

**Group No.: \_\_J\_\_      Project No.: \_\_J\_\_**

**Project Name: Railways Management System**

## **PHASE - I**

### **REQUIREMENTS DOCUMENT**

#### **1. MINIWORLD:**

The Railways Management System is a software system designed to manage the operations of a railway system. The system will model the following entities: stations, tracks, trains, and train schedules. Each station is identified by a unique ID and has a name and location associated with it. Stations are connected by tracks, which are unidirectional and have a source station and a destination station. For simplicity, only one track exists between any two stations, and all the tracks are put together to form a graph.

Each train is identified by a unique ID and has a name, starting station, ending station, and intermediate stops associated with it. The system will keep track of the train schedule, which records what time a train passes through each station on its route. You can assume for simplicity that each train reaches its destination on the same day and that every train runs every day. Passenger bookings are also part of the system, consisting of train, date, from-station, to-station, coach, seat, and passenger name. Each booking is associated with a specific train schedule.

#### **2. DATABASE REQUIREMENTS:**

The system requires a database to store information about stations, tracks, trains, train schedules, and passenger bookings.

Each station will have an entry in the database, with fields for station ID, name, and location.

Each track will have an entry in the database, with fields for track ID, source station ID, and destination station ID.

Each train will have an entry in the database, with fields for train ID, name, starting station ID, ending station ID, and a list of intermediate stops.

Each train schedule will have an entry in the database, with fields for train ID, date, station ID, time-in, time-out (same as time-in if it does not stop), and sequence number.

Each passenger booking will have an entry in the database, with fields for train ID, date, from-station ID, to-station ID, coach, seat, and passenger name.

### **3. Constraints:**

1. Each station must have a unique ID and name.
2. Each track must have a unique ID and connect two stations.
3. Each train must have a unique ID and a valid starting station, ending station, and intermediate stops.
4. Each train schedule must have a unique combination of train ID, date, and station ID.
5. Each passenger booking must have a unique combination of train ID, date, from-station ID, to-station ID, coach, and seat.
6. Train schedules must be ordered by sequence number for a given train and date.
7. A passenger booking can only be made for a valid train schedule.
8. Each train schedule must have a valid time-in and time-out for each station on its route.
9. The database must be consistent and accurate in representing the railway system.

### **4. APPLICATION REQUIREMENTS:**

There must be provision for:

1. Adding, updating, and deleting stations, tracks, trains, and train schedules in the system.
2. Searching for trains based on starting station, ending station, and date.
3. Displaying the train schedule for a specific train and date.
4. Booking and canceling passenger bookings for a specific train and date.

5. Managing the availability of seats for each coach on a train.
6. Providing a user-friendly interface for the application.
7. Ensuring the security and privacy of the data stored in the system.