



Documentation On

## **“ONLINE RAILWAY RESERVATION”**

PG-DAC MAR 2023

*Submitted By:*

PRN NO:- 230360820019

**Kalpesh Amrutkar**

PRN NO:- 230360820053

**Siddhant Dusane**

## **Objective**

Our project introduces railway reservation system with an objective to make the reservation system more efficient, easier and fast. This project explores how computer technology can be used to solve the problem of user. The main objectives provided by this software are as follows:

- We can enquire about availability of trains
- We can reserve and cancel their seats
- We can modify the information related to

### a) Trains

- Train Schedule
- Train Name

### b) Ticket Fare

This project is dedicated to model existing railway reservation systems that aim at development of Railway Reservation System that facilitates the railway customer to manage their reservations and the railway administrator to modify the backend database in a user-friendly manner

## **Introduction**

In this emerging world of computers, almost all-manual system has switched to automated and computerized system. Therefore, we are developing the software for “Railway Reservation System” to model the present system and to remove the drawbacks of the present system. This project explores how computer technology can be used to solve the problem of user.

This being a big step in terms of improvement in the railway system it is widely accepted across the country. Rather than designing manually, we have made use of computer. Use of computer has solved many problems, which are faced during manual calculation. Once data are fed, it can perform accurate functions. Therefore, to reduce the complexity and efficiency a versatile and an outsourcing railway reservation system has been developed. This project introduces railway reservation system. It explains how reservation is being done in Indian Railways. The systematic procedure is explained. This project is developed in java language. All most all the header files have been used in this project. Proper comments have been given at desired locations to make the project user friendly. Various functions and structures are used to make a complete use of this language.

The customers are required to register on the server for getting access to the database and query result retrieval. Upon registration, each user has an account that is essentially the ‘view level’ for the customer. The account contains comprehensive information of the user entered during registration and permits the customer to get access to his/her past reservations, enquire about travel fare and availability of seats, make fresh reservations, and update his account details. Each passenger is allotted a unique ticket no. through which one can access his/her account.

## **Purpose**

The Indian Railways (IR) carries about 5.5 lakhs passengers in reserved accommodation every day. The Computerized Passenger Reservation System (PRS) facilitates the booking and cancellation of tickets from any of the 4000 terminals (i.e. PRS booking window all over the countries). These tickets can be booked or cancelled for journeys commencing in any part of India and ending in any other part, with travel time as long as 72hours and distance up to several thousand kilometers. In the given project we will be developing a website which will help users to find train details, book and cancel tickets and the exact rates of their tickets to the desired destination. With the help of online booking people can book their tickets online through internet, sitting in their home by a single click of mouse. Using their credit cards people can easily get their tickets done within minutes.

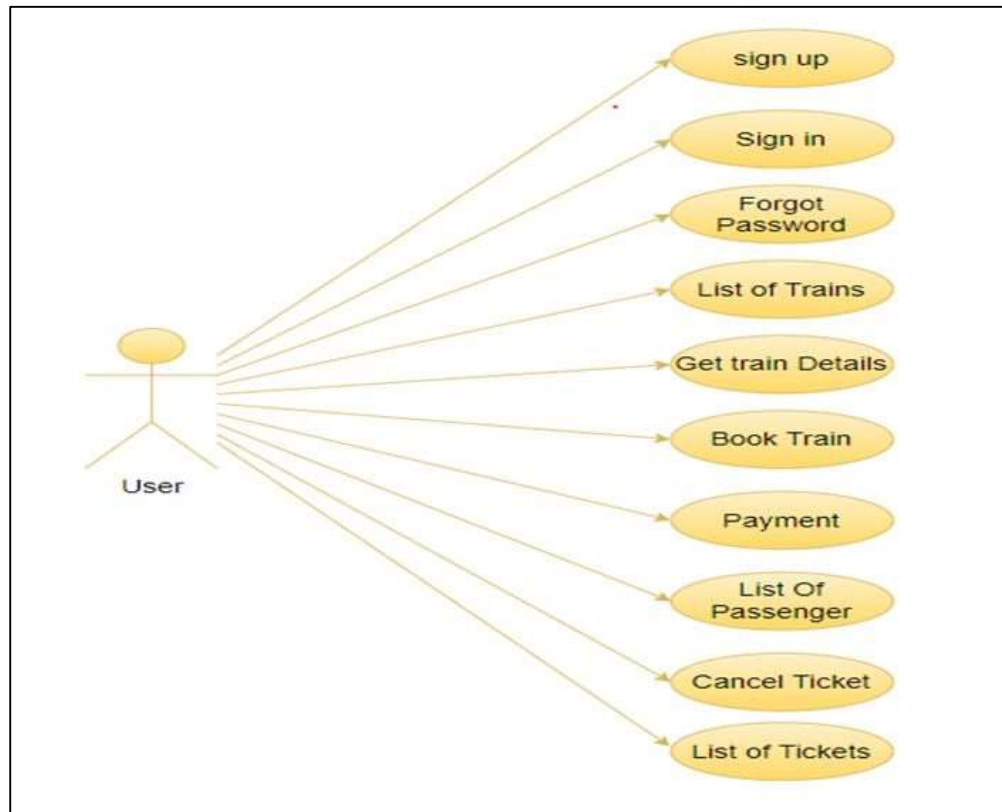
## **Scope**

The purpose of this source is to describe the railway reservation system which provides the train timing details, reservation, billing and cancellation on various types of reservation namely,

- Confirm Reservation for confirm Seat.
- Reservation against Cancellation.
- Online Reservation.
- Freight Revenue enhancement
- Passenger Revenue enhancement
- Improved & optimized service

## Requirements

### 1. Functional Requirements:



### User Account

The passenger, who will henceforth be called the 'user', will be presented with 3 choices by the reservation system, as the first step in the interaction between them. A user can choose one of these and his choice would be governed by whether he is a guest or a registered user and whether he wants to check the availability of tickets or also buy them. The terms 'registered user' and 'guest' are described below.

A user who has traveled by the railway earlier would have been given a user id and a password. This 'personal information' would be henceforth referred to as 'profile'. Such a user with a profile in DB-user shall be called a 'registered user'. A registered user will be able to check the availability of tickets as well as buy a ticket by logging into the system.

A new user, on the other hand, would either have to

- a) register himself with the system by providing personal information or
- b) log into the system as a user.

In case of 'a', the new user becomes a registered user.

A registered user can also act as a guest if he only wants to check the availability of tickets.

'Availability of tickets' always refers to viewing the train schedule for given days, the price of tickets. The system shall present the user with an option to exit from the system at any time during the following processes.

### **Registration and creation of user profile**

The system shall require a user to register, in order to carry out any transactions. It will ask the user for the following information at the least – a user id, a password, first name, last name, address, phone number, email address, sex, age. The system will automatically create a 'role' field and initialize it to 'user' in the user's profile.

### **Search Train**

Here we provided Search facility for any user to search train schedule with login into account. This will provide user an option for searching train and book for their journey. After logging in a user, the system shall request him to enter the following details – origin city and destination city. "City" is a generic term and refers to a city or town as the case may be.

After the origin and destination cities are ascertained, the system shall now access the train schedule database, referred to as 'train\_schedule', and checks if there is a direct operational service between the two cities.

The system shall now ask the user to enter the following details - class departure date and add passengers. 'Class' refers to AC/NON-AC class. This choice shall be made by the user through a dropdown menu indicating all the possible combinations of choices. 'Departure date' refers to a single date, entered through text box.

Having taken all the above input from the user, the system checks for any false entries like the departure date & all. In case of incompatibility, the system will not display any train available.

The system queries the flights database 'train\_schedule' to check which of the train on the schedule have seats available. The system displays the results in a suitable form .(a tabular form) with the following information depicted – for each trainId , departure time in origin city, arrival time in destination city, departure city, arrival city ,Ticket price and the number of seats available on that train.

There can be several trains of different types between two cities and from the Origin City. In case, the user has entered a range of dates, the system shall display all the trains for all those dates in the range. There will be a Book button in front of every row displayed n the table of flights searched.

The system will then ask for personal information of all passengers i.e. one registered user can book for multiple users. So all users will be added in the table.

The system shall now display the price of the ticket for the trip. This will be the sum of the prices for all the members of the travel party being represented by the user.

### **Making Reservations**

After having taken the user through the step 2.2, Checking Availability, The system will now ask the user if he wishes to block/buy the ticket. If yes, and

- a) if the user has been a guest, he will have to first register and become a registered user and then log onto the system.
- b) If the user is already a registered user, and if he has logged on already, he can buy the ticket, but if he has been acting as a guest, he will have to log on.

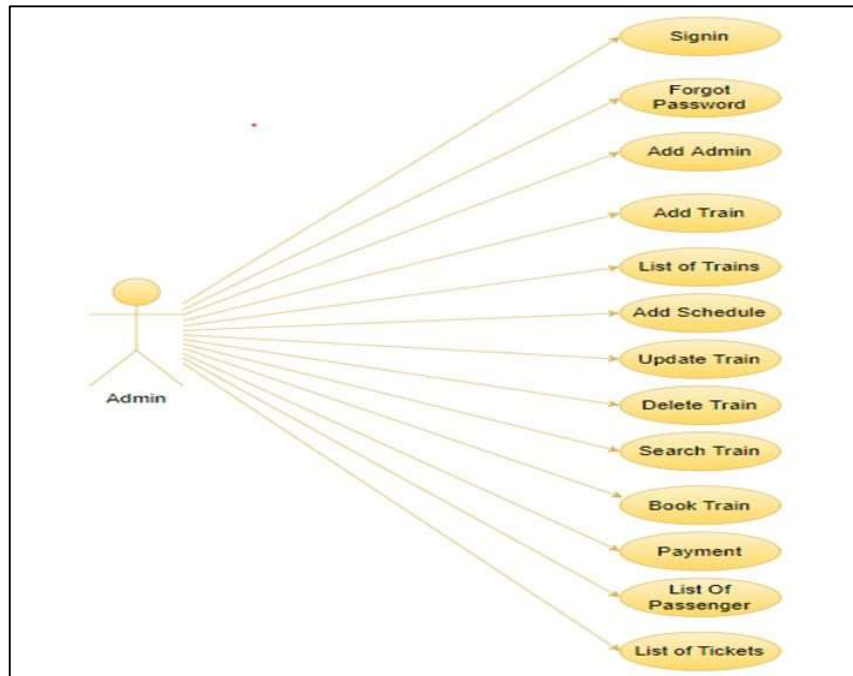
Having ensured that the user is logged on validly according to 3.4.1, the system compares the departure date with the system date. If the departure date falls within 2 weeks of the system date, the system informs the user that he has no option to block the ticket and asks him if he would like to buy it.

If the difference between the departure date and system date is more than 2 weeks, the system asks the user if he would like to buy the ticket. The system informs the user that he can block the ticket at no cost now. It also informs him that if he chooses to block the ticket, he should make a final decision before 2 weeks of the departure date. The system shall send an email to the user.

Having taken the input from the user in 3.4.2, the system shall now proceed to update the reservation database DB-reservation. It will decrement the number of available seats on the particular train for the particular class by the number of travelers being represented by the user. In case the user buys the ticket, the system asks for entering his or her bank card information and then charges the price of the ticket to his debit card number.

### **View Booking History**

The system shall allow a user to view all information about his previous bookings. It accesses Passenger table and retrieves the details of the trip and presents them to the user in a tabular format.



Admin should be able to login, add train information, add flight information, Delete train and see train details.

## 2. Non-Functional Requirements:

- Performance: The system should handle a large number of concurrent users and process transactions quickly.
- Reliability: The system should be highly reliable and available, with minimal downtime.
- Security: Ensure the security of user data, transactions, and prevent unauthorized access.
- Scalability: The system should be able to handle increasing user load and data volume without significant performance degradation.
- Usability: The system should have a user-friendly interface and be easy to navigate and use.
- Accessibility: The system should be accessible to users with disabilities, complying with accessibility standards.
- Compatibility: The system should be compatible with multiple devices, operating systems, and web browsers.
- Data Integrity: Ensure data integrity and accuracy, with appropriate backup and recovery mechanisms.
- Maintenance: The system should be easy to maintain and support, with regular updates and bug fixes.



- **Integration:** The system should integrate with other systems such as payment gateways, CRM, and reporting tools.

**Other Requirements:**

- **Hardware Interfaces**

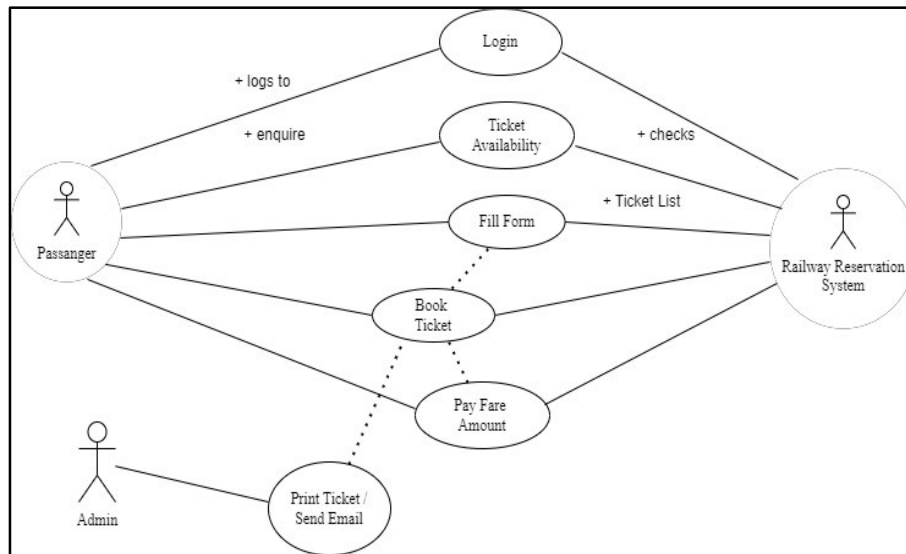
The SPMS is expected to function on Intel PIII 900 MHz Processor equivalent or above, 128 MB RAM, 20 GB HDD.

- **Software Interfaces**

The SPMS shall work on MS Windows operating systems family (MS Windows 98, MS Windows NT Workstation, MS Windows 2000, MS Windows XP). It configures to work with Oracle database. This System works on Apache Tomcat server. It uses browser IE 5.0 & above. It uses IIS 5.0 server.

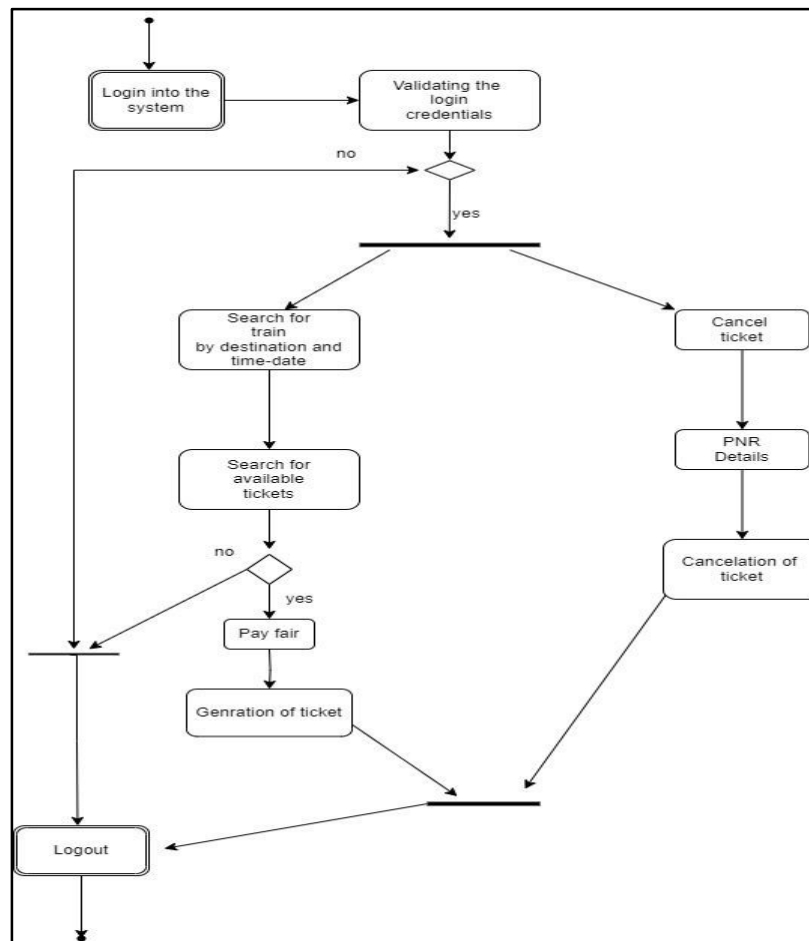
## Diagrams

### Use Case

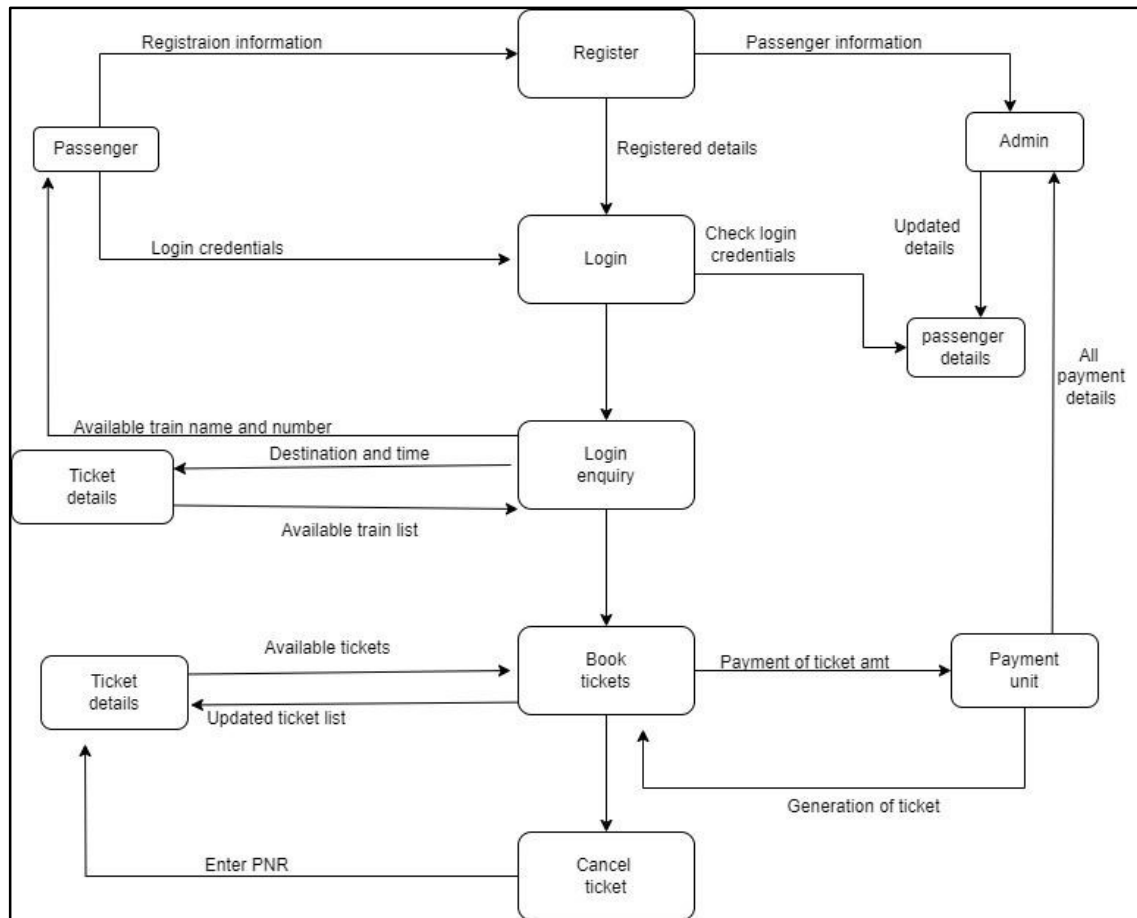


### Activity Diagram

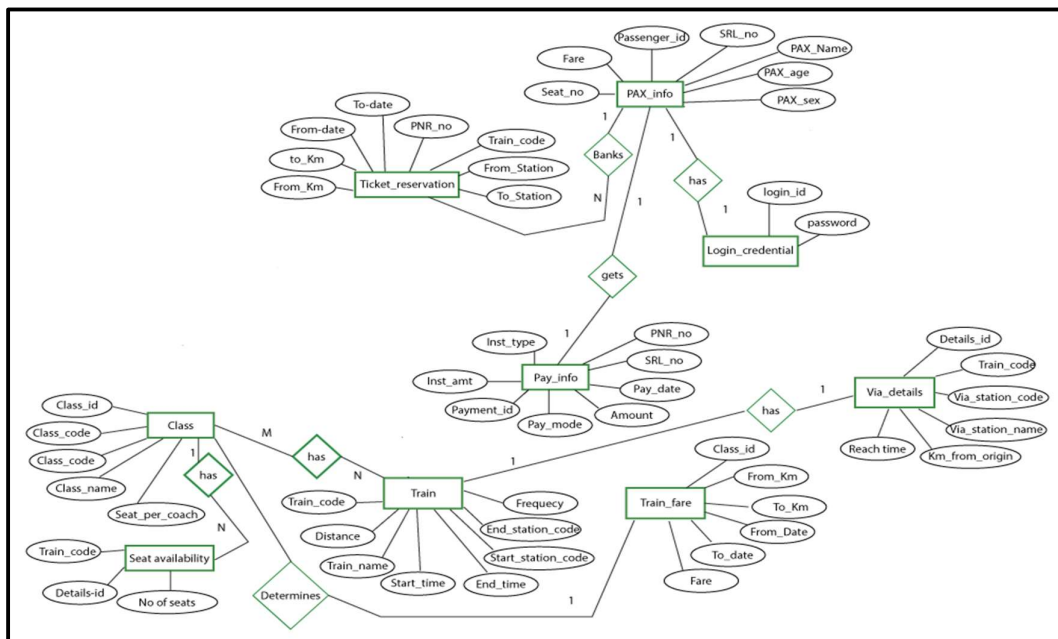
This diagram shows the flow of processes from one to another activity



## Data Flow Diagram



## E-R Diagram



The above ER diagram illustrates the key information about the railway reservation system