**Bus Reservation**

**System**

**DHYAN NANDAN T**

**SF ID: 104952**

**Table of Contents**

***Title Page***

***Table of Contents***

***Chapter 1: Introduction*** ………………………………………………. *3*

1.1 Problem Statement………………… *………………..3*

1.2 Project Description………………. *.…………………3*

***Chapter 2: Requirements*** *……………………………………………..4*

2.1 Hardware Requirements………................................ *4*

2.2 Software Requirements ……………………………... *4*

2.3 Functional Requirements…………………………….. *4*

***Chapter 3 : Design*** …………………………………………………….. *5*

3.1 Basic Design And Working…………………………... *5*

3.2 Flowchart………………………………………………. *5*

3.3 User Diagram…………………………………………..7

***Chapter 4 : Test Plan*** ………………………………………………….. *8*

4.1 Introduction……………………………………………. *9*

4.2 Test Strategy………………………………………….. *9*

4.3 Test Objective…………………………………………. *10*

4.4 Test Criteria…………………………………………… *10*

4.5 Test Environment…………………………………….. *11* ***Chapter 5: Test cases*** …………………………………………………. *11*

***Chapter 6: Expected Results*** ………………………………………… *.12*

***Chapter 7: Conclusion*** ………………………………………………… *13*

# 1. Introduction

## 1.1 Problem Statement

This project aims to create a system that allows the users automatically to reserve bus tickets without their manual presence at the reservation office. Users should be able to choose the desired seats from the available ones, and cancellation must also be allowed. The cancellation must automatically facilitate r efund and the seat should be marked ‘Unreserved’ i.e must be open for further booking. This project mainly focuses on Bus Reservation firms to make their processes automated.

## 1.2 Project Description

This system is an alternative for manually reserving bus tickets. With the help of this system, the user can view the bus list, book tickets, cancel bookings and check the bus status board. While booking tickets, the user has to enter the bus number to select the desired bus. Then the system displays the total number of seats along with the names of already reserved passengers marked corresponding to their respective seat numbers. The user has to enter the number of tickets, seat number, and name of the person to facilitate booking. Upon completion the seats shall be reserved and the total cost will be displayed on screen.

There is also a cancel booking feature that allows users to cancel their reservations whenever required. Here, cancel booking also features refunding.It also has a Login System to make it more secure. Thus, there’s no chance of data misuse or loss.

The whole project is developed in ‘C’ Programming language, different variables and strings have been used for the development of this project. It’s simple and hence it is easy for users to operate and understand this system. (Here the user can be a firm that reserves tickets for Passengers in their required Bus Trips).

# 2. Requirements

## 2.1 Hardware requirements

1. Ram 512MB or higher.
2. Processor speed 1.7 GHZ
3. Hardware that can run applications like CodeBlocks.

## 2.2 Software requirements

1. Windows 8 or higher versions.
2. CodeBlocks IDE.
3. MingGW Compiler.

## 2.3 Functional requirements

The reservation system m ust include the following features like:

1. Proper login system
2. Booking System
3. Refunding (cancel bookings)
4. View Bus Status

### 3. Design

## 3.1 Basic Design and Working

The Bus Reservation System consists of the following functions and features in its design. It has a Reservation System, Cancellation system and a Status Board Viewer that shows the currently unavailable and available seats separately.

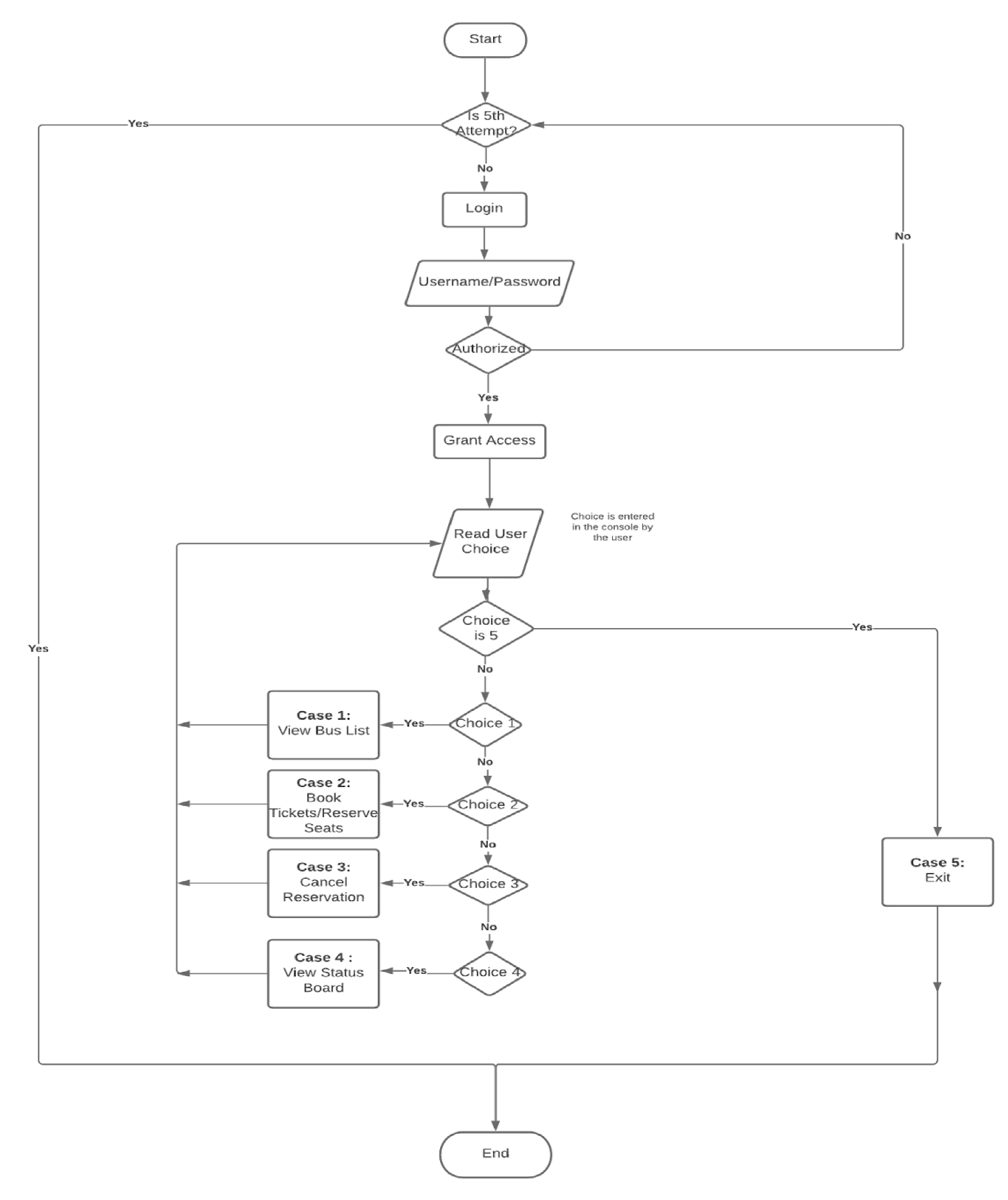
The details of various busses running service are shown in the bus list. These details are entered directly into the source code and cannot be modified by the user. The user can choose the desired bus according to his/her journey schedule. He also can choose the desired seat from the available no. of seats. A user can book any number of tickets provided seats are unreserved previously. Once the booking is done, the booking can be verified by choosing to view the Bus Status Board. If the booking has been completed, the name of passenger will appear alongside the reserved seat number.

Now, if the user needs to cancel the previously reserved seat he can do it by selecting the cancel menu. Filling in the required details and pressing enter will calculate the refund amount and mark the seat empty.

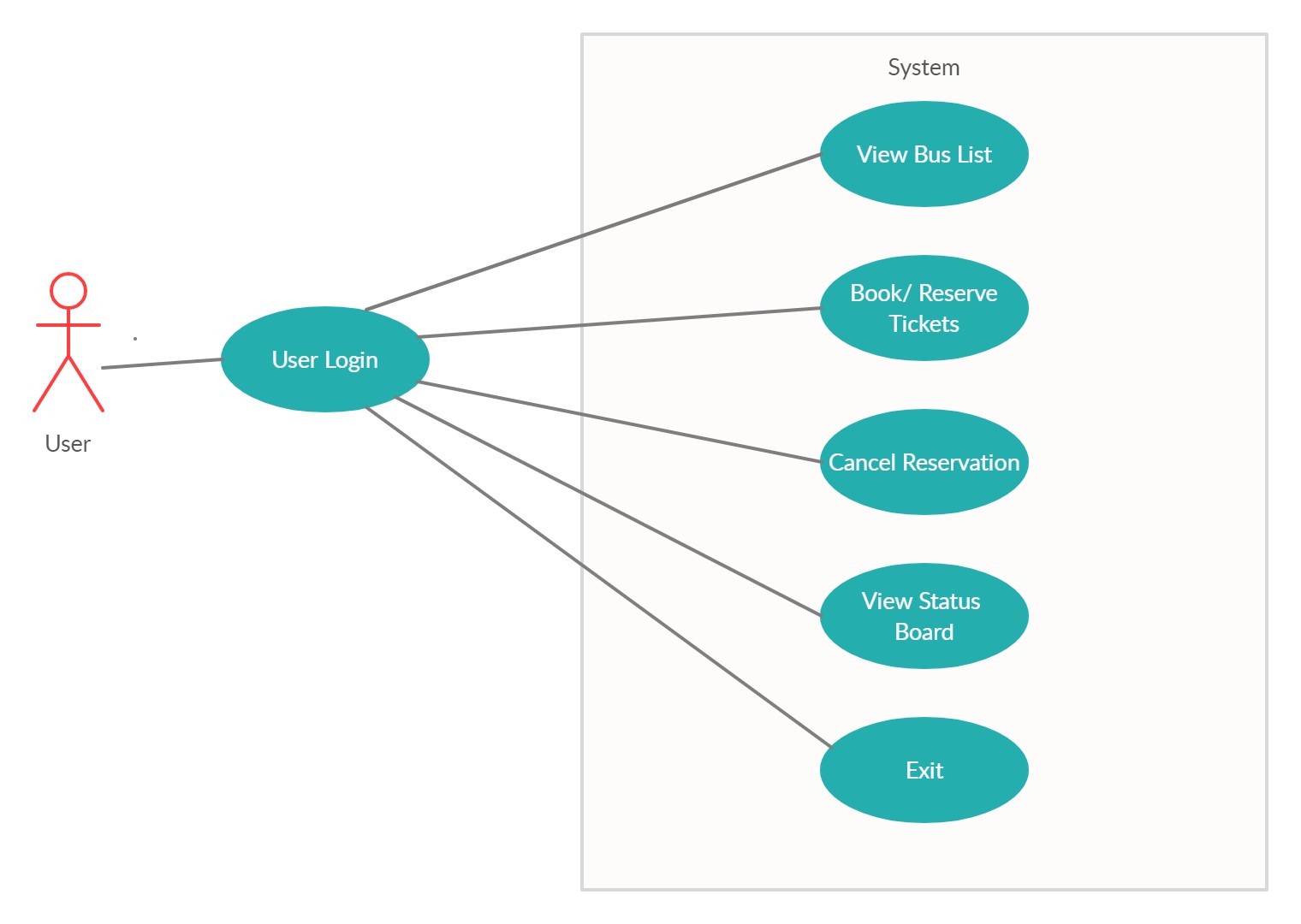
If cancellation is not required, users can directly choose to exit the system using the exit button.

The bus owners can view the Reservation Status to know how many tickets have been sold for a particular trip by checking the Bus Status Board. Hence they will be informed directly about the transactions that are being made. It does not require the bus employees to issue the ticket at the time of the journey.

## 3.2 Flowchart



## 3.3 User Diagram



Users are allowed to carry out various functions like Login, Bus List etc...The various functions and their uses are given in detail below.

**Login**

* *This function allows users to login after registration*

**View Bus List**

* *It allows the user to view the Details of Busses available* .

**Reserve Seats/ Booking**

* *This function allows users to book tickets and Reserve Seats. ● The system will display the total amount at the end.*

**View Status Board**

* *This function allows the user to see the current booking status of the bus.*
* *It shows the seats reserved by the passengers' names and the rest are marked empty.*

**Cancel Booking**

* *This function allows users to cancel the previously reserved tickets.*
* *The cancelled seat will now be marked empty.*

# 

# 4. Test Plan

## 4.1 Introduction

The test plan is designed to identify the items and features to be tested. The method of testing to be done for the bus reservation system is described below. The given test cases are tested for validity and accuracy. If the output of test cases match with the expected results, the test passes.

## 4.2 Test Strategy

#### 4.2.1 Scope of Testing

##### 4.2.1.1 Features to be tested

|  |  |
| --- | --- |
| **Module**  **Name** | **Description** |
| Login | Existing users can login to the system. |
| Bus List | Users can view the List of Busses. |
| Status Board | The user can view the current status of the bus. It also shows the seats that have been already reserved. |
| Booking | Users can book as many desired seats from the available ones. The total cost will be displayed at the end of the booking. |
| Cancel | Users can choose to cancel the previously reserved ticket. |

|  |  |
| --- | --- |
|  | Refunds will be initiated and the refund amount will be displayed. |
| View  Bookings | Users can view the reserved seats by checking the status board. |

##### 4.2.1.2 Features not to be tested

The features that are not to be tested are:

1. User Interfaces
2. Hardware interfaces

## 4.3 Test Objective

The test objectives are to verify the functionality of the bus reservation system. The focus is to be given on testing the functionalities such as Reserving Seats, Displaying Total, Viewing Reserved Seats in the status board ..etc to make sure these functions are working efficiently.

## 4.4 Test Criteria

#### 4.4.1 Suspension Criteria

If more than 45% of test cases are failed then the testing is suspended until all fail cases are fixed.

#### 4.4.2 Exit Criteria

1. 100% Test coverage.
2. All manual test cases are executed.

## 4.5 Test Environment

All test cases are executed in CodeBlocks IDE.

# 5.Test Cases

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Test**  **Case**  **ID** | | **Test Scenario** | | **Test Steps** | | **Test Data** | |
| TU01 | | Check user login with valid data | | 1. Enter Username 2. Enter Password | | Username= user  Password= pass | |
| TU02 | | Check login with invalid data | | 1. Enter Username 2. Enter Password | | Username=user  Password= abc | |
| TU03 | | Check if the user can view the bus list. | | 1.Select the option bus list | | Enter = 1 | |
| TU04 | | Check if user can book tickets. | | Choose Book Tickets Option.  1.Enter Seat No.  2.Enter Name | | Seat no. = 7  Name = “Dhyan” | |
| TU05 | | Check if the user can view the reserved seats. (Check Status) | | Choose bus status option.  1.Enter Bus No. | | Bus No. = 1 | |
| TU06 | | Check if the user can cancel reserved tickets. | | Choose Cancel option  1.Enter bus no.  .2.Enter Seat no. | | Choose Cancel option  Bus number= 1    Seat Number=1 | |
| TU07 | | Check status | | Choose bus status option.  1.Enter Bus No. | | Choose bus status option.  Bus No. = 1 | |

**6. Expected Result**

|  |  |  |
| --- | --- | --- |
| **Test**  **Case**  **ID** | **Test Scenario** | **Expected Result** |
| TU01 | Check login with valid data. | Login should be successful |
| TU02 | Check login with invalid data. | Error message should be displayed |
| TU03 | Check if the user can view the bus list. | Users should be able to view the list of Busses already stored within the system. |
| TU04 | Check if the user can book tickets. | Users should be able to book tickets and the total amount should be displayed at the end of booking. |
| TU05 | Check if the user can view the reserved seats. | Name of the passenger should be given at the reserved seat number. |
| TU06 | Check if the user can cancel the reserved ticket. | Users should be able to cancel the reserved seat and the refund should be issued at the end. |
| TU07 | Check if the cancelled ticket is now marked empty (Checking Status) | The cancelled seats should now be empty. |

# 7. Conclusion

The Bus reservation System is a very useful system which can save a lot of time and manual work done for reserving seats and cancellation.

This system can be made really effective especially during this pandemic era, wherein people can reserve tickets in advance. Overcrowding can be avoided, letting only limited no of reservations. Also there is no need for giving printed tickets, thereby passengers and employees can avoid direct contact between them. This will be especially beneficial at the present time, to reserve bus tickets maintaining social distancing and following existing protocols.

Moreover this system also allows passengers to book tickets, well in advance before their journeys. Thus it wipes out the last minute tensions and uncertainty.