

**MINI PROJECT ON**

**AIRLINE TICKET RESERVATION SYSTEM**

***Submitted by: Dhuma M S***

***SF ID: 104986***

**CONTENTS**

[**1.INTRODUCTION**](#_heading=h.3dy6vkm) **7**

[1.1 Problem Statement](#_heading=h.1t3h5sf) 7

[1.2 Motivation](#_heading=h.4d34og8) 7

[1.3 Project Scope](#_heading=h.3rdcrjn) 8

[**2.DESCRIPTION**](#_heading=h.lnxbz9) **9**

[2.1 Project Perspective](#_heading=h.35nkun2) 9

[2.2 Functional Description](#_heading=h.1ksv4uv) 9

[**3.RESEARCH**](#_heading=h.z337ya) **12**

[3.1 History](#_heading=h.3j2qqm3) 12

[**4.DESIGN**](#_heading=h.1y810tw) **13**

[4.1 User Design](#_heading=h.4i7ojhp) 13

[4.2 Admin Design](#_heading=h.2xcytpi) 13

[4.3 Diagrammatic representation:](#_heading=h.1ci93xb) 14

[4.3.1 Use Case Diagram:](#_heading=h.3whwml4) 14

[**4.3.2 Flow Chart:**](#_heading=h.3as4poj) **15**

[4.4 Functional Requirements](#_heading=h.2p2csry) 16

[4.4.1 Use Case name: User Login](#_heading=h.147n2zr) 16

[4.4.2 Use Case name: Reservation](#_heading=h.3o7alnk) 16

[4.4.3 Use Case name: View Flights](#_heading=h.23ckvvd) 17

[4.4.4 Use Case name: Exit](#_heading=h.ihv636) 17

[4.4.5 Use Case name: Add flight information](#_heading=h.32hioqz) 17

[4.4.6 Use Case name: Printing and Confirming Ticket](#_heading=h.1hmsyys) 18

[**5.REQUIREMENTS**](#_heading=h.4f1mdlm) **19**

[5.1 SOFTWARE REQUIREMENTS:](#_heading=h.2u6wntf) 19

[5.2 HARDWARE REQUIREMENTS/SPECIFICATION:](#_heading=h.19c6y18) 19

[**6.TEST CASES**](#_heading=h.3tbugp1) **20**

[**8. EXPECTED RESULTS**](#_heading=h.3q5sasy) **21**

[**9.REFERENCE**](#_heading=h.25b2l0r) **22**

**LIST OF FIGURES**

1. Interior of a Flight………………………………………………………….…...8
2. Login Page………………………………………………………………….…...9
3. (a) Main Menu……………………………………………………………….…10

(b) Enter Details………………………………………………………………..10

1. Flight Details…………………………………………………………………..10
2. Ticket……………………………………………………………………….…..11
3. Use Case Diagram………………………………………………………….…..14
4. Flow Chart……………………………………………………………………...15

# 

# 

# 

# 

# 

**LIST OF TABLES**

1. Login Page……………………………………………………………..……..20

**LIST OF ABBREVIATIONS**

CISC - Customer Information Control System

DATA - Depth, Air, Time & Awareness

IBM - International Business Machines

OS - Operating System

SABRE - Semi-automated Business Research Environment

TWA - Trans World Airlines

# **1.INTRODUCTION**

Transportation including roads, railways, airways, waterways etc always plays an important role in our daily life. People always choose their suitable means of transportation depending on their safety and convenience. Air transport is an important enabler to achieving economic growth and development. It facilitates integration into the global economy and provides vital connectivity on a national, regional, and international scale. Airways have always been considered as the best option for long distance travel since they have a low travelling time.

Our country is facing severe pandemic conditions nowadays. Transportation is a major challenge faced by the citizens. For long distance travel it is not safe to use train or bus for transportation, so most preferable will be the air transportation system. It is necessary to make sure that we get the desired seat in the desired flight to our destination.

The proposed project is providing an easy means for the airline ticket reservation system. It shows all the available flights along with the time and fare details. A person can reserve a ticket to this destination through this user-friendly platform.

## **1.1 Problem Statement**

The purpose of this document is to build an online system to manage flights and passengers to ease the flight ticket reservation.

## **1.2 Motivation**

Transportation is an inevitable factor in our daily life. The current situation is making long distance travel unsafe. To an extent airlines are safe to depend on for long distance transportation. The project is introducing an airline ticket reservation system so that the users can make sure of their seats and avoid having any kind of physical contact between people.

## 

## 

## **1.3 Project Scope**

## The purpose of the online flight management system is to ease flight management and to create a convenient and easy-to-use application for passengers, trying to buy airline tickets. The system is based on a relational database with its flight management and reservation functions.

Our country is facing severe pandemic conditions which have made our transportation facilities a challenging one, especially long-distance transportation. Choosing airways will be the safest option for a person to travel long-distance.

Assuring our seat is an important factor. The current situation of the country can’t encourage physical contact between individuals. Online reservation of tickets can avoid the individuals from getting in contact with each other and also it can prevent the wastage of paper thus being environment friendly. The proposed project helps the user to reserve their travel tickets in an easy manner.



Fig 1. Interior of a Flight

# **2.DESCRIPTION**

## **2.1 Project Perspective**

* Customer description: It includes the required details of the customer.
* Flight description: It includes the flight name, number, destination, ticket fare per person and the time of departure.
* Reservation Description: It includes customer details, flight name, flight number, destination, total ticket fare and departure time.

## **2.2 Functional Description**

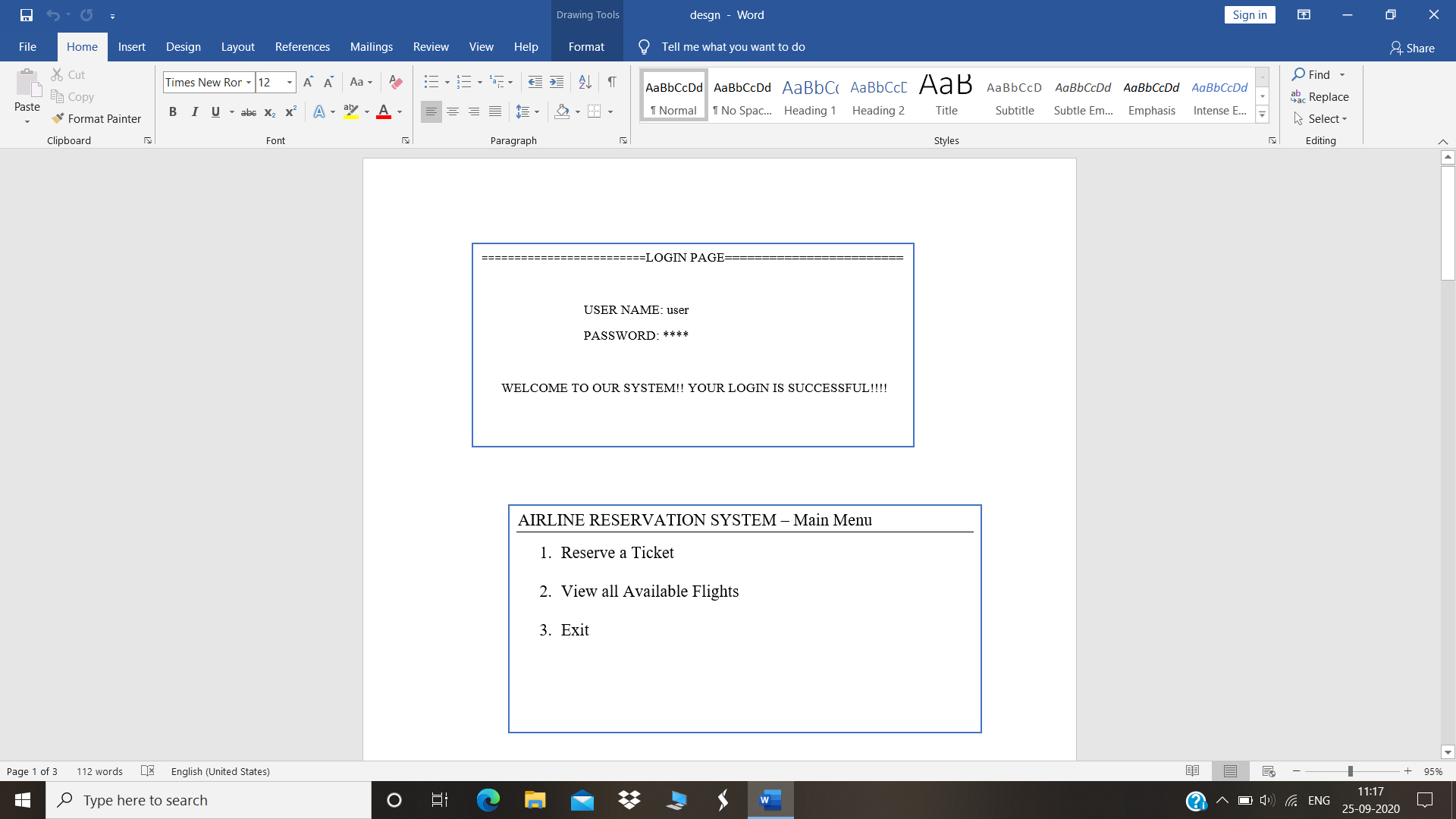
**o**  Login: The user is expected to provide correct user id and password in order to enter the system.

Fig 2. Login Page

o Reservation: The user can reserve seats in his/her desired flight by providing all the necessary details and can confirm the ticket.

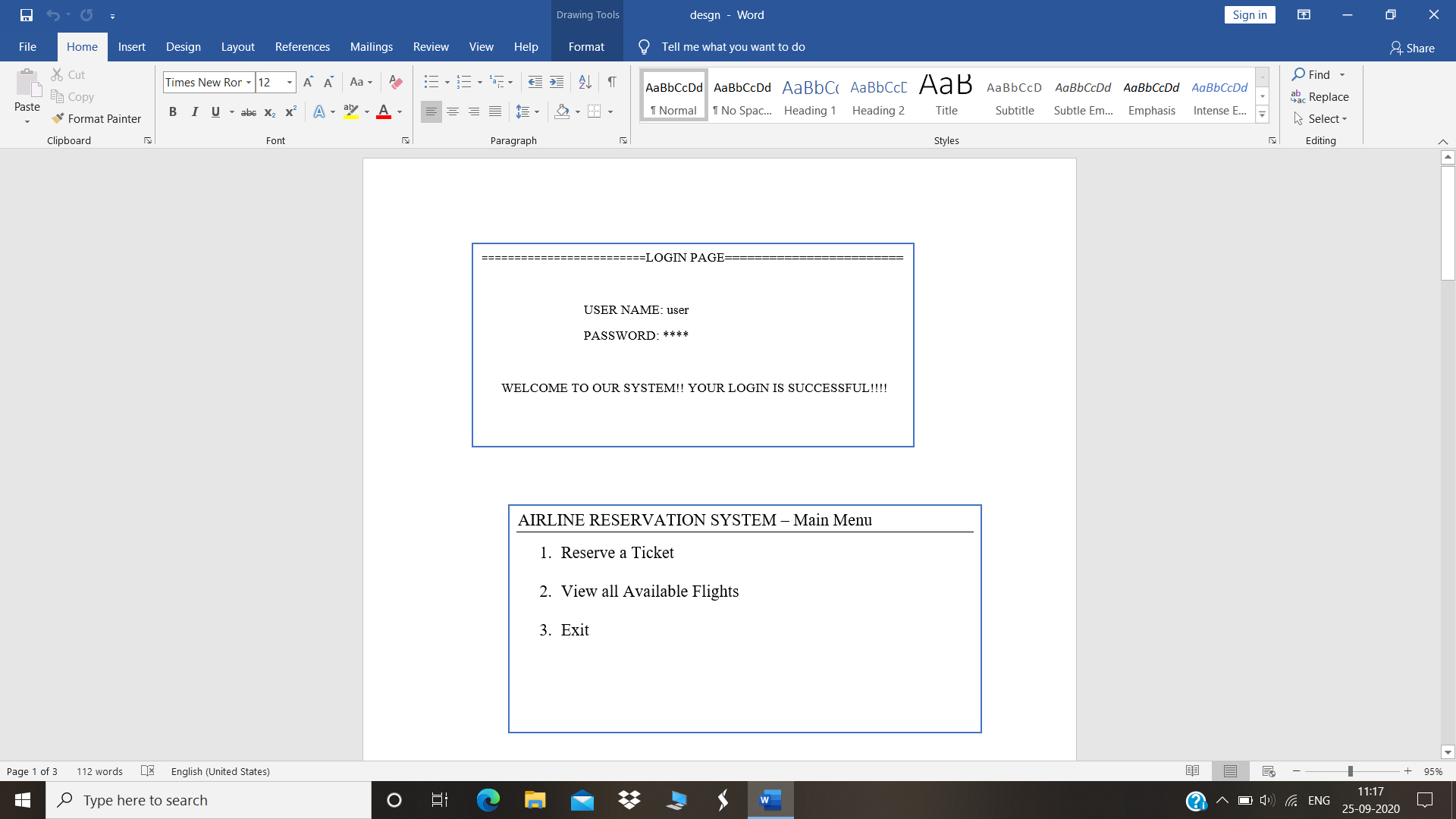
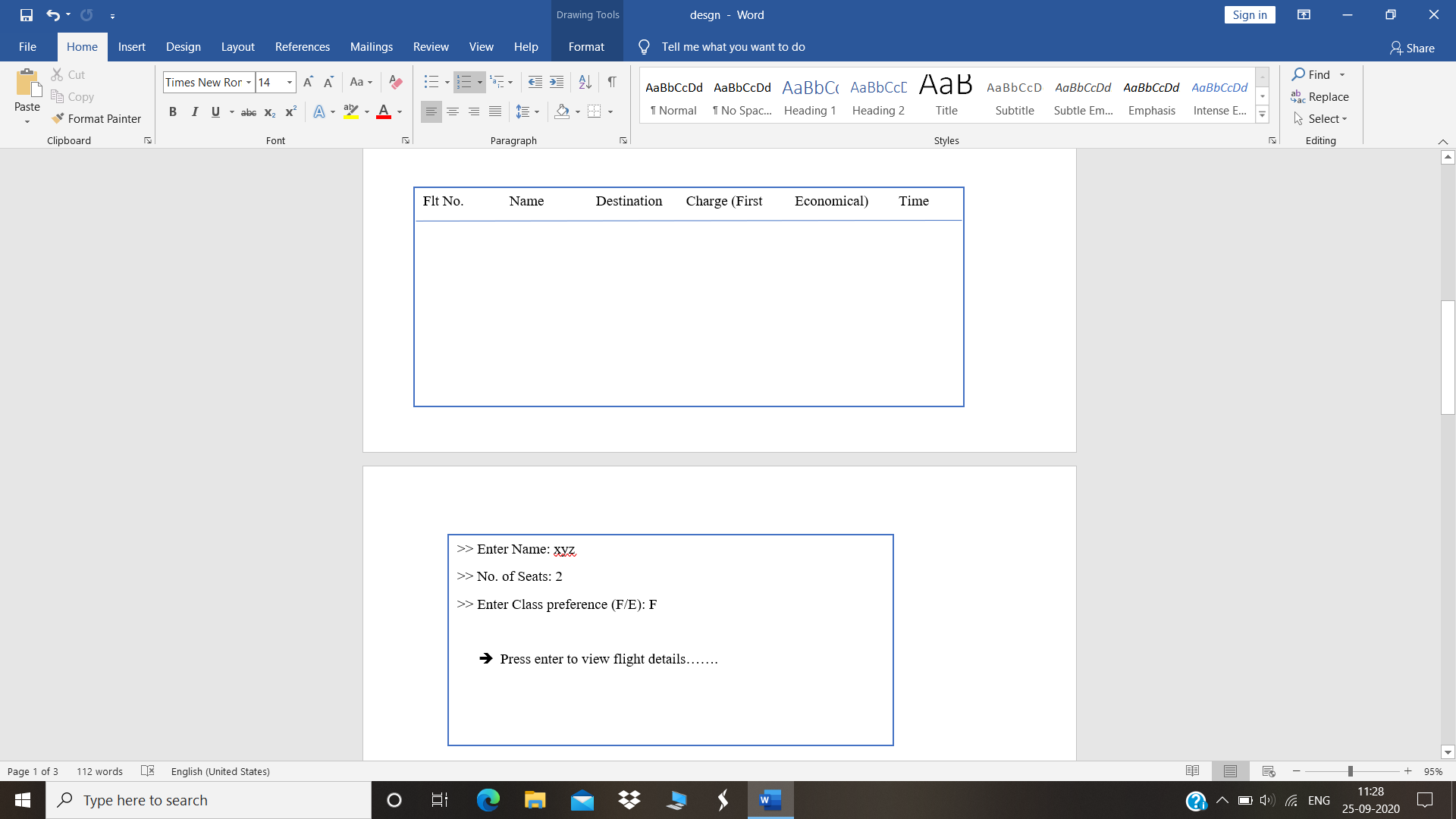
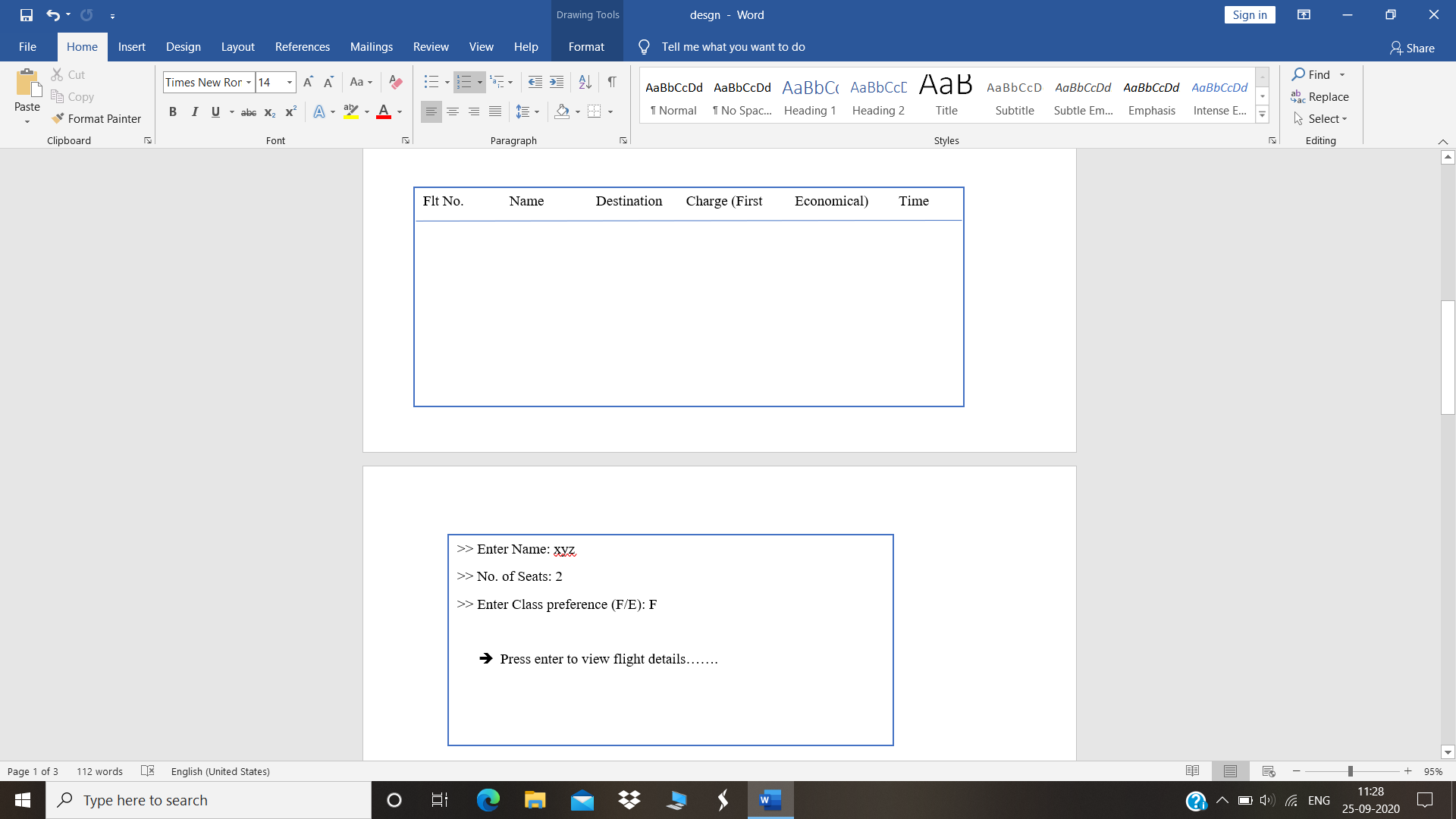
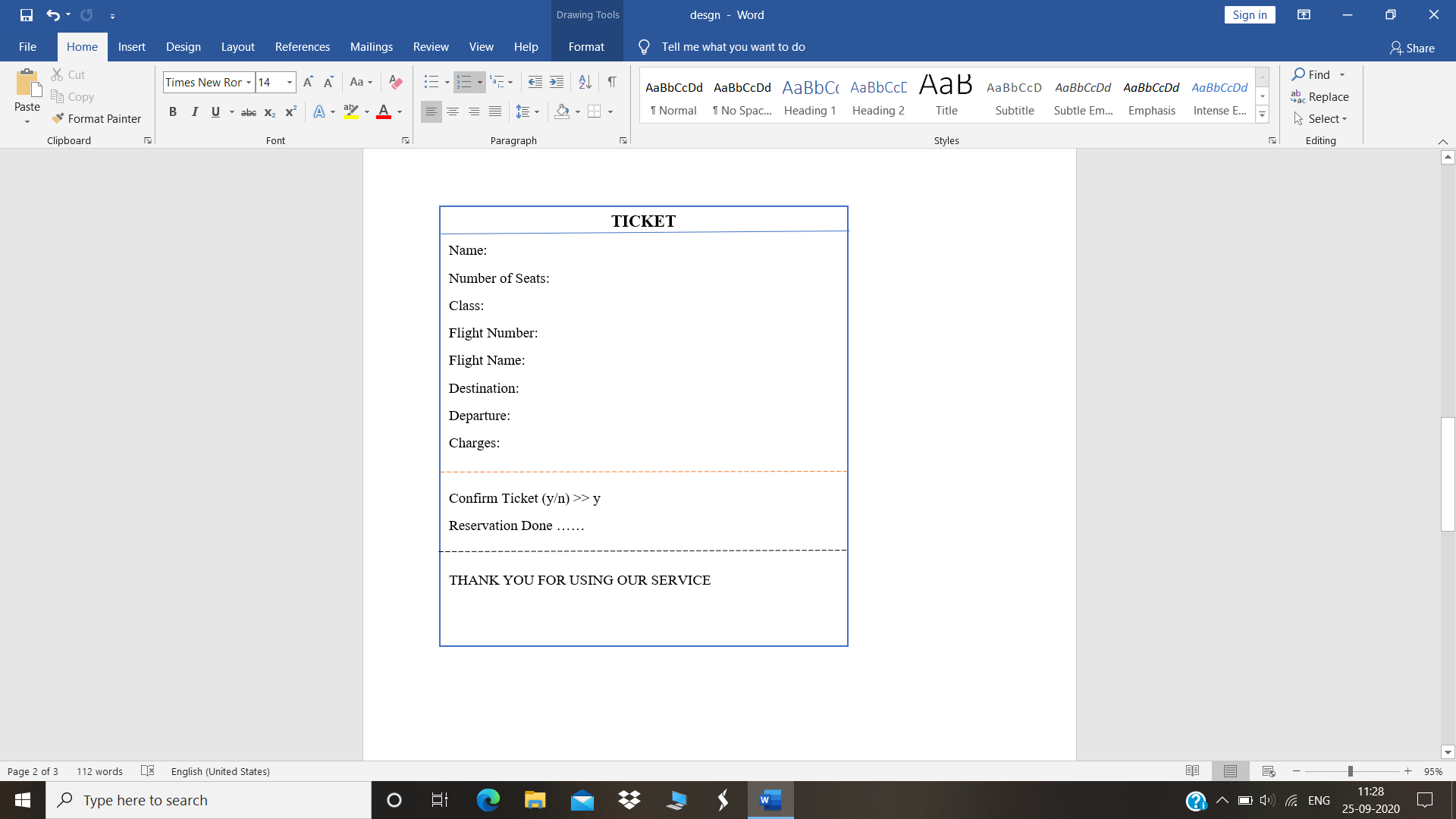
Fig 3(a). Main Menu

Fig 3(b).Enter Details

o View Flight Details: It includes the Flight name, flight number, destination, preferred class and the ticket fare.

o Charge: The ticket fare will be calculated considering the number of seats and the fare per person. The charge will be different for first class and economical class.

o Print Ticket: The ticket will be printed showing all the provided details and the total fare.

Fig 5. Ticket

# 

# 

# **3.RESEARCH**

## **3.1 History**

In earlier times, in order to book a ticket, you would have to visit a travel agent or call the airline directly over the phone. Then, paper tickets would be mailed to you. Agents interacted with the system via CICS terminals which were significantly different to use and program than anything in use today.

Automated reservations systems have their roots back in the late 1940s but it was American Airlines working with IBM that led to the first modern airline reservation system called SABRE. At the time, these were some of the largest applications on the largest mainframe systems of their day. It was not a standard OS, not that many existed at the time. Other airlines developed their own systems, notably United and TWA’s Apollo system and DATAS from Delta. Later, United opened up Apollo to travel agents and non-airlines entered the reservation business.

# **4.DESIGN**

## **4.1 User Design**

The proposed project is a user-friendly platform for the Airline ticket reservation.

- The user can login to his/her personalised login by providing the valid credentials.

- He/she will be provided with 3 choices in the main menu.

- While selecting among the choices 1,2,3 he/she will be directed to the reservation system, viewing the flight chart or exiting the console respectively.

- In the reservation window, the user can enter the required details and the class preference.

- Then the user is directed to the flight chart from where he/she can select the convenient flight.

- The user can confirm the ticket by choosing ‘y’ after the ticket is displayed

- Reservation will be confirmed and the data is stored.

## **4.2 Admin Design**

- The admin is providing the Flight chart showing the details like name of the flight, destination, ticket fare per person (first class and economical class has different rates) and the time of departure.

- The ticket fare is calculated automatically by considering the preferred class.

- Taking the product of the number of seats and fare per person gives the total ticket charge.

- The ticket is displayed by showing all the details such as Name, Flight name, number of seats, destination, class preferred, total ticket fare and the departure time.

- The data will be stored if the user confirms the ticket, else the data is lost.

## **4.3 Diagrammatic representation:**

### **4.3.1 Use Case Diagram:**

### 

### 

### **4.3.2 Flow Chart:**

## 

Fig 7. Flow Chart

## 

## **4.4 Functional Requirements**

The functional requirements of the Airline Reservation System are divided among the customer and the administrator of the application. These functional requirements can be explained in detail as follows:

### **4.4.1 Use Case name: User Login**

· 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.

· Actor: User

· Input: The user or the customer creates a username and password at the time of registering with the system. He then uses them to logon to the system and make reservations or view any information.

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

### **4.4.2 Use Case name: Reservation**

· Description: This use case describes the scenario where the user books airline tickets.

· Actor: User

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

· Output: The system verifies the authenticity of the username and password and then displays information related to various flights to the customer.

### **4.4.3 Use Case name: View Flights**

· Description: This use case describes the scenario where the user can view the available flights.

· Actor: User

· Input: After logging onto the system, the customer looks up the information for all the available flights.

· Output: The system verifies the authenticity of the username and password and then displays information.

### **4.4.4 Use Case name: Exit**

· Description: This use case helps the user to exit the console

· Actor: User

· Input: After logging onto the application, the customer looks up whatever information he needs and then can exit.

· Output: The system exits the user.

The admin activities use cases will be described here:

### **4.4.5 Use Case name: Add flight information**

· Description: This use case describes the scenario where the admin adds flight information like flight name, number, charge, time and destination, in the system.

· Actor: Admin

· Input: The admin of the system accesses code.

· Output: The system displays the page where the administrator can modify the flight information.

### **4.4.6 Use Case name: Printing and Confirming Ticket**

· Description: This use case describes the scenario where the admin verifies the data inputs provided by the user, calculates the fare and then prints and confirms the tickets. The data will be stored only if the ticket is confirmed

· Actor: Admin

· Input: The admin of the system accesses the code.

· Output: The system authenticates the admin, and then displays the printed ticket with confirmation details and then stores the data.

# 

# 

# 

# 

# 

# **5.REQUIREMENTS**

## **5.1 SOFTWARE REQUIREMENTS:**

· CodeBlocks (20.03) is used for C programming.

The entire project is divided into various ***functions*** so that the length of the program can be reduced. It helps to locate and isolate any fault. It also facilitates top-down modular programming style, the high-level logic of the overall problem is solved and then the details of each lower-level program is addressed.

***Switch cases*** used here which enables the programmer to compare a number of values of a variable by a single switch statement and using a number of cases. It makes error detection easier as the program is divided into modules through these cases. It is generally used when many values for a variable are to be compared.

***File Handling*** concept is also used in this program in order to provide proficiency for reusability and large storage capacity. By the application of file handling methodologies we can save time and can easily transfer the contents of a file system to another without losing the data.

· Compiler: GNU GCC Compiler.

The compiler is converting the high level programming languages into machine code.

## 5.2 HARDWARE REQUIREMENTS/SPECIFICATION:

· Operating System: Windows 8 or above

· Processor: Intel core (32-bit or above)

· RAM: 8 GB or above

· Input devices: Keyboard & Touchpad (Mouse)

· Output devices: Standard Monitor

# **6.TEST CASES**

Test case is a set of conditions under which a tester will determine whether an application or software system or one of its features is working as expected. It is used by the testing team, development team as well as the management. The test cases clarify what needs to be done to test a system. It gives us the steps which we execute in a system, the input data values which we enter in the system along with the expected results when we execute a particular test case. Test cases bring together the whole testing process. If the test cases are ready, they are really helpful to measure whether client expectations were fulfilled or not. When we execute the test case we can find out the errors which we may have left behind. We need to validate that the software is working accurately for each use case and evaluate the end to end functionality of the software. The test scenarios in the proposed project are listed below:

· Login Page

**Table 1:**Test Scenario: Login Page

|  |  |  |  |
| --- | --- | --- | --- |
| Sl.No. | TEST INPUT | EXPECTED OUTPUT | ACTUAL OUTPUT |
| 1 | User providing the correct user id and password. | Welcome message displayed. | Welcome message displayed. |
| 2. | User providing an incorrect user name and/or password. | A message,’ Sorry!!! Login is Unsuccessful’ appears and the user can provide the credentials again. | A message,’ Sorry !!! Login is Unsuccessful’ appears and the user can provide the credentials again. |

# **8. EXPECTED RESULTS**

The proposed project is developed for the user to facilitate Airline Ticket Reservation. The system is expected to give the following results:

* Login page displayed.
* Directs the user to the Main Menu when he/she provides the correct user id and password.
* Directs the user to the Reservation Window, where he/she can enter the required details, when Choice 1 is selected.
* Displays the Flight Chart when Choice 2 is selected.
* Exits the window when Choice 3 is selected.
* Calculates the ticket fare automatically by taking the product of number of seats and fare per person, after verifying the preferred class.
* The ticket is displayed showing the entered details and calculated fare.

All systems are expected to compile successfully with 100% run rate and pass rate.

# **9.REFERENCE**

* <https://www.codewithc.com/>
* <https://www.geeksforgeeks.org/c-programming-language/>
* <https://en.wikipedia.org/wiki/Airline_reservations_system>