

CHAPTER 1

INTRODUCTION

1.1 General Overview

Our website has various kinds of information that helps regarding booking of tickets via railways .

Users will be able to search the train availability ,the exact fare ,the arrival and departure time of the train and they can also book the ticket by using the debit ,credit or master card and after booking the ticket if the user want to cancel it then they can easily do it also.

1.2 Survey

Railway passengers frequently need to know about their ticket reservation status, ticket availability on a particular train or for a place, train arrival or departure details, special trains etc.. Customer information centers at the railway stations are unable to serve such queries at peak periods. The number of the reservation counters available to the passengers and customers are very less. On most of the reservation systems there are long queues, so it takes a long time for any individual to book the ticket. As now there are no call centers facilities available to solve the queries of the passengers.

The online railway ticket reservation system aims to develop a web application which aims at providing trains details, trains availability, as well as the facility to book ticket in online for customers.

So, we thought of developing a web based application which would provide the users all these facilities from his terminal only as well as help them in booking their tickets. The Application was to be divided into two parts namely the user part , and the administrator part. And each of these has their corresponding features.

We decided to give the name of the website “ONLINE TRAIN TICKET RESEVATION SYSTEM”.

.

1.3 Objectives: ONLINE TRAIN TICKET RESEVATION SYSTEM PROJECT

The objective of the online train ticket reservation system

Project is to design software to fully automate the process of issuing a railway ticket. That is:- 1. To create a database of the trains

2. To search the trains it's arrival and departure time,distance between source and destination.

3.To check the availability of the ticket.

4. To calculate fare.

5.To book the ticket.

6.To cancel the ticket if necessary.

1.3 Analysis

Online railway ticket reservation is a online ticket booking website, which is capable of booking ticket and serach the train availavility . This website is mainly created to fulfil the following requirements, it comprises of the following properties:-

- ♣ 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.

The report has been organized into seven sections, each section dealing with a different aspect of project and its development. A brief overview of each of the section is in order.

The Software Requirements Specification section will deal with the technical details of the proposed system. It will contain information regarding the developmental and operational environments, user interfaces, functional and requirements specifications and the exception and error handling features of the system.

The Design section will provide a detailed insight into the working of the system, how the system to be framed to make the implementation error free and to avoid a possible rework. All the features mentioned in the SRS section will be dealt with in detail through the means of architectural design, block diagram, detailed data flow diagrams, structure charts, flow chart and any other relevant method.

Through the use of pseudo-codes, the Implementation Details section will provide the final design step before actual implementation of the system. It will list all the functions that will be used in the system, parameters used by them, what results they will produce and how they will interact with the rest of the system.

All queries regarding the actual performance of the developed system will be cleared in the Results and Discussion section. It will list all the test cases, results of those tests and a discussion on whether these tests yielded the desired results or not. If negative, this section will also provide information regarding the failure.

Concluding all this will be the Summary and Conclusion section which will round up the entire discussion.

This will be followed by the Reference/Bibliography section that will tell about the books and site that have been referred by us while developing the website.

CHAPTER.2

SOFTWARE REQUIREMENTS SPECIFICATION- “ONLINE TRAIN TICKET RESEVATION SYSTEM PROJECT”

2.1 Development Environments

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 or higher.
- XAMPP.
- Internet Explorer or Google Chrome.

2.2 Operating environment:

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
- Any standard web browser

CHAPTER.3

SDLC (SOFTWARE DEVELOPMENT LIFE CYCLE) – ONLINE TRAIN TICKET RESEVATION SYSTEM PROJECT

Every activity has a life cycle and software development process is not an exception for the same. Even if you are not aware of SDLC you still must be following it unknowingly. But if a software professional is aware about SDLC he can execute the project in a much controlled fashion. One of the big benefits of this awareness is that hot blooded developers will not start directly execution (coding) which can really lead to project running in an uncontrolled fashion. Second it helps customer and software professional to avoid Confusion by anticipating the problems and issues before hand. In short SDLC defines the various stages in a software life cycle. But before we try to understand what SDLC is all about. We need to get a broader view of the start and end of SDLC. Any project started if it does not have a start and end then its already in trouble. It's like if you go out for a drive you should know where to start and where to end or else you are moving around endlessly.

Below is the figure that shows typical flow in SDLC which has five main models .

- Waterfall - Big Bang and Phased model .
- Iterative - Spiral and Incremental model.

Iterative model

Iterative model was introduced because of problems faced in Waterfall model.

The iterative waterfall model is used in the development of the system. The system is developed in increments, each increments adding some functional capability to the system until the full system is fully implemented.

The advantage of this approach is that it will result in better testing, as testing of each increment is easier than testing the entire system in totality. Furthermore, this approach provided us with important feedback that was very useful in the implementation of the system.

Feasibility Study

Integration and System testing

Requirement analysis and specification

Coding and unit testing

Design

Maintenance

Development Schedule

The work on the proposed ONLINE RAILWAY TICKET RESERVATION was started on 17th June, 2011 and it was estimated to be over by 17th of August, 2011.

The following Gantt chart has explained the estimated duration of the different phases of the software development work diagrammatically:

Activities

.

3.1 Implementation Details

3.1.1 System Overview

Online railway ticket reservation system is a web based application. It has been developed using ASP.NET and C# as the code end programming language. A MS SQL Server database consisting of multiple tables is used for data storage.

This Website is a online travel booking website, which is capable of booking tour of national and international destination with easy steps at the cheapest rate . It allows the Administrator to perform all operations and view bookings. The general user is however given only restricted access.

Travelmasti helps in booking tour packages. The user can't book any tour until he is a registered user.

Feasibility Study

Requirement gathering and specification

Database Design

GUI Design

Code database part

Code GUI part

Unit & System testing

17th-30th june

1st -15th july

25th -5th august

5th-17th august

16th-25th july

3.2MySQL

MySQL server: - ONLINE TRAIN TICKET RESEVATION SYSTEM PROJECT

MySQL server is basically a database server which is mostly used for storing user data into the required database in a specific table for easy access of these data in the future. The database server works when a local application invokes it. Before storing data into a database with the help of database server, PHP sends a SQL query to MySQL database server for establishing a connection to the server by using the loopback address, i.e., 127.0.0.1 along with the user name and password for getting authentication from the database server. Since connection is established locally with the database server by PHP, so there is no chance of getting access to database server for modifying database by the client. Also, when PHP request for information retrieval from the database server by sending SQL query then the database server sends back the result of query after executing it by the database server. Another method of accessing database server is only possible by locally, i.e., only administrator of that particular computer can get all the access facilities of the database server as well as all the database which are not available to restricted users. PHP can update information in the database server if it has the administrator username and password to access the specific database, otherwise connection will be rejected by MySQL server and the database cannot be updated.

Since before the dawn of the computer age, people have been using databases. Before computers, a database may have been a Rolodex containing phone numbers of the important people you knew, or it was a filing cabinet that contained all the personnel records for the company. Today, databases are computer-based and are found virtually everywhere. From desktop databases of your record collection to Web-enabled databases that run large corporations.

CHAPTER.4

IMPLEMENTATION

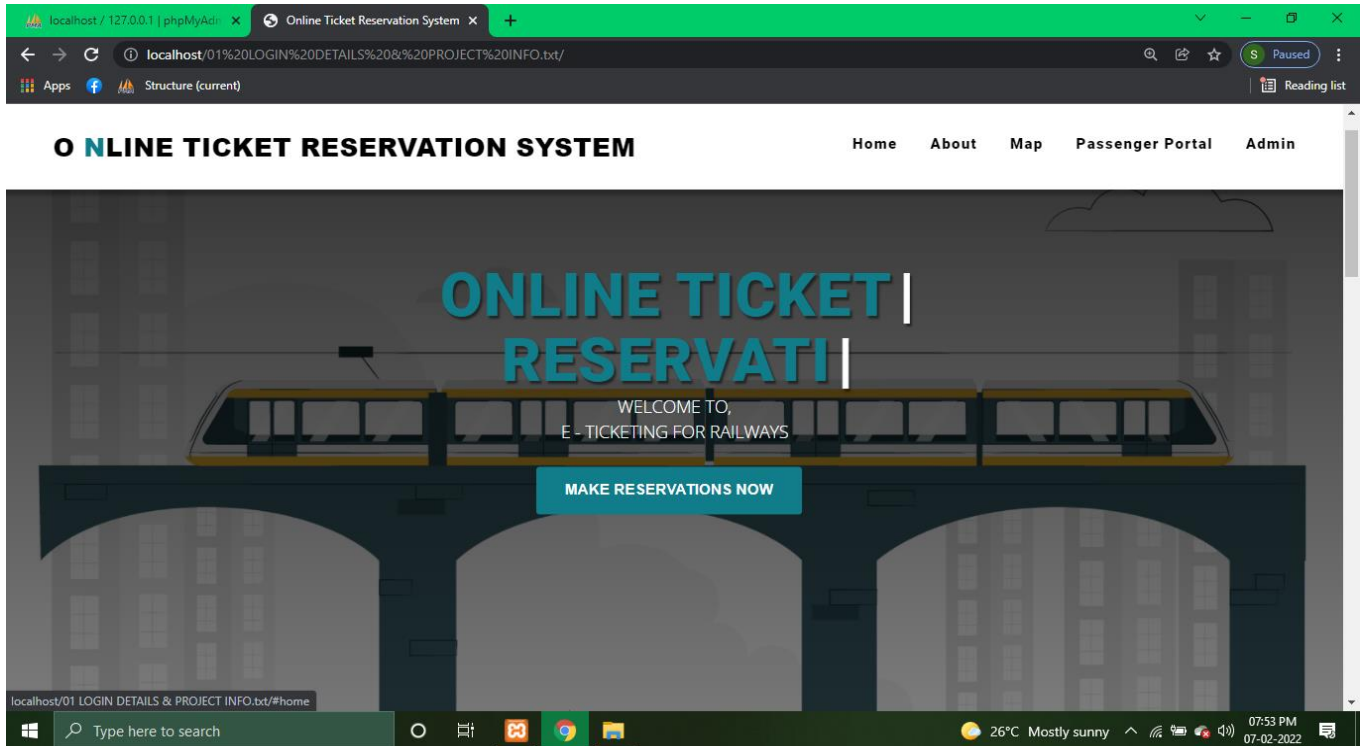
Following are the related files

- account.php: the main page of the website after login
- book.php: for booking seats
- goforcancellation.php, cancel.php: for cancelling seats
- captcha.php: for captcha generation
- changepassword.php, changepasswordfinal.php: for changing passwords
- dblink.php: for establishing link with the database
- findTrains.php: finding trains between station
- footer.php: general footer code for all php files
- generate.php: generating test data and some tables
- getAvailability: getting availability of seats
- guard.php: destroys timed out sessions and redirects to index.php
- header.js: Javascript functions to assist the website on client side
- header.php: general header code for most php files
- history.php: getting the booking history of current user account
- index.php: main page of the website when not logged in
- logout.php: Logs the user out
- pnrStatus.php: echoes back with PNR status related data
- Railways.sql: contains SQL code for generating tables and sequences
- register.php: registers new users into the website
- tdr.php, tdrfiling.php, tdrfinal.php: TDR related files
- triggers.sql: triggers for checking whether validity constraints are satisfied
- updateprofile.php, updateprofilefinal.php: User profile updation
- datagencodes: Python and Bash codes for generating and importing realistic data.

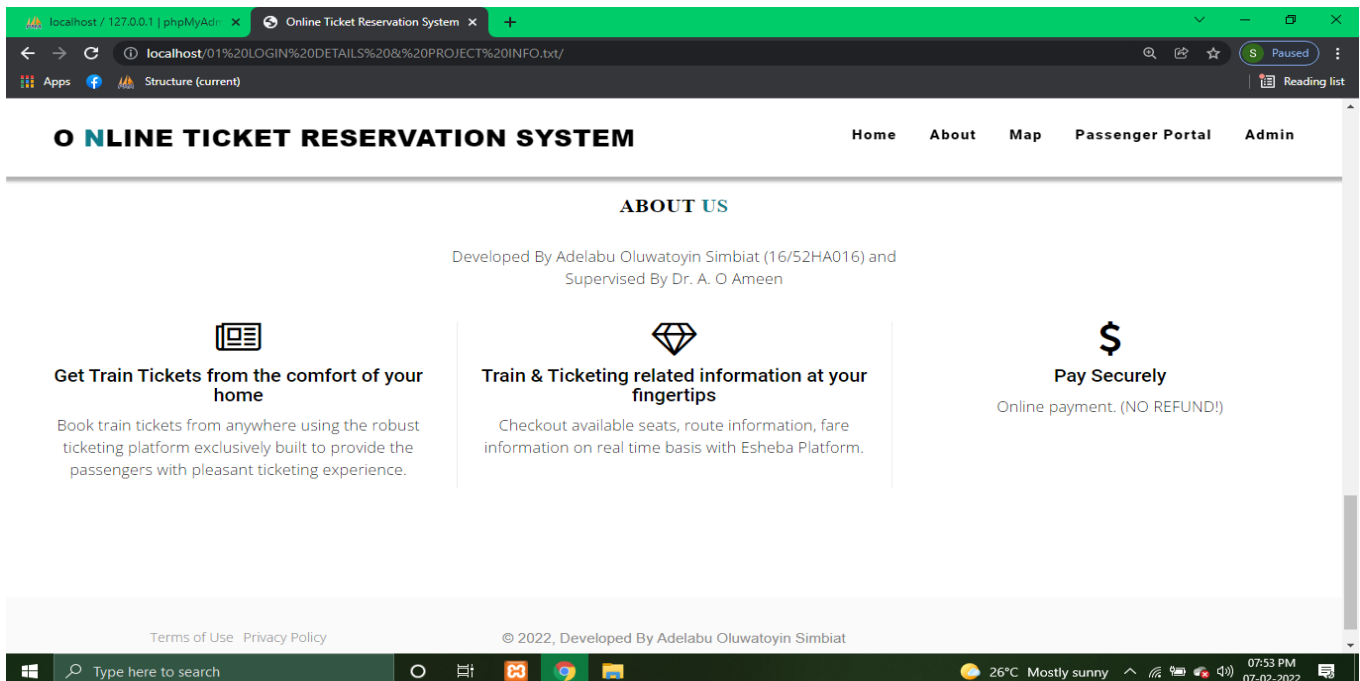
CHAPTER.5

SCREENSHOTS

5.1 HOMEPAGE:



5.2 ABOUT US:



5.3 CUSTOMER PANEL:

Customer Panel

Email Address

admin

Password

SIGN IN

5.4 CUSTOMER REGISTER PAGE:

Create Account

You need to create an account to book/view trains!

Full Name

Contact Number

Email Address

Select Picture

Choose File No file chosen

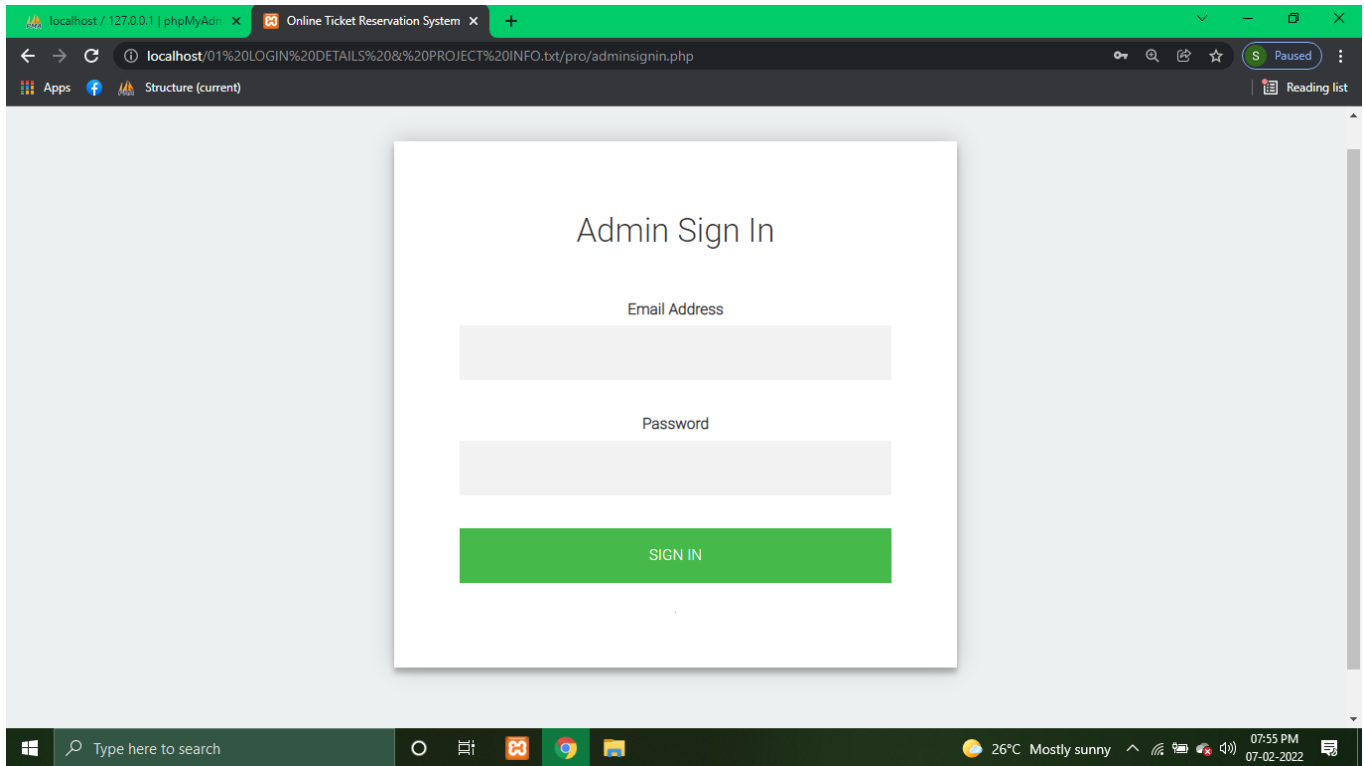
Address

Password

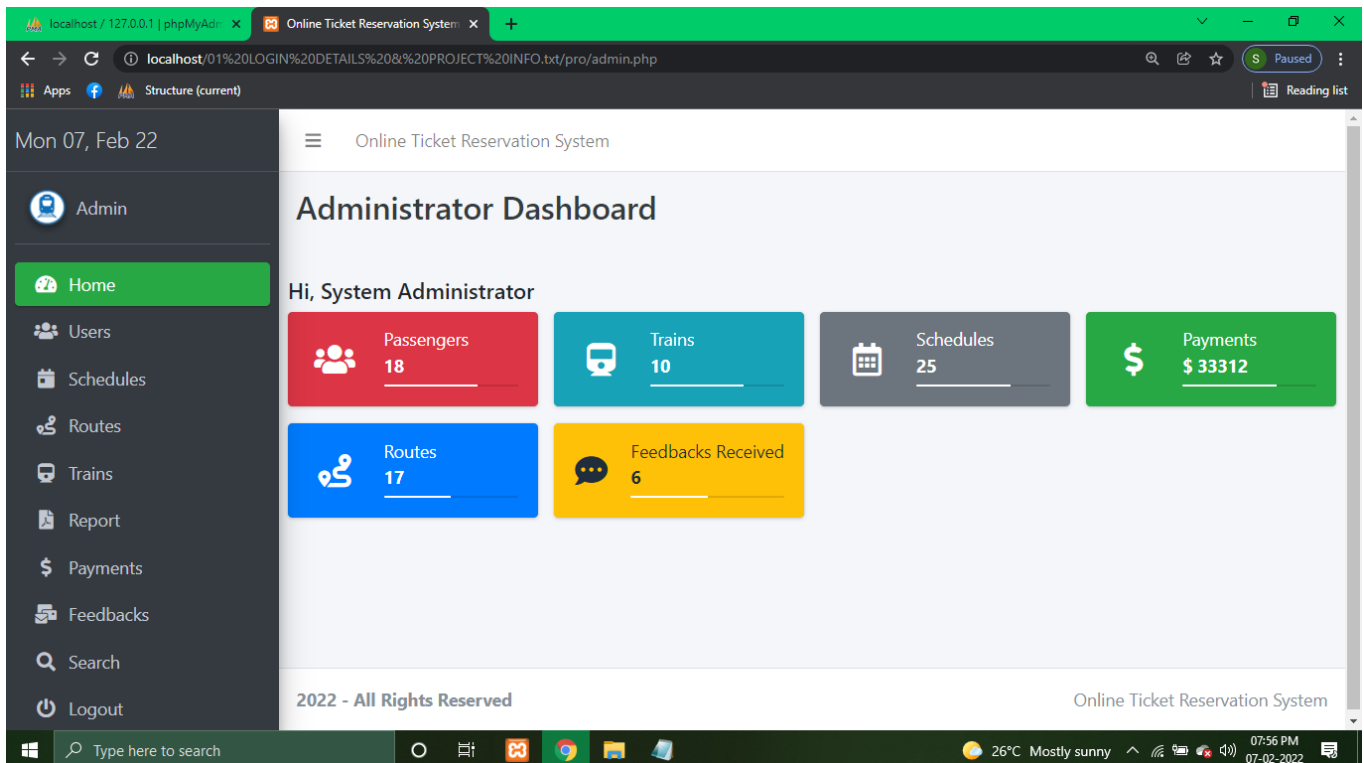
Confirm Password

CREATE ACCOUNT

5.5 ADMIN SIGN IN:



5.6 ADMINISTRATOR DASHBOARD:



5.7 REGISTERED USERS:




localhost / 127.0.0.1 | phpMyAdmin x Online Ticket Reservation System x +

localhost/01%20LOGIN%20DETAILS%20&%20PROJECT%20INFO.txt/pro/admin.php?page=users

Apps Structure (current) Reading list

Registered Users

Show 10 entries Search:

#	Full Name	Email	Contact	Image	Action
1	Monika Rani P	Monika@mail.com	0000000000		Disable Account
2	Akhila Akhila	akhila@mail.com	0111111111		Disable Account
3	Chandhana K	chandhana@mail.com	0222222222		Disable

Type here to search 26°C Mostly sunny 07:56 PM 07-02-2022

localhost / 127.0.0.1 | phpMyAdmin x Online Ticket Reservation System x +








localhost/01%20LOGIN%20DETAILS%20&%20PROJECT%20INFO.txt/pro/admin.php?page=users

Apps Structure (current) Reading list

Administrator Dashboard

Registered Users

Show 10 entries Search:

#	Full Name	Email	Contact	Image	Action
1	Monika Rani P	Monika@mail.com	0000000000		Disable Account
2	Akhila Akhila	akhila@mail.com	0111111111		Disable Account
3	Chandhana K	chandhana@mail.com	0222222222		Disable Account
4	Sachin Sachi	sachi@mail.com	0333333333		Disable Account
5	Enosh Leyandor	enosh@mail.com	0444444444		Disable Account
6	Pradeep BM	pradeep@mail.com	0555555555		Disable Account
7	Keerthana Gowda	keerthana@mail.com	0666666666		Disable Account

Type here to search 25°C Partly cloudy 07:57 PM 07-02-2022

5.8 SCHEDULES:

Administrator Dashboard

All Dynamic Schedules [Add New One-Time Schedule](#) [Add Range Schedule](#)

Show 10 entries Search:

#	Train	Route	F.C Fee	S.C Fee	Total Bookings	Date/Time	Actions
1	Phantom Express	Vlinginia to Onaginia	\$ 160	\$ 72	250 Seat(s) Available for First Class 598 Seat(s) Available for Second Class	24-01-2022 / 10:00 AM	Edit Delete
2	Renfee R	Belagavi to Mysore	\$ 120	\$ 75	397 Seat(s) Available for First Class 850 Seat(s) Available for Second Class	14-01-2022 / 04:45 AM	Edit Delete
3	Lagos Rails	Mysore to Mandya	\$ 105	\$ 65	400 Seat(s) Available for First Class 999 Seat(s) Available for Second Class	14-01-2022 / 11:02 AM	Edit Delete
4	British Railways	Udupi to Mangalore	\$ 105	\$ 55	19 Seat(s) Available for First Class 900 Seat(s) Available for Second Class	12-10-2021 / 10:59 PM	Edit Delete
5	Renfee R	Ramanagara to Mysore	\$ 100	\$ 50	399 Seat(s) Available for First Class 850 Seat(s) Available for Second Class	12-10-2021 / 12:20 AM	Edit Delete
6	Marble Railways	Belagavi to Mysore	\$ 166	\$ 79	394 Seat(s) Available for First Class	12-10-2021 / 11:10 AM	Edit Delete

5.9 EDITING SCHEDULES:

Editing Schedule

Train : Phantom Express

Route : Vlinginia to Onaginia

First Class Charge : 160

Second Class Charge : 72

Date : 24-01-2022

Time : 10:00 AM

[Edit Schedule](#) [Close](#)

5.10 ROUTES:

All Routes [Add New Route](#)

Show 10 entries Search:

#	From	To	Action
1	Banglore	Mysore	Edit Delete
2	Mysore	Mandya	Edit Delete
3	Mysore	Goa	Edit Delete
4	Mysore	Thirupathi	Edit Delete
5	Banglore	Mangalore	Edit Delete
6	Hubballi	Goa	Edit Delete
7	Kolar	Chithradurga	Edit Delete
8	Udupi	Mangalore	Edit Delete

5.11 ALL TRAINS:

All Trains [Add New Train](#)

Show 10 entries Search:

#	Train Name	First Class Seat	Second Class Seat	Action
1	Kano Rails	30	800	Edit Delete
2	British Railways	20	900	Edit Delete
3	Wester Railways	10	600	Edit Delete
4	Lagos Rails	400	1000	Edit Delete
5	Marble Railways	395	780	Edit Delete
6	Renfee R	400	850	Edit Delete
7	Venice Express	500	1200	Edit Delete
8	Orient Express	200	500	Edit Delete

5.12 REPORTS:

All Schedules

Show 10 entries

#	Train	Route	Date/Time	Actions
1	Phantom Express	Vlinginia to Onaginia	24-01-2022 / 10:00 AM	View
2	Renfee R	Belagavi to Mysore	14-01-2022 / 04:45 AM	View
3	Lagos Rails	Mysore to Mandya	14-01-2022 / 11:02 AM	View
4	British Railways	Udupi to Mangalore	12-10-2021 / 10:59 PM	View
5	Renfee R	Ramanagara to Mysore	12-10-2021 / 12:20 AM	View
6	Marble Railways	Belagavi to Mysore	12-10-2021 / 11:10 AM	View
7	Venice Express	Urmkirkey to Longdale	12-10-2021 / 09:00 AM	View
8	Orient Express	Hubballi to Goa	11-10-2021 / 11:05 AM	View

5.13 PAYMENTS:

All Payments

Show 10 entries

Route	Date	First Class	Second Class	Capacity
Banglore to Mangalore	11-08-2020 - 6:30 PM	\$ 2380	\$ 0	400 Seat(s) Available for First Class 994 Seat(s) Available for Second Class
Banglore to Mangalore	11-08-2020 - 6:30 PM	\$ 2380	\$ 0	400 Seat(s) Available for First Class 994 Seat(s) Available for Second Class
Banglore to Mysore	08-11-2020 - 3:13 PM	\$ 0	\$ 1410	394 Seat(s) Available for First Class 780 Seat(s) Available for Second Class
Banglore to Mysore	15-11-2020 - 3:22 PM	\$ 0	\$ 5050	18 Seat(s) Available for First Class 900 Seat(s) Available for Second Class
Banglore to Mysore	18-09-2021 - 09:00 AM	\$ 5260	\$ 0	20 Seat(s) Available for First Class 896 Seat(s) Available for Second Class

5.14 FEEDBACKS:

Administrator Dashboard

All Feedbacks

Show 10 entries

#	Passenger	Message	Status	Action
1	Passenger Two	This is a demo test.		Reply
2	Passenger Three	This is a feedback text		Reply
3	Test Passenger	Test Test Test Test Test		Reply
4	Test Passenger	Demo Test - 2	Are you sure that this is another test?	No action
5	Demo Account	This is a demo test for feedback sections!!!	none	No action
6	Warning: Trying to access array offset on value of type null in C:\xampp\htdocs\01 LOGIN DETAILS &			

5.15 PASSENGER'S DASHBOARD:

Passenger's Dashboard

Quick Tips

Use the links at the left.
 You can see list of schedules by clicking on "New Booking". The system will display list of available schedules for you which you can view and make bookings from.
 Before your bookings are saved, you are redirected to make payment.
 After a successful payment, system generates your ticket ID for you which you are required to bring to the station.
 You are allowed to view all your booking history by clicking on "View Bookings".

2022 - All Rights Reserved

Online Ticket Reservation System

5.16NEW BOOKING:

The screenshot shows the 'Passenger's Dashboard' of the Online Ticket Reservation System. The dashboard has a sidebar with navigation links: Home, New Booking (highlighted), View Bookings, Feedback, and Logout. The main content area is titled 'Book Train Tickets' and displays a table of available routes. The table has columns for #, Route, Status, Date/Time, and Actions. Two routes are listed: Belagavi to Mysore and Vlinginia to Onaginia. Each route shows the number of seats available for First and Second Class. A 'Book' button is present for each route. The dashboard also includes a search bar, a 'Show 10 entries' dropdown, and pagination controls.

#	Route	Status	Date/Time	Actions
1	Belagavi to Mysore	397 Seat(s) Available for First Class 850 Seat(s) Available for Second Class	08-02-2022 / 04:45 AM	Book
2	Vlinginia to Onaginia	250 Seat(s) Available for First Class 598 Seat(s) Available for Second Class	08-02-2022 / 10:00 AM	Book

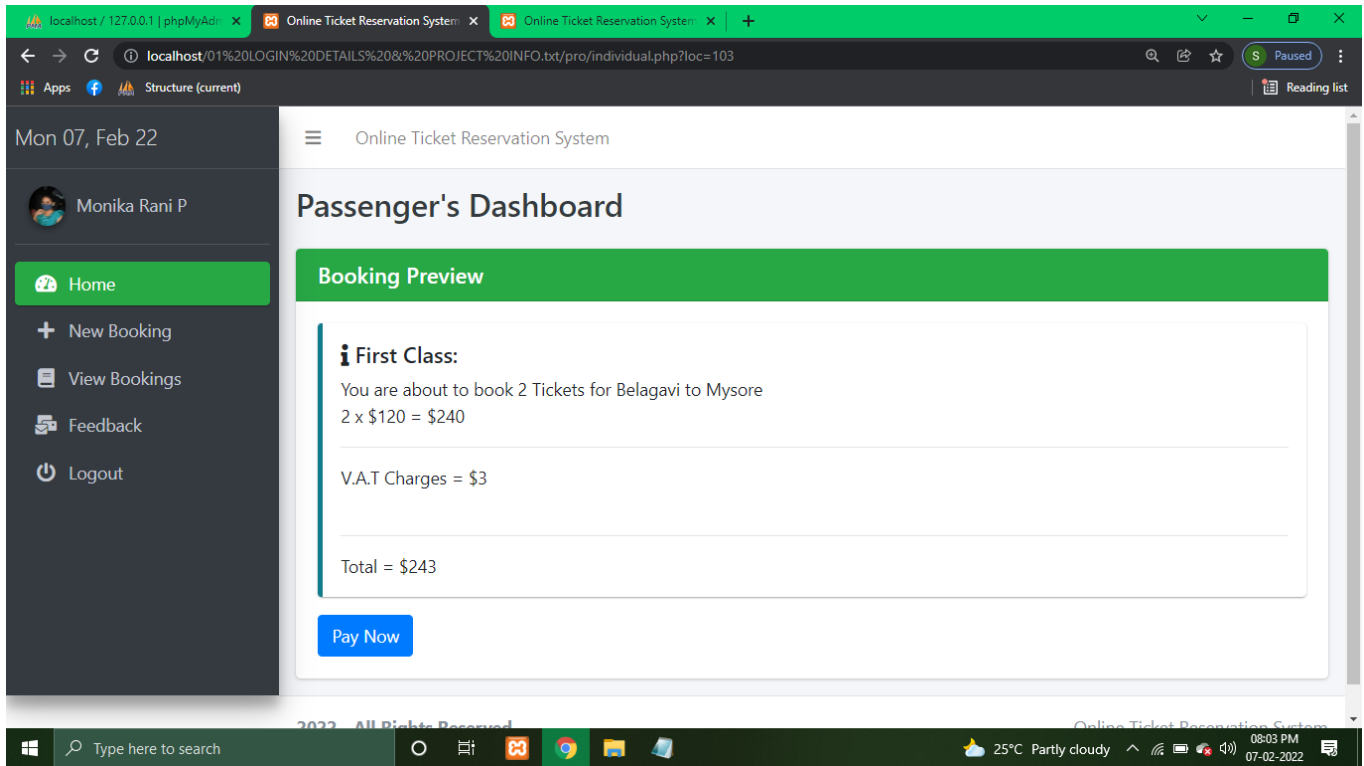
Showing 1 to 2 of 2 entries

The screenshot shows a modal form titled 'Book For Belagavi to Mysore' overlaid on the dashboard. The form contains the following fields and options:

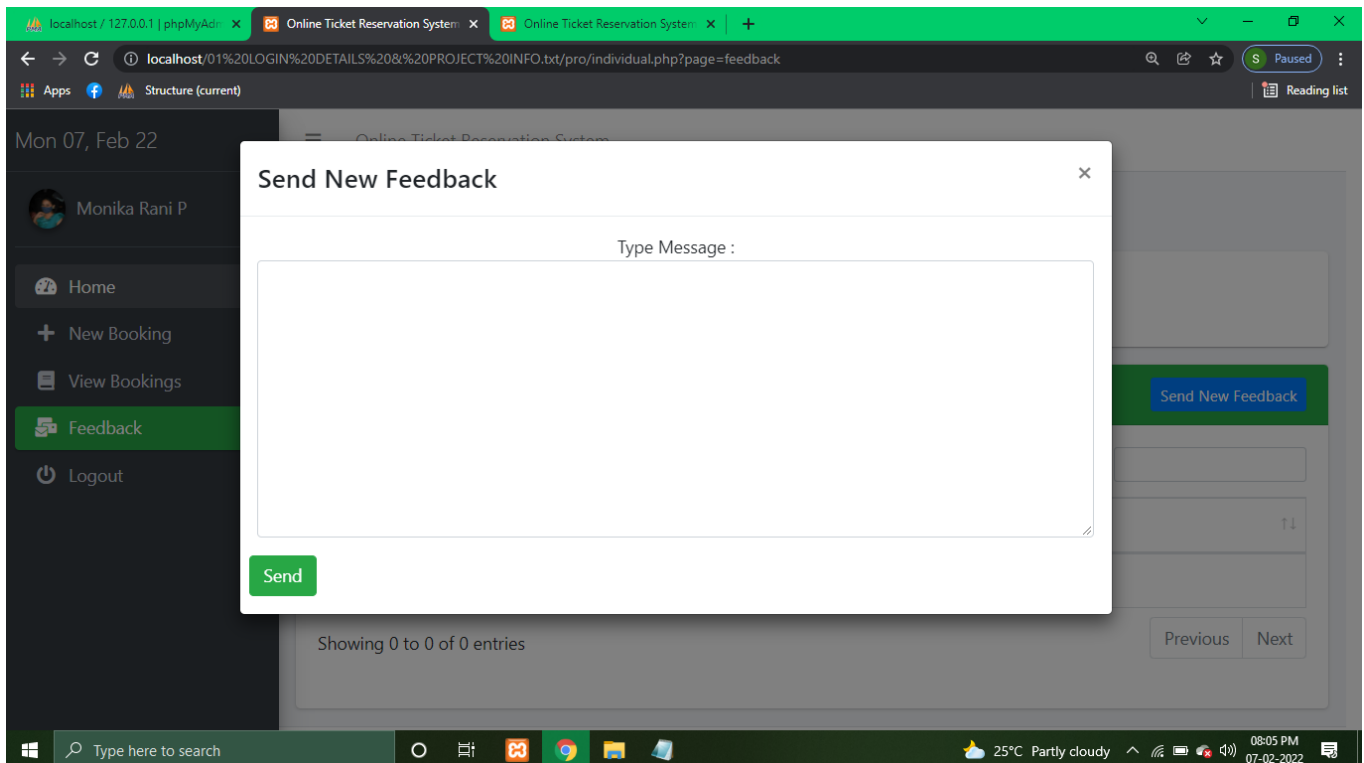
- Number of Tickets (If you are the only one, leave as it is):** A text input field with the value '2'.
- Class:** A dropdown menu with 'First Class (\$ 120)' selected.
- Proceed:** A green button to submit the booking.

The background dashboard is dimmed, showing the same table of routes as in the previous screenshot.

5.17 BOOKING PREVIEW:



5.18 SEND FEEDBACKS:



CHAPTER.6

SOURCE CODE:

Admin:

```
<?php

@session_start();

$file_access = true;

include '../conn.php';

include 'admin_session.php';

include '../constants.php';

if (@$_GET['page'] == 'print' && isset($_GET['code'])) {

    printClearance($_GET['code']);

    // echo "<script>window.location='admin.php'</script>";

}

if (@$_GET['page'] == 'report' && isset($_GET['id'])) {

    printReport($_GET['id']);

    // echo "<script>window.location='admin.php'</script>";

}

$fullname = "System Administrator";

?>

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="utf-8">
```

```
<meta name="viewport" content="width=device-width, initial-scale=1">
<meta http-equiv="x-ua-compatible" content="ie=edge">
<link rel="stylesheet" href="plugins/datatables-bs4/css/dataTables.bootstrap4.css">
```

```
<title><?php echo SITE_NAME, ' - ', ucwords($_SESSION['category']); ?> </title>
```

```
<!-- Font Awesome Icons -->
```

```
<link rel="stylesheet" href="plugins/fontawesome-free/css/all.min.css">
```

```
<!-- Theme style -->
```

```
<link rel="stylesheet" href="dist/css/adminlte.min.css">
```

```
</head>
```

```
<body class="hold-transition sidebar-mini">
```

```
<div class="wrapper">
```

```
<!-- Navbar -->
```

```
<nav class="main-header navbar navbar-expand navbar-white navbar-light">
```

```
<!-- Left navbar links -->
```

```
<ul class="navbar-nav">
```

```
<li class="nav-item">
```

```
<a class="nav-link" data-widget="pushmenu" href="#"><i class="fas fa-bars"></i></a>
```

```
</li>
```

```
<li class="nav-item">
```

```
<a class="nav-link" href="#"><?php echo SITE_NAME ?></a>
```

```
</li>
```

```
</ul>
```

```
<!-- Right navbar links -->
```

```
</nav>
```

```
<!-- /.navbar -->
```

```
<!-- Main Sidebar Container -->
```

```
<aside class="main-sidebar sidebar-dark-success elevation-4">
```

```
<!-- Brand Logo -->
```

```
<a href="admin.php" class="brand-link">
```

```
<span class="brand-text font-weight-light"><?php echo date("D d, M y"); ?></span>
```

```
</a>
```

```
<!-- Sidebar -->
```

```
<div class="sidebar">
```

```
<!-- Sidebar user panel (optional) -->
```

```
<div class="user-panel mt-3 pb-3 mb-3 d-flex">
```

```
<div class="image">
```

```

```

```
</div>
```

```
<div class="info">
```

```
<a href="#" class="d-block">Admin</a>
```

```
</div>
```

```
</div>
```

```
<!-- Sidebar Menu -->
```

```

<nav class="mt-2">

    <ul class="nav nav-pills nav-sidebar flex-column" data-widget="treeview" role="menu"
        data-accordion="false">

        <!-- Add icons to the links using the .nav-icon class
            with font-awesome or any other icon font library -->

        <li class="nav-item">

            <a href="admin.php" class="nav-link <?php if (!isset($_GET['page'])) echo 'active'; ?>">

                <i class="nav-icon fas fa-tachometer-alt"></i>

                <p>
                    Home
                </p>
            </a>

        </li>

        <li class="nav-item">

            <a href="admin.php?page=users" class="nav-link
                <?php
                    echo (@$_GET['page'] == 'users') ? 'active' : '';
                ?>
            ">

                <i class="nav-icon fas fa-users"></i>

                <p>
                    Users
                </p>
            </a>

        </li>
    </ul>

```

```

<li class="nav-item">

    <a href="admin.php?page=dynamic" class="nav-link

    <?php

    echo (@$_GET['page'] == 'dynamic') ? 'active' : ";

    ?>

    ">

    <i class="nav-icon fas fa-calendar-day"></i>

    <p>

        Schedules

    </p>

</a>

</li>

<li class="nav-item">

    <a href="admin.php?page=route" class="nav-link    <?php

        echo (@$_GET['page'] == 'route') ? 'active' : ";

        ?>">

    <i class="nav-icon fas fa-route"></i>

    <p>

        Routes

    </p>

</a>

</li>

</li>

<li class="nav-item">

    <a href="admin.php?page=train" class="nav-link    <?php

        echo (@$_GET['page'] == 'train') ? 'active' : ";

```

```

        ?>">

        <i class="nav-icon fas fa-train"></i>

        <p>

            Trains

        </p>

    </a>

</li>

<li class="nav-item">

    <a href="admin.php?page=report" class="nav-link    <?php

        echo (@$_GET['page'] == 'report') ? 'active' : ";

        ?>">

        <i class="nav-icon fas fa-file-pdf"></i>

        <p>

            Report

        </p>

    </a>

</li>

<li class="nav-item">

    <a href="admin.php?page=payment" class="nav-link    <?php

        echo (@$_GET['page'] == 'payment') ? 'active' : ";

        ?>">

        <i class="nav-icon fas fa-dollar-sign"></i>

        <p>

            Payments

        </p>

```


<li class="nav-item">

<a href="admin.php?page=feedback" class="nav-link" <?php

echo (@\$_GET['page'] == 'feedback') ? 'active' : '';

?>">

<i class="nav-icon fas fa-mail-bulk"></i>

<p>

Feedbacks

</p>

<li class="nav-item">

<a href="admin.php?page=search" class="nav-link" <?php

echo (@\$_GET['page'] == 'search') ? 'active' : '';

?>">

<i class="nav-icon fas fa-search"></i>

<p>

Search

</p>

<li class="nav-item">

Admin sign in:

```
alert("Ensure you fill the form properly.");

</script>

<?php

    } else {

        //Check for login

        $password = md5($password);

        $check = $conn->prepare("SELECT * FROM users WHERE email = ? AND password = ?");

        $check->bind_param("ss", $email, $password);

        if (!$check->execute()) die("Form Filled With Error");

        $res = $check->get_result();

        $no_rows = $res->num_rows;

        if ($no_rows == 1) {

            $row = $res->fetch_assoc();

            $id = $row['id'];

            session_regenerate_id(true);

            $_SESSION['category'] = "super";

            $_SESSION['admin'] = $id;

        }

    }

    ?>

<script>

alert("Access Granted!");

window.location = "admin.php";

</script>

<?php

    } else { ?>
```

```
<script>
alert("Access Denied.");
</script>

<?php
    }
}
}
?>

<div class="signup-page">
    <div class="form">
        <h2>Admin Sign In</h2>
        <br>
        <form class="login-form" method="post" role="form" id="signup-form" autocomplete="off">
            <!-- json response will be here -->
            <div id="errorDiv"></div>
            <!-- json response will be here -->

            <div class="col-md-12">
                <div class="form-group">
                    <label>Email Address</label>
                    <input type="text" required name="email">
                </div>
            </div>

            <div class="col-md-12">
                <div class="form-group">
                    <label>Password</label>
```

```
<input type="password" name="password" id="password">
```

```
<span class="help-block" id="error"></span>
```

```
</div>
```

```
</div>
```

```
<div class="col-md-12">
```

```
<div class="form-group">
```

```
<button type="submit" id="btn-signup">
```

```
    SIGN IN
```

```
</button>
```

```
</div>
```

```
</div>
```

```
<p class="message">
```

```
<a href="#">.</a><br>
```

```
</p>
```

```
</form>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
<script src="assets/js/jquery-1.12.4-jquery.min.js"></script>
```

```
</body>
```

```
</html>
```

Train:

```
<?php
```

```
if (!isset($file_access)) die("Direct File Access Denied");
```

```
$source = 'train';
```

```
$me = "?page=$source";
```

```
?>
```

```
<div class="content">
```

```
<!-- Main content -->
```

```
<section class="content">
```

```
<div class="container-fluid">
```

```
<div class="row">
```

```
<div class="col-sm-12">
```

```
<div class="card card-success">
```

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

```
<h3 class="card-title">
```

```
All Trains</h3>
```

```
<div class='float-right'>
```

```
<button type="button" class="btn btn-info btn-sm" data-toggle="modal"
```

```
data-target="#add">
```

```
Add New Train &#128645;
```

```
</button></div>
```

```
</div>
```

```
<div class="card-body">
```

```
<table id="example1" style="align-items: stretch;"
```

```
class="table table-hover w-100 table-bordered table-striped"><?php //
```

?>">

```

<thead>

<tr>

<th>#</th>

<th>Train Name</th>

<th>First Class Seat</th>

<th>Second Class Seat</th>

<th style="width: 30%;">Action</th>

</tr>

</thead>

<tbody>

<?php
$row = $conn->query("SELECT * FROM train");
if ($row->num_rows < 1) echo "No Records Yet";
$sn = 0;
while ($fetch = $row->fetch_assoc()) {
    $id = $fetch['id'];
    ?>

<tr>

<td><?php echo ++$sn; ?></td>

<td><?php echo $fullname = $fetch['name']; ?></td>

<td><?php echo $fetch['first_seat']; ?></td>

<td><?php echo $fetch['second_seat']; ?></td>

<td>

<form method="POST">

<button type="button" class="btn btn-primary" data-toggle="modal"

```

data-target="#edit<?php echo \$id ?>">

Edit

</button> -

<input type="hidden" class="form-control" name="del_train"

value="<?php echo \$id ?>" required id="">

<button type="submit"

onclick="return confirm('Are you sure about this?')"

class="btn btn-danger">

Delete

</button>

</form>

</td>

</tr>

<div class="modal fade" id="edit<?php echo \$id ?>">

<div class="modal-dialog modal-lg">

<div class="modal-content">

<div class="modal-header">

<h4 class="modal-title">Editing <?php echo \$fullname;

?></h4>

<button type="button" class="close" data-dismiss="modal"

aria-label="Close">

×

</button>


```

</div>

<div class="modal-body">

    <form action="" method="post">

        <input type="hidden" class="form-control" name="id"
            value="<?php echo $id ?>" required id="">

        <p>Train Name : <input type="text" class="form-control"
            name="name" value="<?php echo $fetch['name'] ?>"
            required minlength="3" id=""></p>

        <p>First Class Capacity : <input type="number" min='0'
            class="form-control"
            value="<?php echo $fetch['first_seat'] ?>"
            name="first_seat" required id="">

        </p>

        <p> Class Capacity : <input type="number" min='0'
            class="form-control"
            value="<?php echo $fetch['second_seat'] ?>"
            name="second_seat" required id="">

        </p>

        <p>

            <input class="btn btn-info" type="submit" value="Edit Train"
                name='edit'>

        </p>

    </form>

    <div class="modal-footer justify-content-between">

        <button type="button" class="btn btn-default"
            data-dismiss="modal">Close</button>

```

</div>

</div>

<!-- /.modal-content -->

</div>

<!-- /.modal-dialog -->

</div>

<!-- /.modal -->

<?php

}

?>

</tbody>

</table>

</div>

</div>

</div>

</div>

</div>

</div>

</div>

</div>

</div>

</section>

</div>

<div class="modal fade" id="add">

```
<div class="modal-dialog modal-lg">

  <div class="modal-content" align="center">

    <div class="modal-header">

      <h4 class="modal-title">Add New Train &#128646;

    </h4>

    <button type="button" class="close" data-dismiss="modal" aria-label="Close">

      <span aria-hidden="true">&times;</span>

    </button>

  </div>

  <div class="modal-body">

    <form action="" method="post">

      <table class="table table-bordered">

        <tr>

          <th>Train Name</th>

          <td><input type="text" class="form-control" name="name" required minlength="3"
id=""></td>

        </tr>

        <tr>

          <th>First Class Capacity</th>

          <td><input type="number" min='0' class="form-control" name="first_seat" required
id=""></td>

        </tr>

        <tr>

          <th>Second Class Capacity</th>

          <td><input type="number" min='0' class="form-control" name="second_seat" required
id="">

        </td>

      </tr>

    </table>

  </div>

</form>

</div>
```

```
</tr>
```

```
<tr>
```

```
<td colspan="2">
```

```
<input class="btn btn-info" type="submit" value="Add Train" name='submit'>
```

```
</td>
```

```
</tr>
```

```
</table>
```

```
</form>
```

```
</div>
```

```
</div>
```

```
<!-- /.modal-content -->
```

```
</div>
```

```
<!-- /.modal-dialog -->
```

```
</div>
```

```
<?php
```

```
if (isset($_POST['submit'])) {
```

```
    $name = $_POST['name'];
```

```
    $first_seat = $_POST['first_seat'];
```

```
    $second_seat = $_POST['second_seat'];
```

```
    if (!isset($name, $first_seat, $second_seat)) {
```

```
    alert("Fill Form Properly!");
} else {
    $conn = connect();

    //Check if train exists

    $check = $conn->query("SELECT * FROM train WHERE name = '$name' ")->num_rows;

    if ($check) {
        alert("Train exists");
    } else {
        $ins = $conn->prepare("INSERT INTO train (name, first_seat, second_seat) VALUES (?, ?, ?)");
        $ins->bind_param("sss", $name, $first_seat, $second_seat);
        $ins->execute();
        alert("Train Added!");
        load($_SERVER['PHP_SELF'] . "$me");
    }
}
}
```

```
if (isset($_POST['edit'])) {
    $name = $_POST['name'];
    $first_seat = $_POST['first_seat'];
    $second_seat = $_POST['second_seat'];
    $id = $_POST['id'];

    if (!isset($name, $first_seat, $second_seat)) {
        alert("Fill Form Properly!");
    } else {
        $conn = connect();

        //Check if train exists
```

```

$check = $conn->query("SELECT * FROM train WHERE name = '$name' ")->num_rows;

if ($check == 2) {
    alert("Train name exists");
} else {
    $ins = $conn->prepare("UPDATE train SET name = ?, first_seat = ?, second_seat = ? WHERE id =
?");

    $ins->bind_param("sssi", $name, $first_seat, $second_seat, $id);

    $ins->execute();

    alert("Train Modified!");

    load($_SERVER['PHP_SELF'] . "$me");
}
}
}

if (isset($_POST['del_train'])) {
    $con = connect();

    $conn = $con->query("DELETE FROM train WHERE id = " . $_POST['del_train'] . "");

    if ($con->affected_rows < 1) {
        alert("Train Could Not Be Deleted. This Train Has Been Tied To Another Data!");

        load($_SERVER['PHP_SELF'] . "$me");
    } else {
        alert("Train Deleted!");

        load($_SERVER['PHP_SELF'] . "$me");
    }
}

?>

```

REFERENCES

Website

- Dynamicdrive.com
- Freecsstemplates.com

Books

- Microsoft Official Custom Workshop
- ASP.NET The Complete Reference (Herbert Schild)
- Database Management System, Rameez Elmasri & Shamkant Navathe.

RESULTS

Online railway ticket reservation system was successfully designed and developed as per the specifications. It was extensively tested using a database which contains data similar to what can be expected in an actual database. The system was found to work satisfactorily without any errors under all conditions.