NIRMA UNIVERSITY
INSTITUTE OF TECHNOLOGY
NAAC ACCREDITED 'A' GRADE

Database Management System

Innovative Assignment

Railway Management System

Group members:

Tulsi Patel (19BCE284)

Vaidahi Patel (19BCE290)

Vinit Patel (19BCE295)

Table of content

ABSTRACT

The Railway Management System facilitates the passengers to enquire about the trains available on the basis of source and destination, Booking and Cancellation of tickets, enquire about the status of the booked ticket, etc. The aim of case study is to design and develop a database maintaining the records of different trains, train status, and passengers. This project contains entity relationship model diagram based on railway management system and introduction to relation model.

INTRODUCTION

Database is an organized collection of data. The main purpose of maintaining database for Railway Management System is to reduce the manual errors involved in the booking and cancelling of tickets and make it convenient for the customers and providers to maintain the data about their customers and also about the seats available to them. Due to automation many loopholes that exist in the manual maintenance of the records can be removed. The speed of obtaining and processing the data will also be fast.

PROJECT DESCRIPTION

The aim of our project is to design and develop a database maintaining the records of different trains, train status, and passengers. The record of train includes its number, name, source, destination, and days on which it is available, whereas record of train status includes dates for which tickets can be booked, total number of seats available, and number of seats already booked. Passengers can book their tickets for the train in which seats are available. For this, passenger has to provide the desired train number and the date for which ticket is to be booked. Before booking a ticket for a passenger, the validity of train number and booking date is checked. Once the train number and booking date are validated, it is checked whether the seat is available. If yes, the ticket is booked with confirm status and corresponding ticket ID is generated which is stored along with other details of the passenger. The ticket once booked can be cancelled at any time. For this, the passenger has to provide the ticket ID (the unique key). The ticket ID is searched and the corresponding record is deleted. With this, the first ticket with waiting status also gets confirmed.

List of Entities and Attributes

ENTITY	ATTRIBUTES
User	**User_id** password first_name last_name mobile_no email age gender address
Passenger	**passenger_id** name age seat_number reservation_status gender ticket_id
Ticket	**id** status booked_user no_of_passengers
Station	**no** name arrival_time hault
Train_status	**train_no** b_seats1 b_seats2 a_seats1 a_seats2 w_seats1 w_seats2 fare1 fare2
Train	**train_no** date availability_of_seats arrival_time departure_time train_name source destination

List of Relationships

S.No	Relation Type	Entity Types Involved
1	Books	User, Train_status, Passenger, Ticket
2	Starts	Train, Station
3	Reaches	Train, Station
4	Cancel	User, Ticket
5	Stops_at	Train, Station

Functional Requirements

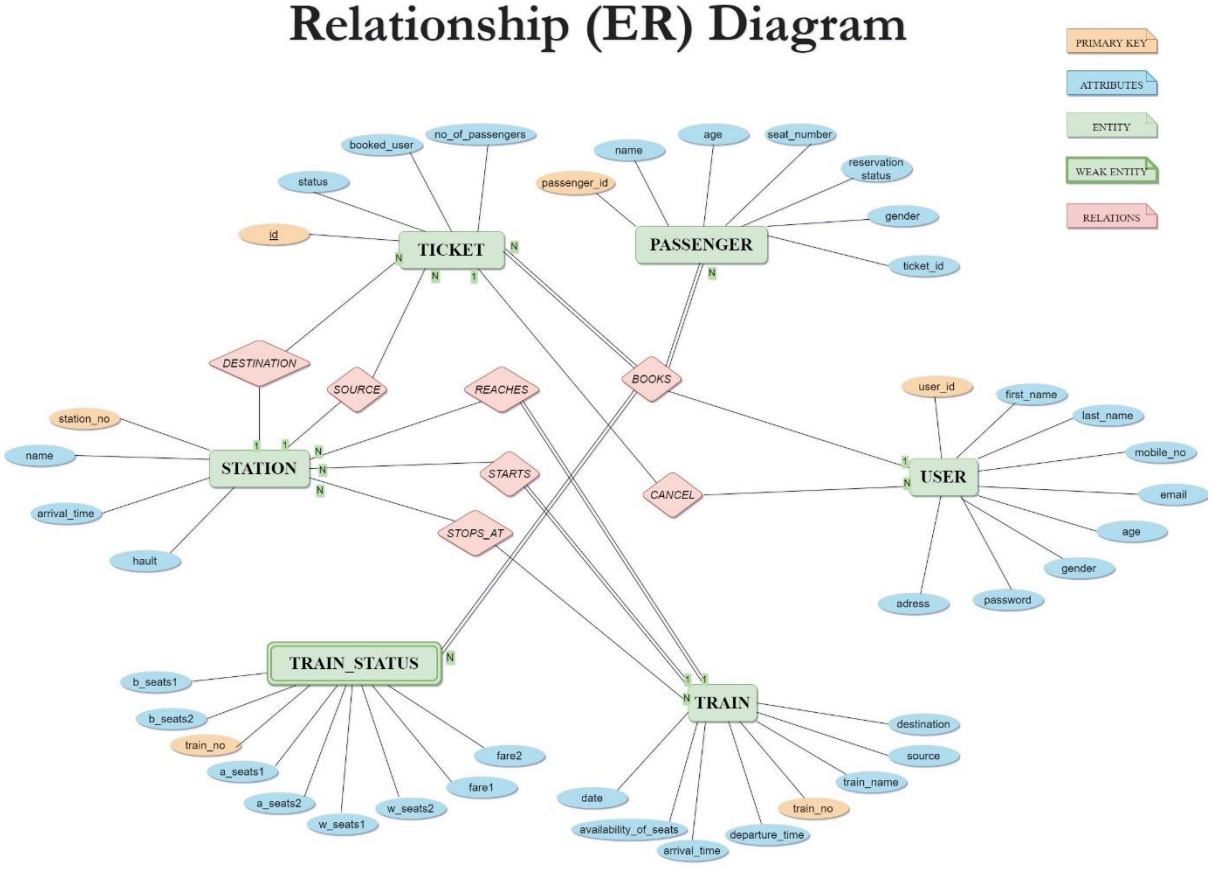
User requirements

- Seat availability: The users through the use of this system can check whether the seats are available in the particular train to travel.
- Train tickets: The users can access their train tickets through the online mode through the use of this application.
- Stations: This application can also contain the details of the various stations from the source to the destination of the passenger.
- Trains: The trains will be having the name and the id number.
- Train schedules: Even the schedules of the different trains can be mentioned through this application.
- Time in and out: For each train at every station there is a time in or time out time. It indicates that passengers should get in and get out of the train only within that particular duration of time.

System requirements

- Any Windows Operating System with DOS Support
- Oracle SQL plus 11g XE
- Oracle SQL Developer

Entity
Relationship (ER) Diagram



Relational Model