if (isSeatBooked(passenger.seatNumber)) {
            cout << "Error: Seat already booked! Please choose a different seat." << endl;
        } else {
```

```
        break;
            }
        }

        outfile << passenger.firstName << " "
                << passenger.lastName << " "
                << passenger.ID << " "
                << passenger.phoneNumber << " "
                << passenger.seatNumber << endl;

        outfile.close();

        cout << "Reservation made successfully!" << endl;
    }

    void cancelReservation() {
        string searchID;
        cout << "Enter ID to search for passenger: ";
        cin >> searchID;

        ifstream infile("passengers.txt");
        if (!infile) {
            cout << "Error opening file!" << endl;
            return;
        }

        ofstream outfile("temp.txt");
        if (!outfile) {
            cout << "Error opening file!" << endl;
            return;
        }

        Passenger passenger;
        bool found = false;

        while (infile >> passenger.firstName
                      >> passenger.lastName
                      >> passenger.ID
                      >> passenger.phoneNumber
                      >> passenger.seatNumber) {
            if (passenger.ID == searchID) {
                found = true;
                cout << "Reservation for passenger with ID "
                     << passenger.ID << " cancelled!" << endl;
                continue; // Skip writing canceled reservation to temp.txt
            }

            outfile << passenger.firstName << " "
                    << passenger.lastName << " "
                    << passenger.ID << " "
                    << passenger.phoneNumber << " "
                    << passenger.seatNumber << endl;
        }
```

```
 infile.close();
    outfile.close();

    if (!found) {
        cout << "Passenger not found!" << endl;
    }
}

void changeReservation() {
    string searchID;
    cout << "Enter ID to search for passenger: ";
    cin >> searchID;

    ifstream infile("passengers.txt");
    if (!infile) {
        cout << "Error opening file!" << endl;
        return;
    }

    ofstream outfile("temp.txt");
    if (!outfile) {
        cout << "Error opening file!" << endl;
        return;
    }

    Passenger passenger;
    bool found = false;

    while (infile >> passenger.firstName
                  >> passenger.lastName
                  >> passenger.ID
                  >> passenger.phoneNumber
                  >> passenger.seatNumber) {
        if (passenger.ID == searchID) {
            found = true;
            cout << "Enter new first name: ";
            cin >> passenger.firstName;
            cout << "Enter new last name: ";
            cin >> passenger.lastName;
            cout << "Enter new phone number: ";
            cin >> passenger.phoneNumber;
            cout << "Enter new seat number: ";
            cin >> passenger.seatNumber;
        }

        outfile << passenger.firstName << " "
                << passenger.lastName << " "
                << passenger.ID << " "
                << passenger.phoneNumber << " "
                << passenger.seatNumber << endl;
    }
```

```
 infile.close();
    outfile.close();

    if (!found) {
        cout << "Passenger not found!" << endl;
    } else {
        cout << "Reservation changed successfully!" << endl;
    }
}

void searchPassenger() {
    string searchID;
    cout << "Enter ID to search for passenger: ";
    cin >> searchID;

    ifstream infile("passengers.txt");
    if (!infile) {
        cout << "Error opening file!" << endl;
        return;
    }

    Passenger passenger;
    bool found = false;

    while (infile >> passenger.firstName
                  >> passenger.lastName
                  >> passenger.ID
                  >> passenger.phoneNumber
                  >> passenger.seatNumber) {
        if (passenger.ID == searchID) {
            found = true;
            cout << "Passenger found!" << endl;
            cout << "First Name: " << passenger.firstName << endl;
            cout << "Last Name: " << passenger.lastName << endl;
            cout << "ID: " << passenger.ID << endl;
            cout << "Phone Number: " << passenger.phoneNumber << endl;
            cout << "Seat Number: " << passenger.seatNumber << endl;
            break;
        }
    }

    infile.close();

    if (!found) {
        cout << "Passenger not found!" << endl;
    }
}

void displayPassengers() {
    ifstream infile("passengers.txt");
    if (!infile) {
        cout << "Error opening file!" << endl;
        return;
```

```
    }

    Passenger passenger;
    int count = 0;

    while (infile >> passenger.firstName
                  >> passenger.lastName
                  >> passenger.ID
                  >> passenger.phoneNumber
                  >> passenger.seatNumber) {
        cout << "Passenger " << ++count << ":" << endl;
        cout << "First Name: " << passenger.firstName << endl;
        cout << "Last Name: " << passenger.lastName << endl;
        cout << "ID: " << passenger.ID << endl;
        cout << "Phone Number: " << passenger.phoneNumber << endl;
        cout << "Seat Number: " << passenger.seatNumber << endl;
        cout << "-------------------------------" << endl;
    }

    infile.close();

    if (count == 0) {
        cout << "No passengers found!" << endl;
    }
}

int main() {
    int choice;

    while (true) {
        cout << "= = = = = =KINGFISHER AIRLINES= = = = = =" << endl;
        cout << "1. MAKE A RESERVATION" << endl;
        cout << "2. CANCEL RESERVATION" << endl;
        cout << "3. MODIFY RESERVATION" << endl;
        cout << "4. SEARCH PASSENGERS" << endl;
        cout << "5. DISPLAY PASSENGERS" << endl;
        cout << "6. QUIT" << endl;
        cout << "ENTER YOUR CHOICE : ";
        cin >> choice;

        switch (choice) {
            case 1:
                makeReservation();
                break;
            case 2:
                cancelReservation();
                break;
            case 3:
                changeReservation();
                break;
            case 4:
                searchPassenger();
                break;
```

```
        case 5:
                displayPassengers();
                break;
            case 6:
                return 0;
            default:
                cout << "INVALID CHOICE...! " << endl;
                break;
        }
    }

    return 0;
}
```

CHAPTER 7

SNAPSHOTS

After running the executable fireroom the project zip or running the project from the compiler you will directly see the following main screen:

you will directly see the following main screen:

Here we've displayed the menu options such as make a reservation, cancel a reservation, search passenger, change a reservation, display the passengers, and quit. If the passenger wants to make a reservation for a flight then he must choose option 1. If the passenger's mind changed and wants to cancel the reservation then he must choose option 2. By choosing option 3 the passenger can Change reservations, if he wants to check who has booked a particular seat.

[1] "Airline Reservation System" by Ashutosh Bajpai, International Journal of Innovative Research in Science, Engineering and Technology (IJIRSET), Volume 6, Issue 6, June 2017. (Link: https://www.ijirset.com/upload/2017/june/22_IJRSET_Article.pdf)

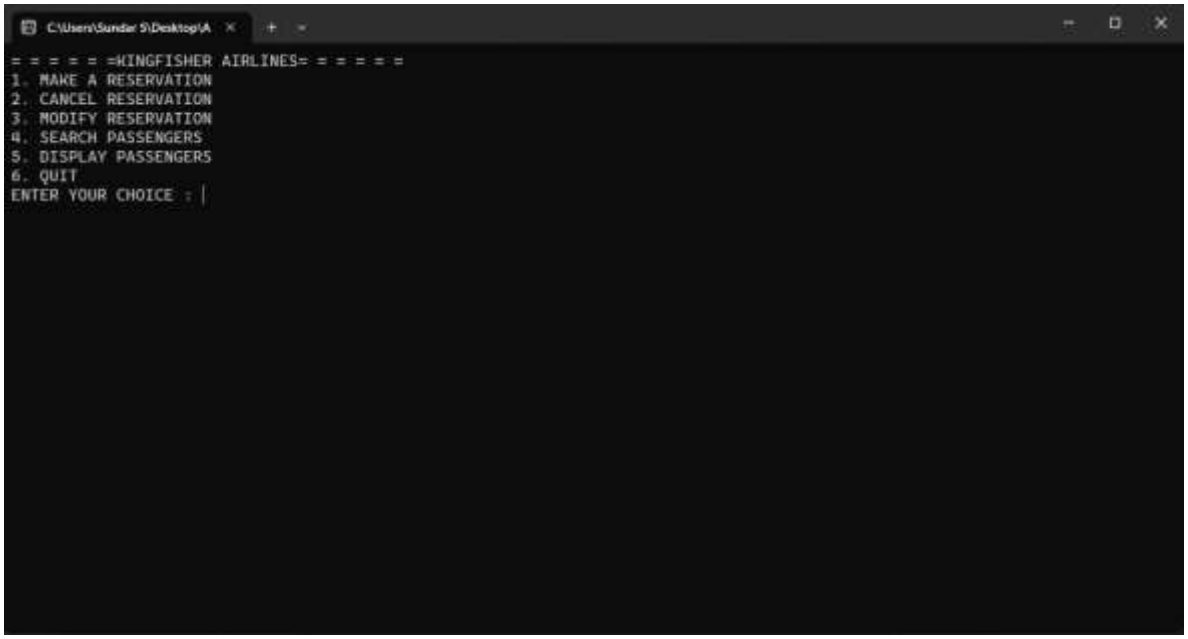
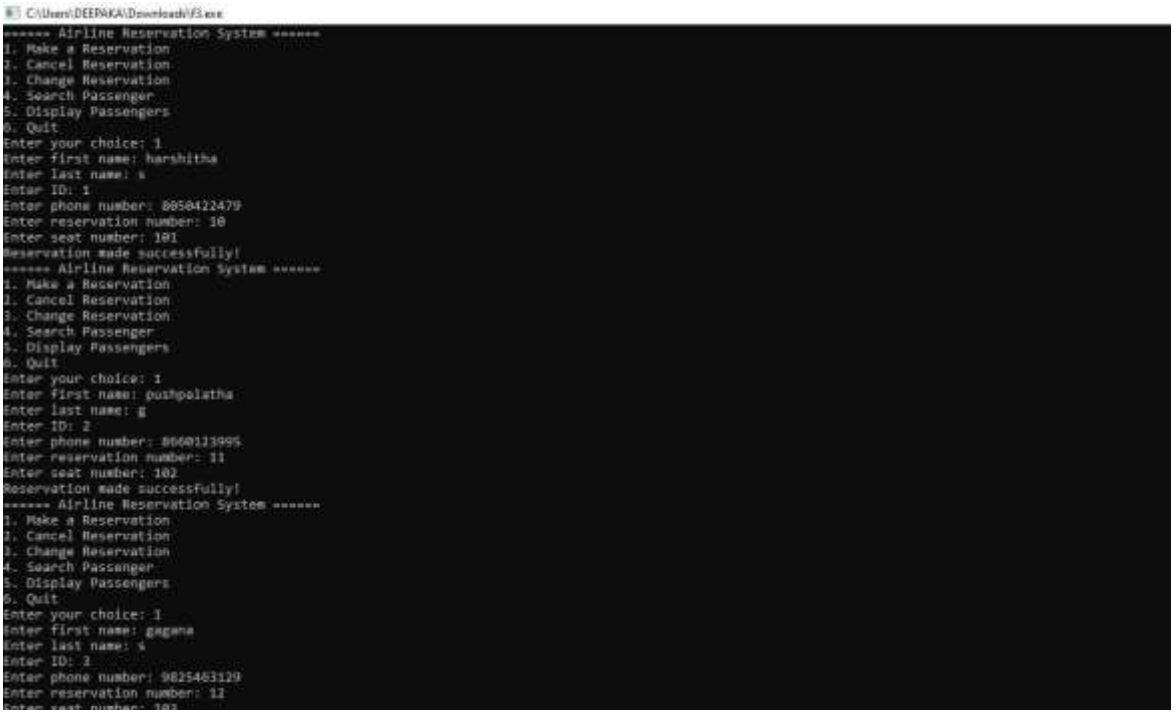


Figure 7.1 Main screen of Airline Reservation System

Make Reservation:

By choosing menu option number 1 passengers will be able to make a fresh reservation for any particular flight. During the reservation software will ask for some basic details which are as follows:

- First name of a passenger.
- Last name of a passenger.
- Id number.
- Phone number.
- Seat number which you want to book.



Gone are the days when people who wanted to travel by air had to make long queues in order to book a flight. The invention of computers and the internet has made things simple because one can book a flight at the touch of a button. This paper explains the path that the automated reservations have followed to get to where they are today and also the challenges that have been met.

Figure 7.2 Making Reservation page

After entering the above details the unique reservation id will be generated for your booking. If the reservation is done successfully then the process completed message will be displayed at the bottom of the project. If the user is trying to book the seat which is already reserved then the software will show the message “Seat is already reserved please choose another seat number from below”. So you’ve to enter the seat number again from the given list then only the seat will be reserved successfully.

Cancel Reservation:

Sometimes what happens passengers’ mind changes and they try to cancel the reservation because of some urgency or personal reason. So in such a situation, he’ll be able to cancel the reservation anytime by choosing option 2 from the options menu.

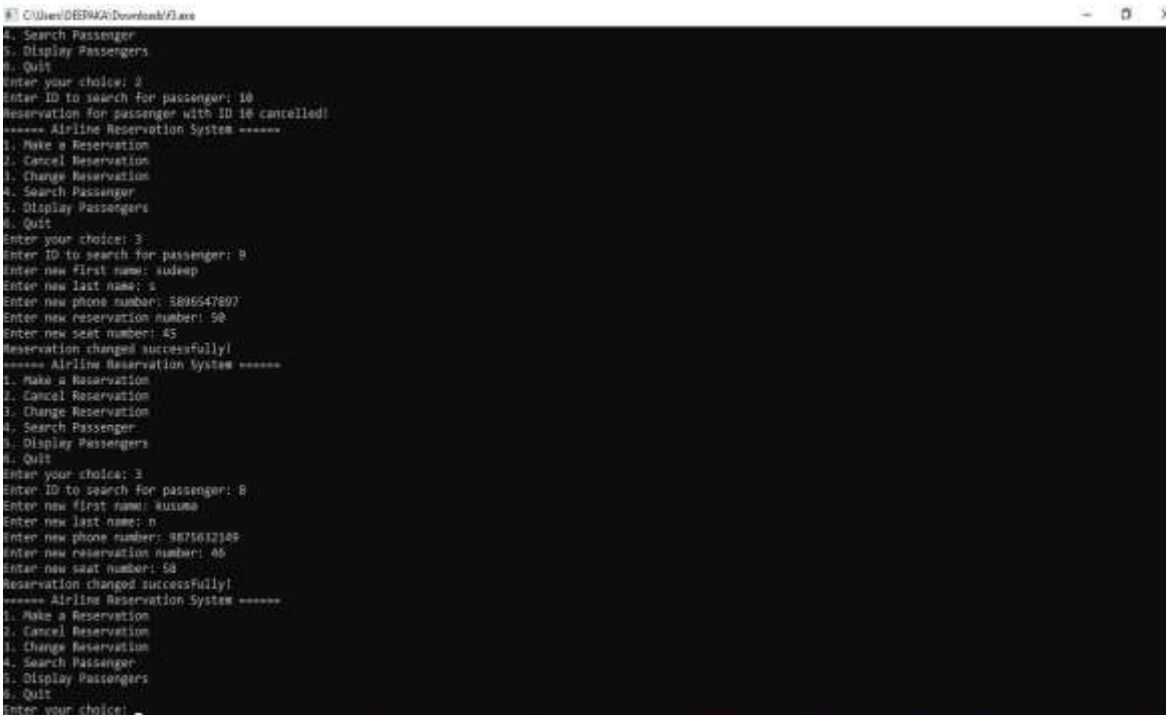
To cancel the reservation you must have your reservation number generated by software during the reservation. Because software asks reservation number during the cancellation. If you forgot the reservation number then you’ll not be able to cancel the tickets.

Figure 7.3 Cancel Reservation page

When you choose menu option 2, then you've to enter the reservation number then the software will fetch the passenger details using this reservation number. This reservation number is uniquely generated and confidential that's why after entering the reservation number, your reservation will be canceled quickly.

Change Reservation:

By choosing menu option number 3 you'll be able to change the reservation or edit the existing reservation. Sometimes the passengers want to change their seat number because they want to sit near the window or the couple wants seats near each other. In such a situation, passenger wants to change the location of their seat by editing the reservation.



After choosing menu option 3, the software asks for your existing seat number first. After entering the existing seat number the software displays the list of available seats which you can choose.

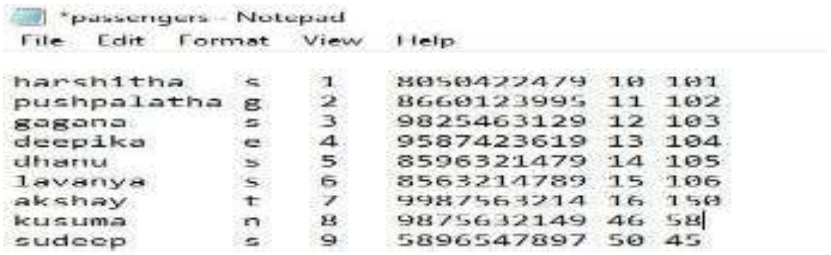


Figure 7.4 Change Reservation page

If you enter the seat number which is already chosen by others then the software will crash and the project will be exited.

Search Passenger:

With this option 4, we can search the passengers we need to search.

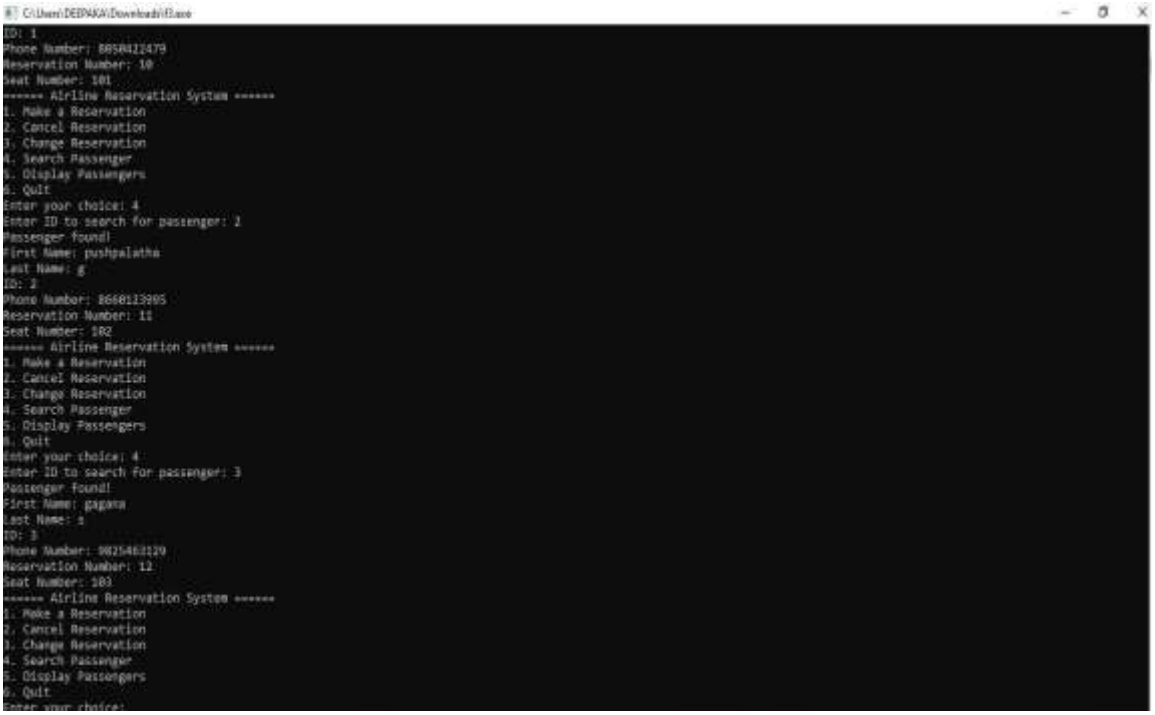


Figure 7.5 Search Passenger page

Display Passengers :

This software provides a facility to display all the passengers of all the reservations done by the passengers. If the user wants to check which seats are available and which seats are actually booked then in such case this list will help the user to make a reservation quickly.

The "Display Passengers" feature in an airline reservation system allows users, such as administrators or staff members, to view the list of passengers who have made reservations. This functionality is crucial for managing and organizing passenger information, ensuring smooth operations and providing a reliable passenger record database.

When the "Display Passengers" option is selected, the system accesses the passenger database, typically stored in a file or database management system. It retrieves the relevant data for each passenger, including their first name, last name, ID, phone number, reservation number, and seat number.

The system then presents this information in a clear and organized manner.

Figure 7.6 Display Passengers

Quit:

By choosing the option number 7, the user will be able to quit the Reservation page of the Airline Reservation System.

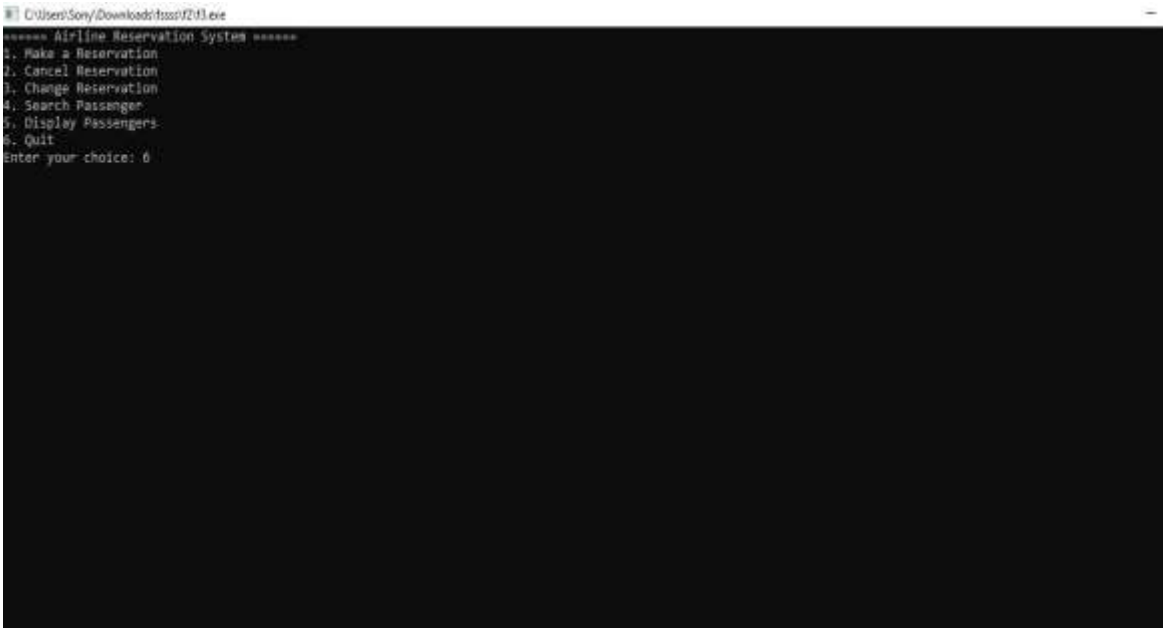


Figure 7.7 Quit page

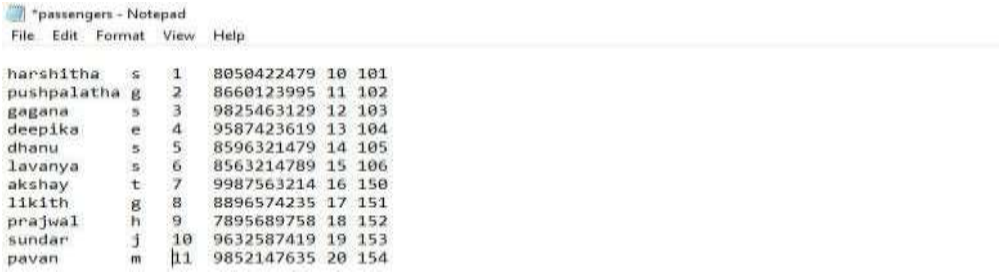


Figure 7.8 Text File

In an airline reservation system, text files can be used for various purposes, such as storing and processing data related to flights, reservations, passengers, and other relevant information. Text files provide a simple and widely compatible format for storing structured or unstructured data. Here are some use cases and considerations regarding text files in an airline reservation system:

1. Data Storage:

Text files can serve as a storage medium for holding data in a structured or semi-structured format. For example, a text file can be used to store flight schedules, airport codes, seat availability, or pricing information. Each line in the file can represent a separate record or entry, and the fields within each record can be delimited by specific characters (e.g., commas or tabs) or follow a fixed-width format.

2. Data Import/Export:

Text files can act as a common interchange format for importing or exporting data from the airline reservation system. For instance, reservation data from external sources or travel agencies can be provided in a text file format for the system to ingest and process. Similarly, system-generated reports or data extracts can be exported as text files for further analysis or sharing with external parties.

Conclusion and Future Enhancement

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 minimize 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 an 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.

FUTURE ENHANCEMENT we have decided to further enhance with a seat reservation available. It is to fulfill passengers request to sit where they prefer. They are allowed to choose their seat whether near to window's seat or in the middle. In this program, we have used fixed length records and Hashing on primary key, so that the user gets the advantage of using these type of records. The program runs well but there is always a scope for improvement. Better search techniques can be implemented and searching can be done based on other aspects too.

REFERENCES

Books:

[i] . File Structures with C++ by Michael J Folk.

[ii]. File Structures using C++ by K R Venugopal, K G Srinivasa, P M Krishnaraj,TataMcGraw-Hill ,2009, Referred Page No-4,39-40.

Here are a few references related to the topic of airline reservation systems:

[1] "Airline Reservation System" by Ashutosh Bajpai, International Journal of Innovative Research in Science, Engineering and Technology (IJIRSET), Volume 6, Issue 6, June 2017. (Link: https://www.ijirset.com/upload/2017/june/22_IJRSET_Article.pdf)