PROJECTS/RAILWAY MANAGEMENT SYSTEM

```
class Account:
    def __init__(self, username, password):
        self.username = username
        self.password = password

class Passenger:
    def __init__(self, name, age, gender, phone_number):
        self.name = name
        self.age = age
        self.gender = gender
        self.phone_number = phone_number

    def __str__(self):
        return f"Name: {self.name}, Age: {self.age}, Gender: {self.gender}, Phone Number: {self.phone_number}"

class Train:
    def __init__(self, train_number, train_name, source, destination, total_seats):
        self.train_number = train_number
        self.train_name = train_name
        self.source = source
        self.destination = destination
        self.total_seats = total_seats
        self.booked_seats = 0
        self.passenger_list = []  # List to store passenger details

    def check_availability(self):
        available_seats = self.total_seats - self.booked_seats
        return available_seats

    def book_ticket(self, num_seats):
        available_seats = self.check_availability()
        if available_seats >= num_seats:
            print(f"Booking {num_seats} seat(s) on train {self.train_name}.")
            for i in range(num_seats):
                passenger = self.get_passenger_details(i+1)
                self.passenger_list.append(passenger)
            self.booked_seats += num_seats
            print(f"Successfully booked {num_seats} seat(s).")
            print("Thank you for booking with us!")
        else:
            print(f"Not enough seats available. Only {available_seats} seat(s) left.")

    def cancel_ticket(self, num_seats):
        if self.booked_seats >= num_seats:
            self.booked_seats -= num_seats
            print(f"Successfully canceled {num_seats} seat(s) on train {self.train_name}.")
        else:
```

```
            print("Error: You are trying to cancel more seats than you have booked.")

    def get_passenger_details(self, seat_number):
        print(f"\nEnter details for passenger {seat_number}:")
        name = input("Enter Name: ")
        age = input("Enter Age: ")
        gender = input("Enter Gender (M/F/O): ")
        phone_number = input("Enter Phone Number: ")
        passenger = Passenger(name, age, gender, phone_number)
        return passenger

    def __str__(self):
        available_seats = self.check_availability()
        return f"Train number: {self.train_number}\nSource Station: {self.source}\nDestination Station: {self.destination}\nTotal seats: {self.total_seats}\nAvailable seats: {available_seats}\n"

class RailwayTicketSystem:
    def __init__(self):
        self.trains = {}
        self.initialize_trains()  # Predefine trains when the system starts

    def initialize_trains(self):
        # Predefined trains
        self.add_train(24450, "Express 101", "Secunderabad", "Bangalore", 200)
        self.add_train(24457, "Bangalore Mysore Express", "Bangalore", "Mysore", 200)

    def add_train(self, train_number, train_name, source, destination, total_seats):
        if train_number in self.trains:
            print("Train with this number already exists.")
        else:
            new_train = Train(train_number, train_name, source, destination, total_seats)
            self.trains[train_number] = new_train
            print(f"Train {train_name} added successfully.")

    def display_trains(self):
        if not self.trains:
            print("No trains available.")
        else:
            print("\nAvailable Trains:")
            for train in self.trains.values():
                print(train)

    def book_ticket(self, train_number, num_seats):
        if train_number in self.trains:
            train = self.trains[train_number]
            train.book_ticket(num_seats)
        else:
            print("Invalid train number.")

```

```
    def cancel_ticket(self, train_number, num_seats):
        if train_number in self.trains:
            train = self.trains[train_number]
            train.cancel_ticket(num_seats)
        else:
            print("Invalid train number.")

class RailwayTicketApp:
    def __init__(self):
        self.accounts = {}  # Dictionary to store user accounts
        self.logged_in_account = None  # Store the logged-in user's account

    def register_account(self):
        username = input("Enter username: ")
        if username in self.accounts:
            print("Username already exists. Please choose another one.")
        else:
            password = input("Enter password: ")
            self.accounts[username] = Account(username, password)
            print("Account successfully created!")

    def login_account(self):
        username = input("Enter username: ")
        password = input("Enter password: ")
        if username in self.accounts and self.accounts[username].password ==
password:
            self.logged_in_account = self.accounts[username]
            print(f"Login successful! Welcome {username}.")
            return True
        else:
            print("Invalid username or password.")
            return False

    def logout_account(self):
        self.logged_in_account = None
        print("You have been logged out.")

    def is_logged_in(self):
        return self.logged_in_account is not None

def main():
    railway_system = RailwayTicketSystem()
    app = RailwayTicketApp()

    while True:
        if app.is_logged_in():
            # Show Railway Ticket Management options
            print("\nRailway Ticket Management System")
            print("1. Display Trains")
            print("2. Book Ticket")
            print("3. Cancel Ticket")
            print("4. Logout")
            choice = input("Enter your choice: ")

```

```
            if choice == "1":
                railway_system.display_trains()

            elif choice == "2":
                train_number = int(input("Enter Train number to book ticket: "))
                num_seats = int(input("Enter number of seats to book: "))
                railway_system.book_ticket(train_number, num_seats)

            elif choice == "3":
                train_number = int(input("Enter Train number to cancel ticket: "))
                num_seats = int(input("Enter number of seats to cancel: "))
                railway_system.cancel_ticket(train_number, num_seats)

            elif choice == "4":
                app.logout_account()

            else:
                print("Invalid choice. Please try again.")

        else:
            # Show login/register menu
            print("\nRailway Ticket Management System")
            print("1. Register Account")
            print("2. Login")
            print("3. Exit")
            choice = input("Enter your choice: ")

            if choice == "1":
                app.register_account()

            elif choice == "2":
                if app.login_account():
                    continue  # If login is successful, go back to the Railway Ticket Management System.

            elif choice == "3":
                print("Exiting Railway Ticket Management System.")
                break

            else:
                print("Invalid choice. Please try again.")

if __name__ == "__main__":
    main()
```