Table of Contents

Introduction

This system is basically concerned with the reservation and cancellation of railway tickets to the passengers. The need of this system arose because as is the known fact that India has the largest railway network in the whole of the world and to handle it manually is quite a tough job. By computerizing it, we will be able to overcome many of its limitations and will be able to make it more efficient. The handling of data and records for such a vast system is a very complex task if done manually but it can be made much easier if the system is computerized.

To make ticketing more convenient for travellers, we start an online reservation system, which helps us in booking tickets from the comfort of our homes or offices. While this is convenient for most people, it's made things particularly easy for people residing in remote locations.

This software comprises reserving and cancelling seats for the passengers, it contains information about trains, information about passenger, it contains the details of reservation fees, any concessions etc. it makes entries for reservation, cancelled tickets, it will update for uptime and downtime trains. All the information is safely stored in the backed i.e. MySql data base.

a). Objective:

The purpose of this to describe the railway reservation system which provides the train timing details, reservation, billing, and cancellation of reservation. We need to provide source and destination names or codes and then we will be given the train names between those two stations, Reservation forms it has the easiest of the user interface making it user friendly and easy to use, for Cancellation one just needs the PNR number which is again easy, Reservation enquiry and Fare records which is a from used for the fare between two station with a specific class. User friendly interface to administrator and customer. The system is so easy that even a native user can use this to book tickets online and the system is such that it stands up to the user expectations. The system is so secured and clerk and manager utilize it. Nobody can able to access the system without his or her permission because of providing login facility to the system.

The origin of most software is in the need of a client, who either wants to automate the existing manual system or desires a new software system. Finally, the end user will use the completed system. Thus, there are three major parties interested in a new system, the client, the user, and the developer. Somehow the requirements for the system that will satisfy the needs of the client.

A few factors that direct us to develop a new system are given below:

1. Faster system
2. Accuracy
3. Reliability
4. Informative
5. Reservation and cancellation from anywhere to any place

b). Scope:

Administrator is a person who updates the Train Information by entering his own Password. The system can handle only reservation and train details efficiently and it doesn’t contain other details about the railway administration. The main scope of this system is:

- Searching train
- Creating reservation
- Cancel reservation
- View reservation status
- View train schedule
- Payment
- Generation tickets
- Update train schedule
- Update reservation details

c). Project & its function:

The Product of this project is Railway Reservation System, which is to create Reservation, Cancel Reservation, Viewing Train Information, Viewing Reservation Details, Updating Train Information, Updating Reservation Details and Generate Reports.

d). Benefits & Goals:

The Benefit of this project is to reduce the work of Reservation Clerk and it is easy to check the remaining seats present in the Train and easily view the Reservation Status of the Train and tells to the passenger.

The Goals of this project is that, from any Railway Station we can Create Reservation, which is updated automatically in all the stations. Hence, there is no confusion to the Reservation Clerk in all the stations to create the Reservation.

Overall Description

To be more specific, our system is limited in such a way that a train starting from a particular source will have a single destination.

The basic functions being performed by our system are reservation and cancellation. These functions will be handled with the help of following sub functions: -

- It reserves and cancels seats for the passenger.
- It contains information about the trains.
- It contains information about the passenger.
- It contains the details of reservation fees, any concessions etc.
- It makes entries for reservation, waiting, cancelled tickets.
- It will update for uptime and downtime trains.

a). Project Description:

The Reservation Admin gets the details from the passenger and he checks whether the seats are remaining in order to the Reservation. If the seats are available, then system allocate the seat to the passenger by giving the Reservation Slip to the passenger and he Update the Reservation details.

The Reservation Admin gets the details from the passenger in order to cancel the Reservation and Cancels Reservation. He also Update the Reservation details after cancelling the Reservation and he creates the Reservation.

b). Project Functions:

Search: This function allows user to search for train that are available between the two travel cities, namely the “Departure city” and “Arrival city” as desired by the traveller. The system initially prompts the user for departure and arrival city, date of departure, preferred time slot and the number of passengers.

Create Reservation: A passenger should be able to reserve seats in the train. A reservation form fills by the passenger and submit to the system, if seats are available then the entries are insert in the system data base with person and train details. Then passenger needs to pay the required fare and tickets print.

Cancel Reservation: If the passenger wishes to cancel the ticket then the passenger required to enter the PNR number and mobile number. Then the entries of that passenger delete from the system data base.

Payment: System asks passenger to enter the various card details for booking the ticket.

Update Train Info: The administrator can modify the train details if he wants to change it.

View Reservation Status: Passengers can see the reservation status online by using the PNR number that printed on his ticket. The system shows the status like confirmed or RAC.

Requirement Specification

a). Hardware Requirements:

➢ Processor	:	Intel i3
➢ RAM	:	128 MB
➢ Hard Disk	:	40 GB
➢ Cache	:	512 KB

b). Software Requirement:

➢ Operating system	:	windows-8 or above.
➢ Browser	:	Google chrome, Mozilla Firefox, IE etc.
➢ Text editor	:	Sublime Text.
➢ Front End	:	HTML, CSS, Bootstrap.
➢ Back End	:	PHP, MYSQL.

Module Specification

a). Admin:

Admin can update the train details, adds the trains, remove the trains, manage the users, and also manage the reservation.

b). Passenger:

Passenger module can view the train details, book the ticket, search train, make payment, cancel reservation and update personal details if want.

c). Search:

This module helps user to find train details from source location to destination location for specific date or day.

d). Ticket:

This is module shows the confirmation of reserve seats of the passenger. Ticket will get to passenger after completing payment.

e). Cancel Reservation:

This module used to cancel the reservation if passenger wants. To cancel the reservation passenger required to provide PRN and Mobile No.

f). Login:

This module used to give the access to the authentication users only. So, it helps from fraud and give security to the system.

g). Register:

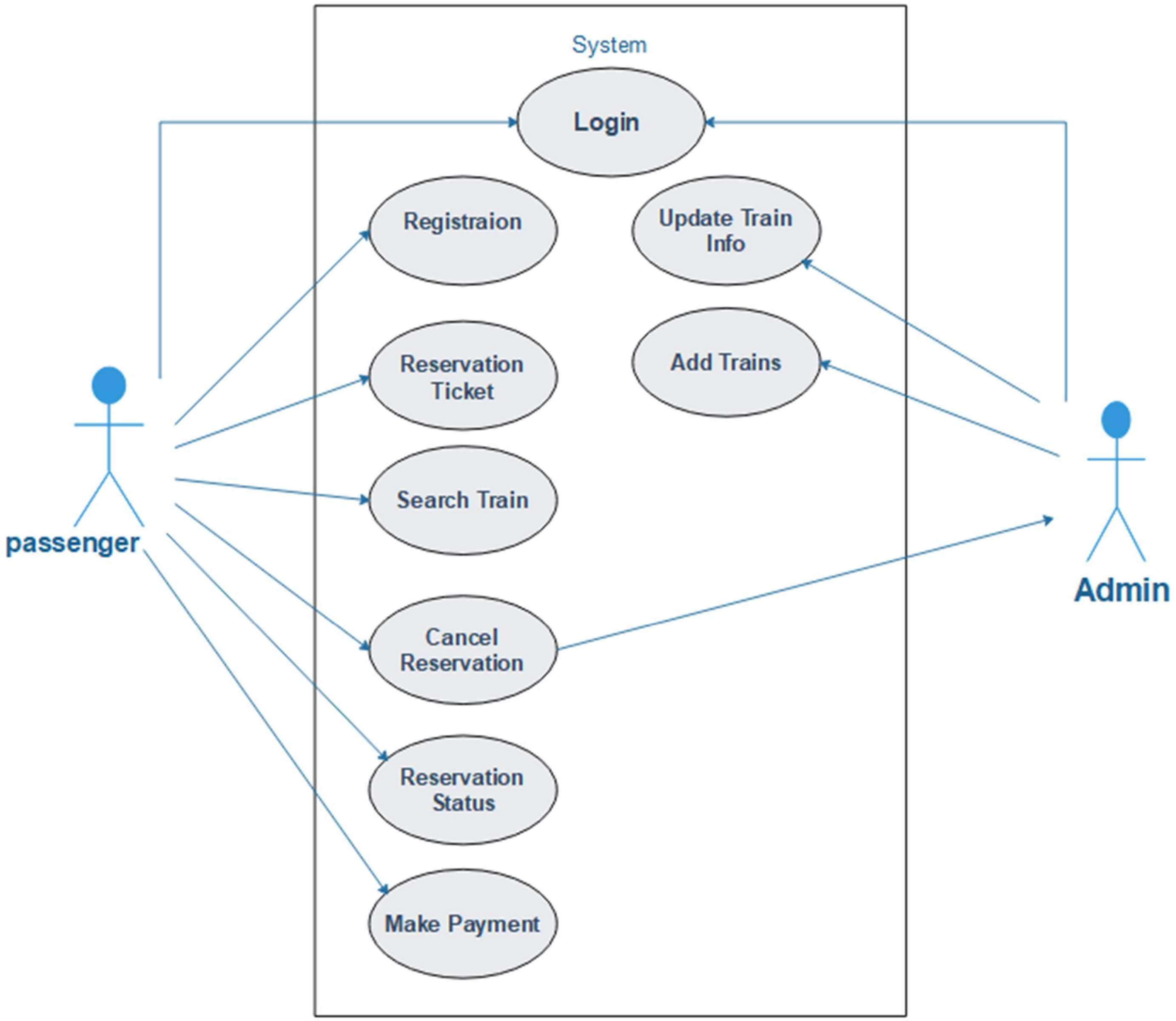
This module show the form to the user so user can register himself by filling the registration form. So, he gets the authentication to access the system.

h). Source & Destination:

This module takes the location and shows the train details to passenger.

Diagram

a). Use Case Diagram

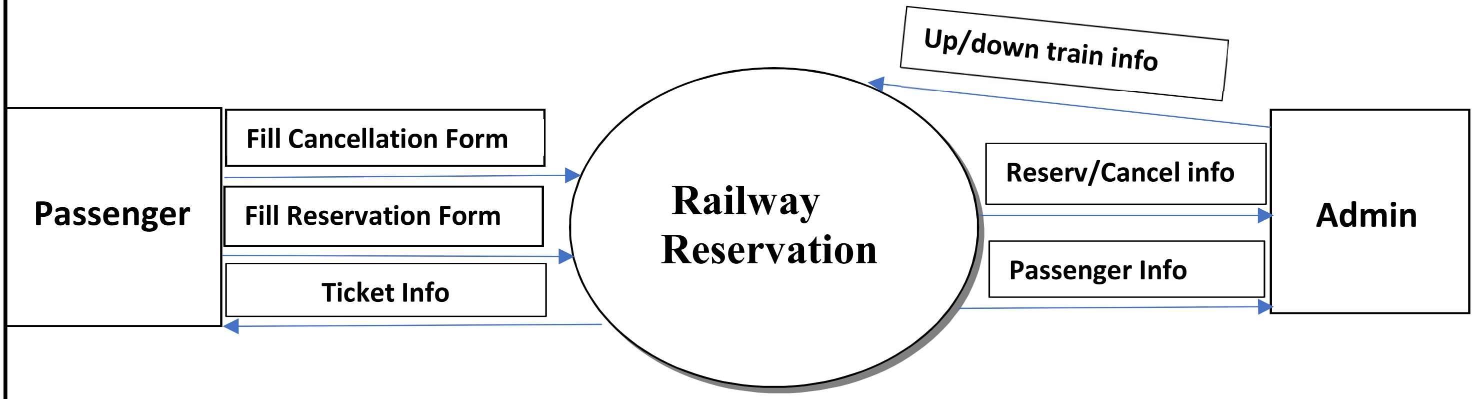


b).DATA FLOW DIAGRAMS

The data flow diagram is a graphical representation that depicts information flow and the transforms that are applied as data moves from input to output. The DFD may be used to represent a system or software at any level of abstraction. In fact, DFD may be partitioned into levels that represent increasing information flow and functional detail.

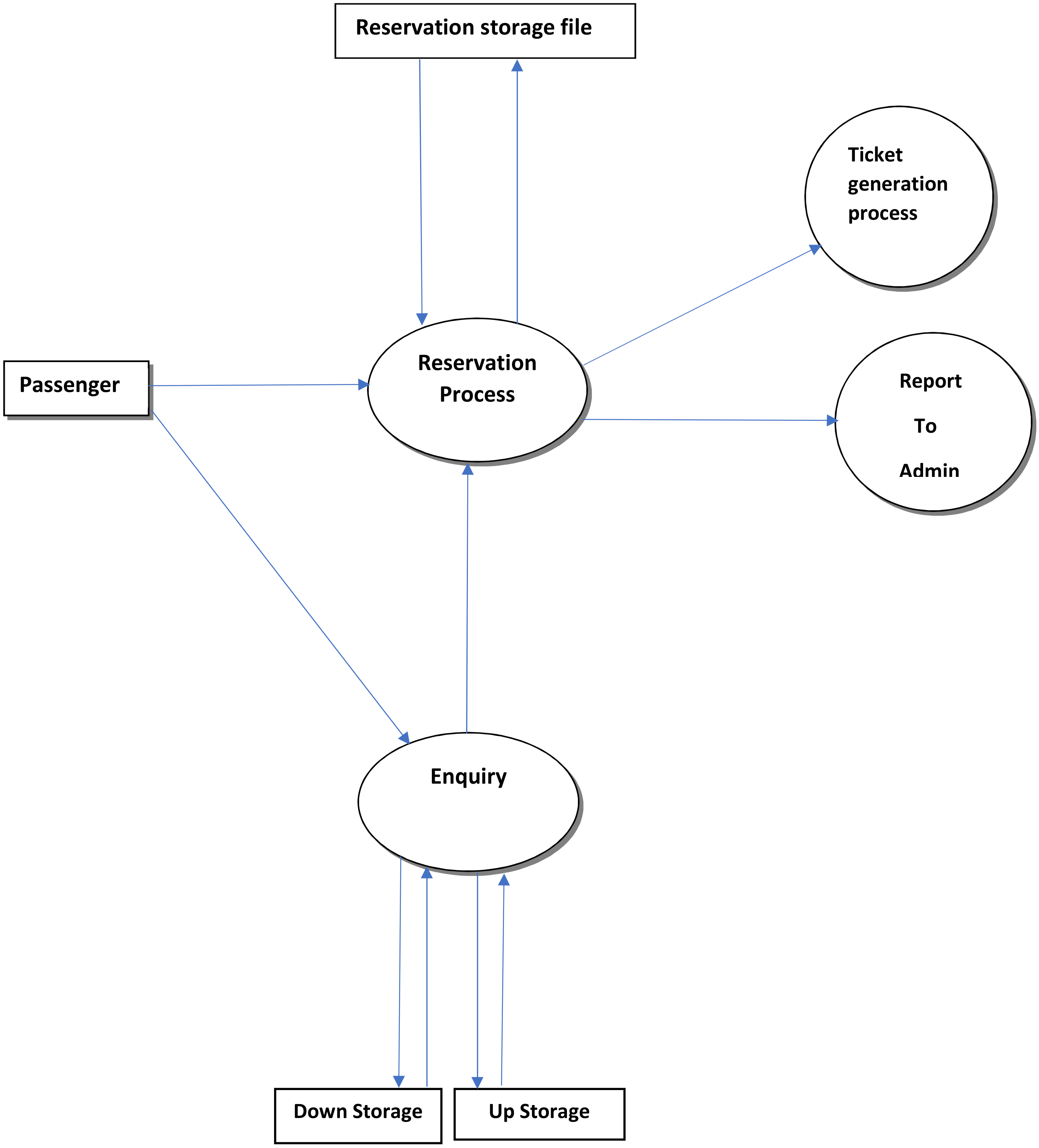
The level 0 DFD or a context model represents the entire software element as a single bubble with input and output data indicated by incoming and outgoing arrows, respectively.

In level 0 diagram shown below, the passenger fills either the reservation or cancellation form as input. He gets the ticket as the output and the report is sent to the administration.



Level 0 DFD Or Context Free Diagram

Level – 1 Data flow diagram



Level -1 DFD

A level 1 DFD is the further refinement of level 0 DFD showing greater details and functionalities. In this, the single bubble of level 0 DFD is refined further . Each of the processes depicted at level 1 is a subfunction of the overall system depicted in the context model.

As shown in the DFD above, the passenger either enquires about the trains or goes directly for the reservation or the cancellation processes as a result of which he gets the ticket generated. The reports are then sent to the administration.

c). FLOW CHART

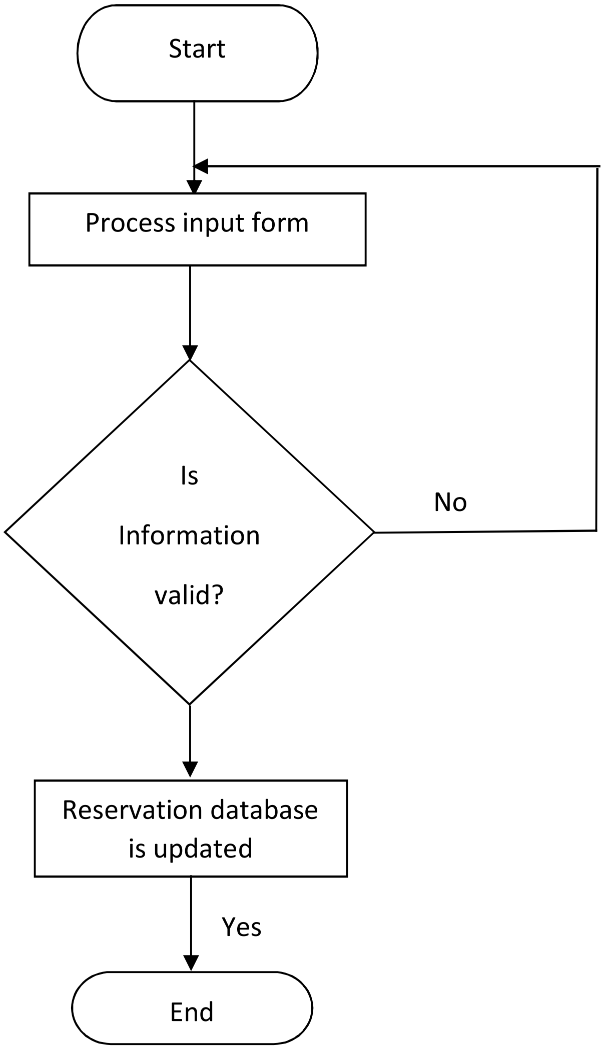
The program flowchart shows how the system proceeds from the input form to the output form of the system. It explains how the system is actually processed step by step. It represents the flow of control as the system is processed.

There are three types of program flow chart:

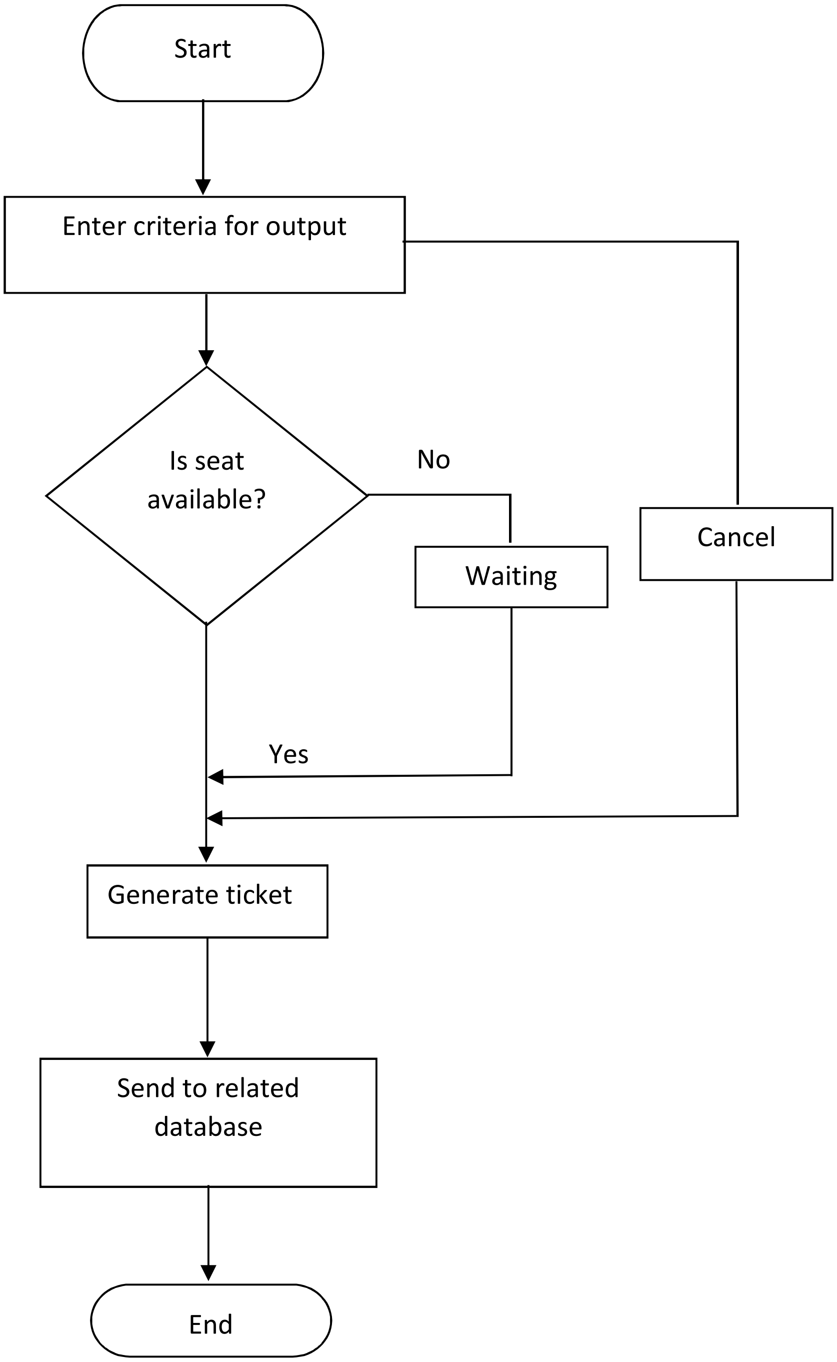
1. Input flowchart: This flowchart depicts the basic input operations in the system. In railway reservation system, first of all the password is checked then if the password is valid then we process the input form if the data is valid then the entries are updated in the data base otherwise the form is refilled.

2.Output flowchart: This flowchart depicts the basic output operations in the system. The user is required to entered criteria for output. If it is for the reservation then the availability if seats is checked. If the seats are available then the confirmed ticket is generated otherwise the user is asked for waiting and if he wants then waiting ticket is generated. If the user wants the seat to be cancelled it is done and the cancelled ticket is generated for the user. The information about all the above transactions is then transferred to the related databases.

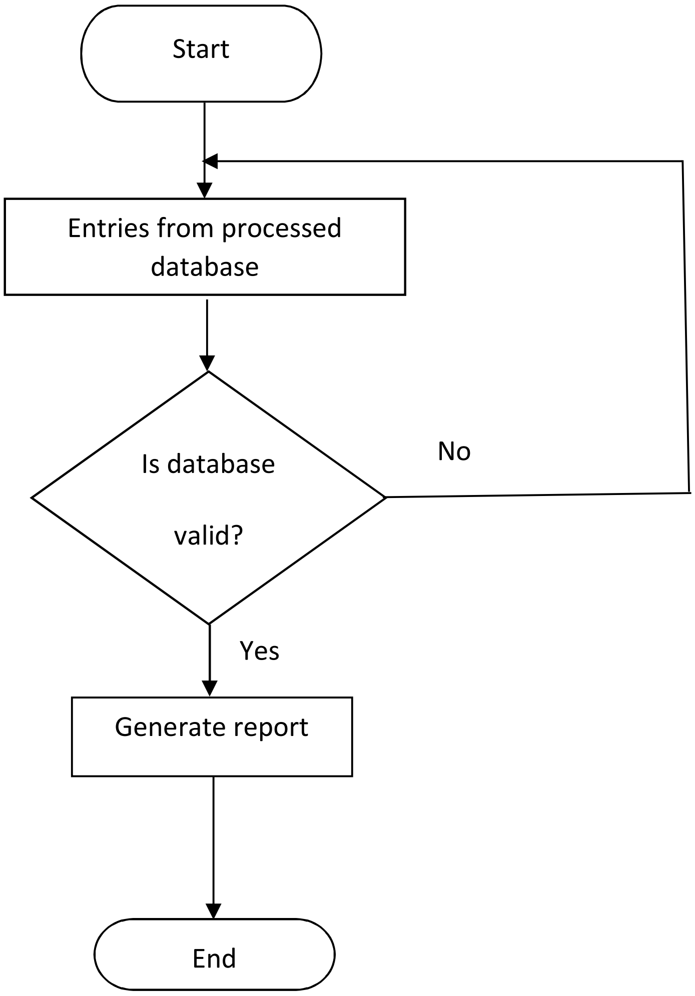
3. Report flow chart: This flowchart depicts the basic operations for the generation of reports. If the entries from the processed database are valid the concerned reports are generated otherwise the process will have to be repeated.



INPUT FLOW CHART

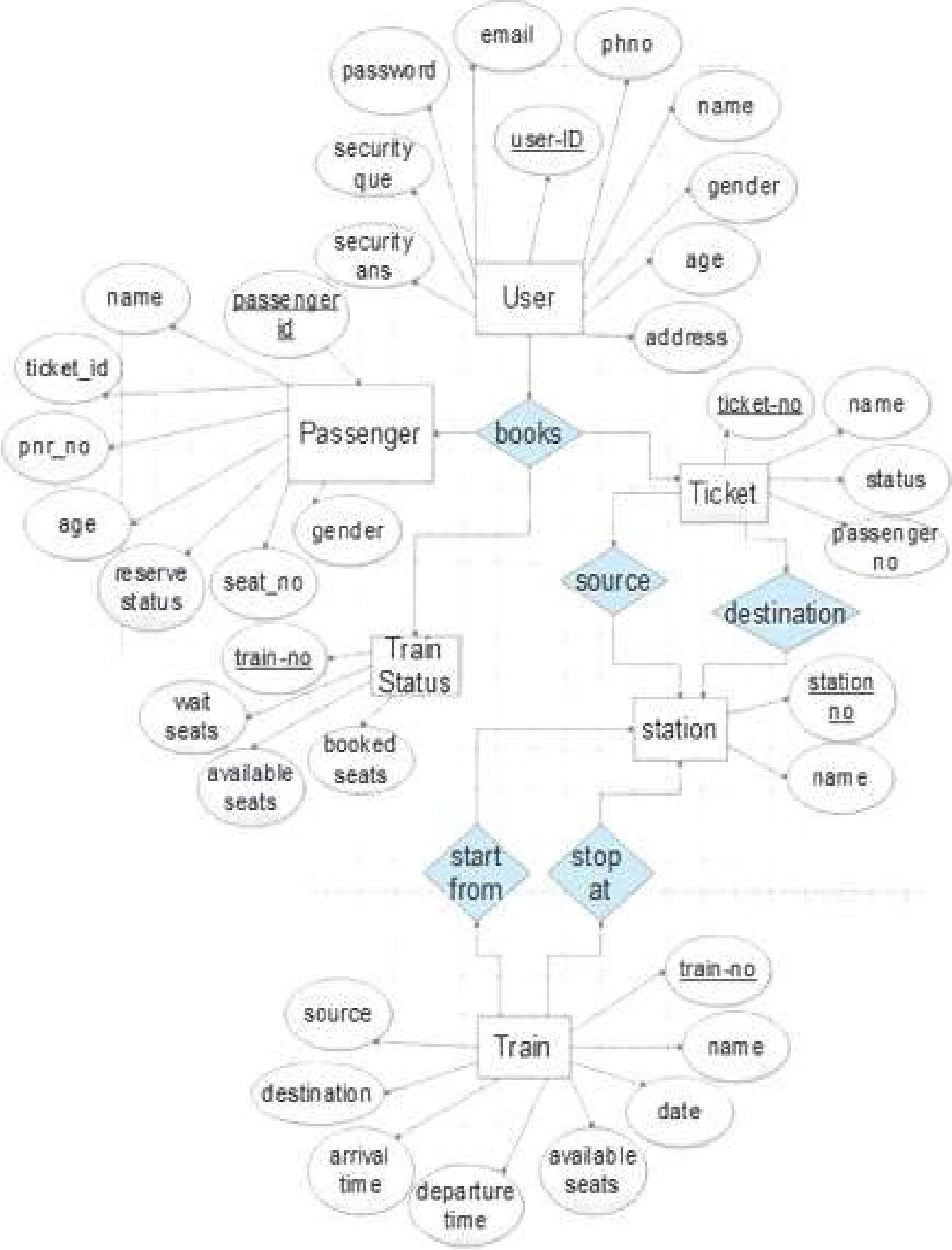


OUTPUT FLOWCHART



REPORT FLOWCHART

d).ER-Diagram


email
phno
password
name
user-ID
security que
gender
security ans
age
User
name
passenger id
address
ticket_id
ticket-no
name
Passenger
books
pnr_no
status
Ticket
age
gender
passenger no
reserve status
seat_no
source
destination
Train Status
train-no
wait seats
station no
station
booked seats
available seats
name
start from
stop at
train-no
source
Train
name
destination
date
arrival time
departure time
available seats

ENTITY RELATIONSHIP DIAGRAM

Entity relationship diagram expresses the overall logical structure of a database graphically. It shows the relationship between different entities. The entities can have composite, multivolume or derived attributes. The entities and their attributes are: -

1 Passenger

#. Name

#. Gender

#. Age

#. Phone no.

#. Reserve details

2 Ticket

#. Ticket no.

#. Name

#. Status

3 Reservation counter

4 Administrator

5 Train

#. Train no

#. Train name

#. Source

#. Destination

The relationships between different entities are: -

1. Fill: The passenger fills the form.
2. Submit: The form is submitted to the reservation counter.
3. Check: The reservation counter checks the seats.
4. Generate: Reservation counter generates the ticket.
5. Issue: Reservation counter issues ticket to the passenger.
6. Send info: The reservation counter sends information to the administrator.
7. Allotted: The seat is allotted in the train.

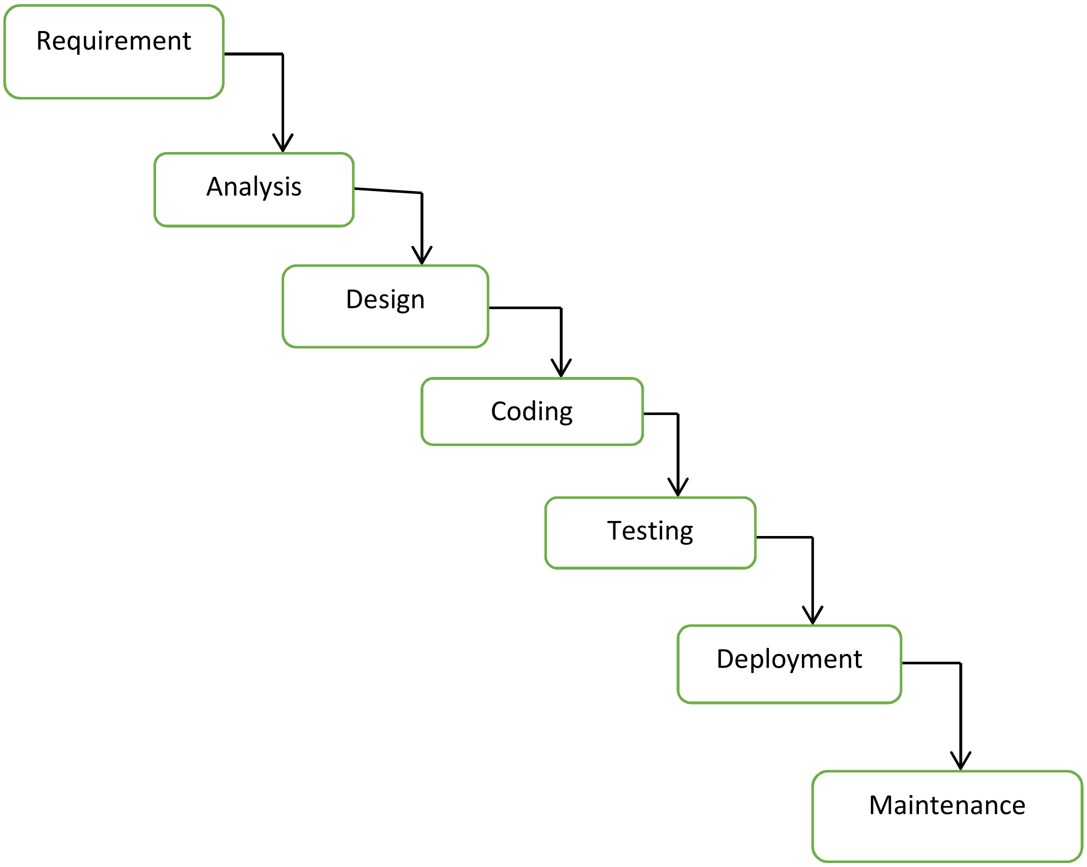
Symbols	Meanings
→	Data flow
○	Process
☰	Data store
▭	Entity

Methodology Adopted

The Waterfall Model was the first Process Model to be introduced. It is very simple to understand and use. In Waterfall Model, it follows in sequential order and so we move to the next step of development if the previous step is completely successful. This model is the successful SDLC approach for small project and requirements are very clear.

As the Waterfall Model illustrates the software development process in a linear sequential flow; hence it is also referred to as a Linear Sequential Life Cycle Model.

Sequential Phases in Waterfall Model



1.Requirement: In this phase, all the requirements are collected and documented.

2.Analysis: In this phase, we analyse all the gathered requirement are valid or not.

3.Design: In this phase, all the system is analysed and specified like hardware system configuration and architecture of system.

4.Coding: In this phase, all the development work is performed and components are handed over to testing team.

5.Testing: In this phase, each unit is tested and it is assured that the development components are working or not as expecting.

6.Deployment: If no bug or defect are found then the project is deployed to production (use).

7.Maintenance: The Team provides all the necessary support to fix the bug or issue if found by end user.

FEASIBILITY STUDY

An initial investigation in a proposal that determines whether an alternative system is feasible. A proposal summarizing the thinking of the analyst is presented to the user for review. When approved, the proposal initiates feasibility study that describes and evaluates candidate systems and provides for the selection of best system that meets system performance requirements.

To do a feasibility study, we need to consider the economic, technical factors in system development. First a project team is formed. The team develops system flowcharts that identify the characteristics of candidate systems, evaluate the performance of each system, weigh system performance and cost data and select the best candidate system for the job. The study culminates in a final report to the management.

By the end of this chapter you should know:

1. The steps in defining system performance.
2. What key considerations are involved in feasibility analysis?
3. How to conduct a feasibility study?

SUMMARY

1. A feasibility study is conducted to select the best system that meets performance requirements. This entails an identification description, an evaluation of candidate systems, and the selection of the best system for the job.

2. A statement of constraints, the identification of specific system objectives and a description of outputs define a system's required performance. The analyst is then ready to evaluate the feasibility of candidate systems to produce these outputs.

3. Three key considerations are involved in feasibility analysis: economic, technical and behavioural.

4. There are eight steps in feasibility study:

a. STATEMENT OF CONSTRAINTS

Constraints are factors that limit the solution of a problem. Some constraints are identified during the initial investigation

b. IDENTIFICATION OF SPECIFIC SYSTEM OBJECTIVES

Once the constraints are spelled out, the analyst proceeds to identify the system's specific performance objectives. They are derived from the general objectives specified in the project directive at the end of the initial investigation. The steps are to state the system's benefits and then translate them into measurable objectives.

c. DESCRIPTION OF OUTPUTS

A final step in system performance definition is describing the output required by the user. An actual sketch of the format and contents of the reports as well as a specification of the media used, their frequency, size and numbers of copies required are prepared at this point.

d. FEASIBILITY CONSIDERATIONS

Three key considerations are involved in the feasibility analysis:

a. Economic feasibility
b. Technical feasibility
c. Behavioural feasibility

Testing and Debugging

Testing:

Testing is the process of exercising software with the intent of finding errors and ultimately correcting them. The following testing techniques have been used to make this project free of errors.

Content Review

The whole content of the project has been reviewed thoroughly to uncover typographical errors, grammatical error and ambiguous sentences.

Navigation Errors

Different users were allowed to navigate through the project to uncover the navigation errors. The views of the user regarding the navigation flexibility and user friendliness were considered and implemented in the project.

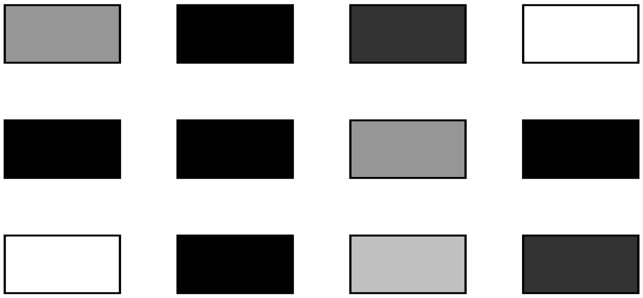
Unit Testing

Focuses on individual software units, groups of related units.

- *Unit – smallest testable piece of software.*

- *A unit can be compiled /assembled / linked/loaded; and put under a test harness.*

- *Unit testing done to show that the unit does not satisfy the application and /or its implemented software does not match the intended designed structure.*

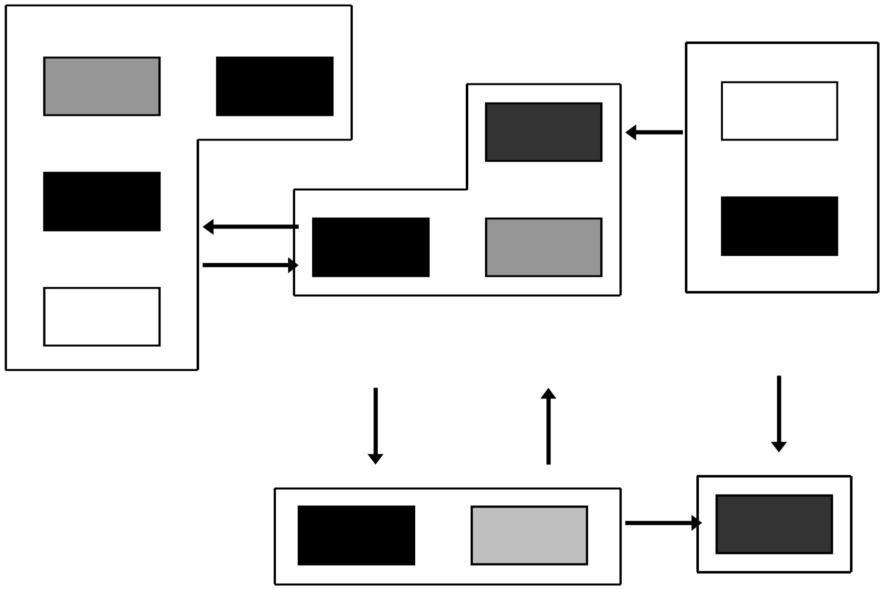


Integration Testing

Focuses on combining units to evaluate the interaction among them

- *Integration is the process of aggregating components to create larger components.*

- *Integration testing done to show that even though components were individually satisfactory, the combination is incorrect and inconsistent.*

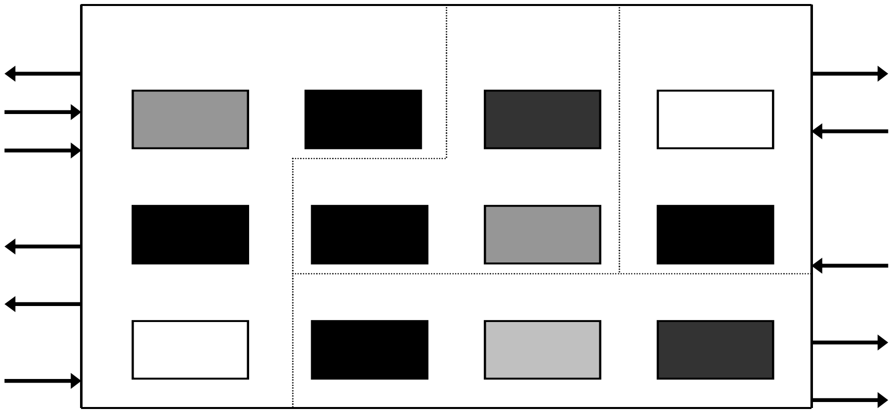


System Testing

Focuses on a complete integrated system to evaluate compliance with specified requirements (test characteristics that are only present when entire system is run)

- *A system is a big component.*

- *System testing is aimed at revealing bugs that cannot be attributed to a component as such, to inconsistencies between components or planned interactions between components.*
- *Concern: issues, behaviours that can only be exposed by testing the entire integrated system (e.g., performance, security, recovery)*



Each form encapsulates (labels, texts, grid etc.). Hence in case of project in V.B. form are the basic units. Each form is tested thoroughly in term of calculation, display etc.

Regression Testing

Each time a new form is added to the project the whole project is tested thoroughly to rectify any side effects. That might have occurred due to the addition of the new form. Thus, regression testing has been performed.

Debugging:

Debugging is a consequence of successful testing. That is when a test care uncovers an error, debugging is the process that results in the removal of errors. There are mainly two types of debugging.

Backtracking

Backtracking is a fairly common debugging approach that can be used successfully in small programs. Beginning at the site where a symptom has been uncovered, the source code is traced backward until the site of the cause is found.

Brute Forcing

This is the most common category of debugging which involves loading the source code with write statements and tracking line by line execution to isolate possible errors.

Future Scope

- User will get Ticket on mobile phone through the SMS.
- Website will use OTP verification so it controls the fraud logins.
- User will have the various option for do payment.
- User will track the train current location using new features.
- User will also book flit ticket using same website.
- User will get the help of AI assistant so user can ask anything related to train.
- In future there will be the android application which based on web-application content.
- More understandable user-interface will be designed.

LIMITATIONS OF EXISTING SYSTEM: -

Data redundancy: It means that same data fields appear in many different files and often in different formats. In manual system, it poses quite a big problem because the data has to be maintained in large volumes but in our system, this problem can be overcome by providing the condition that if the data entered is duplicate, it will not be entered, otherwise, updating will take place.

Difficulty in accessing the data: In manual system, searching information is time consuming but in our system, any information can be accessed by providing the primary key.

Unsatisfactory security measures: In manual system, no security measures were provided but, in this system, password security has been provided. The person can access the system by providing the correct password otherwise he is denied the access.

- User don't get SMS after booking ticket.
- User don't get any email.
- User use only debit card for payment process.
- User need internet connection to access the website

Conclusion

Practical Training is a very important part of the curriculum as it strengthens the concepts and enhances knowledge about the practical implementation of all the theory concepts, we have learnt so far in different subjects.

This summer training helped us learns a lot. In this training we did the project on railway reservation system. This project is used to keep a track on reserving the seat to the passenger. It helps managing the system very efficiently and conveniently.

Finally, this gives us a lot of mental satisfaction that the project we have worked upon is a real time project, which will be installed at the customer site after some more session of regress testing.

Although the project work has been done in a complete and detailed manner but due to the constraint of time, we could not include some more features we wanted to. We left these features as a part of the future development. As soon as we'll get time we'll try to add them to my project.

References

Henry F Korth, Abraham Silberschatz, ‘’Database system concepts’’, McGraw-Hill International editions, Computer Science Series (1991). Second Ed.

Software Engineering, McGraw-Hill International editions

Online References-

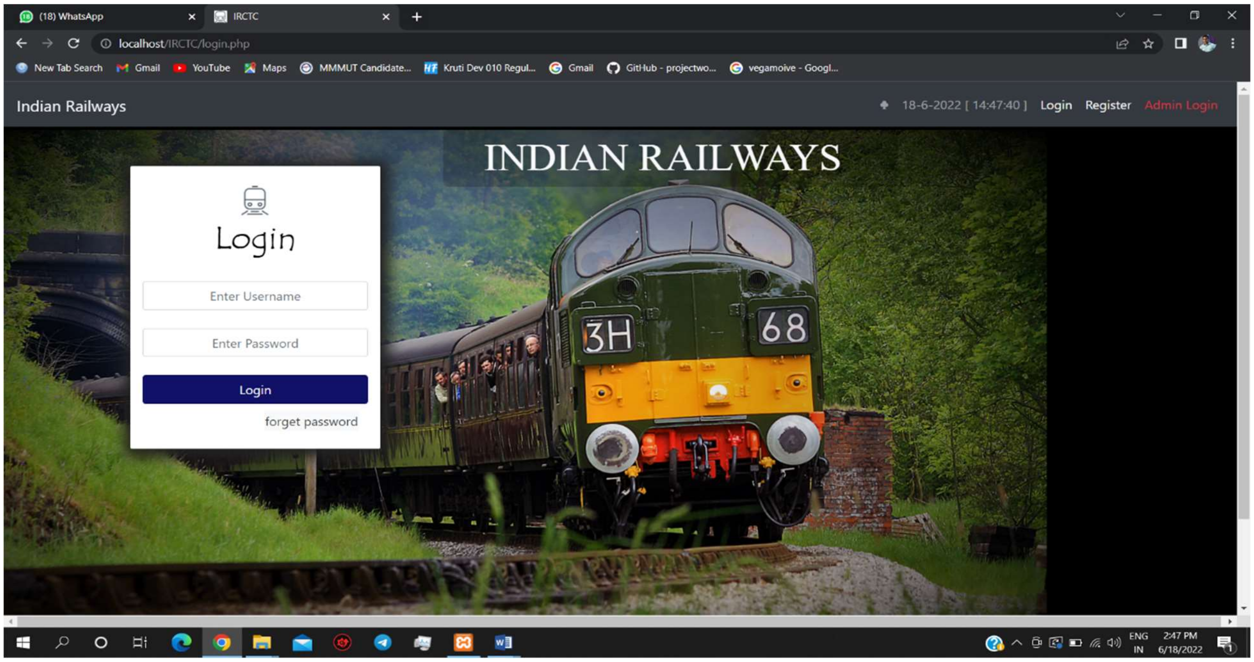
www.javatpoint.com

www.tutorialspoint.com

www.stackoverflow.com

www.youtube.com

Input and Output Screen



Indian Railways

18-6-2022 [14:47:40] Login Register Admin Login

INDIAN RAILWAYS

Login

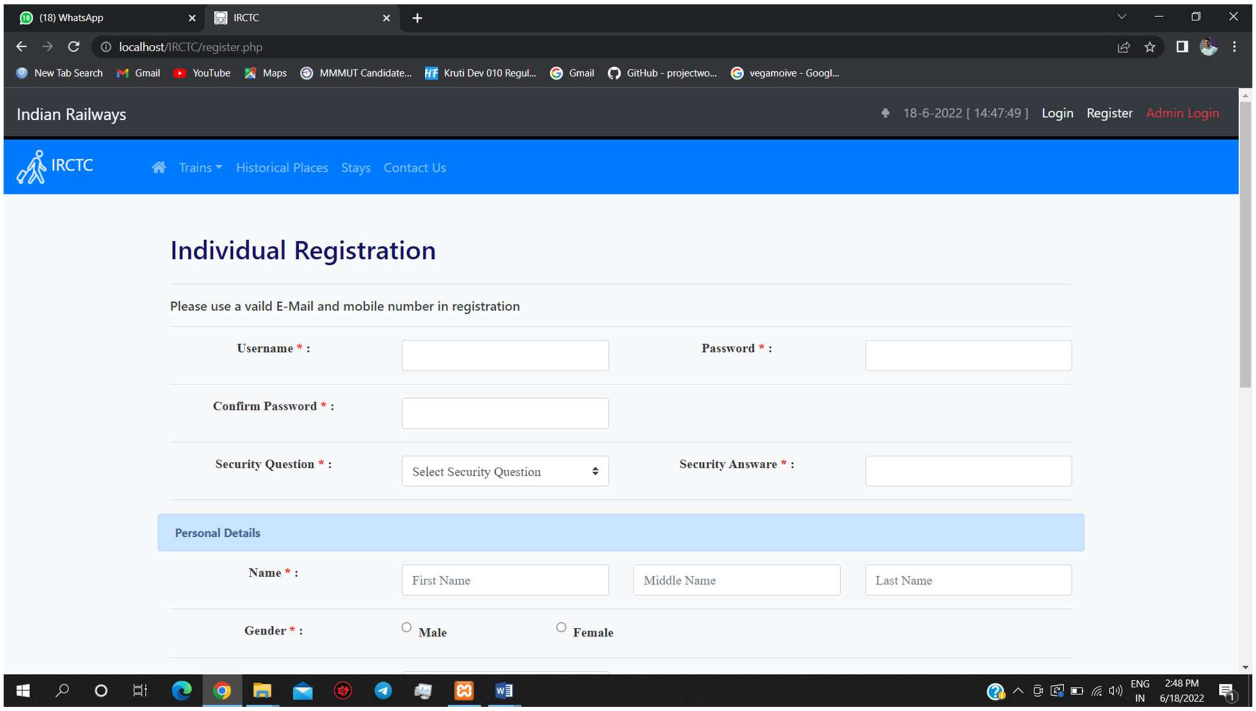
Enter Username

Enter Password

Login

forget password

Login Page



Indian Railways

18-6-2022 [14:47:49] Login Register Admin Login

IRCTC Trains Historical Places Stays Contact Us

Individual Registration

Please use a vaild E-Mail and mobile number in registration

Username * :

Password * :

Confirm Password * :

Security Question * : Select Security Question

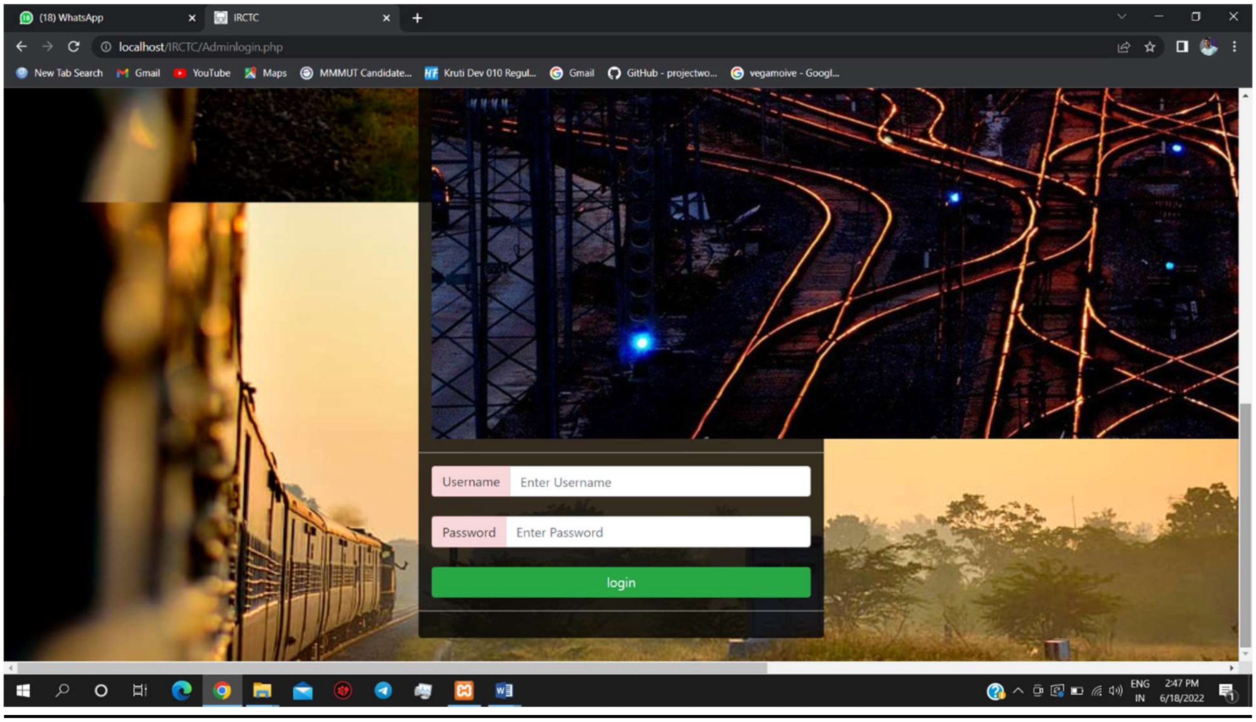
Security Answare * :

Personal Details

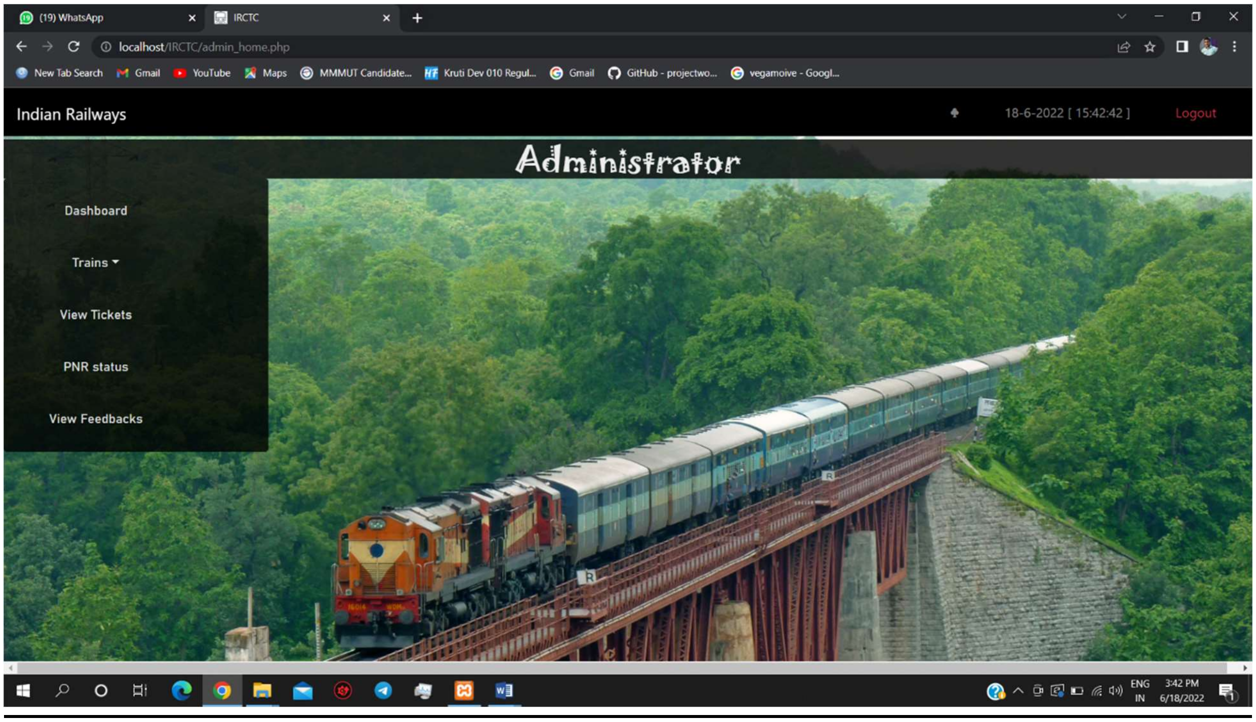
Name * : First Name Middle Name Last Name

Gender * : ○ Male ○ Female

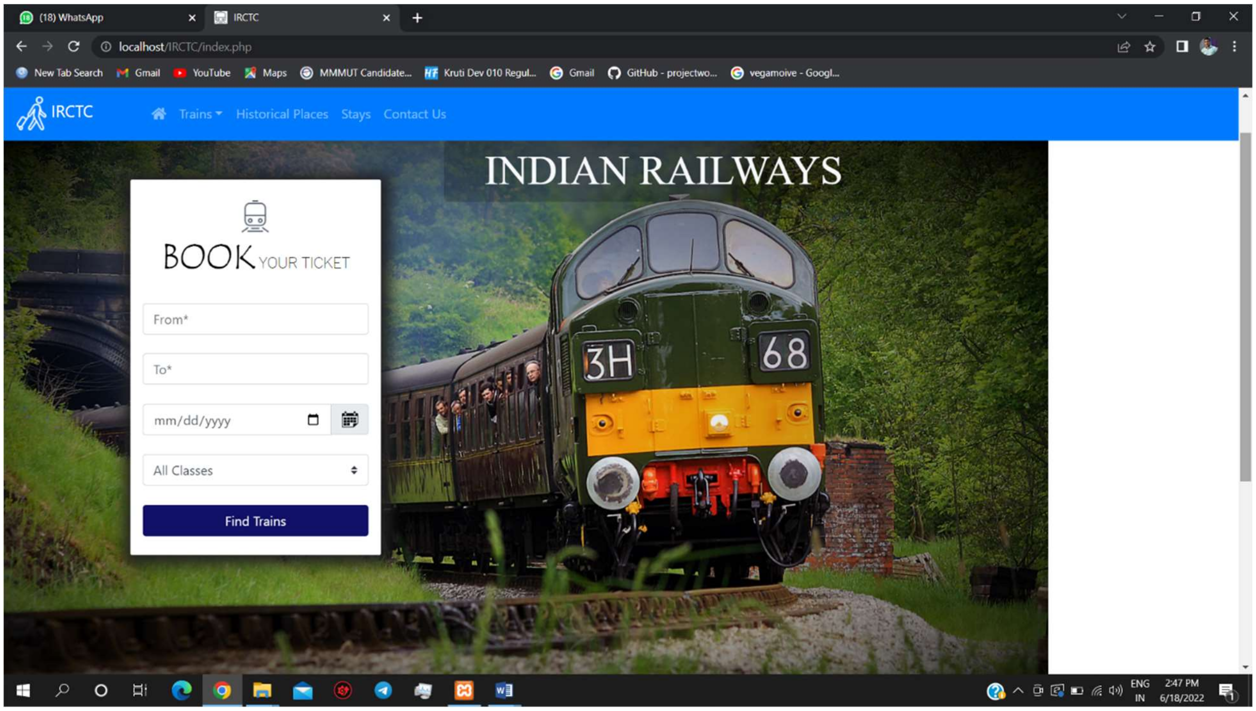
Registration Page



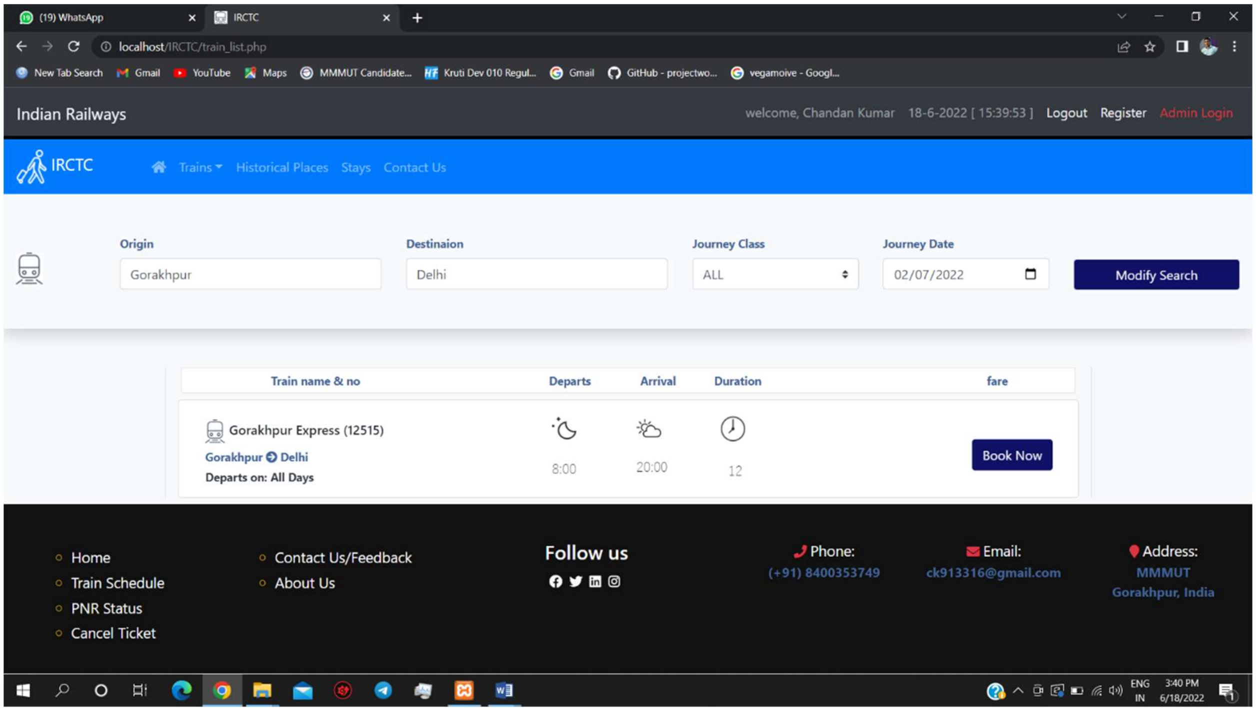
Admin login Page



Admin details



Booking train Page



Indian Railways

welcome, Chandan Kumar 18-6-2022 [15:39:53]

Origin: Gorakhpur

Destinaion: Delhi

Journey Class: ALL

Journey Date: 02/07/2022

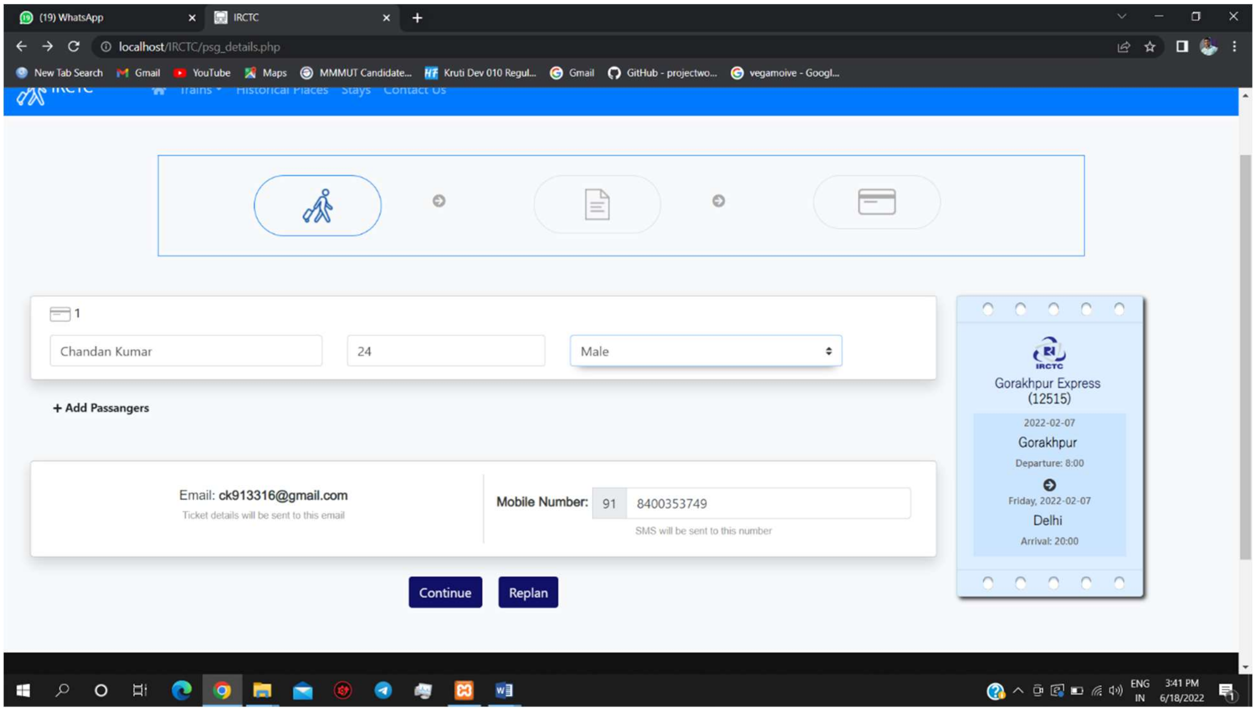
Train name & no	Departs	Arrival	Duration	fare
Gorakhpur Express (12515) Gorakhpur ➔ Delhi Departs on: All Days	8:00	20:00	12	

Phone: (+91) 8400353749

Email: ck913316@gmail.com

Address: MMMUT Gorakhpur, India

Train status



1

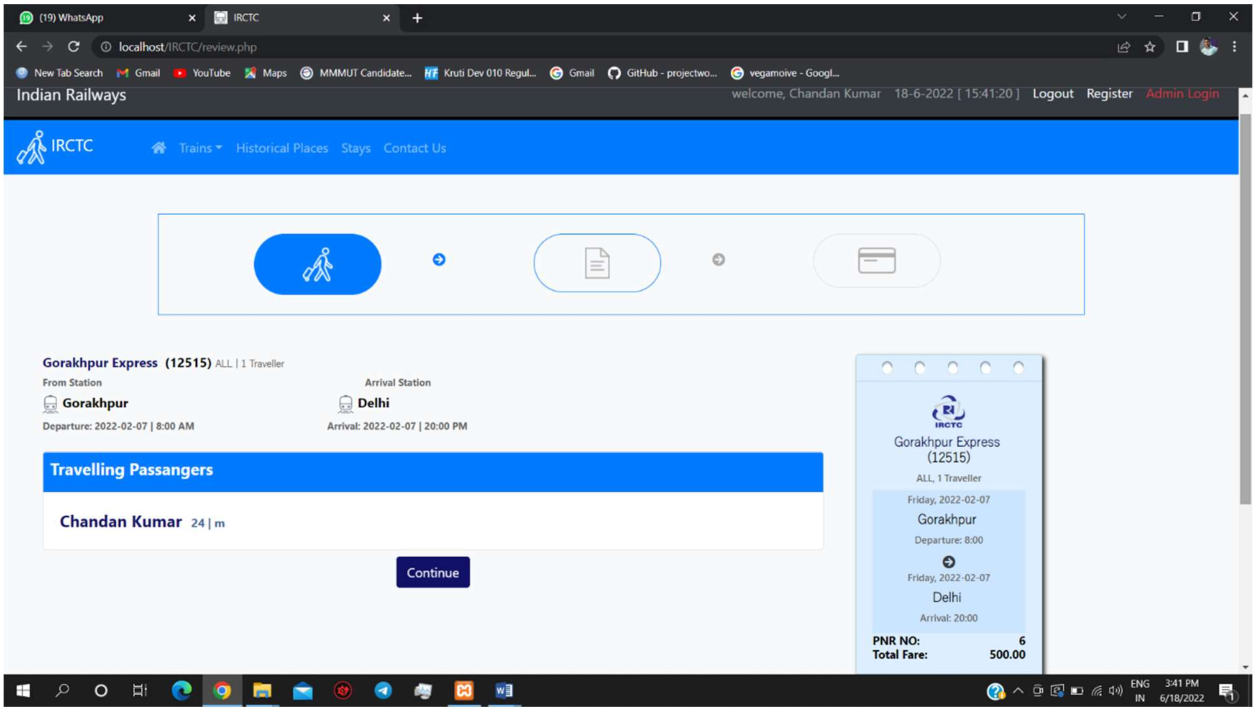
Chandan Kumar | 24 | Male

Email: ck913316@gmail.com
Ticket details will be sent to this email

Mobile Number: 91 8400353749
SMS will be sent to this number

Gorakhpur Express (12515)
2022-02-07
Gorakhpur
Departure: 8:00
Friday, 2022-02-07
Delhi
Arrival: 20:00

Fill the Passenger details



Gorakhpur Express (12515) ALL | 1 Traveller

From Station
Gorakhpur
Departure: 2022-02-07 | 8:00 AM

Arrival Station
Delhi
Arrival: 2022-02-07 | 20:00 PM

Travelling Passangers

Chandan Kumar 24 | m

Gorakhpur Express (12515)
ALL, 1 Traveller
Friday, 2022-02-07
Gorakhpur
Departure: 8:00
Friday, 2022-02-07
Delhi
Arrival: 20:00

PNR NO: 6
Total Fare: 500.00

Train Fair details

CODE

Index.php

```
<?php
   session_start();
   include('Details.php');
   include('DBConnection.php');

   if(isset($_SESSION['update'])){
       unset($_SESSION['update']);
   }
   if (isset($_GET['success']) && $_GET['success'] == 1) {
         echo "<script> alert('your are logged in'); </script>";

   }
   else if (isset($_GET['logout']) && $_GET['logout'] == 1) {
         echo "<script> alert('your are logged out'); </script>";
   }
   if(isset($_SESSION["uname"])){
      $uname = $_SESSION["uname"];
         include("header2.php");
   }
   else{
         include("header.html");
   }

 ?>


<!doctype html>
<html lang="en">
<head>
                  <title>IRCTC</title>
                  <!-- Required meta tags -->
   <meta charset="utf-8">
   <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">

   <link rel="icon" type="icon/png" href="asset/img/logo/rail_icon.png">

   <!-- Bootstrap CSS -->
   <link rel="stylesheet" href="asset/css/bootstrap.min.css">

   <!-- :start of optional css-->

   <!-- font-awesome for icon -->
   <link rel="stylesheet" href="asset/font-awesome/css/all.min.css">

   <!-- animation css -->
   <link rel="stylesheet" href="asset/css/animate.css">

   <!-- hover css animations -->
   <link rel="stylesheet" href="asset/css/hover-min.css">

   <!-- custom css -->
   <link rel="stylesheet" type="text/css" href="asset/css/custom.css">

   <!-- :end of optional css -->


   <!-- Optional JavaScript -->
   <!-- jQuery first, then Popper.js, then Bootstrap JS -->
   <script src="asset/js/jquery-3.4.1.slim.min.js"></script>
   <script src="asset/js/popper.min.js"></script>
   <script src="asset/js/bootstrap.min.js"></script>
   <script src="asset/js/validation.js"></script>
   <style>

       #bg-custom{
         background-color:rgba(2,2,2,0.8);
      }
      #m-cust{
            margin-right: 250px;
            margin-top: 60px;
        }
        .bg-black{
            background-color:black;
        }
        .bg-img{
            background-image: url('asset/img/7.jpg');
            /*background-repeat: no-repeat;*/
            background-size: 100%;
            max-width: 1300px;
            min-height: 700px;
            /*opacity: 0.8;*/
        }
        @media(max-width: 400px){
            .bg-img{
                background-image: url('asset/img/5.jpg');
                background-size: auto;
                background-repeat: no-repeat;
                /*background-position: center*/

            }
        }

        .bg-img2{
            background-image:url('asset/img/5.jpg');
            background-size: 100%;
        }
        .pnr{
            background-color: white;
            color: black;
            /*width: 340px;*/
            padding-top: 10px;
            box-shadow: 2px 2px 18px 10px #222;
            border-radius: 2px;
        }

        .fs-1{
            font-size: 42px;
            font-family: Tempus Sans ITC;
            margin-top: 50px;
        }
        .fs-2{
            font-size: 18px;
            font-family: Yu Gothic Light;
            font-weight: lighter;
            margin-bottom: 50px;
        }
        .main-name{
            font-size: 50px;
            font-family: Arial Rounded MT Bold;
            margin-top: 0px;
            background-color: rgba(2,2,2,0.2);
            /*background-color: #116;*/
            border-radius: 5px;
            width:560px;
            padding-left: 50px;

        }

    </style>

</head>
<body >

    <!-- include header -->
                        <!-- <?php //include('header.html'); ?> -->




    <!-- container -->
    <!-- 1st row -->
    <!-- its show box for serching train and image that used @background -->
    <div class="row bg-img text-light">
        <!-- col 1 -->
        <div class="col-12 col-sm-12 col-md-4 offset-1">
```

```
      <!-- PNR status box -->
      <div class="row pnr m-5 text-center">
          <div class="col-12 mt-3">
            <span><img src="asset/img/logo/rail_icon.png"></span><br>
            <span class="fs-1">BOOK</span>
            <span class="fs-2" >YOUR TICKET</span>
          </div>
          <div class="col-12 mt-4">
           <!-- form tag -->
            <form action="./train_list.php" method="post">
              <div class="input-group">
                <input type="text" name="src" class="form-control hvr-shadow" placeholder="From*" required>
              </div><!-- group1 ends -->
              <br>
              <div class="input-group">
                <input type="text" name="dest" class="form-control hvr-shadow" placeholder="To*" required>
              </div> <!-- group2 ends-->
              <br>
              <div class="input-group">
                <input type="date" name="date" class="form-control hvr-shadow" placeholder="" required>
                <div class="input-group-append">
                  <span class="input-group-text text-dark ">
                     <img src="asset/img/logo/cal.png" width="20" height="20">
                  </span>
                </div>
              </div> <!-- group3 ends-->
              <br>
              <div class="input-group">
                <select name="class" class="custom-select hvr-shadow">
                  <option class="" value="ALL">All Classes</option>
                  <option class="" value="AC">AC</option>
                  <option class="" value="SL">Sleeper(SL)</option>
                </select>
              </div> <!-- group4 ends-->
              <br>
              <div class="input-group">
                <input class="btn text-light bg-blue btn-block hvr-shadow" type="submit" value="Find Trains" >
              </div><br>
            </form>
          </div>
      </div>
    </div>

    <!-- col 2 and title of system-->
    <div class="sm-hide col-sm-6 offset-0">
      <div class="text-left main-name">
        <span>INDIAN RAILWAYS</span>
      </div>
    </div>
  </div> <!-- ends of row first -->


  <div class="jumbotron bg-secondary"></div>
  <div class="jumbotron bg-secondary"></div>
  <div class="jumbotron bg-secondary"></div>



  <!-- include footer -->
  <?php include('footer.html'); ?>


</body>
</html>
```