MINI PROJECT

RAILWAY RESERVATION SYSTEM

ABSTRACT

The Railway Reservation 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 Introduction to the Railways reservation system. It is the computerized system of reserving the seats of train seats in advanced. It is mainly used for long route. On-line reservation has made the process for the reservation of seats very much easier than ever before.

In our country India, there are number of counters for the reservation of the seats and one can easily make reservations and get tickets. Then this project contains entity relationship model diagram based on railway reservation system and introduction to relation model. There is also design of the database of the railway reservation system based on relation model. Example of some SQL queries to retrieves data from rail management database.

INTRODUCTION

Database is an organized collection of data. The data is typically organized to model aspects of reality in a way that supports processes requiring information. A DBMS makes it possible for end users to create, read, update and delete data in a database. The DBMS essentially serves as an interface between the database and end users or application programs, ensuring that data is consistently organized and remains easily accessible. The DBMS manages three important things: the data, the database engine that allows data to be accessed, locked and modified and the database schema, which defines the database's logical structure. These three foundational elements help provide concurrency, security, data integrity and uniform administration procedures. The DBMS can offer both logical and physical data independence. That means it can protect users and applications from needing to know where data is stored or having to be concerned about changes to the physical structure of data.

PROJECT DESCRIPTION

This project is about creating the database about Railway Reservation System.

The railway reservation 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. 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

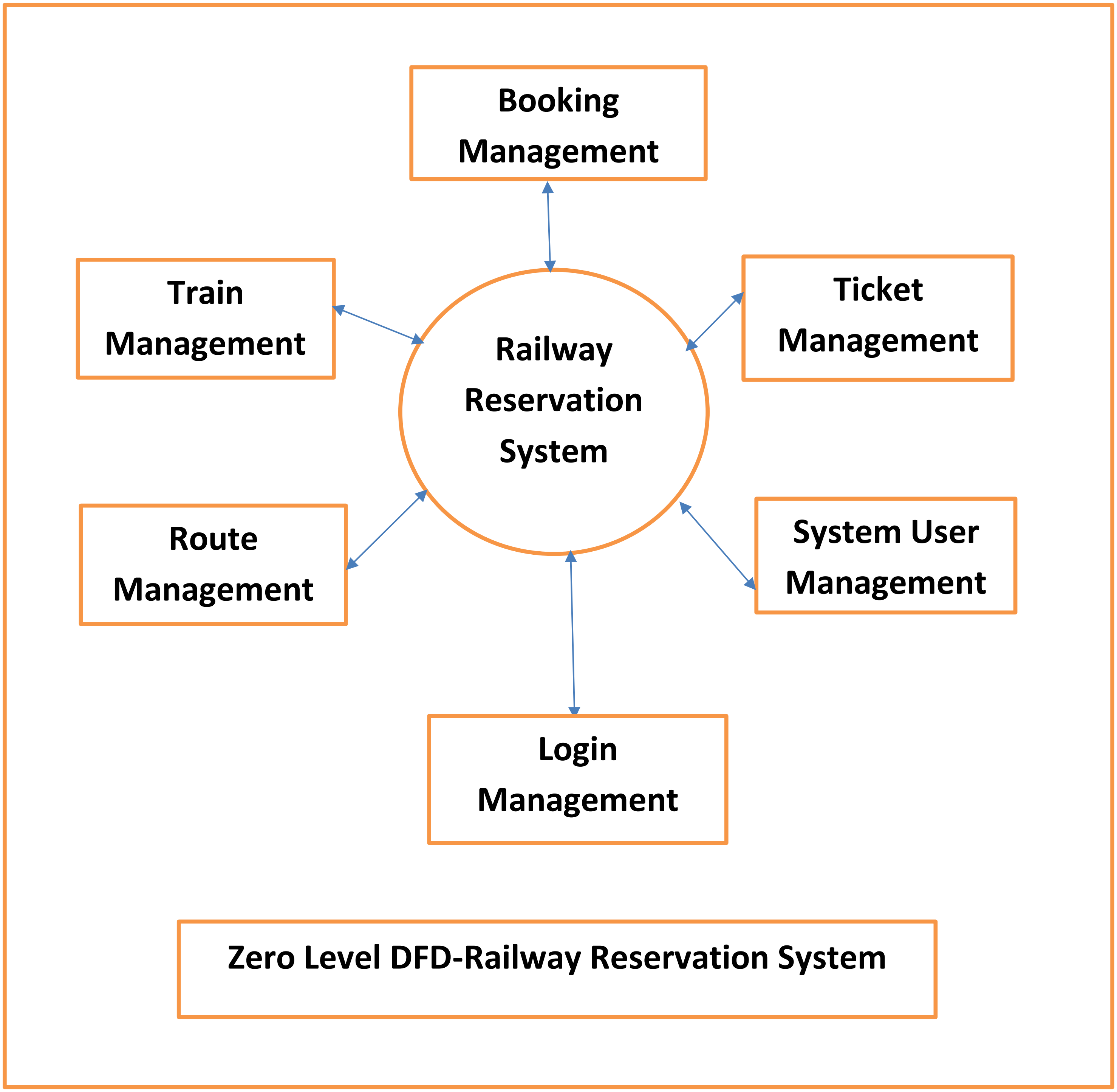
ENTITIES	**ATTRIBUTES**
USERS	**user_id** password first_name last_name gender age email adhar_no mobile_no city state pincode security_ques security_ans

PASSENGER	**passenger_id** name gender age pnr_no seat_no booked_by reservation_status
TRAIN	**Train_no** train_name source destination arrival_time Departure_time availability_of_seats train_no A_seats1 A_seats2 A_seats3 B_seats1 B_seats2 B_seats3 W_seats1 W_seats2 W_seats3
STATION	**Station_no** Station_name train_no arrival_time hault

TICKET	**Ticket_id** train_no booked_user Status no_of_passengers

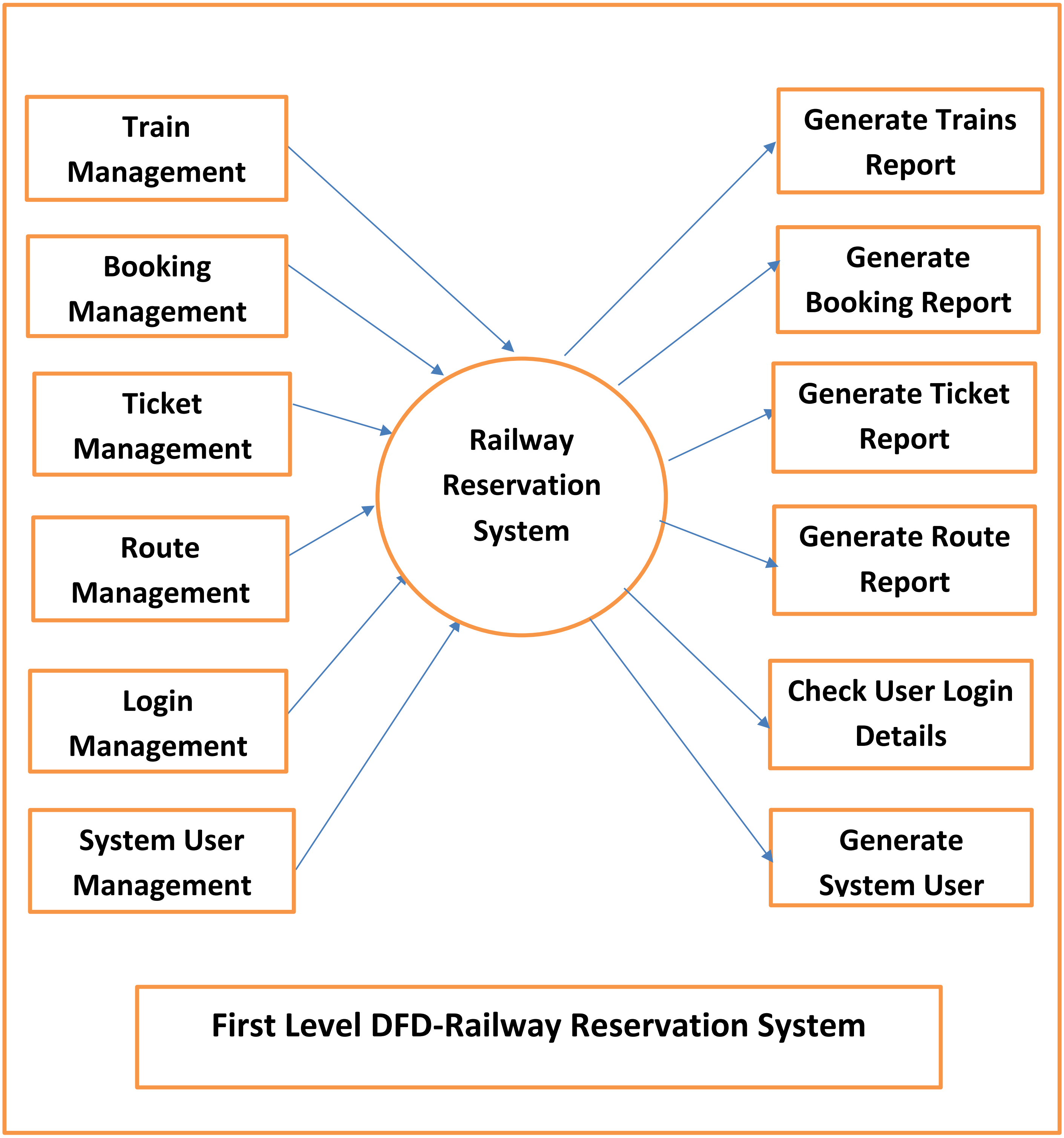
Data Flow Diagram (DFD'S)

ZERO LEVEL DFD: -

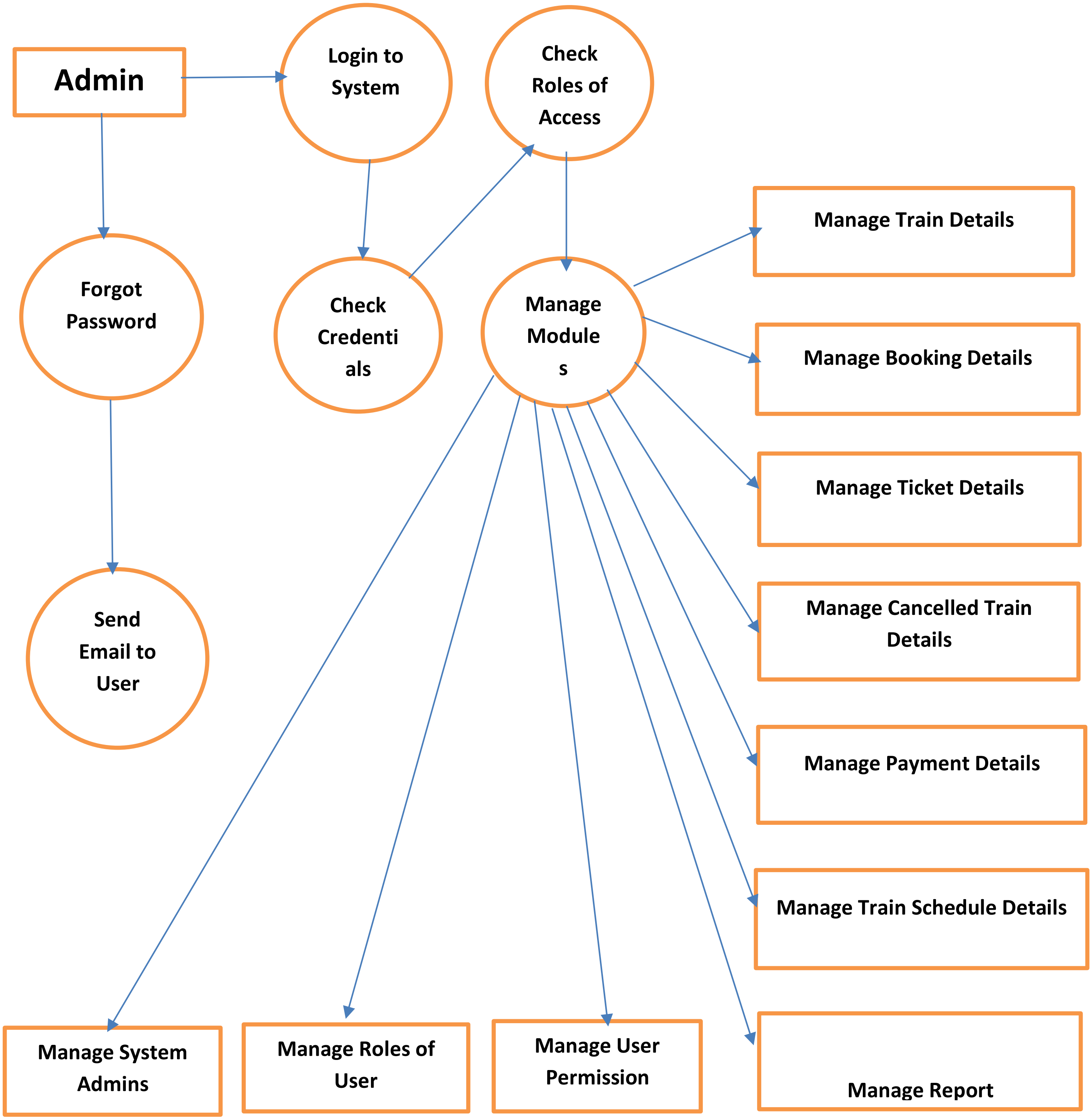


Zero Level DFD-Railway Reservation System

First Level DFD: -

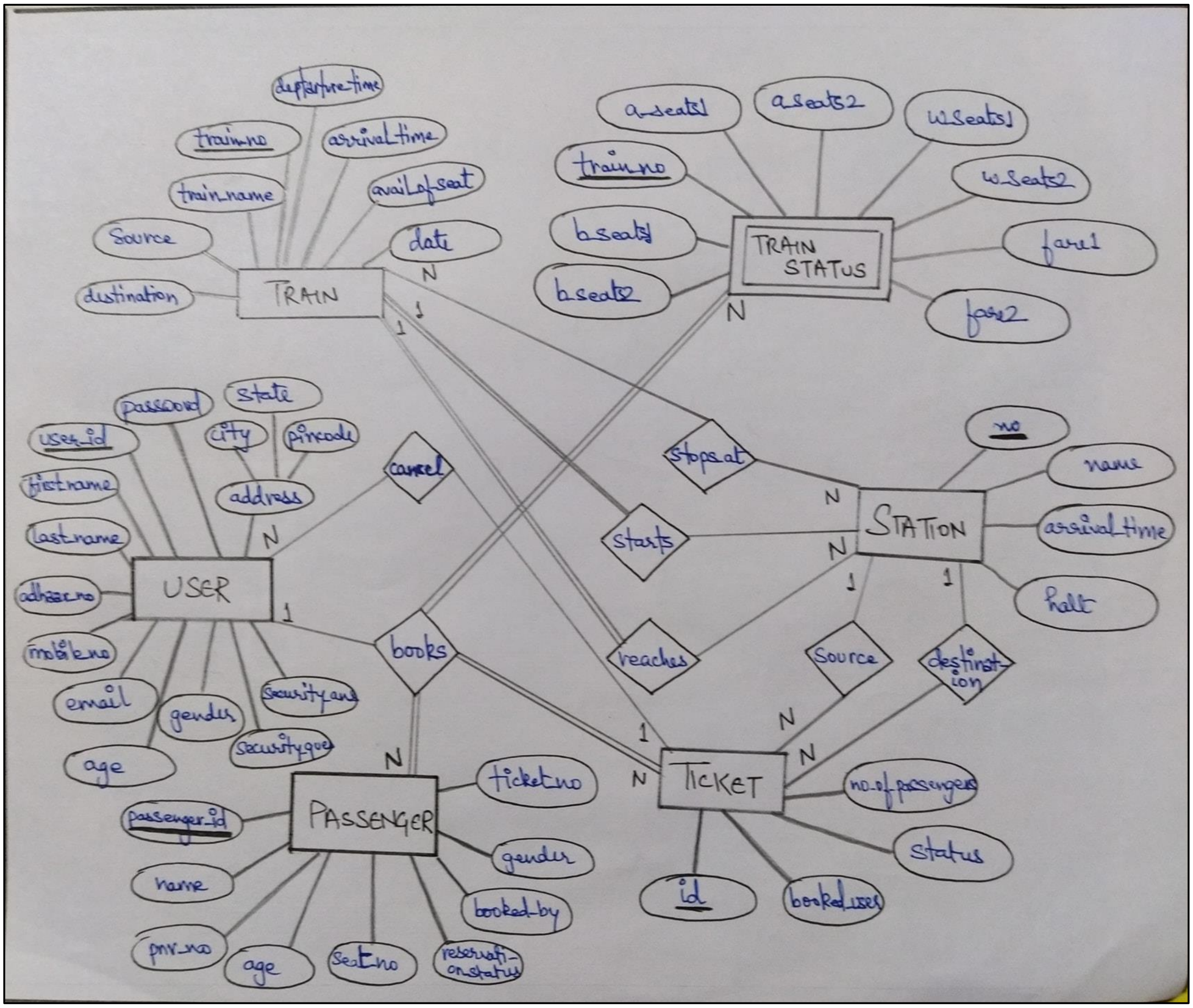


Second Level DFD: -

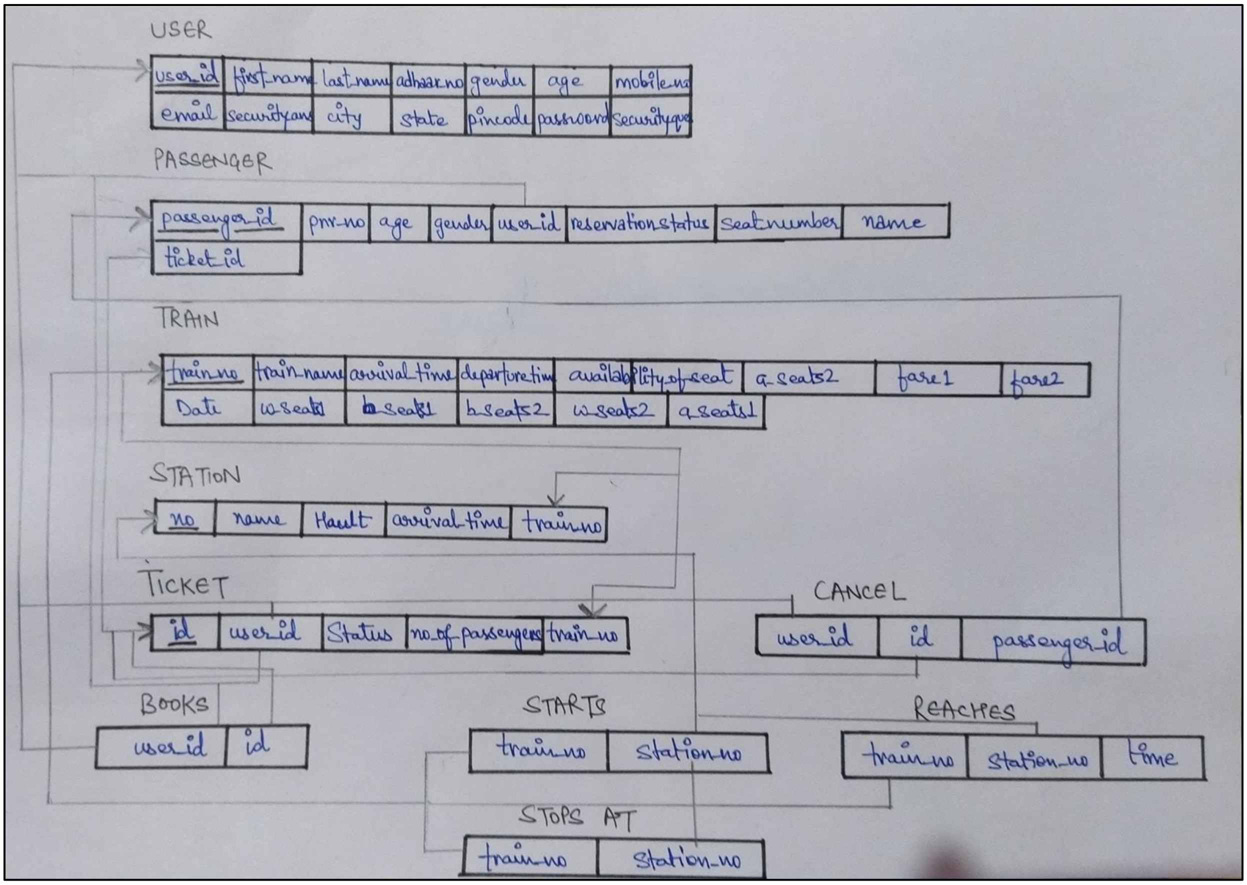


Second Level DFD-Railway Reservation System

ER-DIAGRAM


deptarture-time
train-no
arrival time
train_name
avail.of.seat
Source
date
destination
TRAIN
N
1
1
a-seats1
a-seats2
w-seats1
train-no
w-seats2
b-seats1
TRAIN STATUS
fare1
b-seats2
N
fare2
password
state
user_id
city
pincode
cancel
firstname
address
lastname
N
stops at
no
name
N
STATION
starts
N
arrival time
USER
adhaar no
1
1
1
halt
mobile no
books
reaches
Source
destination
email
gender
security ans
security ques
N
N
age
N
1
ticket no
N
TICKET
no-of-passengers
passenger_id
PASSENGER
gender
status
name
booked-by
id
booked user
pnr_no
age
seat-no
reservation status

SCHEMA DIAGRAM



CREATION QUERIES FOR TABLES

TABLE-1

```
create table USERS(user_id int primary key,first_name varchar2(50),last_name varchar2(50),

adhar_no varchar2(20),gender char,age int,mobile_no varchar2(50),email varchar2(50),city
varchar2(50),

state varchar2(50),pincode varchar2(20),password varchar2(50),security_ques
varchar2(50),security_ans varchar2(50));
```

TABLE-2

```
create table TRAIN(train_no int primary key,train_name varchar2(50),arrival_time
TIMESTAMP,departure_time TIMESTAMP,

availability_of_seats char,DATE_train Date);
```

TABLE-3

```
create table STATION(Station_no int primary key, Station_Name varchar2(50),hault
int,arrival_time TIMESTAMP,train_no int,

FOREIGN KEY(train_no) REFERENCES TRAIN(train_no));
```

TABLE:-4

```
create table TRAIN_STATUS(train_no int primary key,b_seats1 int,b_seats2 int,a_seats
int,a_seats2 int,w_seats int,w_seats2 int,farel float,fare2 float);
```

TABLE-5

```
create table TICKET(Ticket_id int primary key,user_id int,status char,

no_of_passengers int,train_no int,FOREIGN KEY (user_id) REFERENCES USERS(user_id),

FOREIGN KEY(train_no) REFERENCES TRAIN(train_no));
```

TABLE-6

```
create table PASSENGER(passenger_id int primary key,pnr_no int,

age int,gender char,user_id int,reservation_status char,seat_number varchar(5),name varchar(50),ticket_id int,

FOREIGN KEY(user_id) REFERENCES USERS(user_id),

FOREIGN KEY(ticket_id) REFERENCES TICKET(Ticket_id));
```

TABLE-7

```
create table  STARTS( train_no int primary key, station_no int,

FOREIGN KEY(train_no) REFERENCES TRAIN(train_no),

FOREIGN KEY(station_no) REFERENCES STATION(Station_no));
```

TABLE-8

```
create table STOPS_AT( train_no int,station_no int,

FOREIGN KEY(train_no) REFERENCES TRAIN(train_no),

FOREIGN KEY(station_no) REFERENCES STATION(Station_no));
```

TABLE-9

```
create table REACHES(train_no int,station_no int, time TIMESTAMP,

FOREIGN KEY(train_no) REFERENCES TRAIN(train_no),

FOREIGN KEY(station_no) REFERENCES STATION(Station_no));
```

TABLE-10

```
create table BOOKS( user_id int,ticket_id int,FOREIGN KEY(user_id)

REFERENCES USERS(user_id),FOREIGN KEY(ticket_id) REFERENCES TICKET(Ticket_id));
```

TABLE-11

```
create table CANCEL(user_id int,ticket_id int ,passenger_id int,

FOREIGN KEY(ticket_id) REFERENCES TICKET(Ticket_id),FOREIGN KEY(passenger_id)
REFERENCES PASSENGER(passenger_id),

FOREIGN KEY(user_id) REFERENCES USERS(user_id));
```

INSERT QUERIES

```
INSERT INTO
USERS(user_id,first_name,last_name,adhar_no,gender,age,mobile_no,email,city,state,pincode,
password,security_ques,security_ans)

VALUES(1701, 'vijay','sharma',309887340843,'M',34,9887786655,
'vijay@gmail.com','vijayawada', 'andhrapradesh', 520001,'12345@#','favouritecolour', 'red');

INSERT INTO
USERS(user_id,first_name,last_name,adhar_no,gender,age,mobile_no,email,city,state,pincode,
password,security_ques,security_ans)

VALUES (1702,'rohit', 'kumar',
456709871234,'M',45,9809666555,'rohitkumar@gmail.com','guntur', 'andhrapradesh',
522004,'12@#345','favouritebike','bmw');

INSERT INTO
USERS(user_id,first_name,last_name,adhar_no,gender,age,mobile_no,email,city,state,pincode,
password,security_ques,security_ans)

VALUES(1703, 'manasvi','sree',765843210987,
'F',20,9995550666,'manasvi57@gmail.com','guntur','andhrapradesh',
522004,'0987hii','favourite flower','rose');

INSERT INTO

TRAIN(train_no,train_name,arrival_time,departure_time,availability_ofseats,date)
values(12711,'pinakiniexp','113000','114000','A',20170410),(12315,
cormandelexp','124500,125000', 'NA',20170410);
```

```
INSERT INTO

STATION(no,name,hault,arrival_time,train_no)values(111, vijayawada', 10, 113000',
12711),(222, 'tirupathi',S,'114500,12315);

INSERT INTO

TRAIN STATUS(train_no,w_seats 1,b_seats 1,b_seats2.a_seats 1,a_seats2,w_seats 2,fare
1,fare2) values(12711,10,4,0,1,1,0,100,450),(12315,10,5,0,0,2,1,300,600);

INSERT INTO

PASSENGERS(passenger_id,pnr_no,age,gender,user_id,reservation
_status,seat_number,name,ticket_id) values(5001,78965,45,'M',1701,C,B6 45',
'ramesh',4001),(5002,54523,54,F,1701,W,B3-21','surekha,4002);

INSERT INTO

STARTS(train_no,station_no) values(12711,111),(12315,222),

 INSERT INTO

STOPS_AT(train_no,station_no) values(12711,222),(12315,111);

INSERT INTO

REACHES(train_no,station_no,time) values(12711,222,040000'),(12315,111,053500');

INSERT INTO

BOOKS(user_id,id) values(1701,4001), (1702,4002);

INSERT INTO

CANCEL(user_id,id passenger_id) values(1701,4001,5001);
```

SQL QUERIES RELATED TO PROJECT

1.Print user id and name of all those users who booked ticket for pinakini express

```
select u.user_id.concat(u.first_name,u.last_name)as name
from USERS u,train t,ticket te
where u.user_id=tc.user_id and t.train_no-tc.train_no and t.train_name like'pinakini exp';
```

2. Print details of passengers travelling under ticket no 4001

```
select *from PASSENGER where ticket_id like 4001;
```

3. Display all those train no's which reach station no ------

```
select t.*from TRAIN t,station s,reaches r
where t.train_no=r.train_no and r.station_no=s.no and s.name like 'vijayawada';
```

4. Display time at which train no- reaches station no ------

```
select r.*,s.name
from REACHES  r,station s
where r.station_no=s.no;
```

5. Display details of all those users who cancelled tickets for train no------

```
select u.* from USERS u,cancel c,ticket t
where c.user_id-u.user_id and c.id-t.id and t.train_no like 12711;
```

6. Display the train no with increasing order of the fares of class 1

```
select ts.train_no,ts.fare 1,t.train_name
from TRAIN STATUS ts, train t
where t.train_no=ts.train_no order by fare 1 asc;
```

7.Display passenger details for train pinakini.

```
select p.*
from PASSENGER p,train t,ticket tc
where tc.train_no=t.train_no and tc.id=p.ticket_id and t.train_name like 'pinakini exp';
```

8. Display immediate train from tirupathi to Vijayawada

```
select distinct t.* from TRAIN t,station s,starts st,stops at sa
where st.station_no=(select no from station where name like 'tirupathi')
and sa.station_no=(select no from station where name like 'vijayawada')
order by date;
```

9. Display the train no which haults for more time in station no--------

```
select train_no from STATION
having max(hault);
```

10. Display details of all those passengers whose status is confirmed for train no----

```
select t.* from TICKET t
where t.status like 'e' and t.train_no=12711;
```

CONCLUSION

In our project Railway reservation system, we have stored all the information about the Trains scheduled and the users' booking tickets and even status of trains, seats etc. This data base is helpful for the applications which facilitate passengers to book the train tickets and check the details of trains and their status from their place itself it avoids inconveniences of going to railway station for each and every query they get. We had considered the most important requirements only; many more features and details can be added to our project in order to obtain even more user-friendly applications. These applications are already in progress and in future they can be upgraded and may become part of amazing technology.