A MINI PROJECT REPORT

Submitted by

M.S. KOUSHIK [RA2011026010272]
N. VIVEK [RA2011026010269]
R. AKASH [RA2011026010260]

Under the guidance of

Dr.S.Sadagopan

Associate Professor, Department of Computational Intelligence

In partial satisfaction of the requirements for the degree of

BACHELOR OF TECHNOLOGY
in
COMPUTER SCIENCE & ENGINEERING

With specialization in Artificial Intelligence

SCHOOL OF COMPUTING

COLLEGE OF ENGINEERING AND TECHNOLOGY

SRM INSTITUTE OF SCIENCE AND TECHNOLOGY

KATTANKULATHUR – 603203

May 2023

SRM
INSTITUTE OF SCIENCE & TECHNOLOGY
Deemed to be University u/s 3 of UGC Act, 1956

SRM INSTITUTION OF SCIENCE AND TECHNOLOGY
KATTANKULATHUR-603203

BONAFIDE CERTIFICATE

Certified that this Course Project Report titled **"RAILWAY RESERVATION SYSTEM"** is the bonafide work done by **M.S. KOUSHIK [RA2011026010272], N. VIVEK [RA2011026010269] and R. AKASH [RA2011030010226]** who carried out under my supervision. Certified further, that to the best of my knowledge the work reported herein does not form part of any other work.

HEAD OF THE DEPARTMENT
Dr. R. ANNIE UTHRA
Professor and Head,
Department of Computational Intelligence,
SRM Institute of Science and Technology
Kattankulathur Campus, Chennai.

SIGNATURE
Faculty In-Charge
Dr.S.Sadagopan
Assistant Professor
Department of Computational Intelligence
SRM Institute of Science and Technology
Kattankulathur Campus, Chennai

SRM INSTITUTION OF SCIENCE AND TECHNOLOGY
KATTANKULATHUR-603203

BONAFIDE CERTIFICATE

Certified that this Course Project Report titled **"RAILWAY RESERVATION SYSTEM"** is the bonafide work done by **M.S. KOUSHIK [RA2011026010272], N. VIVEK [RA2011026010269] and R. AKASH [RA2011030010226]** who carried out under my supervision. Certified further, that to the best of my knowledge the work reported herein does not form part of any other work.

SIGNATURE
Faculty In-Charge
Dr.S.Sadagopan
Assistant Professor
Department of Computational Intelligence
SRM Institute of Science and Technology
Kattankulathur Campus, Chennai

ABSTRACT

The Railway Reservation System facilitates the passengers to enquire about the trains available based on source and destination, Booking and Cancelation of tickets, enquire about the status of the booked ticket, etc. The aim of case study is to design maintaining the records of different trains, train status, and passengers. This project contains introduction to the Railway reservation system. It is the computerized system of reserving the seats of train seats in advance. It is mainly used for long route. Online reservation has made the process for the reservation of seats very much easier even before.

In our India, there are number of counters for the reservation of the seats and one can easily make reservation 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 relational model.

The Indian Railways (IR) carries about 5.5 lakhs passengers in reserved accommodation every day. The Computerised Passenger Reservation System (PRS) facilates 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 upto 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.

INDEX

LIST OF FIGURES

ABBREVIATIONS

XML	Advanced Encryption Standard
PHP	Hypertext PreProcessor
CSS	Cascading Style Sheet
HTML	HyperText Markup Language
DB	Data Base
RFID	Radio Frequency IDentification
SQL	Structured Query Language
UI	User Interface

1. Introduction:

1.1) Introduction.

Database is an organized collection of data. The data is typically organized to model aspects of reality in a way that supports process requiring information. A DBMS makes, if possible, for end users to create, read, update and delete data in a database.

The main purpose of dbms for Railway Reservation 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 about the seats available at them. Due to automation many loopholes that exist in the manual maintenance of the records can be removed.

1.2) Objectives:

The main objective of the project Railway Reservation System is to manage the details of Train, Ticket, Booking, Customers, Payment. It manages all the information about Train, Seat, Payment, Train. The project is totally built an administrative end and thus only the administrator is guaranteed the access. The purpose of the project is to built an application program to reduce the manual work for managing the Train, Ticket, Seat booking. It tracks all details about the Booking, Customers, Payment.

1.3) Scope of the project:

- A central database that will store all information.
- An online website that will provide real- time information about the availability of tickets their prices.
- Every registered user is able to view his booking id that has been made in his/her name.
- Every registered user can change his password any time he wants to change.
- Every guest user can search train availability, price of the ticket, arrival and departure time, distance between source and destination etc.
- Every registered user has the facilities to print his ticket any time he wishes.
- Administration login
- In admin mode the administrator can make changes in train details.
- He can also view all booking that has been made by different users.
- The booking window contains all the facilities at one place, the

user can simply login to his account and can book his ticket.

1.4) System Requirements:

Hardware:

Intel core 2 duo T6400 2.00 GHz with 2GB RAM, 250 GB hard disk space and other

Standard accessories.

Environment and Applications:

- Microsoft Windows 7.
- Microsoft Visual Studio 2010.
- Microsoft SQL Server 2005.
- Microsoft Internet Explorer.

Hardware configuration:

The minimum configuration for hardware is given below:

- Intel® Pentium® or higher processor.
- 65 MB RAM or higher

Software configuration:

- Microsoft® Windows® XP or later versions
- A standard web browser.
- .Net framework.

2. Literature Survey:

Railways are providing important and mandatory basic facilities to the passengers like (I)Healthy food (II)Good sanitation (III)flexible reservation system (IV) electronic scrolling inside trains and enquiry facilities at all stations and in trains resulting in the convenience of the passengers and which will result in increase of number of passengers. In present system there is no query system for the passengers, by taking this problem PNR status enquiry system was done in which passengers who was in waiting list can enquire about their status if this system is not used have to wait up to TTR arrival and have to provide a bribe, by implementing this system

can check their own status and can utilize it anytime. The action performed can make the travel still more informative and safety. Present reservation is having problem in which passengers cannot choose their seats, members of the family is not getting seats in sequence and therefore seats are in different coaches or different place in same coaches [9]. Indian Railway will continue to play a Crucial role in the economy of the country in the many years to come. The need of the hour is to have an exclusive advanced reservation system,

PNR status checking system, location identification through effective communication system, fire sensing system and catering services in place that would Fulfil the requirements of the whole spectrum of passengers [1]. In present system, there is no passenger intimation in a train that is persons who are travelling during night time are unaware of the exact place now they are in and no prior intimation of when they reach their respective stations. The information about arrival of the respective station can be checked using live status option in application [2].IPMIS model is an inevitable trend, which means a common platform for mobile booking and reservation has become a priority. More also, people have no patience to spend time in queue waiting, therefore the model can be integrated into a decision support system used by operators who make

decisions to change train routes or orders avoiding conflicts and delays

3 Design and Data Diagrams:

2.1 ER Diagram:

ER diagram displays the relations between the various entities(classes and their attributes) stored in the database. ER diagrams are very important for any database project. This diagram shows the communication between entities and their attributes.

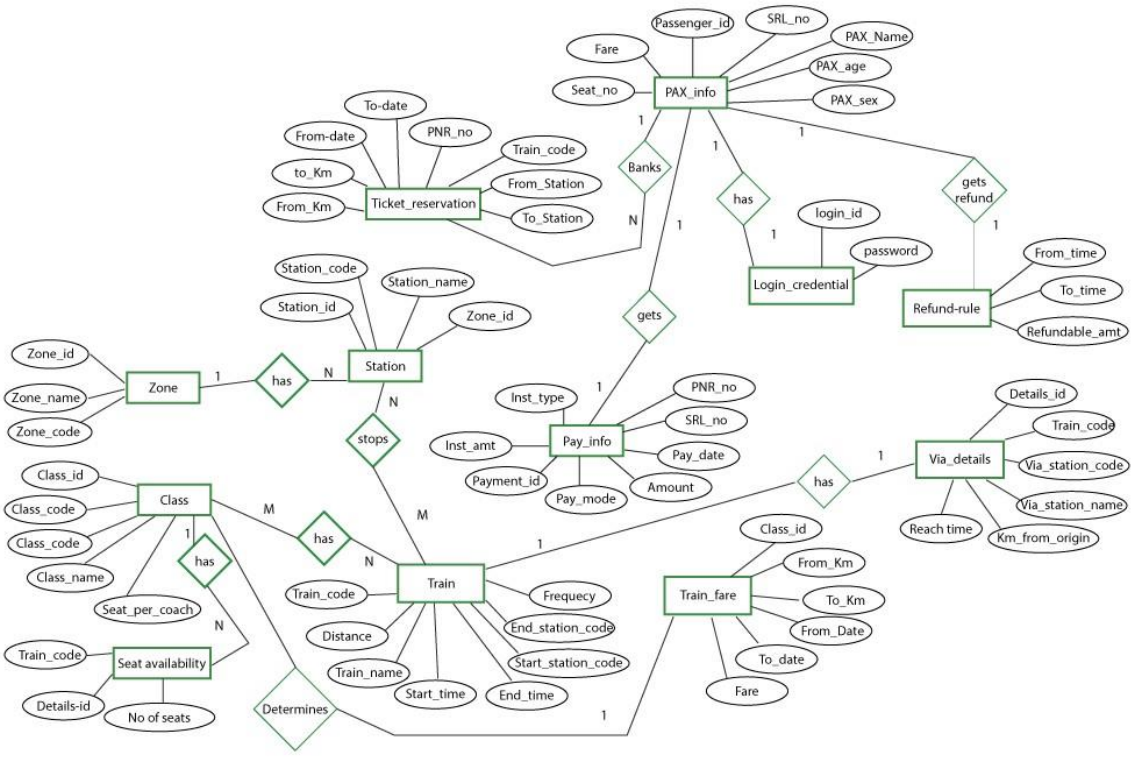


Fig 3.1 Entity Relationship Diagram

The above ER diagram illustrates the key information about the railway reservation system, including entities like PAX_info, Login_credentials, Ticket_reservation, refund_rule, via_details, train_fare, Train, Seat_availability, Class, Zone, station, pay_info. This diagram also shows the relationships between entities.

Entities and their attributes are:

PAX_info: Attributes of PAX_info entity are Passenger_id (primarykey) ,SRL_no ,PAX_name ,PAX_age ,PAX_sex ,fare,seat_no.

- Login_credentials: Attributes of Login_credentials entity are login_id(PK) , password.
- Ticket_reservation: Attributes of Ticket_reservation entity are PNR_no(pk),to-date, from-date, to-km, from-km, to-station, from-station, Train_code.
- Refund_rule: Attributes of refund_rule entity are to-time, from-time, refundable-amt.
- via_details: Attributes of via_details entity are Details_id(PK), Train_code, via_station_code, km_from_origin, Reach_time.
- train_fare: Attributes of train_fare entity are to-date, from-date, to-km, from-km, Fare, Class_id.

- Train: Attributes of Train entity are Train_code(PK), Distance, Train_name, Start_time, End_time, Start_station_code, End_station_code, Frequency.
- Seat_availability: Attributes of Seat_availability entity are Train_code, Class_code, and Number of seats.
- Class: Attributes of Class entity are Class_id(PK), coach_prefix, class_code, Class_name, seat_per_coach.
- Zone: Attributes of Zone entity are zone_id(PK), Zone_name,Zone_code.
- Station: Attributes of station entity are Station_id(PK),Station_code,station_name, zone_id.
- Pay_info: Attributes of Pay_info entity are payment_id(PK), pay_mode, amount, pay_date, srl-no, PNR_no, inst_type, inst_amt.

2.2 Data-Flow Diagram (DFD):

This diagram represents various operations by dataflow movement.

- **Level 0 DFD:**

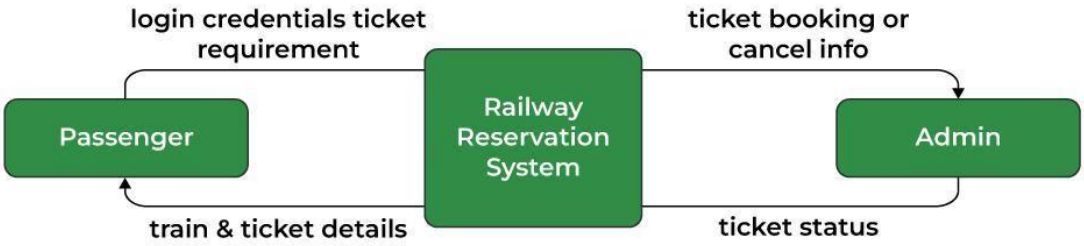


Fig 3.2.1 Level 0 Data Flow Diagram

Level 1 DFD:

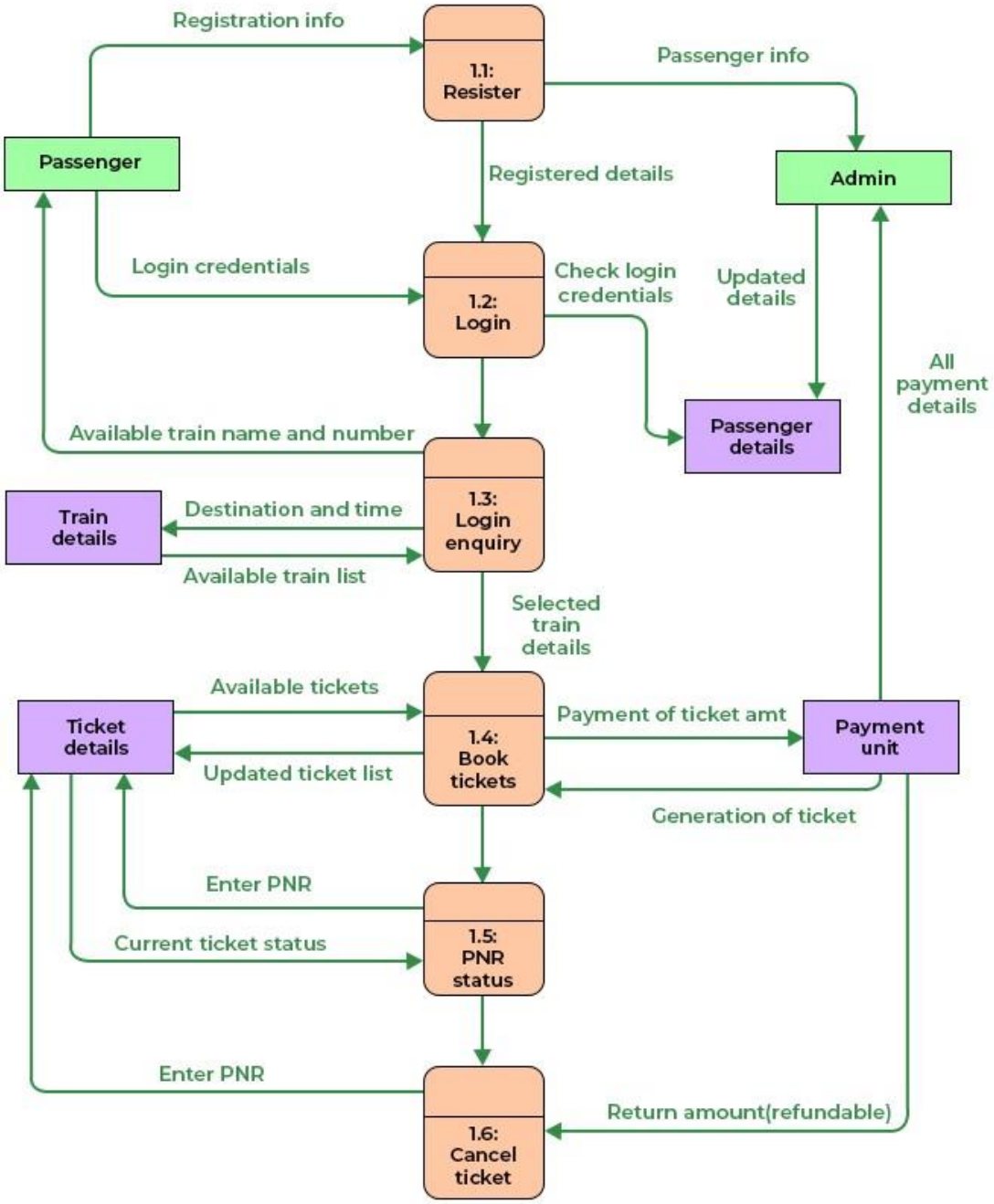


Fig 3.2 Level 1 Data Flow Diagram

3.3 Use Case Diagram:

By using use case diagrams, the interactions between a system and the users within that system will be represented.

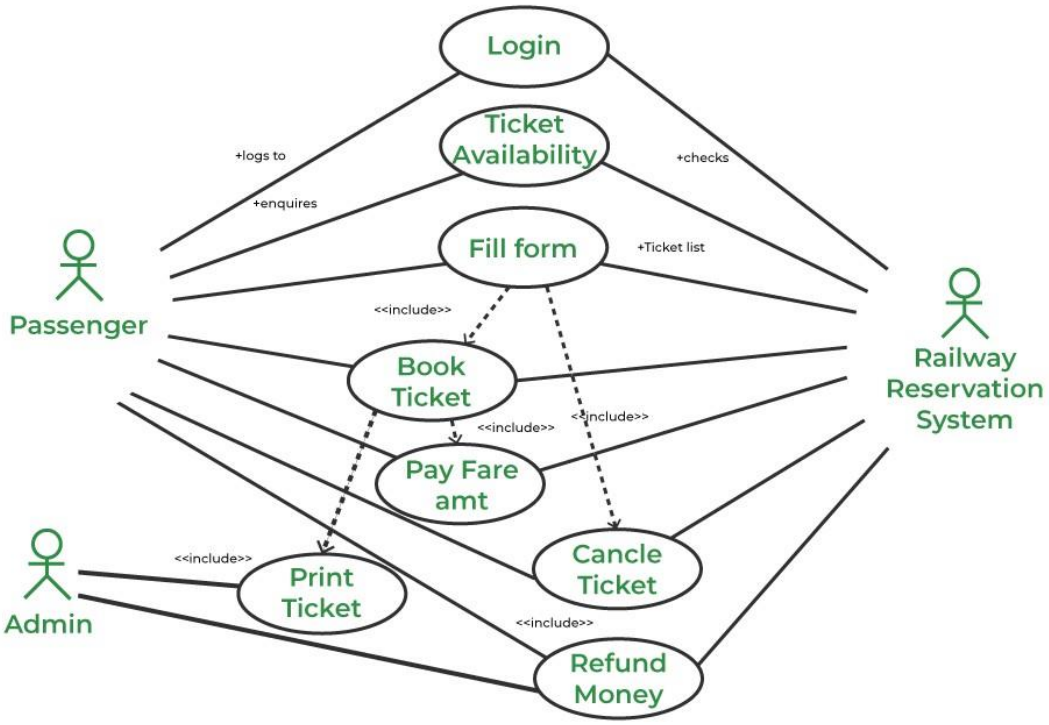


Fig 3.3 Use Case Diagram

Use-Case Descriptions:

The relationship between the actors and the use cases of the online Railway reservation system is given below–>

1. **Passenger Entity:** Use cases of passengers are login, ticket availability, Filling the form, Book ticket, Canceling ticket, and Refund money.
2. **Railway Reservation System:** Use cases of the Railway Reservation System are login, ticket availability, Fill the form, Book ticket, Cancel ticket, and Refunding money.
3. **Admin:** use cases of Admin are Print ticket, refund money. Admin also controls the whole Railway Reservation System in different cases.

3.4 Sequence Diagram:

This diagram shows how and in which order a group of objects works together in a system. This is an interactive diagram and this is mostly used by software developers.

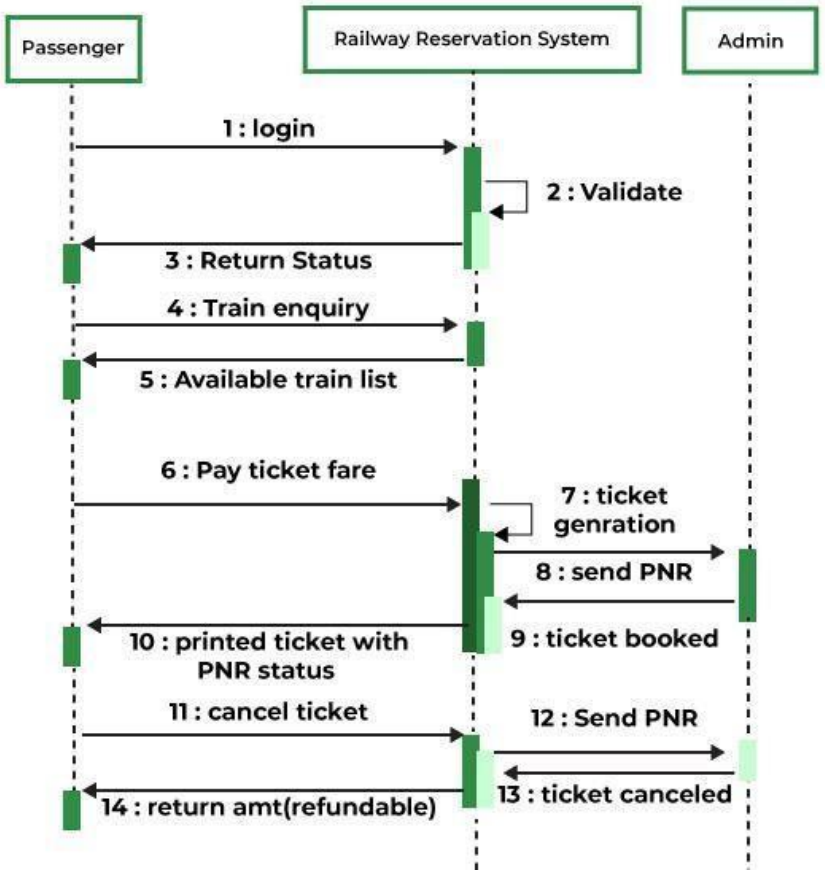


Fig 3.4 Sequence Diagram

4 MODULES AND FUNCTIONALITIES

4.1 User Header Section

```
<header id="header_wrapper">
<div class="container">
<div class="header_box">
<div class="logo"><a href="#"><img src="img/logo.png" alt="logo"></a></div>
<nav class="navbar navbar-inverse" role="navigation">
<div class="navbar-header">
<button type="button" id="nav-toggle" class="navbar-toggle" data-toggle="collapse"
data-target="#main-nav"> <span class="sr-only">Toggle navigation</span> <span
class="icon-bar"></span> <span class="icon-bar"></span> <span
class="iconbar"></span>
</button>
</div>
<div id="main-nav" class="collapse navbar-collapse navStyle">
<ul class="nav navbar-nav" id="mainNav">
<li class="active"><a href="#hero_section" class="scroll-link">Home</a></li>
<li><a href="train-booking.html" class="scroll-link">Book Tickets</a></li>
<li><a href="#aboutUs" class="scroll-link">About Us</a></li>
<li><a href="#service" class="scroll-link">Services</a></li>
<li><a href="#station" class="scroll-link">STATION</a></li>
<li><a href="#clients" class="scroll-link">Meals</a></li>
<li><a href="login.html" class="scroll-link">Login | Signup</a></li>
</ul>
</div>
```

```
            </nav>

          </div>
        </div> </header>
```

The header section which allows user to easily navigate around the page and can select whatever he wants.

4.2 Login Interface

```
            <script>
        const loginText = document.querySelector(".title-text .login");
const loginForm = document.querySelector("form.login");          const
loginBtn = document.querySelector("label.login");          const signupBtn
= document.querySelector("label.signup");          const signupLink =
document.querySelector("form .signup-link a");          signupBtn.onclick
= (()=>{            loginForm.style.marginLeft = "-50%";
          loginText.style.marginLeft = "-50%";
        });
        loginBtn.onclick = (()=>{
loginForm.style.marginLeft = "0%";
          loginText.style.marginLeft = "0%";
        });
        signupLink.onclick = (()=>{
          signupBtn.click();
return false;
        });
      </script>
```

The login page which enables user to enter his details and enter into the page to book the tickets so database can easily organize the information according to his login id's.

4.3 Train Booking Interface

```
          <div class="final animated flip">
   <h2> Here is Your Ticket</h2>
   <div class="ticket" id="ticket">
     <div class="row">
       <div class="col-xs-6 pull-left" id="date"></div>
       <div class="col-xs-6 pull-right ">INDIAN RAILWAYS</div>
     </div>
     <div class="row">
       <div class="col-xs-6 trainno"><span id="number"></span></div>
       <div class="col-xs-6 trainname"></div>
     </div>
     <div class="row">
       <div class="col-xs-6 from">From :<span id="From"></span></div>
```

```
      <div class="col-xs-6 to">To :<span id="To"></span></div>
</div>
    <div class="row">
```

This facility has many windows and categories, which provides an ease to passengers. By giving a few details and some payment, a traveler can choose among their favorite coaches between General, Sleeper, or AC coaches.

5 CODING AND TESTING

Front-End Code for Database

```
<body>


  <style>
.header-top        {
display: none;
   }

   .header-top {
     display: flex;
     background: #09e5ab;
color: #fff;
     padding: 8px 13px;
   }

   *,
   ::after,
::before {
     box-sizing: border-box;
   }

   .goog-te-gadget-simple {
background-color: #09e5ab;       border-left:
1px solid #d5d5d5;       border-top: 1px
solid #9b9b9b;       border-bottom: 1px
solid #e8e8e8;       border-right: 1px solid
#d5d5d5;
     font-size: 10pt;
display: inline-block;
padding-top: 1px;
padding-bottom: 2px;
cursor: pointer;
zoom: 1;       display:
inline;
     margin-right: 30px;
   }
  </style>
```

```
 <div class="header-top">

  <div class="right-top">
   <ul>
    <li>
     <div class="active" id='google_translate_element' /></i>
    </li>
   </ul>
  </div>
 </div>
<div class="index container animated zoomIn">

 <div class="form contactform">
  <div id="page1">
   <h1>Book Your Train Ticket</h1>
   <div class="index row">

    <div class="field col-sm-6">
     <label for="from">From :</label>
     <input id="from" type="text" />
    </div>

    <div class="field col-sm-6">
     <label for="to">To :</label>
     <input id="to" type="text" />
    </div>

    <div class="field col-sm-6 col-centered">
     <label for="datepicker">Date</label>
     <input id="datepicker" type="text" />
    </div>
    <div class="clearfix visible-xs"></div>
    <div class="field col-sm-6 col-centered ">
     <div id="search" class="btn  btn-primary">Search for Trains</div>
</div>
   </div>
  </div>
  <div id="page2" class="animated zoomInDown">
   <div class="row">
    <div class="col-xs-12">
     <div class="table-responsive  text-center">
      <table class="table table-bordered table-hover">
       <caption class="text-center">Available Trains : </caption>
       <thead>
        <tr>
         <th> Train No</th>
         <th>Train Name</th>
         <th>Time</th>
         <th>Cost</th>
        </tr>
```

```
        </thead>
        <tfoot>
          <tr>
            <td colspan='100%'><span class="btn btn-danger book">Book Now</span></td>
          </tr>
        </tfoot>
        <tbody class="member">
          <tr>
            <td class="tnum">78654</td>
            <td id="trainname1" class="tnam"></td>
            <td class="ttime">04:30 AM</td>
            <td class="tcost">₹ 200</td>
          </tr>
          <tr>
            <td class="tnum">54876</td>
            <td id="trainname2" class="tnam"></td>
            <td class="ttime">10:15 AM</td>
            <td class="tcost">₹ 250</td>
          </tr>
          <tr>
            <td class="tnum">25312</td>
            <td id="trainname3" class="tnam"></td>
            <td class="ttime">15:50 PM</td>
            <td class="tcost">₹ 300</td>
          </tr>
          <tr>
            <td class="tnum">45698</td>
            <td id="trainname4" class="tnam"></td>
            <td class="ttime">21:30 PM</td>

            <td class="tcost">₹350</td>
          </tr>
          <tr>
            <td class="tnum">95464</td>
            <td id="trainname5" class="tnam"></td>
            <td class="ttime">23:15 PM</td>

            <td class="tcost">₹ 500</td>
          </tr>

        </tbody>

      </table>
    </div>

  </div>
 </div>

</div>
<!-----page2------>
<div id="invis" class="animated fadeInUpBig invis">
```

```
      <div id="selectclass">
        <h2>Select your class :</h2>
        <div id="trainClass">
          <label class="blue"><input type="radio" name="toggle"><span>GEN</span></label>
          <label class="green"><input type="radio" name="toggle"><span>SL</span></label>
          <label class="yellow"><input type="radio" name="toggle"><span>AC1</span></label>
          <label class="pink"><input type="radio" name="toggle"><span>AC2</span></label>
          <label class="purple"><input type="radio" name="toggle"><span>AC3</span></label>
        </div>
        <div id="bfoot">
          <div class="col-xs-6 ">
            <span class="btn btn-primary booknow">BOOK NOW</span>
          </div>
          <div class="col-xs-6 ">
            <span class="btn btn-danger bookcancel">CANCEL</span>
          </div>
        </div>
      </div>
    </div>
  </div>
 <div class="final animated flip">
  <h2> Here is Your Ticket</h2>
  <div class="ticket" id="ticket">
    <div class="row">
      <div class="col-xs-6 pull-left" id="date"></div>
      <div class="col-xs-6 pull-right ">INDIAN RAILWAYS</div>
    </div>
    <div class="row">
      <div class="col-xs-6 trainno"><span id="number"></span></div>
      <div class="col-xs-6 trainname"></div>
    </div>
    <div class="row">
      <div class="col-xs-6 from">From :<span id="From"></span></div>
      <div class="col-xs-6 to">To :<span id="To"></span></div>
    </div>
    <div class="row">
      <!---- <div class="col-xs-6 barcode">
      <table id="barcodes">
        <tr></tr>
      </table>
    </div>----->
      <div class="col-xs-2 compartment">S3</div>
      <div class="col-xs-4 seatno">Seat No:45</div>
    </div>
  </div>
  </div>
  <!-- partial -->

  <script src='https://code.jquery.com/jquery-2.2.4.min.js'></script>
```

```
    <script src='https://cdnjs.cloudflare.com/ajax/libs/jqueryui/1.11.4/jquery-ui.min.js'></script>
<script src="js/script.js"></script>

    <!------Language Translate script start------>
    <script src="js/language_tran.js"></script>
    <!-------Language Translate script End/----------------->


</body>
```

Back-End Code for Database

```
$(document).ready(function() {

    $(function() {
      $("#datepicker").datepicker();
    });

    $(function() {
      $("#timepicker").timepicker();
    });

    var tnum, tnam, ttime, tcost, classname;
    var from, to, date, availableCities = [
```

```
    "Mumbai",
    "Kolkata",
    "Delhi",
    "Chennai",
    "Bangalore",
    "Hyderabad",
    "Ahmadabad",
    "Pune",
    "Surat",
    "Kanpur",
    "Jaipur",
    "Lucknow",
    "Nagpur",
    "Patna",
    "Indore",
    "Vadodara",
    "Bhopal",
    "Coimbatore",
    "Ludhiana",
    "Kochi",
    "Visakhapatnam",
    "Agra",
    "Varanasi",
    "Madurai",
    "Meerut",
    "Nashik",
    "Jabalpur",
    "Jamshedpur",
    "Asansol",
    "Dhanbad",
    "Faridabad",
    "Allahabad",
    "Amritsar",
    "Vijayawada",
    "Rajkot"
  ];
  $("#from").autocomplete({
    source: availableCities
  });
  $("#to").autocomplete({
    source: availableCities
  });

  $(".invis").hide();
  $("#page2").hide();
  $(".final").hide();

  $("#search").click(function() {
from = $("#from").val();      to =
$("#to").val();      date =
$("#datepicker").val();      if (!(from
```

```
  return False;
    }
    $("#page1").hide();
    $("#page2").show();

    $("#trainname1").html(from + " Express");
    $("#trainname2").html(to + " Express");
    $("#trainname3").html(from + " Passenger");
    $("#trainname4").html(to + " Passenger");
    $("#trainname5").html(from + " - " + to + " Super Fast Train");

    $("tbody > tr").mouseover(function() {
      $(this).css("backgroundColor", "rgba(41, 103, 182, 0.89)");

    }).mouseout(function() {
      $(this).css("backgroundColor", "");
    });

    $("tbody > tr").click(function() {
      $(this).parent().children().removeClass("selected");
      $(this).addClass("selected");
    });

    $(".book").click(function() {
tcost = $(".selected").find(".tcost")
        .text();
      tnum = $(".selected").find(".tnum").text();

      tnam = $(".selected").find(".tnam").text();

      ttime = $(".selected").find(".ttime").text();
      /* alert(tnum);*/
if (!tnum) {
        alert("Please Select Your Train !")
        return False;
      }
      $(".invis").show();

      $(".booknow").click(function() {
        classname = document.querySelector('input[name="toggle"]:checked+span').innerHTML;

        $(".invis").hide(function() {
          $("#page2").hide()
        });
        $(".index").hide();
        $(".final").show();

      });
```

```
        $(".bookcancel").click(function() {

          $(".invis").hide();
        })

        $("#From").html(from);
       $("#To").html(to);
    $(".trainname").html(tnam);
        $("#number").html(tnum);

        var d = new Date();         var n
= d.toLocaleDateString();
        document.getElementById("date").innerHTML = n;

        var code =
'11010010000100111011001011101111011010001110101110011001101110010010111101110111001
01100100100001101100011101011';

        table = $('#barcodes tr');
        for (var i = 0; i < code.length; i++) {
if (code[i] == 1) {
            table.append('<td bgcolor="black">')
          } else {
            table.append('<td bgcolor="white">')
          }
        }
```

Login

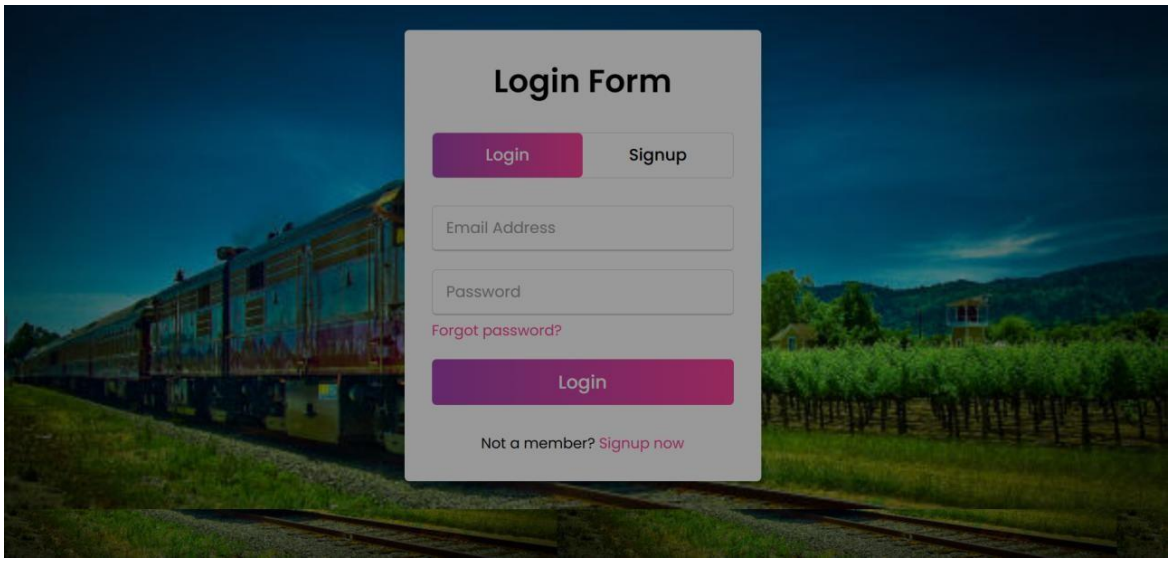
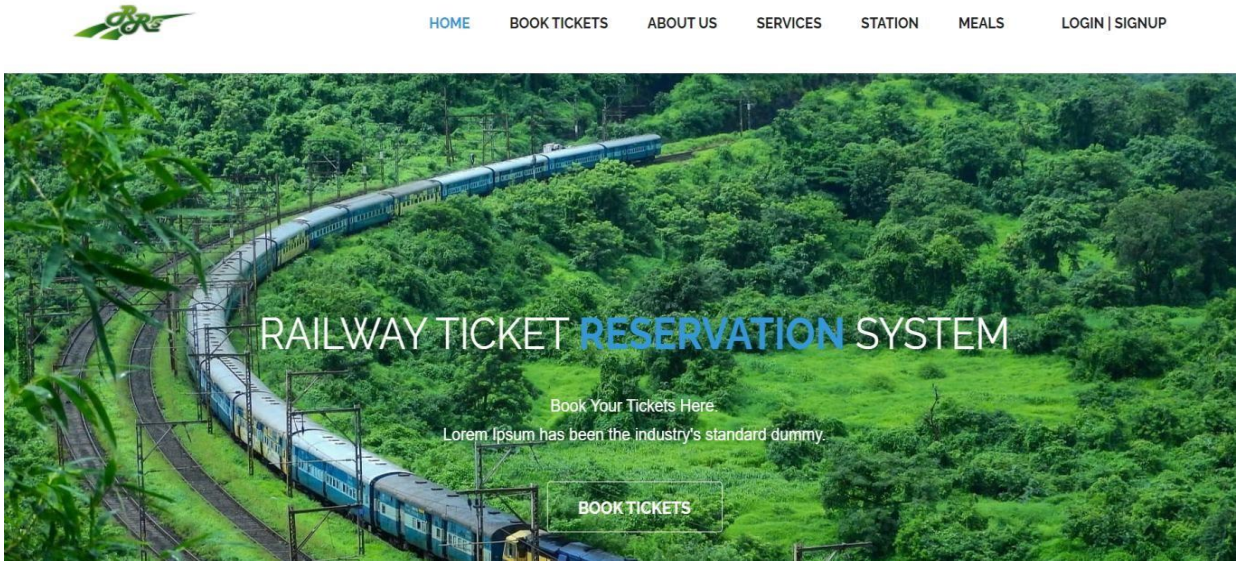


Fig 5.1 Login Page

To book a ticket a user first login with user name and password and if user forget his password then there is another option to change the password.

Fig 5.2 Home page

User Interface



For searching the train user should enter the source(to) and destination(from),date and class in which the the user want to go.To press the “search now” user can know the train details.

Ticket

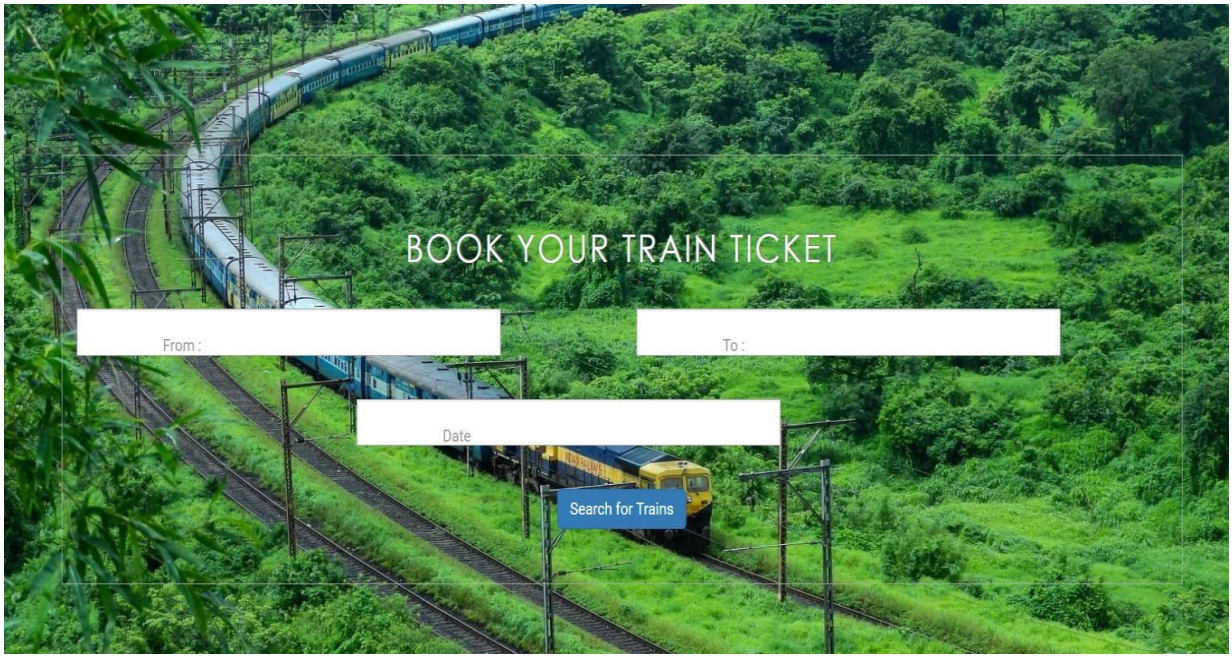


Fig 5.3 Ticket booking page

Ticket booking interface where user can enter the source and destination details and date on which he is gonna travel.

Here user find the train detail and to press the “book ticket” option for booking the ticket.

Database of trains

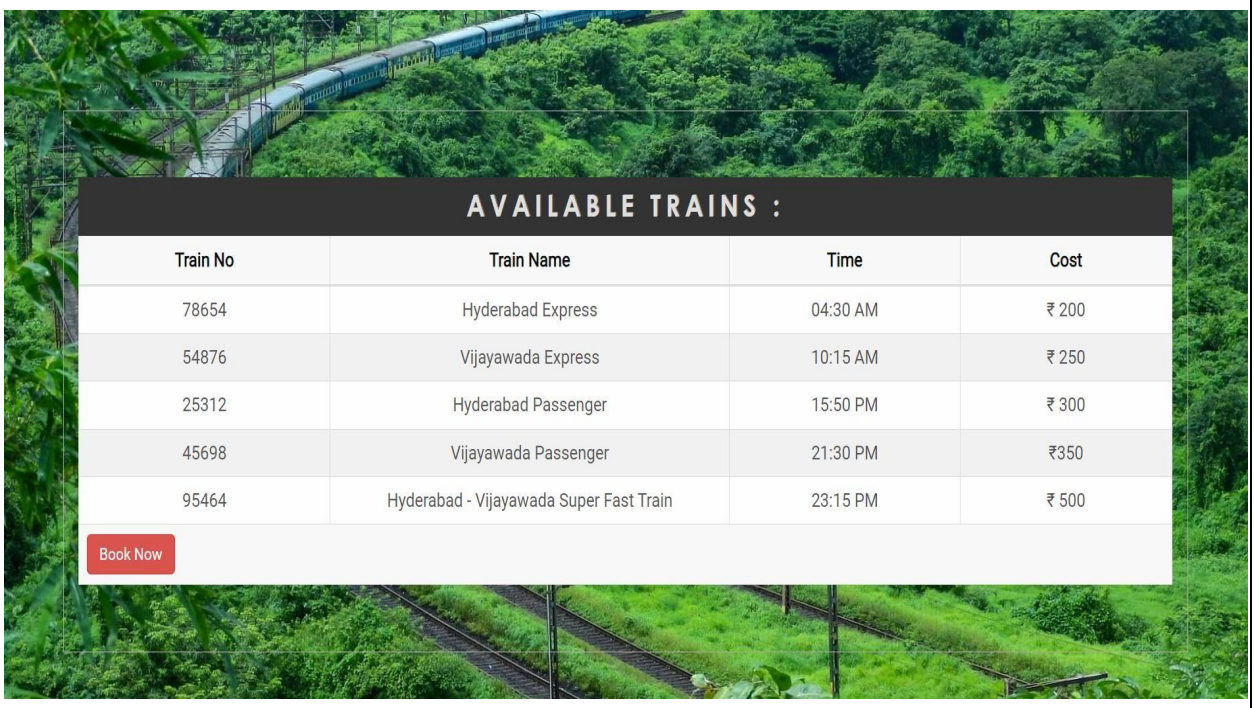


Fig 5.4 Available running tarins from source to destination

After filling up all the information in the payment page user press the “continue to pay” button and the following page is coming.
In this page user find his ticket no.
If the user wants to cancel the ticket then he/she press the cancel ticket option to cancel the
ticket .In that case user should fill the name and ticket no to find the train and then click the cancel ticket hyperlink which delete the ticket from the datadase.

Generated E-Ticket

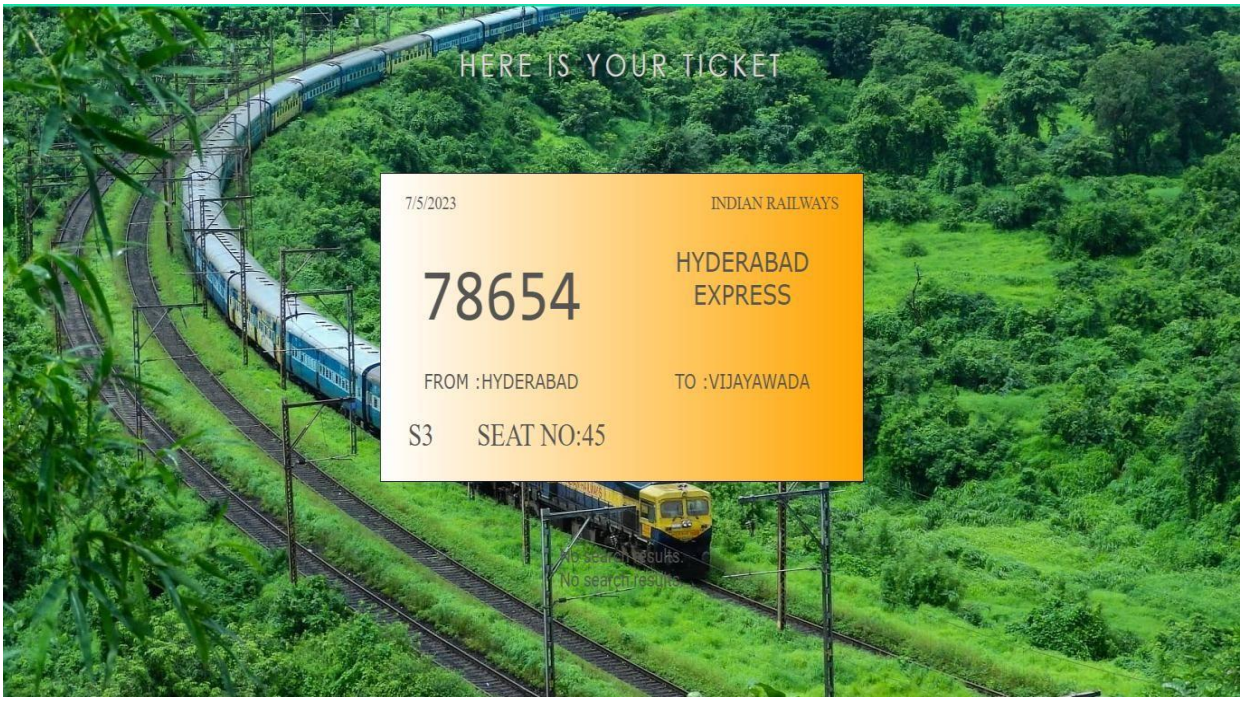


Fig 5.5 E-ticket

6 Result

UNIT	PURPOSE	VERIFIED
Search Train	This unit search the trains.	Yes
Train details	This unit shows the trains of a particular source to destination in a particular date and a specific seat .	Yes
Book ticket.	This unit user can select a particular train and book ticket .	Yes
Login.	This unit login the registered user and create an account for a new user.	Yes
Fill the booking form.	User fill the form to book ticket.	Yes
payment.	User fill the form and pay the money with the help of credit card.	Yes
Ticket no	This unit allows to show his ticket no.	Yes
Cancel ticket	This unit allows user to cancel ticket.	Yes

Starting from the bottom the first test level is "Unit Testing". It involves checking that each feature specified in the "Component Design" has been implemented in the component.

In theory an independent tester should do this, but in practice the developer usually does it, as they are the only people who understand how a component works. The problem with a component is that it performs only a small part of the functionality of a system, and it relies on co-operating with other parts of the system, which may not have been built yet. To overcome this, the developer either builds, or uses special software to trick the component into believe it is working in a fully functional system.

7 CONCLUSION

In our project railway system we have all the information saved regarding the train, passengers, tracks, where and how The Train moves, station, schedules, routes and what it consists of. 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

The field of technology is becoming more in advance. Considering of Railway department, e-ticketing facility was introduced where users browse through the governmental website and book their long journey tickets which is later printed to show to the checker when needed. After that a new technique was introduced called M-ticketing where user messaged to the web portal through mobile phone after which a complete web page was downloaded to the mobile phone after that user can perform all the booking process as like in eticketing facility.

8. REFERENCES

- Tutorials from YouTube
- Tutorials from Google
- https://www.javatpoint.com/
- https://www.studocu.com/in/document/dwarkadas-j-sanghvi-college-ofengineering/computer-science/railway-reservation-system-project-
- https://www.slideshare.net/shashankkarnati/railway-management-systemdatabase-mini-project
- https://www.slideshare.net/shashankkarnati/railway-management-systemdatabase-mini-project
- https://www.scribd.com/document/544322912/Railway-ManagementSystem-Final-Project
- https://www.freeprojectz.com/project-report/2362
- http://www.muengineers.in/computer-project-list/php-project-list/railwayreservation-system
- https://itsourcecode.com/uml/railway-management-system-er-diagram/