Train Seat Reservation System

Project Overview

The **Train Seat Reservation System** is a web-based application designed to manage seat reservations in a train coach dynamically. The application provides an interactive interface to visualize available, booked, and reserved seats. It handles user requests for booking between 1 to 7 seats while ensuring efficient allocation and error handling for edge cases.

Features

- **Dynamic Seat Allocation**:
  - Allows users to book between 1 to 7 seats in a single request.
  - Seats are booked contiguously if possible; otherwise, non-contiguous seats are allocated.
- **Real-Time Feedback**:
  - Users receive immediate feedback via notifications on booking success or failure.
- **Visualization**:
  - Seat availability is displayed with color-coded markers:
    - Green: Vacant
    - Red: Booked
    - Blue: Reserved
- **Reset Functionality**:
  - Allows resetting all seats to vacant state.

Technology Stack

- **Frontend Framework**: React.js
- **Styling**: Tailwind CSS
- **Notifications**: React Toastify

Components

1. **`App` Component**
   - Manages the state of the seat map, number of seats to book, and booked seats.
   - Contains logic for booking and resetting seats.
   - Integrates React Toastify for user notifications.

2. **`SeatMap` Component**
    - Displays the seat layout using a grid structure.
    - Dynamically updates seat colors based on their state (vacant, booked, reserved).

Core Functionalities

1. **Seat Booking**
    - Users can input the number of seats they wish to book (1-7).
    - The algorithm first attempts to allocate seats contiguously within a row.
    - If contiguous seats are unavailable, it falls back to booking non-contiguous seats.
2. **Validation**
    - Ensures user inputs are within the allowed range (1-7).
    - Displays appropriate error messages for invalid inputs or insufficient seat availability.
3. **Resetting**
    - Resets the seat map to its initial vacant state.
    - Clears the list of booked seats and provides confirmation to the user.
4. **User Notifications**
    - Utilizes Toastify to provide feedback for:
        - Successful bookings.
        - Errors (e.g., invalid input, insufficient seats).
        - Reset operations.

Seat Structure

The train coach contains:

- **Rows 1-10**: 7 seats per row.
- **Rows 11-12**: 3 seats per row.

Key Algorithms

1. **Contiguous Seat Allocation**:
    - Scans each row to find a set of contiguous vacant seats.
    - Allocates them in a single step if enough seats are available.
2. **Fallback to Non-Contiguous Allocation**:
    - Iterates through all rows to allocate remaining seats individually if contiguous seats are not found.

Error Handling

- Validates user input for seat requests (1-7 seats).
- Displays error if:
    - Insufficient vacant seats are available.
    - Invalid inputs are provided.

Assumptions

- The train layout is fixed with 12 rows.
- Users can only book up to 7 seats in a single transaction.
- Seat states are:
    - `0`: Vacant
    - `1`: Booked
    - `2`: Reserved (future use case).

Submission Requirements

1. **GitHub Repo Link**: **https://github.com/AbhisekhNayek/Train-Booking-System**
2. **Working URL**: **https://train-booking-system-cyan.vercel.app**

Future Enhancements

- Add user authentication for personalized bookings.
- Include a backend to persist seat states.
- Implement advanced seat preferences (e.g., window, aisle).
- Optimize seat allocation algorithms for larger seat maps.