PROJECT REPORT

TITLE:BUS SEAT BOOKING SYSTEM

Submitted by :Kummara Pallavi

Submitted to :Global Quest Technologies

Date :February 13 2026

Abstract:

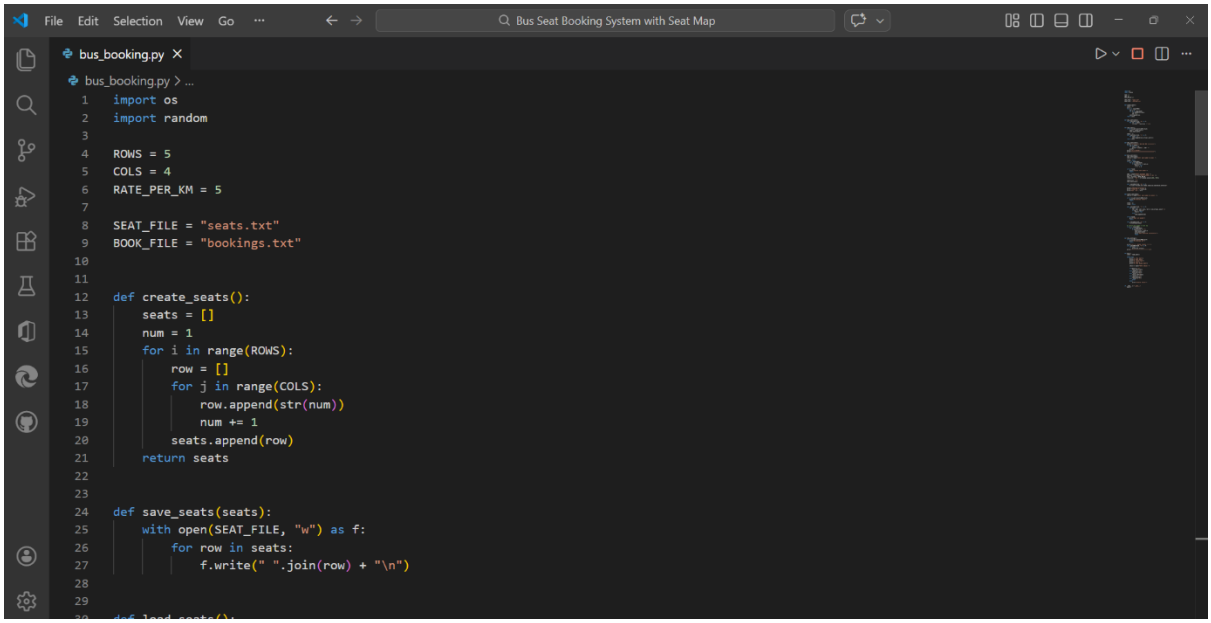
The Bus Seat Booking System is a console-based application developed using Python that simulates a real-time seat reservation process for bus travel. The system allows users to view available seats, book seats, cancel bookings, and view booked ticket details through a simple terminal interface. The primary objective of this project is to demonstrate the implementation of file handling, data persistence, and user interaction in a console environment.

The application maintains a seat map representing the bus layout and updates the seat status dynamically based on user actions. When a user books a seat, the system records passenger details such as name, seat number, travel distance, and fare. The fare is calculated based on the distance traveled using a predefined rate per kilometer. All booking information is stored in text files to ensure that the data is preserved even after the program is terminated.

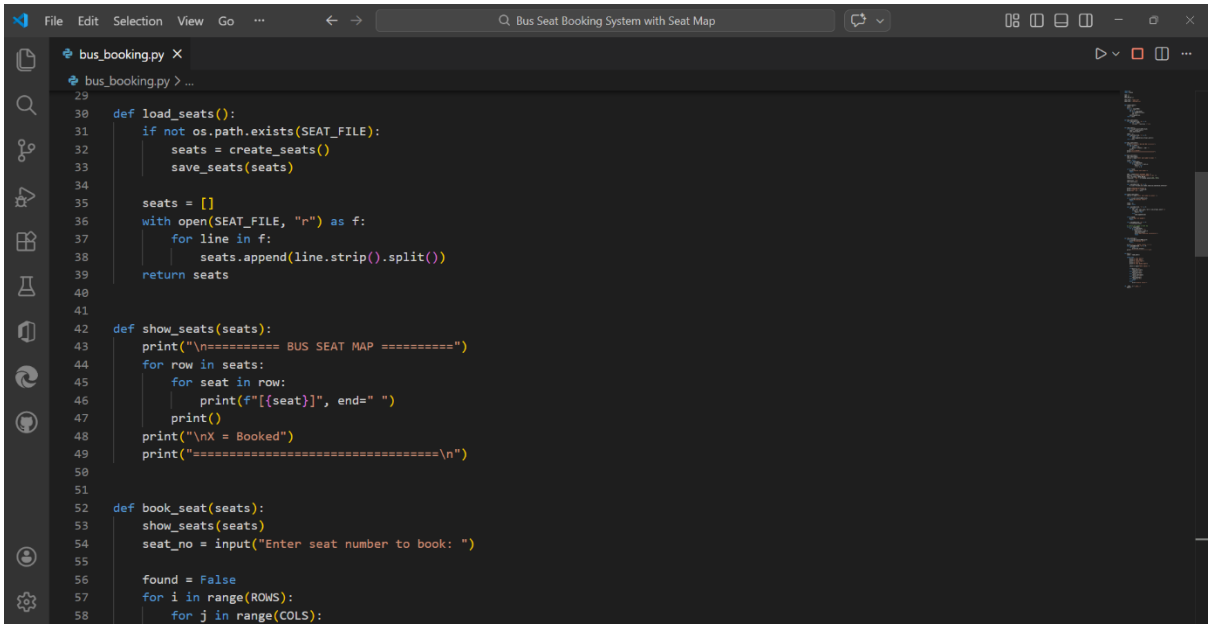
The system also provides functionality to cancel a booked seat, which updates both the seat map and the booking records accordingly. Additionally, users can view the list of all booked tickets at any time. This project emphasizes the practical use of Python concepts such as lists, loops, conditional statements, functions, file handling, and string manipulation.

Overall, the Bus Seat Booking System offers a simple yet effective demonstration of how reservation systems work in real-world scenarios while operating entirely through a console interface.

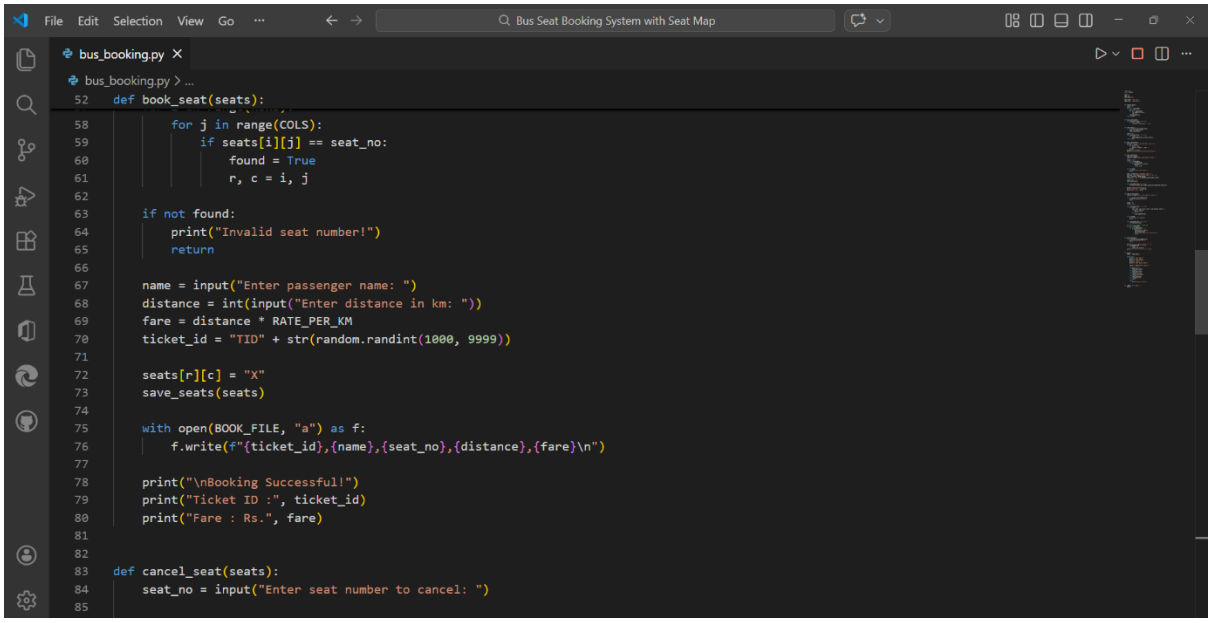
SOURCE CODE:

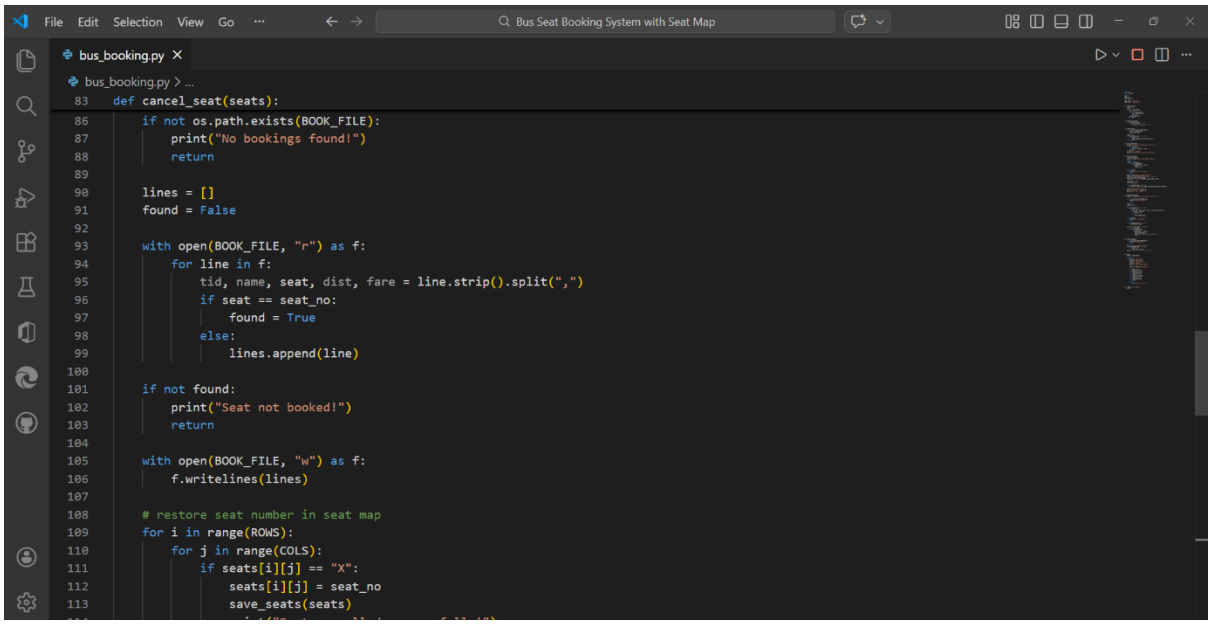


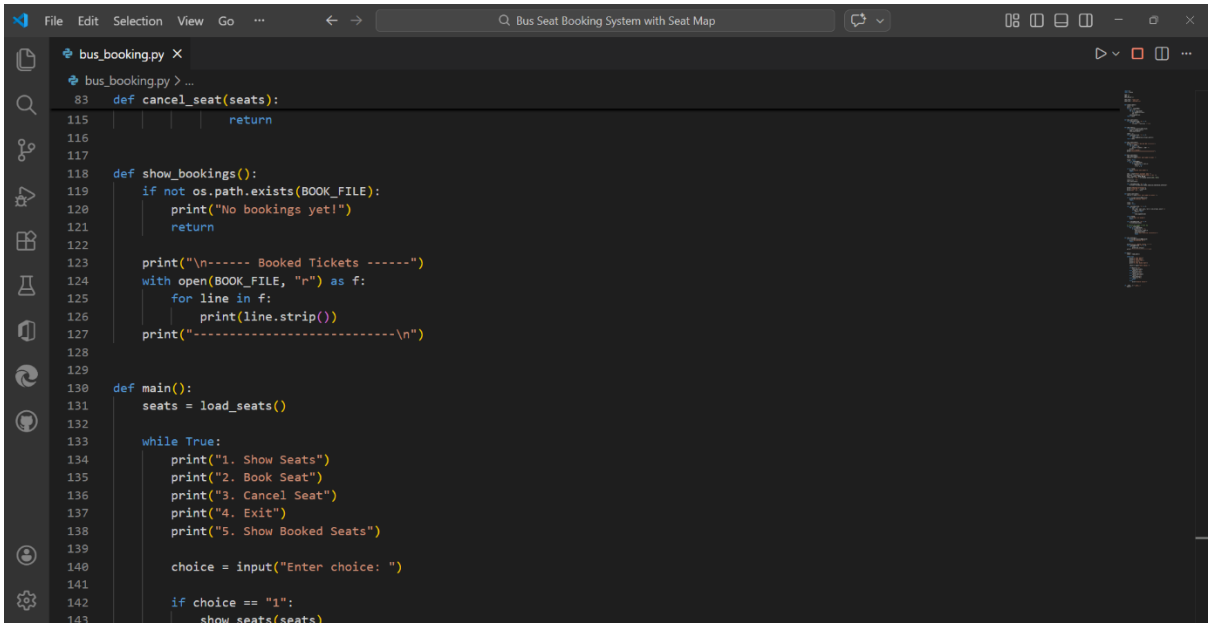
```python
import os
import random

ROWS = 5
COLS = 4
RATE_PER_KM = 5

SEAT_FILE = "seats.txt"
BOOK_FILE = "bookings.txt"


def create_seats():
    seats = []
    num = 1
    for i in range(ROWS):
        row = []
        for j in range(COLS):
            row.append(str(num))
            num += 1
        seats.append(row)
    return seats


def save_seats(seats):
    with open(SEAT_FILE, "w") as f:
        for row in seats:
            f.write(" ".join(row) + "\n")
```

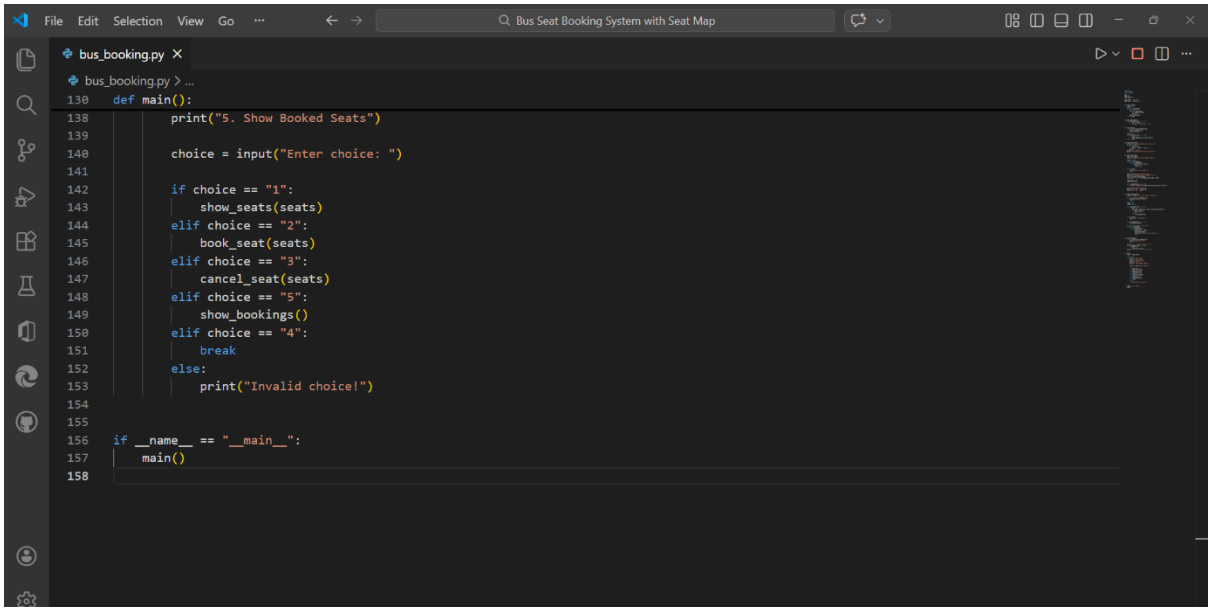


```python
def load_seats():
    if not os.path.exists(SEAT_FILE):
        seats = create_seats()
        save_seats(seats)

    seats = []
    with open(SEAT_FILE, "r") as f:
        for line in f:
            seats.append(line.strip().split())
    return seats


def show_seats(seats):
    print("\n========== BUS SEAT MAP ==========")
    for row in seats:
        for seat in row:
            print(f"[{seat}]", end=" ")
        print()
    print("\nX = Booked")
    print("==================================\n")


def book_seat(seats):
    show_seats(seats)
    seat_no = input("Enter seat number to book: ")

    found = False
    for i in range(ROWS):
        for j in range(COLS):
```

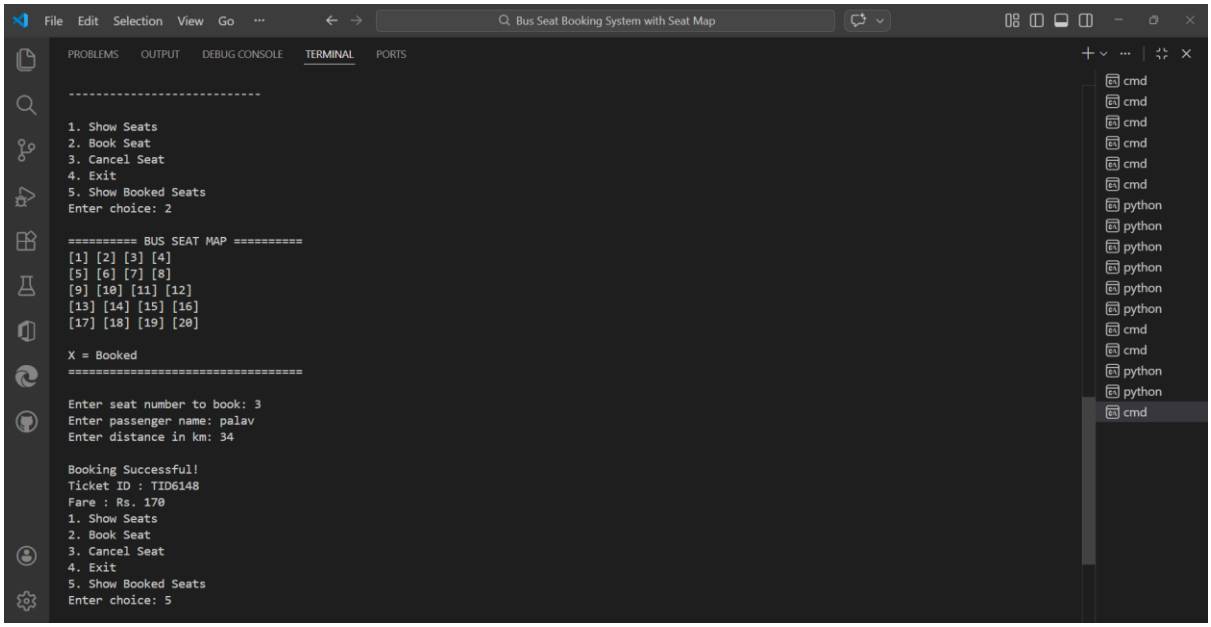

File Edit Selection View Go
Bus Seat Booking System with Seat Map
bus_booking.py
bus_booking.py > ...
def book_seat(seats):
for j in range(COLS):
if seats[i][j] == seat_no:
found = True
r, c = i, j
if not found:
print("Invalid seat number!")
return
name = input("Enter passenger name: ")
distance = int(input("Enter distance in km: "))
fare = distance * RATE_PER_KM
ticket_id = "TID" + str(random.randint(1000, 9999))
seats[r][c] = "X"
save_seats(seats)
with open(BOOK_FILE, "a") as f:
f.write(f"{ticket_id},{name},{seat_no},{distance},{fare}\n")
print("\nBooking Successful!")
print("Ticket ID :", ticket_id)
print("Fare : Rs.", fare)
def cancel_seat(seats):
seat_no = input("Enter seat number to cancel: ")

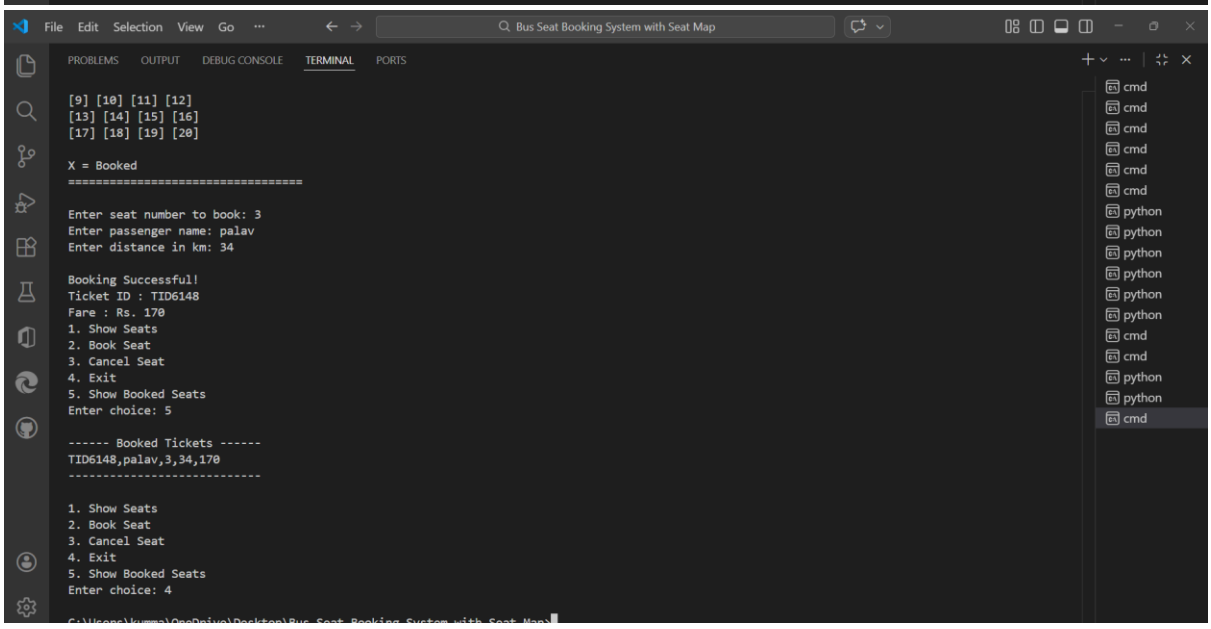

File Edit Selection View Go
Bus Seat Booking System with Seat Map
bus_booking.py
bus_booking.py > ...
def cancel_seat(seats):
if not os.path.exists(BOOK_FILE):
print("No bookings found!")
return
lines = []
found = False
with open(BOOK_FILE, "r") as f:
for line in f:
tid, name, seat, dist, fare = line.strip().split(",")
if seat == seat_no:
found = True
else:
lines.append(line)
if not found:
print("Seat not booked!")
return
with open(BOOK_FILE, "w") as f:
f.writelines(lines)
# restore seat number in seat map
for i in range(ROWS):
for j in range(COLS):
if seats[i][j] == "X":
seats[i][j] = seat_no
save_seats(seats)


File Edit Selection View Go
Bus Seat Booking System with Seat Map
bus_booking.py
bus_booking.py > ...
def cancel_seat(seats):
return
def show_bookings():
if not os.path.exists(BOOK_FILE):
print("No bookings yet!")
return
print("\n------ Booked Tickets ------")
with open(BOOK_FILE, "r") as f:
for line in f:
print(line.strip())
print("----------------------------\n")
def main():
seats = load_seats()
while True:
print("1. Show Seats")
print("2. Book Seat")
print("3. Cancel Seat")
print("4. Exit")
print("5. Show Booked Seats")
choice = input("Enter choice: ")
if choice == "1":


File Edit Selection View Go
Bus Seat Booking System with Seat Map
bus_booking.py
bus_booking.py > ...
def main():
print("5. Show Booked Seats")
choice = input("Enter choice: ")
if choice == "1":
show_seats(seats)
elif choice == "2":
book_seat(seats)
elif choice == "3":
cancel_seat(seats)
elif choice == "5":
show_bookings()
elif choice == "4":
break
else:
print("Invalid choice!")
if __name__ == "__main__":
main()

OUTPUT:


File Edit Selection View Go
Bus Seat Booking System with Seat Map
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
---
1. Show Seats
2. Book Seat
3. Cancel Seat
4. Exit
5. Show Booked Seats
Enter choice: 2
========== BUS SEAT MAP ==========
[1] [2] [3] [4]
[5] [6] [7] [8]
[9] [10] [11] [12]
[13] [14] [15] [16]
[17] [18] [19] [20]
X = Booked
==================================
Enter seat number to book: 3
Enter passenger name: palav
Enter distance in km: 34
Booking Successful!
Ticket ID : TID6148
Fare : Rs. 170
1. Show Seats
2. Book Seat
3. Cancel Seat
4. Exit
5. Show Booked Seats
Enter choice: 5


File Edit Selection View Go
Bus Seat Booking System with Seat Map
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
[9] [10] [11] [12]
[13] [14] [15] [16]
[17] [18] [19] [20]
X = Booked
==================================
Enter seat number to book: 3
Enter passenger name: palav
Enter distance in km: 34
Booking Successful!
Ticket ID : TID6148
Fare : Rs. 170
1. Show Seats
2. Book Seat
3. Cancel Seat
4. Exit
5. Show Booked Seats
Enter choice: 5
------ Booked Tickets ------
TID6148,palav,3,34,170
---
1. Show Seats
2. Book Seat
3. Cancel Seat
4. Exit
5. Show Booked Seats
Enter choice: 4

GLOBAL QUEST
TECHNOLOGIES

Global Quest Technologies

#324,2nd Floor, 3 A Cross, Near
Seshadripuram First Grade College,
Above City Union Bank,
Yelahanka New Town,
Bengaluru-560064

+91 9448 403 469 | 080-49720009
info@gqtech.in | www.gqtech.in