

## BUS MANAGEMENT:

### **ABSTRACT**

The Student Bus Management project is a website aimed at students and college administration to maintain college bus transport facility. Handling this task manually can be inefficient and disorganized. To overcome this disadvantage we have come up with this project. The system takes information of students, buses, drivers, routes etc. as input source and attempts to maintain the bus services. It allows flexibility during processes like adding, deleting and updating of records for the admin. Students can view records of drivers, buses and their routes. They can also give suggestions and voice out their complaints regarding the transportation system.

This project sought to present a website intended to handle the college bus transport system easily. The outcome of this project will be a simple interface that can be easily used to manage college bus transportation without much training.

# CONTENTS

S.No	Topics	Page. No
	<b>List of Figures</b>	Vi
	<b>Abbreviations</b>	X
	<b>Abstract</b>	Iv
1	<b>Introduction</b>	
	1.1 Motivation	1
	1.2 Objective of the Project	1
	1.3 Problem Statement	1
2	<b>Existing System</b>	
	2.1 Literature survey	2
3	<b>Proposed Methodology</b>	
	3.1 System Specifications	4
	3.2 System Design	5
	3.3 Proposed Work	5
4	<b>Implementation and Results</b>	7
5	<b>Conclusion and Future Scope</b>	47
	<b>Bibliography</b>	48

## **List of Figures**

<b>Figure No</b>	<b>Name of the Figure</b>	<b>Page No</b>
2.1	CBIT transport page	2
2.2	CBIT transport contact details	3
2.1	CBIT Bus route detail	3
3.2	System Design	5
4.1	Database tables	7
4.2	Admin database schema	7
4.3	Driver database schema	8
4.4	Bus database schema	8
4.5	Route database schema	8
4.6	Complaint/Suggestion box database schema	8
4.7	Code to register 1	8
4.8	Code to register 2	9
4.9	Code to register 3	9
4.10	Code to login 1	10
4.11	Code to login 2	10
4.12	Code to login 3	11
4.13	Code to stay logged in	11
4.14	Logout code	11
4.15	Database connection to the website	12
4.16	Admin module front page code 1	12
4.17	Admin module front page code 2	12
4.18	Admin module front page code 3	13
4.19	Admin module front page code 4	13
4.20	Admin module front page code 5	13
4.21	Admin module sidebar code 1	14
4.22	Admin module sidebar code 2	14
4.23	Code to show student data 1	14

4.24	Code to show student data 2	14
4.25	Code to delete student data 1	15
4.26	Code to delete student data 2	15
4.27	Code to edit student data 1	15
4.28	Code to edit student data 2	15
4.29	Code to edit student data 3	16
4.30	Code to edit student data 4	16
4.31	Code to edit student data 5	16
4.32	Code to show bus data	17
4.33	Code to add bus data 1	17
4.34	Code to add bus data 2	17
4.35	Code to delete bus data 1	18
4.36	Code to delete bus data 2	18
4.37	Code to edit bus data 1	18
4.38	Code to edit bus data 2	18
4.39	Code to edit bus data 3	19
4.40	Code to show driver data 1	19
4.41	Code to show driver data 2	19
4.42	Code to add driver data 1	20
4.43	Code to add driver data 2	20
4.44	Code to delete driver data 1	20
4.45	Code to delete driver data 2	20
4.46	Code to edit driver data 1	21
4.47	Code to edit driver data 2	21
4.48	Code to edit driver data 3	21
4.49	Code to edit driver data 4	21
4.50	Code to show route data 1	22
4.51	Code to show route data 2	22
4.52	Code to add route data	23

4.53	Code to delete route data 1	23
4.54	Code to delete route data 2	23
4.55	Code to edit route data 1	24
4.56	Code to edit route data 2	24
4.57	Code to edit route data 3	24
4.58	Code to show complaints and suggestions 1	25
4.59	Code to show complaints and suggestions 2	25
4.60	Student module front page 1	26
4.61	Student module front page 2	26
4.62	Student module front page 3	26
4.63	Student module front page 4	26
4.64	Student module sidebar code	27
4.65	Code to show student's personal details 1	27
4.66	Code to show student's personal details 2	28
4.67	Code to show student's fee details	28
4.68	Code to show bus details	29
4.69	Code to show driver details	29
4.70	Code to show route details	30
4.71	Code to show style complaint/suggestion webpage	30
4.72	Code to send a complaint/suggestion	30
4.73	Code to save complaint/suggestion into database	31
4.74	Login page	31
4.75	Front page	31
4.76	Register page	32
4.77	Student front page	32
4.78	Student update page	33
4.79	Student update page 1	33
4.80	Student delete page	33

4.81	Student fee front page	34
4.82	Student fee update page	34
4.83	Student fee update page 1	35
4.84	Student fee delete page 1	35
4.85	Bus front page	36
4.86	Bus add page	36
4.87	Bus update page	36
4.88	Bus update page 1	37
4.89	Bus delete page	37
4.90	Driver front page	38
4.91	Driver add page	38
4.92	Driver update page	38
4.93	Driver update page 1	39
4.94	Driver delete page	39
4.95	Route front page	40
4.96	Route add page	40
4.97	Route update page 1	40
4.98	Route update page	41
4.99	Route delete page	41
4.100	Complaint/Suggestion box page	42
4.101	Homepage	42
4.102	Student page	43
4.103	Student fee page	43
4.104	Bus page	44
4.105	Driver page	44
4.106	Route page	45
4.107	Route pdf	45
4.108	Complaint/Suggestion page	46

## **List of Abbreviations**

Acronym	Abbreviation
HTML	Hypertext Markup Language
CSS	Cascading Style Sheets
PHP	Hypertext Preprocessor
XAMMP	Cross-platform, Apache, MYSQL, PHP and Perl
MySQL	My Structured Query Language
RAM	Random Access Memory

# **1. INTRODUCTION**

## **1.1 MOTIVATION**

The bus transport systems in colleges are managed manually. Hence they come across various limitations like lack of security of data, time consuming, low efficiency, more man power and lots of paperwork.

In order to overcome these limitations we have come up with this project. The data stored will be consistent and accurate. The time consumed to complete tasks like adding or deleting or updating a record will be a lot less than when done manually. The efficiency increases. Man power required to maintain the system will be less and paper work is reduced.

## **1.2 OBJECTIVE OF THE PROJECT**

To develop a software such that everybody working in college bus management can handle it easily. Admin can store, retrieve and manipulate data easily. Keeping these major target segments in focus, the system was developed. The objective of developing a tool for technical as well as non-technical user hence got achieved.

Through the development of this project, we also aimed at learning some technical skills such as PHP, database connection to webpages created, CSS frontend development and backend development which would definitely be helpful to us in the future.

## **1.3 PROBLEM STATEMENT**

The world is getting digitalized in all aspects. This allows us to work in an efficient manner. Coming to the bus transport management of students in colleges, they are done manually. This brings up a lot of complications while maintaining this facility. Hence to overcome the complications, this project has been introduced.

## 2. EXISTING SYSTEM

### 2.1 LITERATURE SURVEY

In the existing system colleges have to manually maintain information regarding bus transport facility. Information relating to student details, driver details, bus details and their routes have to be maintained separately. Colleges do use website for their bus transportation system but for limited tasks. They just provide information on the bus routes and transport contact details. Information of students availing the bus transport, driver, bus and their route details are all handled manually with much manpower and paper work.

CBIT has a transport webpage as shown in figure 2.1. Figure 2.2 and 2.3 shows the cbit transport contact details and bus routes.

### Transport

[Fee Payment procedure](#) | [Bus Routes](#) | [Circulars](#) | [Contact Details](#) | [Application Form](#)

#### INSTRUCTIONS TO THE PARENTS

1. College reserves the right to modify, merge or cancel any proposed route(s) without assigning any reason.
2. The seats will be allotted on the first come first serve basis for every academic year after receipt of Transport Fee.
3. New routes/buses will be introduced subjected to the number of students and availability of buses. Decision of the Institute will be final in this regard.
4. Request for the extension or diversion of route(s) and additional stops will not be considered for the existing buses/routes.
5. Transport fee once paid is non-refundable.
6. Parents are requested to lodge their suggestions / requests / complaints, if any in writing, with the I/c, Students Transport Centre or Principal only.

#### INSTRUCTIONS TO THE STUDENTS

1. Must always carry the College and Transport Identity Cards, and show the same to the faculty in charges /driver on demand.
2. Ragging and indecent behaviour is strictly prohibited.
3. To be in their respective bus stops at the scheduled time, chasing or forcing the driver to stop the bus is not permitted.
4. Activating loudspeakers of cell phones / I pods , clapping, whistling, shouting etc. are prohibited.
5. Should not keep hands / any other parts of the body out of windows.
6. Boys and girls must occupy separate seats.

Fig. 2.1: CBIT transport page



### Transport Contact details

1. Dr. R.P. Chowdary  
Associate Professor of Mechanical Department  
In charge, Students Transport centre.  
Chaitanya Bharathi Institute of Technology  
Hyderabad – 500 075.  
Phone: 9177220635 (4.30 to 6.30PM)  
e-mail: cbtstudenttransport@gmail.com
2. S. Sateesh Reddy (Transport Supervisor)  
Ph: 99081 76475



Fig. 2.2: CBIT transport contact details

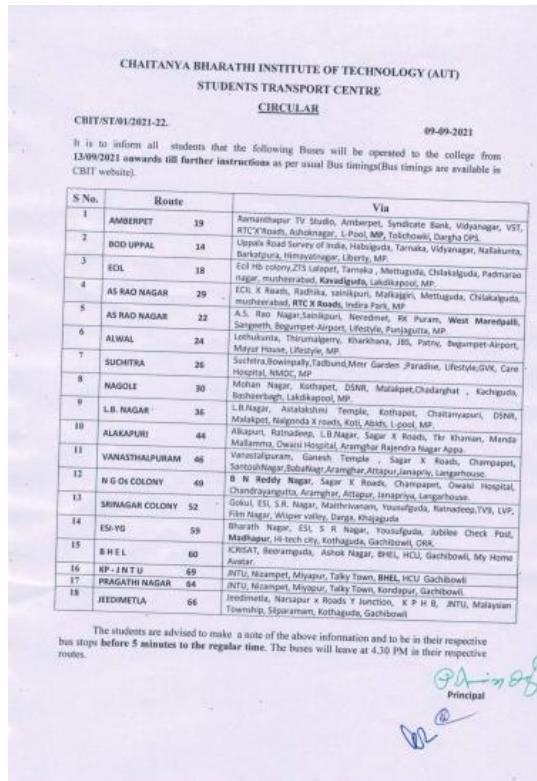


Fig. 2.3: CBIT bus route details

### **3. PROPOSED METHODOLOGY**

#### **3.1 SYSTEM SPECIFICATIONS**

##### **Software Requirements**

- Operating system: Windows7/8/10
- Tools: Visual studio code
- Front end:

HTML: It is the code that is used to structure a web page and its contents.

CSS: It is a style sheet used for describing the presentation of a document written in a markup language such as HTML. It creates an attractive layout.

Bootstrap: It focuses on simplifying the development of informative web pages.

It helps in creating a responsive design mobile friendly site.

JavaScript: JavaScript is a text-based programming language used both on the client-side and server-side that allows us to make web pages interactive. It creates a dynamic and interactive experience for the user.

- Back end:

PHP: It is a software to create dynamically generated web pages in HTML. It makes it easier to add functionality to web pages without needing to call external files for data.

XAMPP Server: It is a platform that furnishes a suitable environment to test and verify the working of projects.

MySQL Server: The MySQL server provides a database management system with querying and connectivity capabilities

##### **Hardware Requirements**

- Processor: Intel Pentium

- Processor Speed: 1.5 GHz
- RAM: 2GB or above
- Hard Disk: 50 GB or above

### 3.2 SYSTEM DESIGN

The block diagram of the proposed system i.e student bus management system is shown in the below figure 3.1. It gives a brief overview of our project. It shows the two modules and their consisting records.

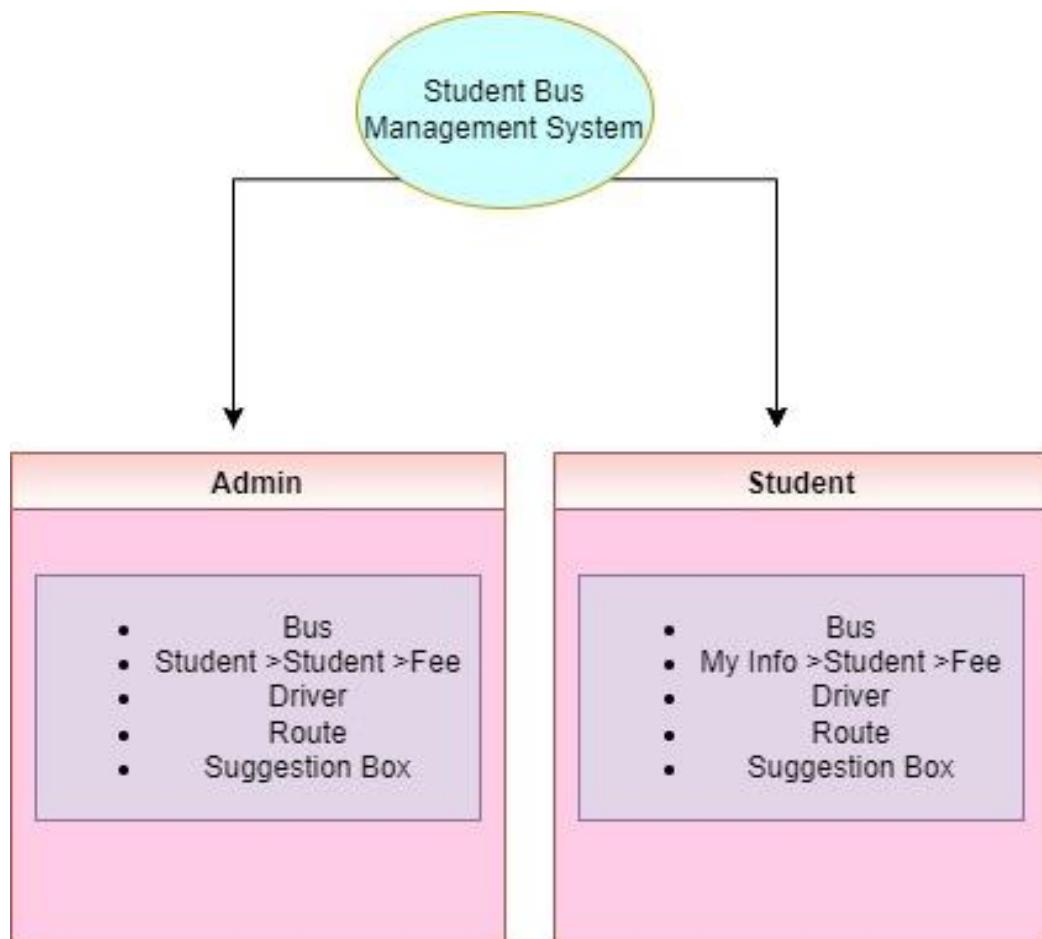


Fig. 3.1: Block diagram of the proposed system

### 3.3 PROPOSED WORK

We've introduced two modules in this website namely:

- Admin Module
- Student Module

The admin module consists of different records where we can view them and also perform tasks like add, delete, and update on tuples. Basically here the person has the ability to manipulate information in the database and manage the website.

The admin module consists of the following records:

- Bus records: route number, route name, driver id, seating capacity, occupancy
- Student records: 1)Student: roll number, name, gender, branch, sem, mobile number, email, bus number, seat number, pickup point, guardian name, guardian number
  - 2)Fee: roll number, name, mobile number, route number, fee, amount paid, pay method
- Driver records: driver id, driver name, mobile number, email, Aadhaar number, bus number, employ date, salary
- Route records: route number, route name, enroute stages, start time, departure
- Complaint/Suggestion box: rollnumber, type, subject

The student module consists of different records where we can just view their information. Only in the complaint/suggestion box we can send information regarding the bus facilities.

The student module consists of the following records:

- Bus records: route number, route name, driver id, seating capacity, occupancy
- Student records: 1)Student: roll number, name, gender, branch, sem, mobile number, email, bus number, seat number, pickup point, guardian name, guardian number
  - 2)Fee: roll number, name, mobile number, route number, fee, amount paid, pay method
- Driver records: driver id, driver name, bus number
- Route records: route number, route name, enroute stages, start time, departure
- Complaint/Suggestion box: roll number, type, subject

## 4. IMPLEMENTATION AND RESULTS

### Database:

Figure 4.1 shows the overall database. Figures 4.2, 4.3, 4.4, 4.5, 4.6 show the database schema of admin, driver, bus, route and complaint/suggestion box respectfully.

	Table	Action
<input type="checkbox"/>	admin	
<input type="checkbox"/>	bus	
<input type="checkbox"/>	csbox	
<input type="checkbox"/>	driver	
<input type="checkbox"/>	route	
5 tables		

Fig. 4.1: Database Tables

busscheduledb admin	
#	id : int(10)
#	rnum : bigint(10)
②	name : varchar(255)
②	gender : varchar(10)
②	program : varchar(255)
②	branch : varchar(255)
②	sem : varchar(255)
#	mobilenum : bigint(10)
#	route_number : int(10)
#	seat_number : int(20)
②	pickup_point : varchar(100)
②	pname : varchar(100)
#	pnum : bigint(10)
②	fee : varchar(50)
#	amount_paid : bigint(20)
②	pay_method : varchar(50)
②	user : varchar(255)
②	password : varchar(255)
②	confirmpassword : varchar(255)
②	status : varchar(20)
②	file_name : varchar(255)

Fig. 4.2: Admin Database Schema

busscheduledb driver	
DRIVER_ID	int(20)
DRIVER_NAME	varchar(45)
MOBILE_NUMBER	int(10)
EMAIL	varchar(45)
AADHAAR_NUMBER	int(20)
BUS_NUM	int(10)
EMPLOY_DATE	date
SALARY	int(20)

Fig. 4.3: Driver Database Schema

busscheduledb bus	
ROUTE_NUMBER	int(5)
ROUTE_NAME	varchar(50)
DRIVER_ID	int(10)
SEATING_CAPACITY	int(10)
OCCUPANCY	int(25)
REG_NUMBER	varchar(40)

Fig. 4.4: Bus Database Schema

busscheduledb route	
ROUTE_NUMBER	int(5)
ROUTE_NAME	varchar(45)
ENROUTE_STAGES	varchar(1000)
START_TIME	time
DEPARTURE	time

Fig. 4.5: Route Database Schema

busscheduledb csbox	
id	int(11)
Rollnumber	bigint(20)
Type	varchar(50)
Subject	text

Fig. 4.6: Complaint/Suggestion Box Database Schema

## Code Snippet

**Register:** The code shown in figures 4.7, 4.8 and 4.9 helps in creating a page where students availing bus transport can be registered and their details stored in admin db.

```

<body class="bg-dark" style="background-image: linear-gradient(blue,white);>
<div class="container">
<div class="card card-register mx-auto mt-5">
<div class="card-header" style="width:638px">

<div class="card-body">
<form method="POST" action="success1.php">
<div class="form-group">
<div class="form-row">
<div class="col-md-6">
<div class="form-label-group">
<input type="text" id="rnum" class="form-control" placeholder="rnum" name="rnum" required="required" autofocus="autofocus">
<label for="rnum">Rollnumber</label>
</div>
</div>
<div class="col-md-6">
<div class="form-label-group">
<input type="text" id="name" class="form-control" placeholder="name" name="name" required="required">
<label for="name">Name</label>
</div>
</div>
</div>
</div>
<div class="form-group">
<div class="form-row">
<div class="col-md-6">
<div class="form-label-group">
<input type="program" id="program" class="form-control" placeholder="program" name="program" required="required">
<label for="program">Program</label>
</div>
</div>
<div class="col-md-6">
<div class="form-label-group">
<input type="branch" id="branch" class="form-control" placeholder="branch" name="branch" required="required">
</div>
</div>
</div>
</div>

```

Fig. 4.7: code to register 1

```

0   |     </div>
1   |     </div>
2   |   </div>
3   |   <div class="form-group">
4   |     <div class="form-label-group">
5   |       <input type="email" id="inputEmail" class="form-control" placeholder="Email address" name="email" required="required">
6   |       <label for="inputEmail">Email address</label>
7   |     </div>
8   |   </div>
9   |   <div class="form-group">
0   |     <div class="form-row">
1   |       <div class="col-md-6">
2   |         <div class="form-label-group">
3   |           <input type="password" id="inputPassword" class="form-control" placeholder="Password" name="password" pattern="(?=.*\d)(?=.*[a-z])(?=.*[A-Z]).{8,}" title="Must contain at least one number, one uppercase,lowercase letter and at least 8 or more characters" required>
4   |           <label for="inputPassword">Password</label>
5   |         </div>
6   |       </div>
7   |       <div class="col-md-6">
8   |         <div class="form-label-group">
9   |           <input type="password" id="confirmPassword" class="form-control" placeholder="Confirm password" name="confirm" required="required">
0   |           <label for="confirmPassword">Confirm password</label>
1   |         </div>
2   |       </div>
3   |     </div>
4   |     <button type="submit" class="btn btn-primary btn-block" name="register" style="background-image: linear-gradient(, #2144);">Register</button>
5   |   </form>
6   |   <div class="text-center">
7   |     <a class="d-block small mt-3" href="register.php" Login Page</a>
8   |   </div>
9   | </div>
0   | </div>
1   | </div>
2   | </div>
3   | </div>
4   | </div>

```

Fig. 4.8: code to register 2

```

<?php
$servername = "127.0.0.1";
$username = "root";
$password = "";
$dbname = "busscheduledb";
$rnnum=$_POST['rnnum'];
$name=$_POST['name'];
$gender=$_POST['gender'];
$program=$_POST['program'];
$branch=$_POST['branch'];
$sem=$_POST['sem'];
$mobilenum=$_POST['mobilenum'];
$pickup_point=$_POST['pickup_point'];
$pname=$_POST['pname'];
$pnum=$_POST['pnum'];
$user=$_POST['email'];
$password=$_POST['password'];
$confirm=$_POST['confirm'];

try {
    $conn = new PDO("mysql:host=$servername;dbname=$dbname", $username, $Password);
    $conn->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
    $SQL="INSERT INTO admin (rnnum,name,gender,program,branch,sem,mobilenum,pickup_point,pname,pnum,user,password,confirmppassword,status)VALUES('$rn
    $conn->exec($SQL);
}
catch(PDOException $e)
{
}
$conn = null;

include "login.php";
?>

```

Fig. 4.9: code to register 3

**Login:** The code shown in figures 4.10, 4.11, 4.12 and 4.13 helps students and admin to log into the student bus transport management website.

```

login.php
1  <?php require('session.php');?>
2  <!DOCTYPE html>
3  <html lang="en">
4
5  <head>
6
7      <meta charset="utf-8">
8      <meta http-equiv="X-UA-Compatible" content="IE=edge">
9      <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
10     <meta name="description" content="">
11     <meta name="author" content="">
12
13     <title>SB Admin - Login</title>
14
15 </head>
16 <?php if(logged_in()){ ?>
17     <script type="text/javascript">
18         window.location = "index.php";
19     </script>
20 <?php
21 } ?>
22
23 <body class="bg-dark" style="background-image: linear-gradient(blue,white);>
24
25 <div class="container">
26     <div class="card card-login mx-auto mt-5">
27         <div class="card-header" style="width:410px">
28             
29         <div class="card-body" style="">
30             <form method="POST" action="success.php">
31                 <div class="form-group">
32                     <div class="form-label-group">
33                         <input type="email" id="inputEmail" class="form-control" placeholder="Email address" name="user" required="required" autofocus="autofocus">
34                         <label for="inputEmail">Email address</label>
35                     </div>
36                 </div>
37             <div class="form-group">

```

Fig. 4.10: code to login 1

```

<form method="POST" action="success.php">
<div class="form-group">
    <div class="form-label-group">
        <input type="email" id="inputEmail" class="form-control" placeholder="Email address" name="user" required="required" autofocus="autofocus">
        <label for="inputEmail">Email address</label>
    </div>
</div>
<div class="form-group">
    <div class="form-label-group">
        <input type="password" id="inputPassword" class="form-control" placeholder="Password" name="pass" required="required">
        <label for="inputPassword">Password</label>
    </div>
</div>
<div class="form-group">
    <div class="checkbox">
        <label>
            <input type="checkbox" value="remember-me">
            Remember Password
        </label>
    </div>
</div>
<button type="submit" class="btn btn-primary btn-block" style="background-image: linear-gradient(black,blue); id="login">Login</button>
</form>
<div class="text-center">
    <!--<a class="d-block small mt-3" href="register.php">Register an Account</a>-->
</div>
</div>
</div>

```

Fig. 4.11: code to login 2

```

success.php
1  <?php
2  require('connection.php');
3  session_start();
4
5
6  $user = trim($_POST['user']);
7  $upass = trim($_POST['pass']);
8  $h_pass = MD5($upass);
9  $sql = "SELECT * FROM `admin` WHERE `user` = '" . $user. "' AND `password` = '" . $h_pass . "' ";
10 $result = $db->query($sql);
11  if($result->num_rows > 0){
12      $found_user = mysqli_fetch_array($result);
13      $_SESSION['MEMBER_ID'] = $found_user['id'];
14      $_SESSION['position'] = $found_user['status'];
15      if ($_SESSION['position']=='ADMIN') {
16          ?> <script type="text/javascript">
17          |     window.location = "index.php"
18      </script>
19      <?php
20      }else{
21          ?> <script type="text/javascript">
22          |     window.location = "homepage.php"
23      </script>
24      <?php
25      }
26  }else{
27      ?> <script type="text/javascript">
28      alert("Username or Password Not Registered! Contact Your administrator.");
29      window.location = "login.php";
30  </script>
31  <?php
32  }
33
34 $db->close();
35 ?>

```

Fig. 4.12: code to login 3

```

session.php
1  <?php
2  session_start();
3  function logged_in(){
4      return isset($_SESSION['MEMBER_ID']);
5  }
6
7  function confirm_logged_in(){
8      if (!logged_in()) {?>
9          <script type="text/javascript">
10         |     window.location = 'login.php';
11     </script>
12
13
14
15  }
16 ?>

```

Fig. 4.13: code to stay logged in

**Logout:** The code shown in figure 4.14 allows us to log out of the website.

```

logout.php
1  <?php
2  require 'connection.php';
3  require 'session.php';
4
5  session_destroy();
6  ?>
7  <script type="text/javascript">
8      alert("Successfully logout!");
9      window.location = "login.php";
10 </script>
11

```

Fig. 4.14: logout code

**Database Connection:** Figure 4.15 tells us how to connect to the database

```
<?php
$connection = mysqli_connect('localhost', 'root', '') or
    die ('Unable to connect. Check your connection parameters.');
mysqli_select_db($connection, 'busschedule') or die(mysqli_error($connection));
?>
```

Fig. 4.15: database connection to the website

## ADMIN MODULE:

**Front page:** The front page of the admin module contains a sidebar, drop down box and footer consisting of transport contact details. Figures 4.16, 4.17, 4.18, 4.19 and 4.20 shows us the code using which the front page was created.

```
<?php
include 'connection.php';
include 'topnav.php';
?>

<!-- Breadcrumbs-->
<ol class="breadcrumb">
    <li class="breadcrumb-item" style="background-image: linear-gradient();">
        <a href="index.php"> 

## BUS SERVICES </h2> </a> </li> </ol> <head> <style> body {font-family: Arial, Helvetica, sans-serif;}


```

#myImg {  
border-radius: 5px;  
cursor: pointer;  
transition: 0.3s;  
}  
  
#myImg:hover {opacity: 0.7;}

/\* The Modal (background) \*/  
.modal {  
display: none; /\* Hidden by default \*/  
position: fixed; /\* Stay in place \*/  
z-index: 1; /\* Sit on top \*/  
padding-top: 100px; /\* Location of the box \*/  
left: 0;  
top: 0;  
width: 100%; /\* Full width \*/  
height: 100%; /\* Full height \*/  
overflow: auto; /\* Enable scroll if needed \*/  
background-color: #f0f0f0; /\* Fallback color \*/  
background-color: rgba(0,0,0,0.9); /\* Black w/ opacity \*/

Fig. 4.16: admin module front page code 1

```
margin: auto;
display: block;
width: 80%;
max-width: 700px;
}

/* Caption of Modal Image */
#caption {
margin: auto;
display: block;
width: 80%;
max-width: 700px;
text-align: center;
color: #ccc;
padding: 10px 0;
height: 150px;
}

/* Add Animation */
.modal-content, #caption {
-webkit-animation-name: zoom;
-webkit-animation-duration: 0.6s;
animation-name: zoom;
animation-duration: 0.6s;
}

@-webkit-keyframes zoom {
from {-webkit-transform:scale(0)}
to {-webkit-transform:scale(1)}}

@keyframes zoom {
from {transform:scale(0)}
to {transform:scale(1)}}

/* The Close Button */
.close {
```

Fig. 4.17: admin module front page code 2

```

.close {
  position: absolute;
  top: 15px;
  right: 35px;
  color: #f1f1f1;
  font-size: 40px;
  font-weight: bold;
  transition: 0.3s;
}

.close:hover,
.close:focus {
  color: #bbb;
  text-decoration: none;
  cursor: pointer;
}

/* 100% Image Width on Smaller Screens */
@media only screen and (max-width: 700px){
  .modal-content {
    width: 100%;
  }
}

</style>
</head>
</body>




<h2>Contact Details</h2>
<p>
1. Dr. R.P. Chowdary
<br>
Associate Professor of Mechanical Department.
<br>
Phone: 9177228635( 4.30 to 6.30PM)

```

Fig. 4.18: admin module front page code 3

```

Associate Professor of Mechanical Department.
<br>
Phone: 9177228635( 4.30 to 6.30PM)
<br>
e-mail: cbitsstudenttransport@gmail.com</p>
<p>
2. S. Sateesh Reddy (Transport Supervisor)
<br>
Ph: 99081 76475</p>

<!-- The Modal -->
<div id="myModal" class="modal">
  <span class="close">&times;</span>
  <img class="modal-content" id="img01">
  <div id="caption"></div>
</div>

<script>
// Get the modal
var modal = document.getElementById("myModal");

// Get the image and insert it inside the modal - use its "alt" text as a caption
var img = document.getElementById("myImg");
var modalImg = document.getElementById("img01");
var captionText = document.getElementById("caption");
img.onclick = function(){
  modal.style.display = "block";
  modalImg.src = this.src;
  captionText.innerHTML = this.alt;
}

// Get the <span> element that closes the modal
var span = document.getElementsByClassName("close")[0];

// When the user clicks on <span> (x), close the modal
span.onclick = function() {
  modal.style.display = "none";
}

```

Fig. 4.19: admin module front page code 4

```

topnav.php
1  <?php
2  require 'session.php';
3  confirm_logged_in();
4  | ?
5  <!DOCTYPE html>
6  <html lang="en">
7  | <head>
8  |   <meta charset="utf-8">
9  |   <meta http-equiv="X-UA-Compatible" content="IE=edge">
10 |   <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
11 |   <meta name="description" content="">
12 |   <meta name="author" content="">
13 |   <title>BUS</title>
14 |   <link href="vendor/bootstrap/css/bootstrap.min.css" rel="stylesheet">
15 |   <link href="vendor/fontawesome-free/css/all.min.css" rel="stylesheet" type="text/css">
16 |   <link href="vendor/datatables/dataTables.bootstrap4.css" rel="stylesheet">
17 |   <link href="css/sb-admin.css" rel="stylesheet">
18 </head>
19 <body id="page-top">
20   <nav class="navbar navbar-expand navbar-dark bg-dark static-top" style="background-image: linear-gradient(to left, #42275a, #734b6d);">
21     <a class="navbar-brand mr-1" href="index.php">BUS</a>
22     <button class="btn btn-link btn-sm text-white order-1 order-sm-0" id="sidebarToggle" href="#">
23       <i class="fas fa-bars"></i> </button>
24     <form class="d-none d-md-inline-block form-inline ml-auto mr-0 mr-md-3 my-2 my-md-0">
25       </form>
26     <ul class="navbar-nav ml-auto ml-md-0">
27       <li class="nav-item dropdown no-arrow">
28         <a class="nav-link dropdown-toggle" href="#" id="userDropdown" role="button" data-toggle="dropdown" aria-haspopup="true" aria-expanded="false">
29           <i class="fas fa-user-circle fa-fw"></i>
30         </a>
31         <div class="dropdown-menu dropdown-menu-right" aria-labelledby="userDropdown">
32           <a class="dropdown-item" href="register.php">Register</a>
33           <div class="dropdown-divider"></div>
34           <a class="dropdown-item" href="logout.php" data-target="#logoutModal">Logout</a></div>
35         </li>
36       </ul>

```

Fig. 4.20: admin module front page code 5

**Sidebar:** The side bar allows admin to reach the webpages of different records when clicked. Figures 4.21 and 4.22 shows the code used to create sidebar.

```

<li class="nav-item">
  <a class="nav-link" href="index.php">
    <i class="fas fa-fw fa-home"></i>
    <span>HOMEPAGE</span>
  </a>
</li>
<li class="nav-item dropdown">
  <a class="nav-link dropdown-toggle" href="#" id="pagesDropdown" role="button">
    <i class="fas fa-fw fa-user-circle"></i>
    <span>Student</span>
  </a>
  <div class="dropdown-menu" aria-labelledby="pagesDropdown">
    <a class="dropdown-item" href="student.php">Students</a>
    <a class="dropdown-item" href="studentfee.php">Fee Details</a>
  </div>
</li>
<li class="nav-item">
  <a class="nav-link" href="Bus.php">
    <i class="fas fa-fw fa-bus"></i>
    <span>Bus</span>
  </a>
</li>
<li class="nav-item">
  <a class="nav-link" href="driver.php">
    <i class="fas fa-fw fa-users"></i>
    <span>Driver</span>
  </a>
</li>
<li>
  <a class="nav-link" href="route.php">
    <i class="fas fa-fw fa-route"></i>
    <span>Route</span>
  </a>
</li>
<li class="nav-item">
  <a class="nav-link" href="csbox.php">
    <i class="fas fa-fw fa-comment-alt"></i>
  </a>
</li>

```

Fig. 4.21: admin module sidebar code 1

```

<li class="nav-item">
  <i class="fas fa-fw fa-bus"></i>
  <span>Bus</span>
</li>
<li class="nav-item">
  <a class="nav-link" href="driver.php">
    <i class="fas fa-fw fa-users"></i>
    <span>Driver</span>
  </a>
</li>
<li>
  <a class="nav-link" href="route.php">
    <i class="fas fa-fw fa-route"></i>
    <span>Route</span>
  </a>
</li>
<li class="nav-item">
  <a class="nav-link" href="csbox.php">
    <i class="fas fa-fw fa-comment-alt"></i>
    <span>Suggestion Box</span>
  </a>
</li>

```

```

<div id="content-wrapper">
  <div class="container-fluid">

```

Fig. 4.22: admin module sidebar code 2

**Student Records:** The student records in the admin module consist of the student details and allows to manipulate it.

- **Student:** Figures 4.23, 4.24 shows the code which helps in viewing student data.

```

<?php include 'connection.php'; <?
include 'topnav.php'; <?
  <div class="col-lg-12">
    <div>
      <h1>Student Records</h1>
      <div>
        <a href="register.php?action=add" style="background-image: linear-gradient(); type="button" class="btn btn-xs btn-info">Add</a>
        <a href="studenteditpopup.php?action=edit" style="background-image: linear-gradient(); type="button" class="btn btn-xs btn-warning">Edit</a>
        <a href="studentdelpopup.php?action=delete" style="background-image: linear-gradient(); type="button" class="btn btn-xs btn-danger">Delete</a>
      </div>
    </div>
    <div class="table-responsive">
      <table class="table table-bordered" id="datatable" width="100%" cellspacing="0">
        <thead>
          <tr>
            <th>Roll No.</th>
            <th>Name</th>
            <th>Gender</th>
            <th>Branch</th>
            <th>Sem</th>
            <th>Mobile No.</th>
            <th>Email</th>
            <th>Bus No.</th>
            <th>Seat No.</th>
            <th>Pick-up Point</th>
            <th>Guardian Name</th>
            <th>Guardian No.</th>
          </tr>
        </thead>
        <tbody>
        </tbody>
      </table>
    </div>
  </div>
  <?php
$query = 'SELECT * FROM admin WHERE status="STUDENT"'; 
$result = mysqli_query($db, $query) or die (mysqli_error($db));
while ($row = mysqli_fetch_assoc($result)) {

```

Fig. 4.23: code to show student data 1

```

    <tr>
      <td>$row['rnum']</td>
      <td>$row['name']</td>
      <td>$row['program']</td>
      <td>$row['branch']</td>
      <td>$row['sem']</td>
      <td>$row['mobilenum']</td>
      <td>$row['user']</td>
      <td>$row['route_number']</td>
      <td>$row['seat_number']</td>
      <td>$row['pickup_point']</td>
      <td>$row['pname']</td>
      <td>$row['pnum']</td>
    </tr>
  </tbody>
</table>
</div>
<?php include 'footer.php'; >

```

Fig. 4.24: code to show student data 2

- **Student Delete:** Figures 4.25, 4.26 shows the code which allows to delete a particular student's data.

```

<?php
    include('connection.php');
    include('header.php');

    ?>

<body>
<?php
    $query = 'DELETE FROM admin
    WHERE
        rnum = ' . $_POST['rnum'];
    $result = mysqli_query($db, $query) or die(mysqli_error($db));

    ?>
        <script type="text/javascript">
            alert("Successfully Deleted.");
            window.location = "student.php";
        </script>

</body>
</html>

```

Fig. 4.25: code to delete student data 1

```

<?php include 'connection.php';
include 'topnav.php'; ?>

<div class="container">
<div class="card card-register mx-auto mt-5">
<div class="card-header" style="background-image: linear-gradient(pink,#2144)"><h2>Delete Record</h2> </div>
<div class="card-body" style="background-image: linear-gradient(#2144,pink);">

<form action="studentdel.php?action=edit" method="post">
    <div>
        <label for="rnum">Roll Number:</label>
        <input type="rnum" id="rnum" name="rnum" />
    </div>
    <button type="submit" class="btn btn-default"> <h6> Delete </h6> </button>
</form>
</div>
</div>

<br> <br>
<br> <br>
    <?php include 'footer.php'; ?>

```

Fig. 4.26: code to delete student data 2

- **Student edit:** Figures 2.27, 2.28, 2.29, 2.30, 2.31 code allows to edit data.

```

studentedit.php
1 <?php include 'connection.php';
2 include 'topnav.php'; ?>
3
4 <div class="container">
5 <div class="card card-register mx-auto mt-5">
6
7 <?php
8 $query = 'SELECT * FROM admin
9 WHERE
10    rnum = $_POST["rnum"]';
11 $result = mysqli_query($db, $query) or die(mysqli_error($db));
12 while($row = mysqli_fetch_array($result))
13 {
14     $rnum= $row['rnum'];
15     $name= $row['name'];
16     $program= $row['program'];
17     $branch= $row['branch'];
18     $sem= $row['sem'];
19     $mobilenum= $row['mobilenum'];
20     $user= $row['user'];
21     $route_number= $row['route_number'];
22     $seat_number= $row['seat_number'];
23     $pickup_point= $row['pickup_point'];
24     $phone= $row['phone'];
25     $prnum= $row['prnum'];
26 }
27
28 $id = $_POST['rnum'];
29
30 <div class="container">
31 <div class="card card-register mx-auto mt-5">
32 <div class="card-header" style="background-image: linear-gradient(pink,#2144)"><h2>Edit Record</h2> </div>
33 <div class="card-body" style="background-image: linear-gradient(#2144,pink);">
34
35 <form action="studentedit.php?action=add" method="post">

```

Fig. 4.27: code to edit student data 1

```

36
37 <style>
38 .containers {
39     display: flex;
40     flex-flow: column;
41     position: top;
42     top: 0;
43     left: 100;
44     height: 50;
45     width: 100;
46 }
47 .containers .row {
48     margin: 0 auto;
49 }
50 body {
51     font-size: 100%;
52 }
53 h2 { font-family: sans-serif; }
54 h6 { font-family: sans-serif; }
55
56 </style>
57 <div class="containers">
58 <div class="card card-register mx-auto mt-0">
59 <div class="card-header" style="background-image: linear-gradient(to bottom, #eacda3, #d8ae7b)"><h2>Edit Record</h2> </div>
60 <div class="card-body" style="background-image: linear-gradient(to top, #eacda3, #d8ae7b);">
61 <form action="studentedit.php?action=add" method="post">
62
63 <div>
64     <label for="rnum">Roll Number:</label>
65     <input type="rnum" id="ROUTE_NUMBER" name="rnum" value=<?php echo $rnum; ?>>;
66 </div>
67
68 <div>
69     <label for="name">Name:</label>
70     <input type="name" id="name" name="name" value=<?php echo $name; ?>>;
71 </div>
72
73 <div>
74     <label for="program">Program:</label>
75     <input type="program" id="program" name="program" value=<?php echo $program; ?>>;
76 </div>

```

Fig. 4.28: code to edit student data 2

```

<label for="user">User:</label>
  <input type="user" id="user" name="user" value=<?php echo $user;?>>
</div>
</div>
<label for="route_number">Routenum:</label>
  <input type="route_number" id="route_number" name="route_number" value=<?php echo $route_number;?>>
</div>
</div>
<label for="seat_number">Seat number:</label>
  <input type="seat_number" id="seat_number" name="seat_number" value=<?php echo $seat_number;?>>
</div>
</div>
<label for="pickup_point">pickup point:</label>
  <input type="pickup_point" id="pickup_point" name="pickup_point" value=<?php echo $pickup_point;?>>
</div>
</div>
<label for="pname">Guardian Name:</label>
  <input type="pname" id="pname" name="pname" value=<?php echo $pname;?>>
</div>
</div>
<label for="pnum">Guardian Number:</label>
  <input type="pnum" id="pnum" name="pnum" value=<?php echo $pnum;?>>
</div>
</div>
<button type="submit" class="btn btn-default"> h6> Update  </button><br>
</form>
</div>
</div>
</div>
</div>
<?php include 'footer.php'; ?>

```

Fig. 4.29: code to edit student data 3

```

<?php include 'connection.php';>
<?php include 'topnav.php'; ?>
<?php
$route = $POST['route'];
$seat = $POST['name'];
$program = $POST['program'];
$branch = $POST['branch'];
$sem = $POST['sem'];
$mobilenum = $POST['mobilenum'];
$seav = $POST['user'];
$route_number = $POST['route_number'];
$seat_number = $POST['seat_number'];
$pickup_point = $POST['pickup_point'];
$phone = $POST['name'];
$prnum = $POST['prnum'];
$prnum = $POST['prnum'];

$query = "UPDATE admin set route = '$route',";
$query .= "name = '$name', program = '$program', branch = '$branch', sem = '$sem', mobilenum = '$mobilenum',";
$query .= "user = '$user', route_number = '$route_number', seat_number = '$seat_number',";
$query .= "pickup_point = '$pickup_point', phone = '$phone', prnum = '$prnum'";
$query .= "WHERE route = '$route' ";
$result = mysqli_query($db, $query) or die(mysqli_error($db));

```

<?php
<script type="text/javascript">
 alert("Update Successful.");
 window.location = "student.php";
</script>
<?php include 'footer.php'; ?>

Fig. 4.30: code to edit student data 4

```

  width: 100%,
}
.containers .row {
  margin: 0 auto;
}
body {
  font-size: 100%;
}
h2 { font-family: sans-serif; }
h6 { font-family: sans-serif; }
</style>
<div class="containers">
<div class="card card-register mx-auto mt-5">
<div class="card border-dark" >
<body>
<div class="card-header" style="background-image: linear-gradient(to bottom, #eacda3, #d6ae7b)">
<h2>Update Record:</h2> </div>
<div class="card-body" style="background-image: linear-gradient(to top, #eacda3, #d6ae7b)">
<form action="studentedit.php?action=edit" method="post">
<br>
  <div>
    <label for="rnum">ROLL NUMBER:</label>
    <input type="rnum" id="rnum" name="rnum" />
  </div>
  <br>
  <button type="submit" class="btn btn-default"> h6> Update  </button>
</form>
</div>
</div>
</div>
<br> <br>
<br> <br>
<?php include 'footer.php'; ?>

```

Fig. 4.31: code to edit student data 5

**Bus records:** The bus records in the admin module consist of bus details and allows to manipulate it.

- **Bus:** Figure 4.32 shows us the code with which we can view bus details.

```

<?php include 'connection.php';
include 'topnav.php'; ?>
<div class="col-lg-12">
<div>
<h1>Bus Records </h1>
<br> <br>
<a href="busadd.php?action=add" style="background-image: linear-gradient();>
type="button" class="btn btn-xs btn-info">Add</a>
<a href="buseditpopup.php?action=edit" style="background-image: linear-gradient();>
type="button" class="btn btn-xs btn-info">Update</a>
<a href="busdelpopup.php?action=delete" style="background-image: linear-gradient();>
type="button" class="btn btn-xs btn-info">Delete</a>
</div>
<br> <br>
<div class="table-responsive">
<table class="table table-bordered" id="dataable" width="100%" cellspacing="0">
<thead>
<tr>
<th>Route Number</th>
<th>Route Name</th>
<th>Driver ID</th>
<th>Seating Capacity</th>
<th>Occupancy</th>
<th>Registration Number</th>
</tr>
</thead>
<tbody>
<?php
$query = 'SELECT * FROM bus';
$result = mysqli_query($db, $query) or die (mysqli_error($db));
while ($row = mysqli_fetch_assoc($result)) {
    echo '<tr>';
    echo '<td>'. $row['ROUTE_NUMBER']. '</td>';
    echo '<td>'. $row['ROUTE_NAME']. '</td>';
    echo '<td>'. $row['DRIVER_ID']. '</td>';
    echo '<td>'. $row['SEATING_CAPACITY']. '</td>';
    echo '<td>'. $row['OCCUPANCY']. '</td>';
    echo '<td>'. $row['REG_NUMBER']. '</td>';
    echo '</tr> ';}
    </tbody>
</table>
</div>
<?php include 'footer.php'; ?>

```

Fig. 4.32: code to show bus data 1

- **Bus add:** Figures 4.33 and 4.34 shows code which allows to add new bus data.

```

<?php include 'connection.php';
include 'topnav.php'; ?>
<div class="card card-register mx-auto mt-5">
<div class="card-header" style="background-image: linear-gradient(pink, #2144)">
<h2>Add new Record</h2> </div>
<div class="card-body" style="background-image: linear-gradient(pink, #2144, pink);>
<form action="bustrans.php?action=add" method="post">
<div>
    <label for="ROUTE_NUMBER">ROUTE NUMBER:</label>
    <input type="ROUTE_NUMBER" id="ROUTE_NUMBER" name="ROUTE_NUMBER" />
</div>
<div>
    <label for="ROUTE_NAME">ROUTE NAME:</label>
    <input type="ROUTE_NAME" id="ROUTE_NAME" name="ROUTE_NAME" />
</div>
<div>
    <label for="DRIVER_ID">DRIVER ID:</label>
    <input type="DRIVER_ID" id="DRIVER_ID" name="DRIVER_ID" />
</div>
<div>
    <label for="SEATING_CAPACITY">SEATING CAPACITY:</label>
    <input type="SEATING_CAPACITY" id="SEATING_CAPACITY" name="SEATING_CAPACITY" />
</div>
<div>
    <label for="OCCUPANCY">OCCUPANCY:</label>
    <input type="OCCUPANCY" id="OCCUPANCY" name="OCCUPANCY" />
</div>
<div>
    <label for="REG_NUMBER">REGISTRATION NUMBER:</label>
    <input type="REG_NUMBER" id="REG_NUMBER" name="REG_NUMBER" />
</div>
<button type="submit" class="btn btn-default"> <h6> Submit </h6> </button>
</div> </div> </div> </div>
<?php include 'footer.php'; ?>

```

Fig. 4.33: code to add bus data 1

```

<?php include 'header.php' ;?>
<?php include 'connection.php' ;?>

<div class="col-lg-12">
<?php
$broute= $_POST['ROUTE_NUMBER'];
$bname= $_POST['ROUTE_NAME'];
$bdriver= $_POST['DRIVER_ID'];
$capacity= $_POST['SEATING_CAPACITY'];
$occupancy= $_POST['OCCUPANCY'];
$regnum= $_POST['REG_NUMBER'];
switch($_GET['action']){
    case 'add':
        $query = "INSERT INTO bus
        (ROUTE_NUMBER,ROUTE_NAME,DRIVER_ID,
        SEATING_CAPACITY, OCCUPANCY,REG_NUMBER)
        VALUES ('".$broute."','".$$bname."','".$$bdriver."',
        '".$$capacity."','".$$occupancy."','".$$regnum."')";
        mysqli_query($db,$query) or die (mysqli_error($db));
        break;
    }
<script type="text/javascript">
    alert("Successfully added.");
    window.location = "bus.php";
</script>
</div>

```

Fig. 4.34: code to add bus data 2

- **Bus delete:** Figures 4.35 and 4.36 shows code which allows to delete a particular bus's data.

```

<?php
    include('connection.php');
    include('header.php');
?>
<body>
<?php
    $query = 'DELETE FROM bus
        WHERE ROUTE_NUMBER = ' . $_POST['ROUTE_NUMBER'];
    $result = mysqli_query($db, $query) or die(mysqli_error($db));
?>
    <script type="text/javascript">
        alert("Successfully Deleted.");
        window.location = "bus.php";
    </script>
<?php
?>
</body>
</html>

```

Fig. 4.35: code to delete bus data 1

```

<?php include 'connection.php';
include 'topnav.php'; ?>

<div class="container">
<div class="card card-register mx-auto mt-5">
<div class="card-header" style="background-image:
    linear-gradient(pink,#2144)"><h2>Delete Record</h2> </div>
<div class="card-body" style="background-image:
    linear-gradient(#2144,pink);">

<form action="busdel.php?action=edit" method="post">
    <div>
        <label for="ROUTE_NUMBER">ROUTE NUMBER:</label>
        <input type="ROUTE_NUMBER" id="ROUTE_NUMBER" name="ROUTE_NUMBER" />
    </div>
    <button type="submit" class="btn btn-default"> <h6> Delete </h6> </button>
</div>
</div>
</div>

<br> <br>
<br> <br>
<?php include 'footer.php'; ?>

```

Fig. 4.36: code to delete bus data 2

- **Bus edit:** Figures 4.37 and 4.38 and 4.39 shows code which allows to edit a particular bus's data.

```

<?php include 'connection.php';
include 'topnav.php'; ?>
<div class="container">
<div class="card card-register mx-auto mt-5">
<div class="card-header" style="background-image:
    linear-gradient(pink,#2144)"><h2>Update Record</h2> </div>
<div class="card-body" style="background-image:
    linear-gradient(#2144,pink);">
<form action="busedit.php?action=edit" method="post">
    <div>
        <label for="ROUTE_NUMBER">ROUTE NUMBER:</label>
        <input type="ROUTE_NUMBER" id="ROUTE_NUMBER" name="ROUTE_NUMBER" />
    </div>
    <button type="submit" class="btn btn-default"> <h6> Update </h6> </button>
</form>
</div>
</div>
<br> <br>
<br> <br>
<?php include 'footer.php'; ?>

```

Fig. 4.37: code to edit bus data 1

```

<?php include 'connection.php';
include 'topnav.php'; ?>
<body>
<?php
    $broute= $_POST['ROUTE_NUMBER'];
    $bname= $_POST['ROUTE_NAME'];
    $bdriver= $_POST['DRIVER_ID'];
    $capacity= $_POST['SEATING_CAPACITY'];
    $occupancy= $_POST['OCCUPANCY'];
    $regnum= $_POST['REG_NUMBER'];
    $query = 'UPDATE bus set ROUTE_NUMBER ="'.$broute.'",
        ROUTE_NAME ="'.$bname.'", DRIVER_ID ="'.$bdriver.'",
        SEATING_CAPACITY="'.$capacity.'", OCCUPANCY="'.$occupancy.'",
        REG_NUMBER="'.$regnum.'"
        WHERE ROUTE_NUMBER=' . $_POST['ROUTE_NUMBER'];
    $result = mysqli_query($db, $query) or die(mysqli_error($db));
?>
    <script type="text/javascript">
        alert("Update Successfull.");
        window.location = "bus.php";
    </script>
<?php include 'footer.php'; ?>

```

Fig. 4.38: code to edit bus data 2

```

<?php include 'connection.php';
include 'topnav.php';>
<div class="container">
<div class="card card-register mx-auto mt-5">
<?php
$query = 'SELECT * FROM bus
          WHERE ROUTE_NUMBER = "'.$_POST['ROUTE_NUMBER'].'";
$result = mysqli_query($db, $query) or die(mysqli_error($db));
while($row = mysqli_fetch_array($result))
{
    $route= $row['ROUTE_NUMBER'];
    $bname= $row['ROUTE_NAME'];
    $bdriver= $row['DRIVER_ID'];
    $capacity= $row['SEATING_CAPACITY'];
    $occupancy= $row['OCCUPANCY'];
    $regnum= $row['REG_NUMBER'];
    $id = $row['ROUTE_NUMBER'];
}
$did = $_POST['ROUTE_NUMBER'];
?>
<div class="Container">
<div class="card card-register mx-auto mt-5">
<div class="card-header" style="background-image: linear-gradient(pink, #2144);><h2>Edit Record</h2> </div>
<div class="card-body" style="background-image: linear-gradient(#2144, pink);>
<form action="busedit.php?action=add" method="post">
<div>
    <label for="ROUTE_NUMBER">ROUTE NUMBER:</label>
    <input type="ROUTE_NUMBER" id="ROUTE_NUMBER" name="ROUTE_NUMBER" value=<?php echo $broute; ?>">
</div>
<div>
    <label for="ROUTE_NAME">ROUTE NAME:</label>
    <input type="ROUTE_NAME" id="ROUTE_NAME" name="ROUTE_NAME" value=<?php echo $bname; ?>">
</div>
<div>
    <label for="DRIVER_ID">DRIVER:</label>
    <input type="DRIVER_ID" id="DRIVER_ID" name="DRIVER_ID" value=<?php echo $bdriver; ?>">
</div>
<div>
    <label for="SEATING_CAPACITY">SEATING CAPACITY:</label>
    <input type="SEATING_CAPACITY" id="SEATING_CAPACITY" name="SEATING_CAPACITY" value=<?php echo $capacity; ?>">
</div>
<div>
    <label for="OCCUPANCY">OCCUPANCY:</label>
    <input type="OCCUPANCY" id="OCCUPANCY" name="OCCUPANCY" value=<?php echo $occupancy; ?>">
</div>
<div>
    <label for="REG_NUMBER">REGISTRATION NUMBER:</label>
    <input type="REG_NUMBER" id="REG_NUMBER" name="REG_NUMBER" value=<?php echo $regnum; ?>">
</div>
<div>
    <button type="submit" class="btn btn-default"> <b>Update</b> </button>
</div>
</form>
</div>
</div>
</div>
<?php include 'footer.php'; ?>

```

Fig. 4.39: code to edit bus data 3

**Driver records:** The driver records in the admin module consist of driver details and allows to manipulate it. Figures 4.40 and 4.41 shows code which helps in viewing driver details.

- **Driver:**

```

<?php include 'connection.php';
include 'topnav.php';>
<div class="col-lg-12">
<div>
<h2>Driver Records </h2>
<br> <br>
<a href="driveradd.php?action=add" style="background-image: linear-gradient();>
    <type="button" class="btn btn-xs btn-info">Add</a>
<a href="drivereditpopup.php?action=edit" style="background-image: linear-gradient();>
    <type="button" class="btn btn-xs btn-info">Update</a>
<a href="driverdelpopup.php?action=delete" style="background-image: linear-gradient();>
    <type="button" class="btn btn-xs btn-info">Delete</a>
</div>
<br> <br>
<div class="table-responsive">
    <table class="table table-bordered" id="dataable" width="100%" cellspacing="0">
        <thead>
            <tr>
                <th>Driver ID</th>
                <th>Driver Name</th>
                <th>Mobile Number</th>
                <th>Email</th>
                <th>Aadhaar Number</th>
                <th>Bus Number</th>
                <th>Employ Date</th>
                <th>Salary</th>
            </tr>
        </thead>
        <tbody>
            <?php
$query = 'SELECT * FROM driver';
$result = mysqli_query($db, $query) or die (mysqli_error($db));
while ($row = mysqli_fetch_assoc($result)) {
    echo '<tr>';
    echo '<td>'. $row['DRIVER_ID']. '</td>';
    echo '<td>'. $row['DRIVER_NAME']. '</td>';
    echo '<td>'. $row['MOBILE_NUMBER']. '</td>';
    echo '<td>'. $row['EMAIL']. '</td>';
    echo '<td>'. $row['AADHAAR_NUMBER']. '</td>';
    echo '<td>'. $row['BUS_NUM']. '</td>';
    echo '<td>'. $row['EMPLOY_DATE']. '</td>';
    echo '<td>'. $row['SALARY']. '</td>';
    echo '</tr> ' ; ?>
}
</tbody>
</table>
</div>
<?php include 'footer.php'; ?>

```

Fig. 4.40: code to show driver data 1

```

<?php include 'header.php' ?>
<?php include 'connection.php' ?>
<div class="col-lg-12">
<?php
$did = $_POST['DRIVER_ID'];
$dname= $_POST['DRIVER_NAME'];
$drum= $_POST['MOBILE_NUMBER'];
$demail= $_POST['EMAIL'];
$daadhaar= $_POST['AADHAAR_NUMBER'];
$dbus= $_POST['BUS_NUM'];
$edate= $_POST['EMPLOY_DATE'];
$dsal= $_POST['SALARY'];
switch($_GET['action']){
    case 'add':
        $query = "INSERT INTO driver
                  (DRIVER_ID, DRIVER_NAME, MOBILE_NUMBER,
                   EMAIL, AADHAAR_NUMBER, BUS_NUM, EMPLOY_DATE, SALARY)
                  VALUES ('". $did. "','" . $dname. "','" . $drum. "'",
                  "' . $demail. "','" . $daadhaar. "','" . $dbus. "'",
                  "' . $edate. "','" . $dsal. "')";
        mysqli_query($db, $query) or die (mysqli_error($db));
        break; ?>
<script type="text/javascript">
    alert("Successfully added.");
    window.location = "driver.php";
</script>
</div>

```

Fig. 4.41: code to show driver data 2

- **Driver add:** Figures 4.42 and 4.43 shows code which allows to add new driver data.

```
<?php include 'connection.php';
include 'topnav.php'; >
<div class="container">
<div class="card card-register mx-auto mt-5">
<div class="card-header" style="background-image: linear-gradient(pink,#2144)">
    <h2>Add New Record</h2> </div>
<div class="card-body" style="background-image: linear-gradient(#2144,pink);">
<form action="drivertrans.php?action=add" method="post">
    <div>
        <label for="DRIVER_ID">DRIVER ID:</label>
        <input type="DRIVER_ID" id="DRIVER_ID" name="DRIVER_ID" />
    </div>
    <div>
        <label for="DRIVER_NAME">DRIVER NAME:</label>
        <input type="DRIVER_NAME" id="DRIVER_NAME" name="DRIVER_NAME" />
    </div>
    <div>
        <label for="MOBILE_NUMBER">MOBILE NUMBER:</label>
        <input type="MOBILE_NUMBER" id="MOBILE_NUMBER" name="MOBILE_NUMBER" />
    </div>
    <div>
        <label for="EMAIL">EMAIL:</label>
        <input type="EMAIL" id="EMAIL" name="EMAIL" />
    </div>
    <div>
        <label for="AADHAAR_NUMBER">AADHAAR NUMBER:</label>
        <input type="AADHAAR_NUMBER" id="AADHAAR_NUMBER" name="AADHAAR_NUMBER" />
    </div>
    <div>
        <label for="BUS_NUM">BUS NUMBER:</label>
        <input type="BUS_NUM" id="BUS_NUM" name="BUS_NUM" />
    </div>
    <div>
        <label for="EMPLOY_DATE">EMPLOY DATE:</label>
        <input type="EMPLOY_DATE" id="EMPLOY_DATE" name="EMPLOY_DATE" />
    </div>
    <div>
        <label for="SALARY">SALARY:</label>
        <input type="SALARY" id="SALARY" name="SALARY" />
    </div>
    <button type="submit" class="btn btn-default"> <h6> Submit </h6> </button>
</form> </div> </div> </div>
<?php include 'footer.php'; ?>
```

Fig. 4.42: code to add driver data 1

```
<?php include 'header.php' ;?>
<?php include 'connection.php' ;?>
<div class="col-lg-12">
    <?php
        $did= $_POST['DRIVER_ID'];
        $dname= $_POST['DRIVER_NAME'];
        $dnum= $_POST['MOBILE_NUMBER'];
        $demail= $_POST['EMAIL'];
        $daadhaar= $_POST['AADHAAR_NUMBER'];
        $dbus= $_POST['BUS_NUM'];
        $sedate= $_POST['EMPLOY_DATE'];
        $dsal= $_POST['SALARY'];
        switch($_GET['action']){
            case 'add':
                $query = "INSERT INTO driver
                (DRIVER_ID,DRIVER_NAME,MOBILE_NUMBER,
                EMAIL,AADHAAR_NUMBER,BUS_NUM,EMPLOY_DATE,SALARY)
                VALUES ('".$did."','".$dname."','".$dnum."',
                '".$demail."','".$daadhaar."','".$dbus."',
                '".$sedate."','".$dsal."')";
                mysqli_query($db,$query)or die (mysqli_error($db));
                break;}}>
    <script type="text/javascript">
        alert("Successfully added.");
        window.location = "driver.php";
    </script>
</div>
```

Fig. 4.43: code to add driver data 2

- **Driver delete:** Figures 4.44 and 4.45 shows code which allows to delete a particular driver's data.

```
<?php include 'connection.php';
include 'topnav.php'; >
<div class="container">
<div class="card card-register mx-auto mt-5">
<div class="card-header" style="background-image:
    linear-gradient(pink,#2144)"><h2>Delete Record</h2> </div>
<div class="card-body" style="background-image:
    linear-gradient(#2144,pink);">
<form action="driverdel.php?action=edit" method="post">
    <div>
        <label for="DRIVER_ID">DRIVER ID:</label>
        <input type="DRIVER_ID" id="DRIVER_ID" name="DRIVER_ID" />
    </div>
    <button type="submit" class="btn btn-default"> <h6> Delete </h6> </button>
</form>
</div>
</div>
<?php include 'footer.php'; ?>
```

Fig. 4.44: code to delete driver data 1

```
<?php
    include('connection.php');
    include('header.php');
?>
<body>
<?php
    $query = 'DELETE FROM driver
    WHERE DRIVER_ID = ' . $_POST['DRIVER_ID'];
    $result = mysqli_query($db, $query) or die(mysqli_error($db));
?>
    <script type="text/javascript">
        alert("Successfully Deleted.");
        window.location = " driver.php";
    </script>
<?php
?>
</body>
</html>
```

Fig. 4.45: code to delete driver data 2

- **Driver edit:** Figures 4.46, 4.47, 4.48 and 4.49 shows code which allows to edit a particular driver's data.

```

<?php include 'connection.php';
include 'topnav.php'; ?>
<div class="container">
<div class="card card-register mx-auto mt-5">
<div class="card-header" style="background-image: linear-gradient(pink,#2144)"><h2>Update Record</h2> </div>
<div class="card-body" style="background-image: linear-gradient(#2144,pink);">
<form action="drivedit.php?action=edit" method="post">
    <div>
        <label for="DRIVER_ID">DRIVER ID:</label>
        <input type="DRIVER_ID" id="DRIVER_ID" name="DRIVER_ID" />
    </div>
    <button type="submit" class="btn btn-default"> <h6> Update </h6> </button>
</form>
</div>
</div>
<br> <br> <br>
<?php include 'footer.php'; ?>

```

Fig. 4.46: code to edit driver data 1

```

<?php include 'connection.php';
include 'topnav.php'; ?>
<body>
<?php
$did= $_POST['DRIVER_ID'];
$dname= $_POST['DRIVER_NAME'];
$dnum= $_POST['MOBILE_NUMBER'];
$demail= $_POST['EMAIL'];
$daadhaar= $_POST['AADHAAR_NUMBER'];
$dbus= $_POST['BUS_NUM'];
$edate= $_POST['EMPLOY_DATE'];
$dsal= $_POST['SALARY'];
$query = 'UPDATE driver set DRIVER_ID="'.$did.'",';
DRIVER_NAME="'.$dname.'", MOBILE_NUMBER="'.$dnum.'",
EMAIL="'.$demail.'",AADHAAR_NUMBER="'.$daadhaar.'",
BUS_NUM="'.$dbus.'",EMPLOY_DATE="'.$edate.'",SALARY="'.$dsal.'"
WHERE DRIVER_ID= ' . $_POST['DRIVER_ID'];
$result = mysqli_query($db, $query) or die(mysqli_error($db));
?>
<script type="text/javascript">
    alert("Update Successfull.");
    window.location = "driver.php";
</script>
<?php include 'footer.php'; ?>

```

Fig. 4.47: code to edit driver data 2

```

<?php include 'connection.php';
include 'topnav.php'; ?>
<div class="container">
<div class="card card-register mx-auto mt-5">
<?php
$query = 'SELECT * FROM driver
WHERE DRIVER_ID ='.$_POST['DRIVER_ID'];
$result = mysqli_query($db, $query) or die(mysqli_error($db));
while($row = mysqli_fetch_array($result))
{
$did= $row['DRIVER_ID'];
$dname= $row['DRIVER_NAME'];
$dnum= $row['MOBILE_NUMBER'];
$demail= $row['EMAIL'];
$daadhaar= $row['AADHAAR_NUMBER'];
$dbus= $row['BUS_NUM'];
$edate= $row['EMPLOY_DATE'];
$dsal= $row['SALARY'];
}
$id = $_POST['DRIVER_ID'];?>
<div class="container">
<div class="card card-register mx-auto mt-5">
<div class="card-header" style="background-image:
linear-gradient(pink,#2144)"><h2>Edit Record</h2> </div>
<div class="card-body" style="background-image:
linear-gradient(#2144,pink);">

```

Fig. 4.48: code to edit driver data 3

```

<form action="drivedit1.php?action=add" method="post">
    <div>
        <label for="DRIVER_ID">DRIVER ID:</label>
        <input type="DRIVER_ID" id="DRIVER_ID" name="DRIVER_ID" value="<?php echo $did; ?>">
    </div>
    <div>
        <label for="DRIVER_NAME">DRIVER NAME:</label>
        <input type="DRIVER_NAME" id="DRIVER_NAME" name="DRIVER_NAME" value="<?php echo $dname; ?>">
    </div>
    <div>
        <label for="MOBILE_NUMBER">MOBILE NUMBER:</label>
        <input type="MOBILE_NUMBER" id="MOBILE_NUMBER" name="MOBILE_NUMBER" value="<?php echo $dnum; ?>">
    </div>
    <div>
        <label for="EMAIL">EMAIL:</label>
        <input type="EMAIL" id="EMAIL" name="EMAIL" value="<?php echo $demail; ?>">
    </div>
    <div>
        <label for="AADHAAR_NUMBER">AADHAAR NUMBER:</label>
        <input type="AADHAAR_NUMBER" id="AADHAAR_NUMBER" name="AADHAAR_NUMBER" value="<?php echo $daadhaar; ?>">
    </div>
    <div>
        <label for="BUS_NUM">BUS NUMBER:</label>
        <input type="BUS_NUM" id="BUS_NUM" name="BUS_NUM" value="<?php echo $dbus; ?>">
    </div>
    <div>
        <label for="EMPLOY_DATE">EMPLOY DATE:</label>
        <input type="EMPLOY_DATE" id="EMPLOY_DATE" name="EMPLOY_DATE" value="<?php echo $edate; ?>">
    </div>
    <div>
        <label for="SALARY">SALARY:</label>
        <input type="SALARY" id="SALARY" name="SALARY" value="<?php echo $dsal; ?>">
    </div>
    <button type="submit" class="btn btn-default"> <h6> Update </h6> </button>
</form> </div> </div> </div>
<?php include 'footer.php'; ?>

```

Fig. 4.49: code to edit driver data 4

**Route records:** The route records in the admin module consist of route details and allows to manipulate it.

- **Route:** Figures 4.50 and 4.51 shows the code which helps us in viewing route details in tabular format.

```

<?php include 'connection.php';
include 'topnav.php'; ?>
<div class="col-lg-12">
<div>
    <h1>Route Records </h1>
    <br> </br>
    <a href="routeadd.php?action=add" style="background-image: linear-gradient();>
        type="button" class="btn btn-xs btn-info">Add</a>
    <a href="routeeditpopup.php?action=edit" style="background-image: linear-gradient();>
        type="button" class="btn btn-xs btn-info">Update</a>
    <a href="routedelpopup.php?action=delete" style="background-image: linear-gradient();>
        type="button" class="btn btn-xs btn-info">Delete</a>
    </div> <br> </br>
    <div class="table-responsive">
        <table class="table table-bordered" id="dataable" width="100%" cellspacing="0">
            <thead>
                <tr>
                    <th>Route Number</th>
                    <th>Route Name</th>
                    <th>Enroute Stages</th>
                    <th>Start Time</th>
                    <th>Departure</th>
                </tr> </thead> <tbody>
            <tr: nth-child(even) {>
                background-color: #D6EEEE; </style>
            <td>
                <?php
                    $query = 'SELECT * FROM route';
                    $result = mysqli_query($db, $query) or die (mysqli_error($db));
                    while ($row = mysqli_fetch_assoc($result)) {
                        echo '<tr>';
                        echo '<td>'. $row['ROUTE_NUMBER']. '</td>';
                        echo '<td>'. $row['ROUTE_NAME']. '</td>';
                        echo '<td>'. $row['ENROUTE_STAGES']. '</td>';
                        echo '<td>'. $row['START_TIME']. '</td>';
                        echo '<td>'. $row['DEPARTURE']. '</td>';
                        echo '</tr> ' ; } >
            </td>
        </tr>
    </tbody> </table> </div>
<?php include 'footer.php'; ?>

```

Fig. 4.50: code to show route data 1

```

<?php include 'header.php' ;?>
<?php include 'connection.php' ;?>
<div class="col-lg-12">
    <?php
        $rroute= $_POST['ROUTE_NUMBER'];
        $rname= $_POST['ROUTE_NAME'];
        $rstage= $_POST['ENROUTE_STAGES'];
        $stime= $_POST['START_TIME'];
        $departure= $_POST['DEPARTURE'];
        switch($_GET['action']){
            case 'add':
                $query = "INSERT INTO route
                    (ROUTE_NUMBER,ROUTE_NAME,ENROUTE_STAGES, START_TIME, DEPARTURE)
                    VALUES ('".$rroute."','".$rname."','".$rstage."','".$stime."','".$departure."')";
                mysqli_query($db,$query) or die (mysqli_error($db));
                break;?>
        <script type="text/javascript">
            alert("Successfully added.");
            window.location = "route.php";
        </script>
    </div>

```

Fig. 4.51: code to show route data 2

- **Route add:** Figure 4.52 shows code which allows to add new bus route data.

```

<?php include 'connection.php';
include 'topnav.php'; ?>
<div class="container">
<div class="card card-register mx-auto mt-5">
<div class="card-header" style="background-image: linear-gradient(pink,#2144)"><h2>Add new Record</h2> </div>
<div class="card-body" style="background-image: linear-gradient(#2144,pink);">
<form action="routetrans.php?action=add" method="post">
    <div>
        <label for="ROUTE_NUMBER">ROUTE NUMBER:</label>
        <input type="ROUTE_NUMBER" id="ROUTE_NUMBER" name="ROUTE_NUMBER" />
    </div>
    <div>
        <label for="ROUTE_NAME">ROUTE NAME:</label>
        <input type="ROUTE_NAME" id="ROUTE_NAME" name="ROUTE_NAME" />
    </div>
    <div>
        <label for="ENROUTE_STAGES">ENROUTE STAGES:</label>
        <input type="ENROUTE_STAGES" id="ENROUTE_STAGES" name="ENROUTE_STAGES" />
    </div>
    <div>
        <label for="START_TIME">START TIME:</label>
        <input type="START_TIME" id="START_TIME" name="START_TIME" />
    </div>
    <div>
        <label for="DEPARTURE">DEPARTURE:</label>
        <input type="DEPARTURE" id="DEPARTURE" name="DEPARTURE" />
    </div>
    <button type="submit" class="btn btn-default"> <h6> Submit </h6> </button>
</form>    </div>    </div>    </div>
<?php include 'footer.php'; ?>

```

Fig. 4.52: code to add route data

- **Route delete:** Figures 4.53 and 4.54 shows code which allows to delete a particular bus route's data.

```

<?php include 'connection.php';
include 'topnav.php'; ?>
<div class="container">
<div class="card card-register mx-auto mt-5">
<div class="card-header" style="background-image: linear-gradient(pink,#2144)">
    <h2>Delete Record</h2> </div>
<div class="card-body" style="background-image: linear-gradient(#2144,pink);">
<form action="routedel.php?action=edit" method="post">
    <div>
        <label for="ROUTE_NUMBER">ROUTE NUMBER:</label>
        <input type="ROUTE_NUMBER" id="ROUTE_NUMBER" name="ROUTE_NUMBER" />
    </div>
    <button type="submit" class="btn btn-default"> <h6> Delete </h6> </button>
</form>
</div>
</div>
<br> <br>
<?php include 'footer.php'; ?>

```

Fig. 4.53: code to delete route data 1

```

<?php
    include('connection.php');
    include('header.php');
    ?>
<body>
<?php
    $query = 'DELETE FROM route
    WHERE ROUTE_NUMBER = ' . $_POST['ROUTE_NUMBER'];
    $result = mysqli_query($db, $query) or die(mysqli_error($db));
?>
<script type="text/javascript">
    alert("Successfully Deleted.");
    window.location = " route.php";
</script>
<?php?>
</body>
</html>

```

Fig. 4.54: code to delete route data 2

- **Route edit:** Figures 4.55 and 4.56 shows code allows to edit a particular bus route's data.

```

<?php include 'connection.php';
include 'topnav.php'; ?>
<div class="container">
<div class="card card-register mx-auto mt-5">
<div class="card-header" style="background-image: linear-gradient(pink,#2144)">
  <h2>Update Record</h2> </div>
<div class="card-body" style="background-image: linear-gradient(#2144,pink);">
  <form action="routeedit.php?action=edit" method="post">
    <div>
      <label for="ROUTE_NUMBER">ROUTE NUMBER:</label>
      <input type="ROUTE_NUMBER" id="ROUTE_NUMBER" name="ROUTE_NUMBER" />
    </div>
    <button type="submit" class="btn btn-default"> <h6> Update </h6> </button>
  </form>
</div>
<br> <br>
<br> <br>
<?php include 'footer.php'; ?>
<?php include 'connection.php';
include 'topnav.php'; ?>
<body>
<?php
  $rroute= $_POST['ROUTE_NUMBER'];
  $rname= $_POST['ROUTE_NAME'];
  $rstages= $_POST['ENROUTE_STAGES'];
  $stime= $_POST['START_TIME'];
  $departure= $_POST['DEPARTURE'];

  $query = 'UPDATE route set ROUTE_NUMBER = "'.$rroute.'",
            ROUTE_NAME = "'.$rname.'", ENROUTE_STAGES = "'.$rstages.'",
            START_TIME = "'.$stime.'", DEPARTURE = "'.$departure.'"
            WHERE ROUTE_NUMBER = ' . $_POST['ROUTE_NUMBER'];
  $result = mysqli_query($db, $query) or die(mysqli_error($db));
?>
<script type="text/javascript">
  alert("Update Successfull.");
  window.location = "route.php";
</script>
<?php include 'footer.php'; ?>

```

Fig. 4.55: code to edit route data 1

Fig. 4.56: code to edit route data 2

```

<?php include 'connection.php';
include 'topnav.php'; ?>
<div class="container">
<div class="card card-register mx-auto mt-5">
<?php
  $query = 'SELECT * FROM route
            WHERE ROUTE_NUMBER = "'.$_POST['ROUTE_NUMBER'];
  $result = mysqli_query($db, $query) or die(mysqli_error($db));
  while($row = mysqli_fetch_array($result))
  {
    $rroute= $row['ROUTE_NUMBER'];
    $rname= $row['ROUTE_NAME'];
    $rstages= $row['ENROUTE_STAGES'];
    $stime= $row['START_TIME'];
    $departure= $row['DEPARTURE'];
  }
  $id = $_POST['ROUTE_NUMBER'];
?>
<div class="container">
<div class="card card-register mx-auto mt-5">
<div class="card-header" style="background-image: linear-gradient(pink,#2144)"><h2>Edit Record</h2> </div>
<div class="card-body" style="background-image: linear-gradient(#2144,pink);">
  <form action="routeedit1.php?action=add" method="post">
    <div>
      <label for="ROUTE_NUMBER">ROUTE NUMBER:</label>
      <input type="ROUTE_NUMBER" id="ROUTE_NUMBER" name="ROUTE_NUMBER" value="<?php echo $rroute; ?>">
    </div>
    <div>
      <label for="ROUTE_NAME">ROUTE NAME:</label>
      <input type="ROUTE_NAME" id="ROUTE_NAME" name="ROUTE_NAME" value="<?php echo $rname; ?>">
    </div>
    <div>
      <label for="ENROUTE_STAGES">ENROUTE STAGES:</label>
      <input type="ENROUTE_STAGES" id="ENROUTE_STAGES" name="ENROUTE_STAGES" value="<?php echo $rstages; ?>">
    </div>
    <div>
      <label for="START_TIME">START TIME:</label>
      <input type="START_TIME" id="START_TIME" name="START_TIME" value="<?php echo $stime; ?>"> </div>
    <div> <label for="DEPARTURE">DEPARTURE:</label>
      <input type="DEPARTURE" id="DEPARTURE" name="DEPARTURE" value="<?php echo $departure; ?>"> </div>
    <button type="submit" class="btn btn-default"> <h6> Update </h6> </button>
  </form>
</div> </div> </div>
<?php include 'footer.php'; ?>

```

Fig. 4.57: code to edit route data 3

**Complaint/Suggestion box:** This record helps the admin in viewing complaints and suggestions regarding bus transport facility.

- **Csbox:** Figures 4.58 and 4.59 shows code shows the complaints and suggestions received in a tabular format.

```

<?php include 'connection.php';
include 'topnav.php'; ?>
    <div class="col-lg-12">
<div>
    <h1>Complaint/Suggestion Box </h1>
    <br> </br>
    <div class="table-responsive">
        <table class="table table-bordered" id="dataable" width="100%" cellspacing="0">
<style>
tr:nth-child(even) {
    background-color: #D6EEEE;
</style>
<thead>
    <tr>
        <th>Rollnumber</th>
        <th>Type</th>
        <th>Subject</th>
    </tr>
</thead>
<tbody>
<?php
$query = 'SELECT * FROM csbox';
$result = mysqli_query($db, $query) or die (mysqli_error($db));
while ($row = mysqli_fetch_assoc($result)) {
    echo '<tr>';
    echo '<td>'. $row['Rollnumber']. '</td>';
    echo '<td>'. $row['Type']. '</td>';
    echo '<td>'. $row['Subject']. '</td>';
    echo '</tr> ' ; } ?>
</tbody>
</table>
</div>
<?php include 'footer.php'; ?>

```

Fig. 4.58: code to show complaints and suggestions 1

```

csboxdel.php
1  <?php
2  include('connection.php');
3  ?>
4  <body>
5  <?php
6  if (!isset($_GET['do']) || $_GET['do'] != 1) {
7  switch ($_GET['type']) {
8      case 'csbox':
9          $query = 'DELETE FROM csbox
10             WHERE
11                 id = ' . $_GET['id'];
12             $result = mysqli_query($db, $query) or die(mysqli_error($db));
13
14 ?>
15             <script type="text/javascript">
16                 alert("Successfully Deleted.");
17                 window.location = " csbox.php";
18             </script>
19             <?php
20                 //break;
21                 }
22             }
23             ?>
24
25 </body>
26 </html>

```

Fig. 4.59: code to show complaints and suggestions 2

## STUDENT MODULE:

- **Front page:** The Figures 4.60 and 4.61 shows code front page of the student module contains a sidebar, drop down box and footer consisting of transport contact details.

```
homepage.php
1  <?php
2  include 'connection.php';
3  require 'session.php';
4  confirm_logged_in();
5  |?>
6  <!DOCTYPE html>
7  <html lang="en">
8  |<head>
9  |<meta charset="utf-8">
10 |<meta http-equiv="X-UA-Compatible" content="IE=edge">
11 |<meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
12 |<meta name="description" content="">
13 |<meta name="author" content="">
14 |<title>BUS</title>
15 |<link href="vendor/bootstrap/css/bootstrap.min.css" rel="stylesheet">
16 |<link href="vendor/fontawesome-free/css/all.min.css" rel="stylesheet" type="text/css">
17 |<link href="vendor/datatables/dataTables.bootstrap4.css" rel="stylesheet">
18 |<link href="css/sb-admin.css" rel="stylesheet">
19 |<style>
20 |  body {font-family: Arial, Helvetica, sans-serif;}
```

● 21 #myImg {  
22 border-radius: 5px;  
23 cursor: pointer;  
24 transition: 0.3s;  
25 }  
26 #myImg:hover {opacity: 0.7;}  
27 /\* The Modal (background) \*/  
28 .modal {  
29 display: none; /\* Hidden by default \*/  
30 position: fixed; /\* Stay in place \*/  
31 z-index: 1; /\* Sit on top \*/  
32 padding-top: 100px; /\* Location of the box \*/  
33 left: 0;  
34 top: 0;  
35 width: 100%; /\* Full width \*/  
36 height: 100%; /\* Full height \*/  
37 overflow: auto; /\* Enable scroll if needed \*/

```
1  /* Modal Content (image) */
2  .modal-content {
3      margin: auto;
4      display: block;
5      width: 80%;
6      max-width: 700px;
7  }
8
9
10 /* Caption of Modal Image */
11 #caption {
12     margin: auto;
13     display: block;
14     width: 80%;
15     max-width: 700px;
16     text-align: center;
17     color: #ccc;
18     padding: 10px 0;
19     height: 150px;
20 }
21
22 /* Add Animation */
23 .modal-content, #caption {
24     -webkit-animation-name: zoom;
25     -webkit-animation-duration: 0.6s;
26     animation-name: zoom;
27     animation-duration: 0.6s;
28 }
29
30 @-webkit-keyframes zoom {
31     from {-webkit-transform:scale(0)}
32     to {-webkit-transform:scale(1)}
33 }
34
35 @keyframes zoom {
36     from {transform:scale(0)}
37     to {transform:scale(1)}
```

Fig. 4.60: student module front page 1

Fig. 4.61: student module front page 2

```
<style>
</head>

<body id="page-top">

<nav class="navbar navbar-expand navbar-dark bg-dark static-top" style="background-image: linear-gradient(to left, #42278a, #7349ad);">

  <a class="navbar-brand mr-1" href="homepage.php" style="color: white; font-weight: bold; font-size: 1.2em; font-family: sans-serif; text-decoration: none; text-transform: uppercase; letter-spacing: 0.1em; border-bottom: 2px solid transparent; border-bottom: 2px solid white; padding-bottom: 0.1em; margin-bottom: 0.1em; transition: border-bottom 0.3s ease; position: relative; z-index: 1;*>CET BUS TRANSPORT</a>

  <button class="btn btn-link btn-sm text-white order-1 order-sm-0" id="sidebarToggle" href="#"><i class="fa fa-bars" style="font-size: 1.2em; color: white;*></i></button>

  <div class="d-none d-md-inline-block form-inline ml-auto mr-0 mr-md-3 my-2 my-md-0" style="position: relative; z-index: 0;*>

    <!-- Navbar Search -->
    <form class="d-none d-md-inline-block form-inline ml-auto mr-0 mr-md-3 my-2 my-md-0" style="position: relative; z-index: 0;*>

      <i>Home</i>

      <!-- Navbar -->
      <ul class="navbar-nav ml-auto ml-md-0" style="position: relative; z-index: 0;*>
        <li class="nav-item dropdown no-arrow" style="position: relative; z-index: 0;*>
          <a class="nav-link dropdown-toggle" href="#" id="userDropdown" role="button" data-toggle="dropdown" aria-haspopup="true" aria-expanded="false" style="color: white; font-weight: bold; font-size: 0.9em; font-family: sans-serif; text-decoration: none; text-transform: uppercase; letter-spacing: 0.1em; border-bottom: 2px solid transparent; border-bottom: 2px solid white; padding-bottom: 0.1em; margin-bottom: 0.1em; transition: border-bottom 0.3s ease; position: relative; z-index: 1;*>
            <i class="fa fa-user-circle fa-fw" style="font-size: 1.1em; color: white; margin-right: 0.1em;*></i>User
          <div class="dropdown-menu dropdown-menu-right" aria-labelledby="userDropdown" style="position: absolute; top: 100%; left: 0; width: 100%; background-color: #42278a; color: white; font-weight: bold; font-size: 0.9em; font-family: sans-serif; text-decoration: none; text-transform: uppercase; letter-spacing: 0.1em; border-radius: 0.2em; padding: 0.5em; margin-top: 0.1em; position: relative; z-index: 1;*>
            <a class="dropdown-item" href="logout.php" data-target="#logoutModal" style="color: white; font-weight: bold; font-size: 0.9em; font-family: sans-serif; text-decoration: none; text-transform: uppercase; letter-spacing: 0.1em; border-bottom: 1px solid transparent; border-bottom: 1px solid white; padding-bottom: 0.1em; margin-bottom: 0.1em; transition: border-bottom 0.3s ease; position: relative; z-index: 1;*>Logout</a>
          </div>
        </li>
        <li>
          <div style="border: 1px solid #42278a; border-radius: 0.2em; padding: 0.2em; display: inline-block; color: white; font-weight: bold; font-size: 0.8em; font-family: sans-serif; text-decoration: none; text-transform: uppercase; letter-spacing: 0.1em; position: relative; z-index: 1;*>
            <span style="font-size: 0.9em; color: white; margin-right: 0.1em;*><i class="fa fa-search fa-fw" style="font-size: 1.1em;*></i></span>Search
          </div>
        </li>
      </ul>
    </form>
  </div>
</nav>
```

```
<!-- Breadcrumbs-->
<ol class="breadcrumb">
  <li class="breadcrumb-item" style="background-image: linear-gradient();">
    <a href="homepage.php"> <h2> BUS SERVICES </h2> </a>
  </li>
</ol>

<!-- Page Content -->
<div class="bus_logo">



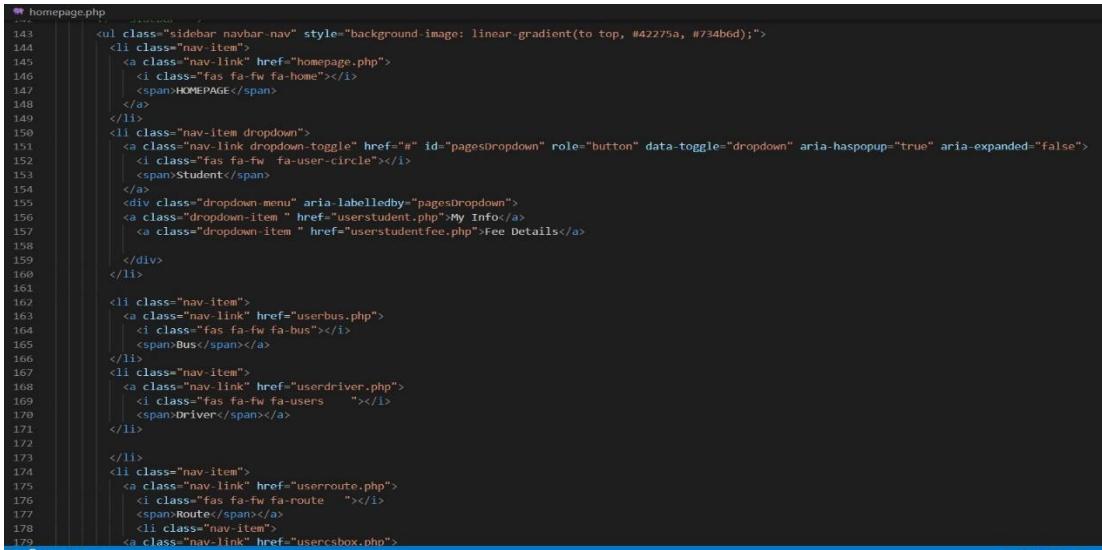

<br> <br>
<h2>Contact Details</h2>
<p>
  1. Dr. R.P. Chowdary
  <br>
  Associate Professor of Mechanical Department.
  <br>
  Phone: 9177220635( 4.30 to 6.30PM)
  <br>
  e-mail: cbitsudentrtransport@gmail.com</p>
<p>
  2. S. Sateesh Reddy (Transport Supervisor)
  <br>
  Ph: 99081 76475</p>
<p>
  The Students Transport Centre is located in the ground floor of K Block (near Canteen)
</p>

<!-- The Modal -->
<div id="myModal" class="modal">
  <span class="close">&times;</span>
  <img class="modal-content" id="img01">
  <div id="caption"></div>
```

Fig. 4.62: student module front page 3

Fig. 4.63: student module front page 4

- **Sidebar:** Figures 4.64 side bar allows student to reach the webpages of different records when clicked.



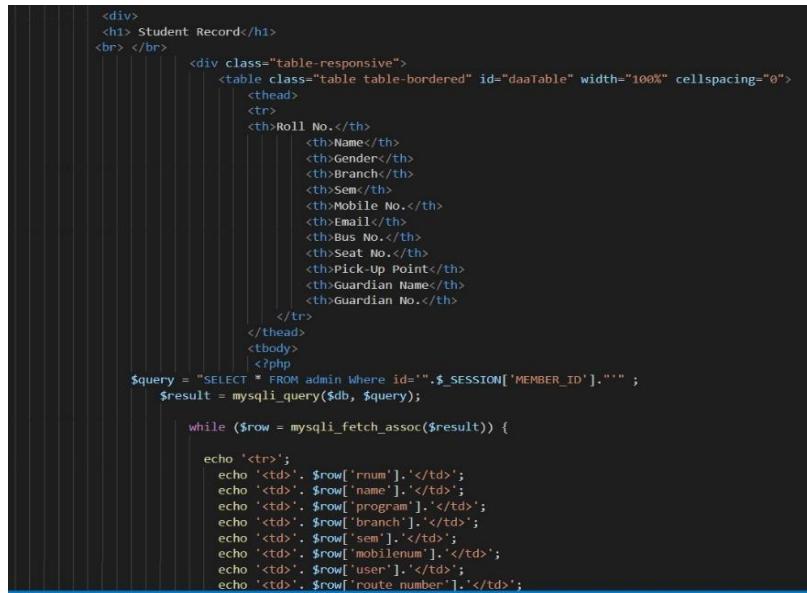
```

143     <ul class="sidebar navbar-nav" style="background-image: linear-gradient(to top, #42275a, #734b6d);>
144     <li class="nav-item">
145         <a class="nav-link" href="homepage.php">
146             <i class="fas fa-fw fa-home"></i>
147             <span>HOMEPAGE</span>
148         </a>
149     </li>
150     <li class="nav-item dropdown">
151         <a class="nav-link dropdown-toggle" href="#" id="pagesdropdown" role="button" data-toggle="dropdown" aria-haspopup="true" aria-expanded="false">
152             <i class="fas fa-fw fa-user-circle"></i>
153             <span>Student</span>
154         </a>
155         <div class="dropdown-menu" aria-labelledby="pagesdropdown">
156             <a class="dropdown-item" href="userstudent.php">My Info</a>
157             <a class="dropdown-item" href="userstudentfee.php">Fee Details</a>
158         </div>
159     </li>
160
161     <li class="nav-item">
162         <a class="nav-link" href="userbus.php">
163             <i class="fas fa-fw fa-bus"></i>
164             <span>Bus</span></a>
165     </li>
166     <li class="nav-item">
167         <a class="nav-link" href="userdriver.php">
168             <i class="fas fa-fw fa-users"></i>
169             <span>Driver</span></a>
170     </li>
171
172     </li>
173     <li class="nav-item">
174         <a class="nav-link" href="userroute.php">
175             <i class="fas fa-fw fa-route"></i>
176             <span>Route</span></a>
177         <li class="nav-item">
178             <a class="nav-link" href="usercsbox.php">
179

```

Fig. 4.64: student module sidebar code

**Student:** The Figures 4.65 and 4.66 shows code shows student's personal data.



```

<div>
<h1> Student Record</h1>
<br> </br>
<div class="table-responsive">
    <table class="table table-bordered" id="daaTable" width="100%" cellspacing="0">
        <thead>
            <tr>
                <th>Roll No.</th>
                <th>Name</th>
                <th>Gender</th>
                <th>Branch</th>
                <th>Sem.</th>
                <th>Mobile No.</th>
                <th>Email</th>
                <th>Bus No.</th>
                <th>Seat No.</th>
                <th>Pick-Up Point</th>
                <th>Guardian Name</th>
                <th>Guardian No.</th>
            </tr>
        </thead>
        <tbody>
            <?php
                $query = "SELECT * FROM admin Where id='".$_SESSION['MEMBER_ID']."' ";
                $result = mysqli_query($db, $query);
                while ($row = mysqli_fetch_assoc($result)) {
                    echo '<tr>';
                    echo '<td> ' . $row['rnum'] . '</td>';
                    echo '<td> ' . $row['name'] . '</td>';
                    echo '<td> ' . $row['program'] . '</td>';
                    echo '<td> ' . $row['branch'] . '</td>';
                    echo '<td> ' . $row['sem'] . '</td>';
                    echo '<td> ' . $row['mobilenum'] . '</td>';
                    echo '<td> ' . $row['user'] . '</td>';
                    echo '<td> ' . $row['route number'] . '</td> ';

```

Fig. 4.65: code to show student's personal details 1

Fig. 4.66: code to show student's personal details 2

**Student fee:** Figures 4.67 shows code shows student's fee details.

```
        <h1> Student Record</h1>
        <br> </br>
        <div class="table-responsive">
            <table class="table table-bordered" id="dataTable" width="100%" cellspacing="0">
                <thead>
                    <tr>
                        <th>Roll No.</th>
                        <th>Name</th>
                        <th>Mobile No.</th>
                        <th>Bus No.</th>
                        <th>Fee</th>
                        <th>Amount Paid</th>
                        <th>Payment Method</th>
                    </tr>
                </thead>
                <tbody>
                    <?php
$query = $query = "SELECT * FROM admin Where id='".$_SESSION['MEMBER_ID']."'"; ;
$result = mysqli_query($db, $query) or die (mysqli_error($db));
while ($row = mysqli_fetch_assoc($result)) {
                echo '<tr>';
                echo '<td>'. $row['rnum']. '</td>';
                echo '<td>'. $row['name']. '</td>';
                echo '<td>'. $row['mobilenumber']. '</td>';
                echo '<td>'. $row['route_number']. '</td>';
                echo '<td>'. $row['fee']. '</td>';
                echo '<td>'. $row['amount_paid']. '</td>';
                echo '<td>'. $row['pay_method']. '</td>';
                // echo '<td>'. $row['file_name']. '</td>';
                echo '</tr> ';
            }
        </?>
```

Fig. 4.67: code to show student's fee details

**Bus:** The Figures 4.68 shows code shows bus details in a tabular format.

```
<?php include 'connection.php';
include 'topnav.php'; >
    <div class="col-lg-12">
        <div>
            <h1>Bus Records </h1>
        </div>
        <br> </br>
        <div class="table-responsive">
            <table class="table table-bordered" id="dataable" width="100%" cellspacing="0">
<style>
tr:nth-child(even) {
    background-color: #D6EEEE;
}
</style>
<thead>
    <tr>
        <th>Route Number</th>
        <th>Route Name</th>
        <th>Driver ID</th>
        <th>Seating Capacity</th>
        <th>Occupancy</th>
    </tr>
</thead>
<tbody>
<?php
    $query = 'SELECT * FROM bus';
    $result = mysqli_query($db, $query) or die (mysqli_error($db));
    while ($row = mysqli_fetch_assoc($result)) {
        echo '<tr>';
        echo '<td>'. $row['ROUTE_NUMBER']. '</td>';
        echo '<td>'. $row['ROUTE_NAME']. '</td>';
        echo '<td>'. $row['DRIVER_ID']. '</td>';
        echo '<td>'. $row['SEATING_CAPACITY']. '</td>';
        echo '<td>'. $row['OCCUPANCY']. '</td>';
        echo '</tr>'; } ?>
</tbody>    </table>    </div>
<?php include 'footer.php'; ?>
```

Fig. 4.68: code to show bus details

**Driver:** The Figures 4.69 shows driver details in a tabular format.

```
<?php include 'connection.php';
include 'topnav.php'; >
    <div class="col-lg-12">
        <div>
            <h1>Driver Records </h1>
        </div>
        <br> </br>
        <div class="table-responsive">
            <table class="table table-bordered" id="dataable" width="100%" cellspacing="0">
<style>
tr:nth-child(even) {
    background-color: #D6EEEE;
}
</style>
<thead>
    <tr>
        <th>Driver ID</th>
        <th>Driver Name</th>
        <th>Bus Number</th>
    </tr>
</thead>
<tbody>
<?php
    $query = 'SELECT * FROM driver';
    $result = mysqli_query($db, $query) or die (mysqli_error($db));
    while ($row = mysqli_fetch_assoc($result)) {
        echo '<tr>';
        echo '<td>'. $row['DRIVER_ID']. '</td>';
        echo '<td>'. $row['DRIVER_NAME']. '</td>';
        echo '<td>'. $row['BUS_NUM']. '</td>';
        echo '</tr>'; } ?>
</tbody>    </table>    </div>
<?php include 'footer.php'; ?>
```

Fig. 4.69: code to show driver details

**Route:** The Figures 4.70 shows code shows bus route details in a tabular format.

```

<?php include 'connection.php';
include 'topnav.php'; ?>
<div class="col-lg-12">
<div>
<h1>Route Records </h1>
</div> <br> <br>
<div class="table-responsive">
<table class="table table-bordered" id="dataable" width="100%" cellspacing="0">
<thead>
<tr>
<th>Route Number</th>
<th>Route Name</th>
<th>Enroute Stages</th>
<th>Start Time</th>
<th>Departure</th>
</tr>
</thead>
<tbody>
<?php
$query = 'SELECT * FROM route';
$result = mysqli_query($db, $query) or die (mysqli_error($db));
while ($row = mysqli_fetch_assoc($result)) {
    echo '<tr>';
    echo '<td>'. $row['ROUTE_NUMBER']. '</td>';
    echo '<td>'. $row['ROUTE_NAME']. '</td>';
    echo '<td>'. $row['ENROUTE_STAGES']. '</td>';
    echo '<td>'. $row['START_TIME']. '</td>';
    echo '<td>'. $row['DEPARTURE']. '</td>';
    echo '</tr> ' ; } ?>
</tbody>
</table> </div>
<?php include 'footer.php'; ?>

```

Fig. 4.70: code to show route details

**Complaint/Suggestion box:** The Figures 4.71,4.72 and 4.73 shows code helps students to send complaints and suggestions regarding bus transport facility.

```

<?php include 'connection.php';
include 'topnav.php'; ?>
<!DOCTYPE html>
<html>
<head>
<meta name="viewport" content="width=device-width, initial-scale=1">
<style>
body {font-family: Arial, Helvetica, sans-serif;}
* {box-sizing: border-box;}
input[type=Rollnumber], select, textarea {
width: 100%;
padding: 12px;
border: 1px solid #ccc;
border-radius: 4px;
box-sizing: border-box;
margin-top: 6px;
margin-bottom: 16px;
resize: vertical;
}
input[type=submit] {
background-color: #04AA6D;
color: white;
padding: 12px 20px;
border: none;
border-radius: 4px;
cursor: pointer;
}
input[type=submit]:hover {
background-color: #45a049;
}
.container {
border-radius: 5px;
background-color: #f2f2f2;
padding: 20px;
}
</style>
</head>
<body>

```

```

<h3>Complaint/Suggestion Box</h3>
<div class="container">
<form action="usersboxadd.php?action=add" method="post">
<label for="Rollnumber">Rollnumber</label>
<input type="Rollnumber" id="Rollnumber" name="Rollnumber" placeholder="Your rollnumber..">

<label for="Type">Type</label>
<select id="Type" name="type">
<option value="Suggestion">Suggestion</option>
<option value="Complaint">Complaint</option>
</select>

<label for="Subject">Subject</label>
<textarea id="Subject" name="Subject" placeholder="Write something.." style="height:200px"></textarea>

<input type="submit" value="Submit">
</form>
</div>
</body>
</html>
<?php include 'footer.php'; ?>

```

Fig. 4.71: code to style complaint/suggestion webpage

Fig. 4.72: code to send a complaint/suggestion

```

<?php include'header.php' ;?>
<?php include'connection.php' ;?>

<div class="col-lg-12">
    <?php
        $csrollnum= $_POST['Rollnumber'];
        $cstype= $_POST['Type'];
        $cssub= trim($_POST['Subject']);

        $query = "INSERT INTO csbox
        (Rollnumber,Type,Subject)
        VALUES ('".$csrollnum."','".$cstype."','".$cssub."')";
        mysqli_query($db,$query)or die (mysqli_error($db));
    ?>
    <script type="text/javascript">
        alert("Successfully sent.");
        window.location = "complaint.php";
    </script>
</div>

```

Fig. 4.73: code to save complaint/suggestion into database

## RESULT:

- **Login page:** The figures 4.74 helps students and admin to log into the student bus transport management website.

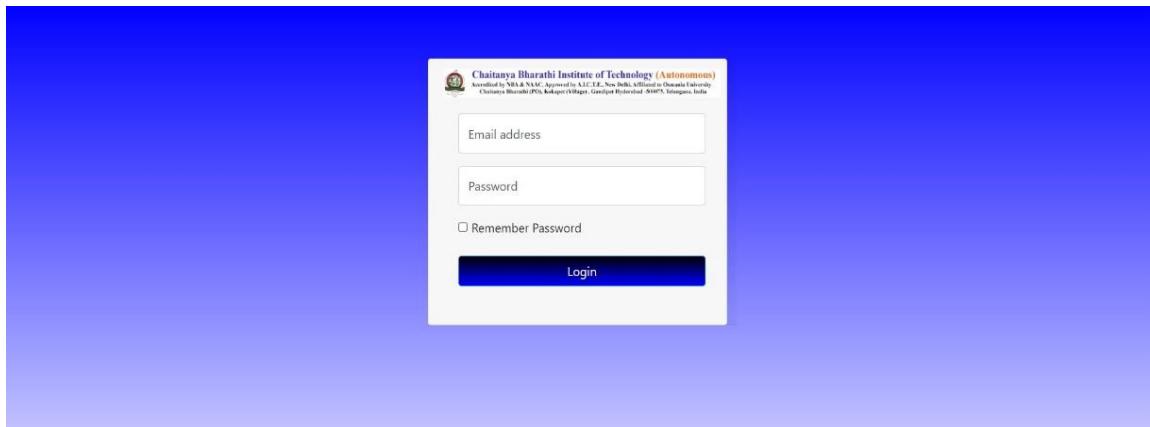


Fig.4.74: Login page

## ADMIN:

**Front page:** The front page of the admin module contains a sidebar, drop down box and footer consisting of transport contact details. Figure 4.75 shows us the frontpage.

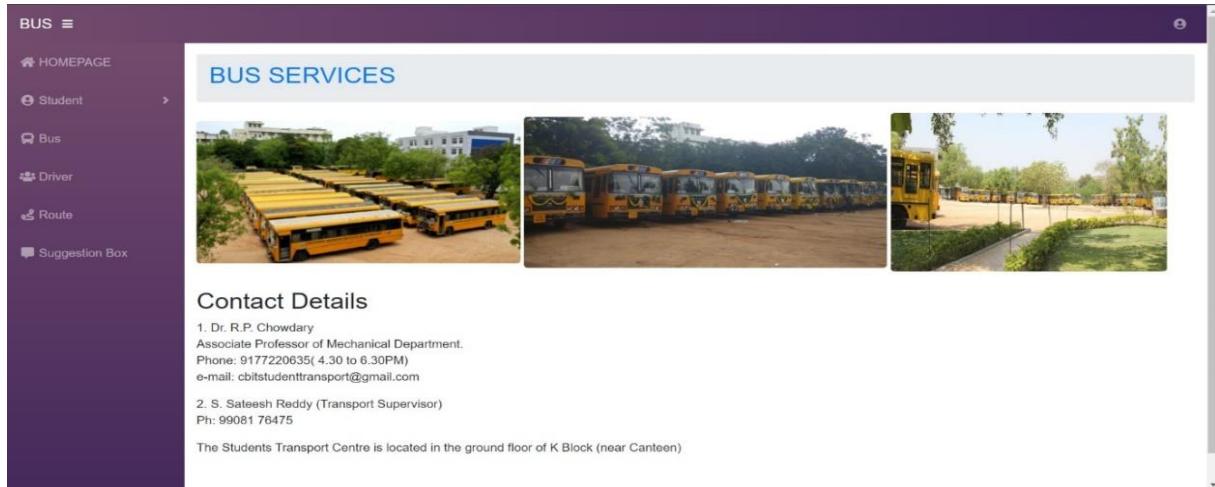


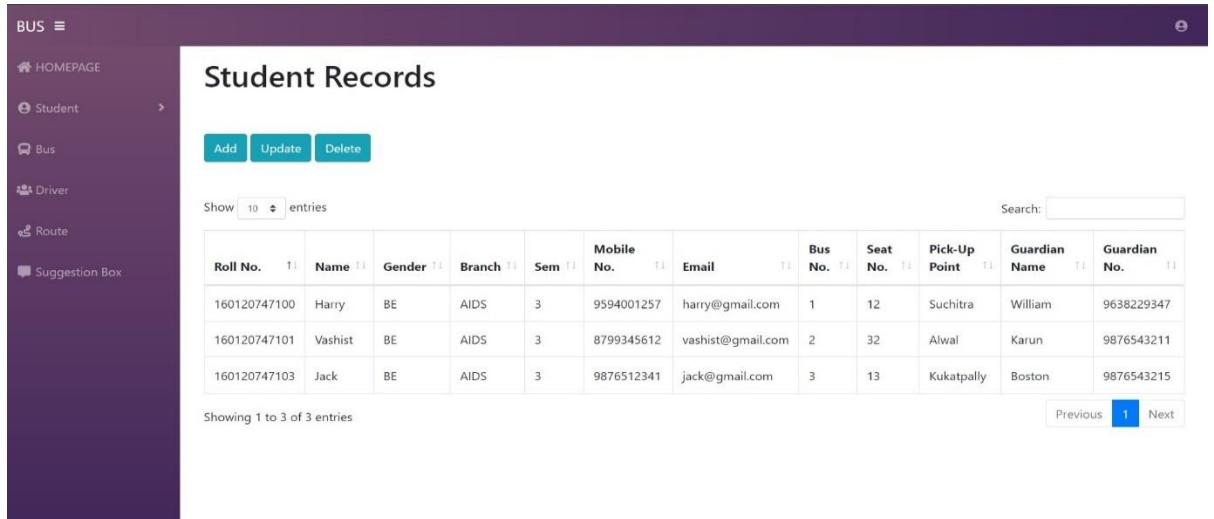
Fig.4.75: Front page

- **Register page:** The figure 4.76 helps in creating a page where students availing bus transport can be registered and their details stored in admin db.

A screenshot of a registration form for 'Chaitanya Bharathi Institute of Technology (Autonomous)'. The header includes the institute's name and accreditation information: 'Accredited by NBA & NAAC. Approved by A.I.C.T.E., New Delhi. Affiliated to Osmania University. Chaitanya Bharathi (P.O), Kokapet (Village), Gandipet Hyderabad -500075, Telangana, India'. The form consists of several input fields arranged in a grid: Rollnumber, Name, Program, Branch, Sem, Mobile Number, Gender, Pickup Point, Guardian Name, Guardian number, Email address, Password, and Confirm password. A large blue 'Register' button is at the bottom.

Fig.4.76: Register page

- **Student:** The figures 4.77 shows student records, the student database is stored in this webpage.



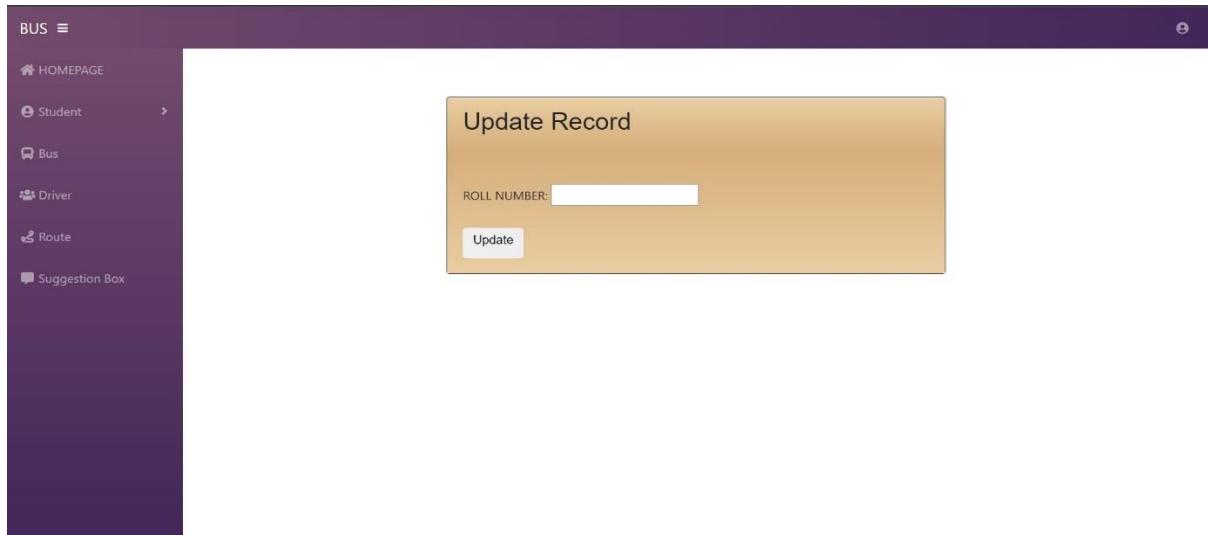
The screenshot shows a 'Student Records' page with a table of student data. The table has columns for Roll No., Name, Gender, Branch, Sem, Mobile No., Email, Bus No., Seat No., Pick-Up Point, Guardian Name, and Guardian No. The data in the table is as follows:

Roll No.	Name	Gender	Branch	Sem	Mobile No.	Email	Bus No.	Seat No.	Pick-Up Point	Guardian Name	Guardian No.
160120747100	Harry	BE	AIDS	3	9594001257	harry@gmail.com	1	12	Suchitra	William	9638229347
160120747101	Vashist	BE	AIDS	3	8799345612	vashist@gmail.com	2	32	Alwal	Karun	9876543211
160120747103	Jack	BE	AIDS	3	9876512341	jack@gmail.com	3	13	Kukatpally	Boston	9876543215

Showing 1 to 3 of 3 entries

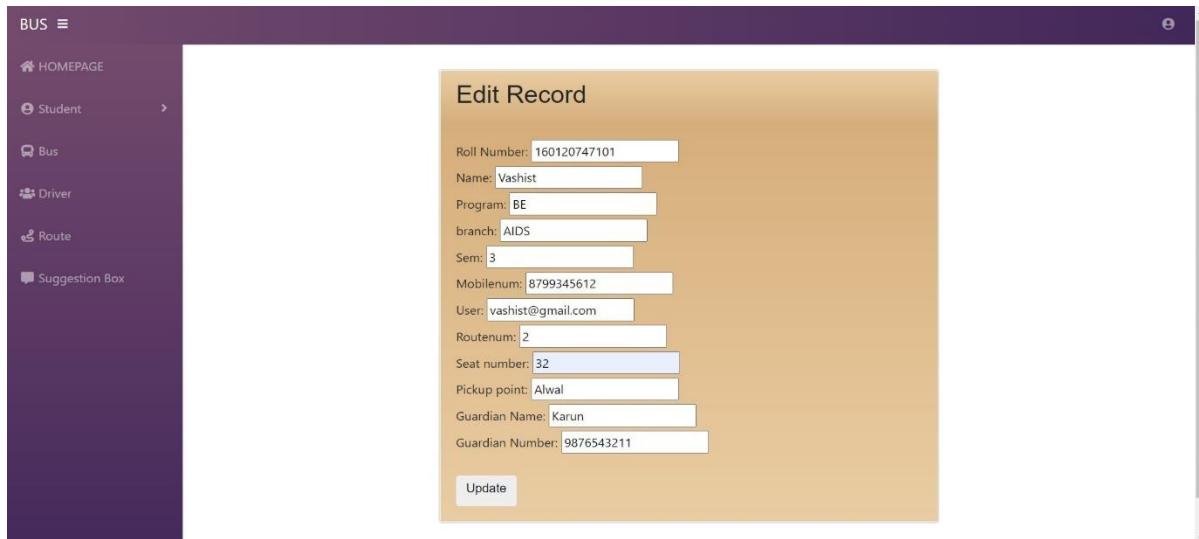
Fig.4.77: Student Front page

1. **Student Update:** The figures 4.78 and 4.79 shows update record, the student database is updated in this webpage.



The screenshot shows a 'Update Record' page. It has a form with a 'ROLL NUMBER:' input field and an 'Update' button.

Fig.4.78: Student Update page



The screenshot shows a web application interface for managing student records. On the left is a dark purple sidebar with a navigation menu:

- HOME PAGE
- Student
- Bus
- Driver
- Route
- Suggestion Box

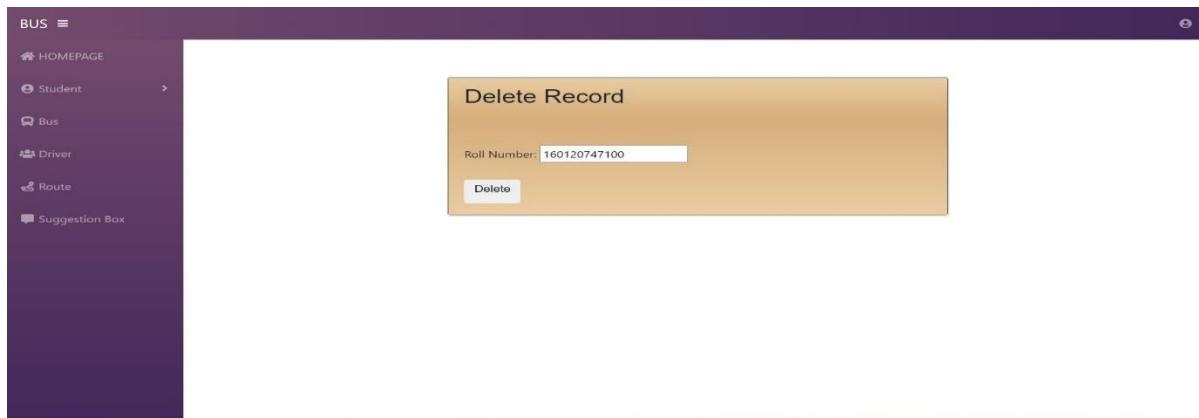
The main content area has a light orange background and is titled "Edit Record". It contains a form with the following fields and their values:

Roll Number:	160120747101
Name:	Vashist
Program:	BE
branch:	AIDS
Sem:	3
Mobilenum:	8799345612
User:	vashist@gmail.com
Route num:	2
Seat number:	32
Pickup point:	Alwal
Guardian Name:	Karun
Guardian Number:	9876543211

At the bottom of the form is a "Update" button.

Fig.4.79: Student Update page1

2. Student Delete: The figures 4.80 shows delete records, the student database is deleted in this webpage.



The screenshot shows a web application interface for managing student records. On the left is a dark purple sidebar with a navigation menu:

- HOME PAGE
- Student
- Bus
- Driver
- Route
- Suggestion Box

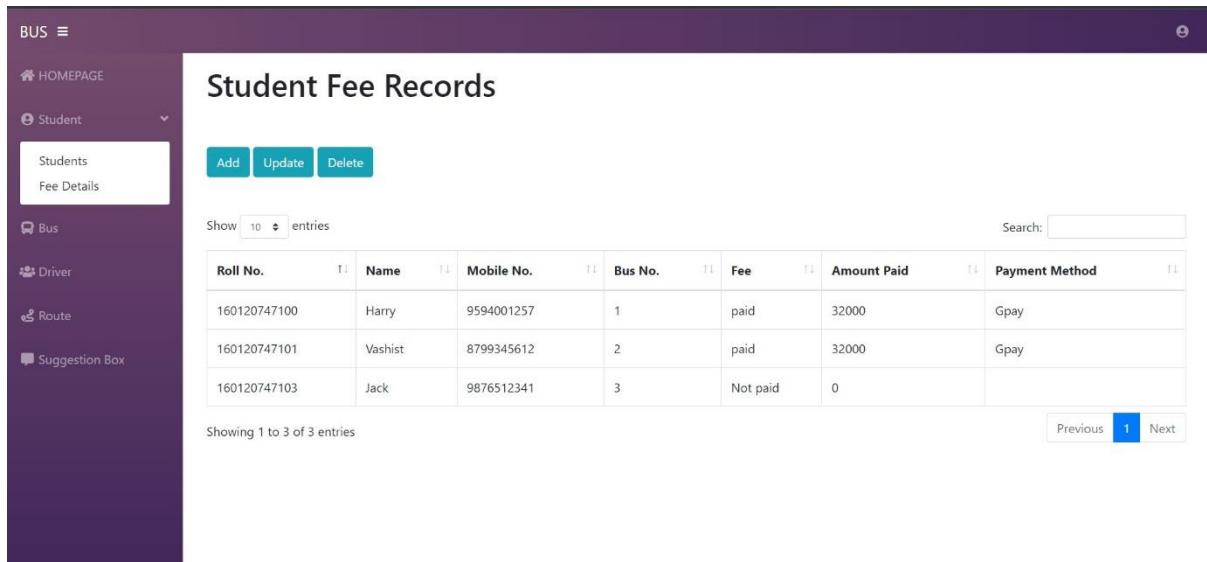
The main content area has a light orange background and is titled "Delete Record". It contains a form with a single field:

Roll Number:	160120747100
--------------	--------------

At the bottom of the form is a "Delete" button.

Fig.4.80: Student delete page

- Student Fee records: The figure 4.81 shows student fee records, the student fee database is stored in this webpage.



**Student Fee Records**

Add Update Delete

Show 10 entries Search:

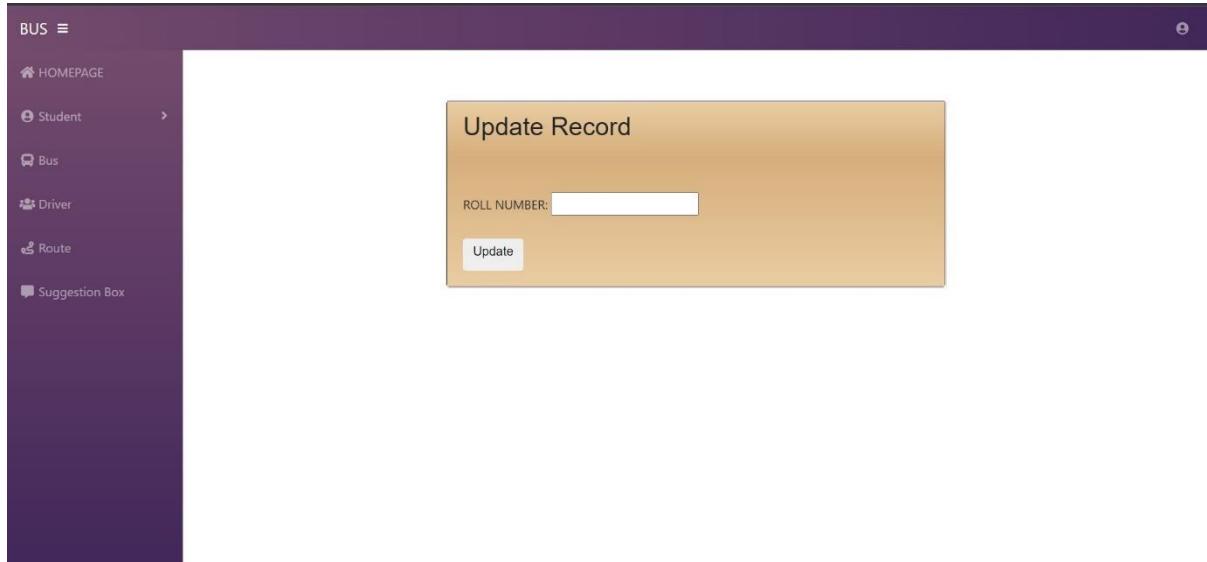
Roll No.	Name	Mobile No.	Bus No.	Fee	Amount Paid	Payment Method
160120747100	Harry	9594001257	1	paid	32000	Gpay
160120747101	Vashist	8799345612	2	paid	32000	Gpay
160120747103	Jack	9876512341	3	Not paid	0	

Showing 1 to 3 of 3 entries

Previous 1 Next

Fig.4.81: Student Fee Front page

1. Student fee update: The figures 4.82 and 4.83 shows student fee records, the student fee database is updated in this webpage.



**Update Record**

ROLL NUMBER:

Update

Fig.4.82: Student Fee Update page

Roll Number: 160120747103  
Name: Jack  
Mobilenum: 9876512341  
Routenum: 3  
Fee: Not paid  
Amount Paid:  
Pay Method:  
File Name:  
Update

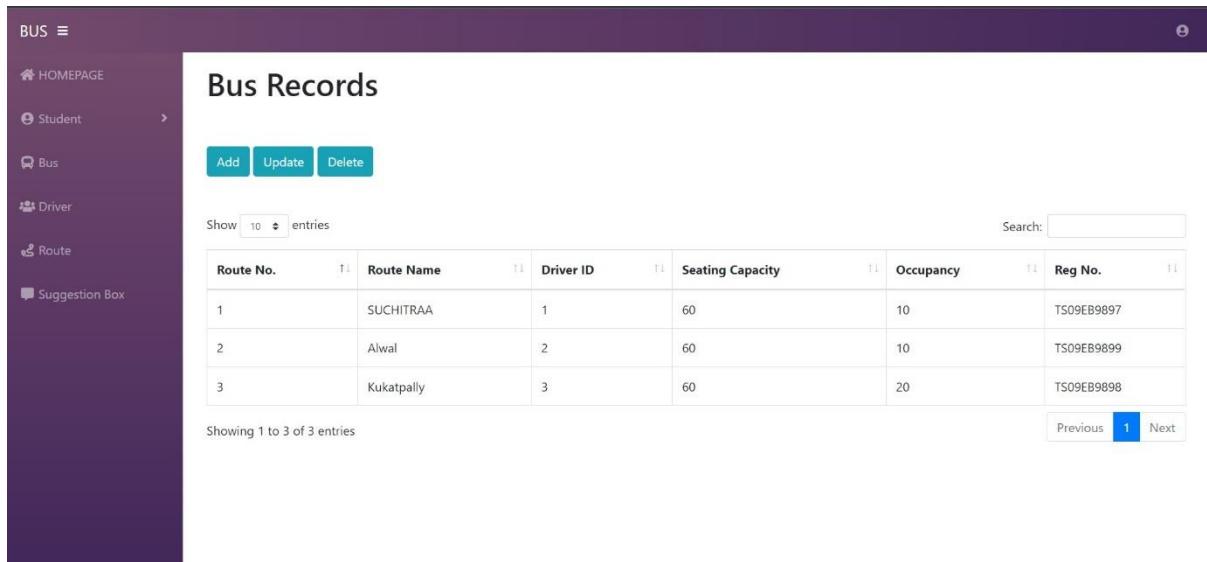
Fig.4.83: Student Fee Update page1

2. Student Fee Delete: The figure 4.84 shows student fee records, the student database is deleted in this webpage.

Roll Number: 160120747100  
Delete

Fig.4.84: Student Fee delete page1

- **Bus:** The figures 4.85 shows bus records, the bus database is stored in this webpage.



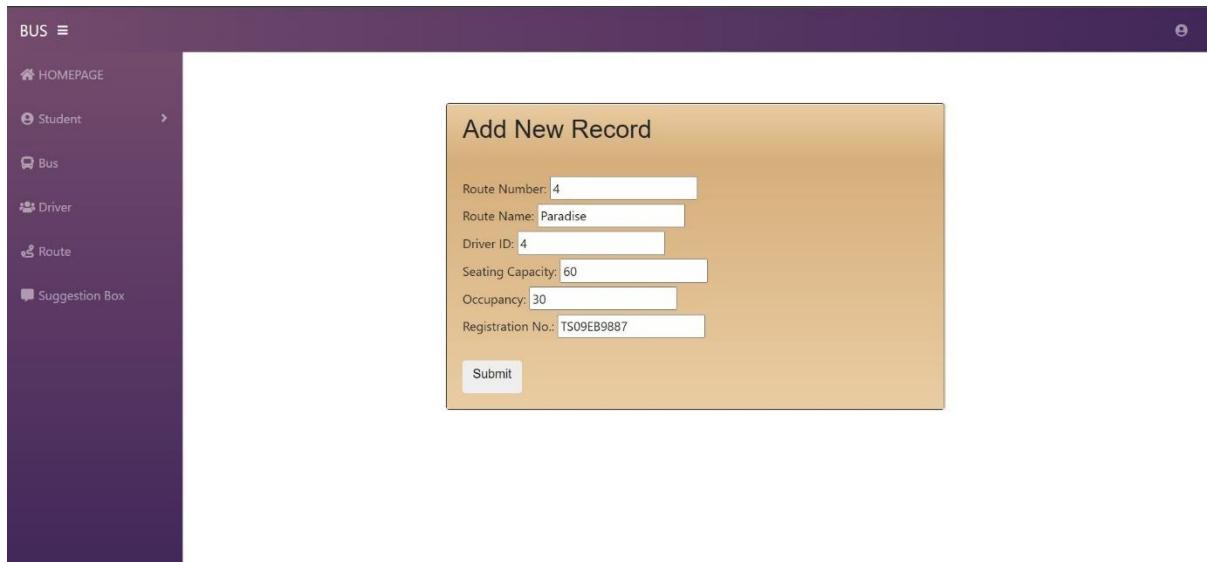
The screenshot shows a web application interface for managing bus records. The left sidebar has links for HOMEPAGE, Student, Bus, Driver, Route, and Suggestion Box. The main content area is titled 'Bus Records' and contains a table of bus routes. The table has columns: Route No., Route Name, Driver ID, Seating Capacity, Occupancy, and Reg No. The data in the table is as follows:

Route No.	Route Name	Driver ID	Seating Capacity	Occupancy	Reg No.
1	SUCHITRAA	1	60	10	TS09EB9897
2	Alwal	2	60	10	TS09EB9899
3	Kukatpally	3	60	20	TS09EB9898

Showing 1 to 3 of 3 entries

Fig.4.85: Bus Front page

1. **Bus Add:** The figures 4.86 shows bus add page, the bus details can be stored in the database.



The screenshot shows a 'Add New Record' form within a modal window. The form contains the following fields:

- Route Number: 4
- Route Name: Paradise
- Driver ID: 4
- Seating Capacity: 60
- Occupancy: 30
- Registration No.: TS09EB9887

At the bottom of the form is a 'Submit' button.

Fig.4.86: Bus Add page

2. Bus update: The figure 4.87 and 4.88 shows bus update records, the bus database is stored in this webpage.

Update Record

ROUTE NUMBER: 1

Update

Fig.4.87: Bus Update page

Edit Record

ROUTE NUMBER: 1

ROUTE NAME: SUCHITRAA

DRIVER: 1

SEATING CAPACITY: 60

OCCUPANCY: 10

REGISTRATION NUMBER: TS09EB9897

Update

Fig.4.88: Bus Update page1

3. Bus Delete: The figures 4.89 shows bus delete records, the bus database is deleted in this webpage.

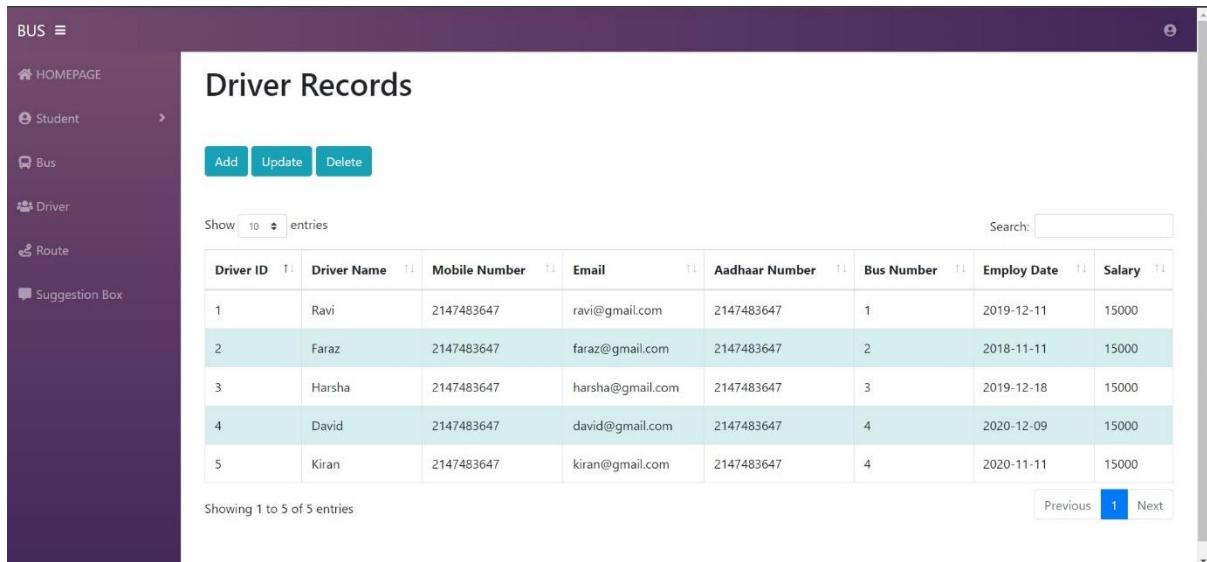
Delete Record

ROUTE NUMBER: 1

Delete

Fig.4.89: Bus Delete page

- **Driver:** The figures 4.90 shows delete records, the delete database is stored in this webpage.



**Driver Records**

Add Update Delete

Show 10 entries Search:

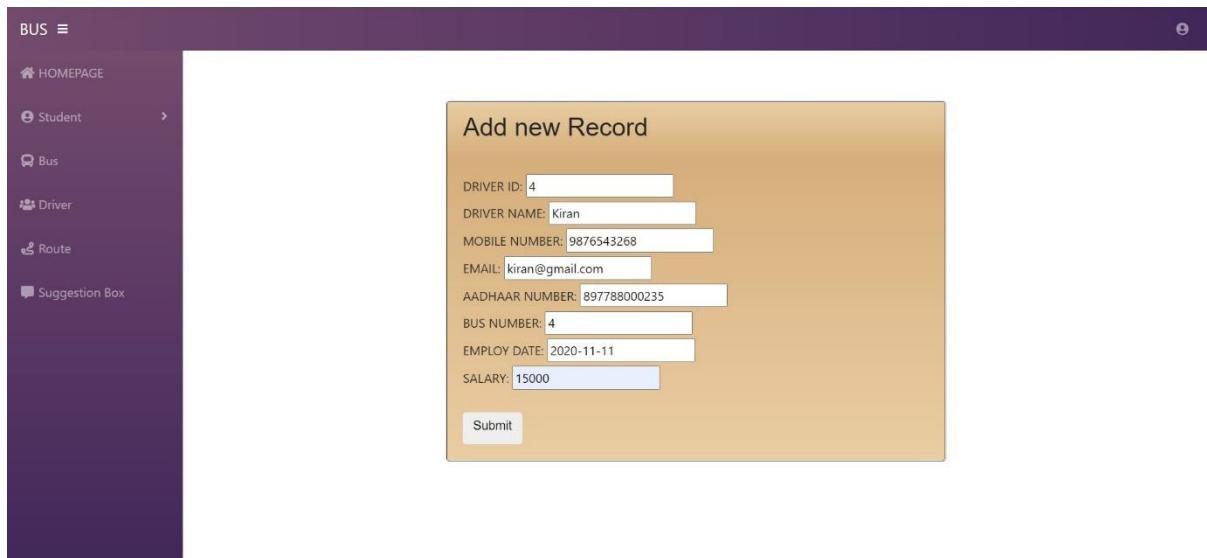
Driver ID	Driver Name	Mobile Number	Email	Aadhaar Number	Bus Number	Employ Date	Salary
1	Ravi	2147483647	ravi@gmail.com	2147483647	1	2019-12-11	15000
2	Faraz	2147483647	faraz@gmail.com	2147483647	2	2018-11-11	15000
3	Harsha	2147483647	harsha@gmail.com	2147483647	3	2019-12-18	15000
4	David	2147483647	david@gmail.com	2147483647	4	2020-12-09	15000
5	Kiran	2147483647	kiran@gmail.com	2147483647	4	2020-11-11	15000

Showing 1 to 5 of 5 entries

Previous 1 Next

Fig.4.90: Driver Front page

1. Driver Add: The figures 4.91 shows driver records



**Add new Record**

DRIVER ID:

DRIVER NAME:

MOBILE NUMBER:

EMAIL:

AADHAAR NUMBER:

BUS NUMBER:

EMPLOY DATE:

SALARY:

Submit

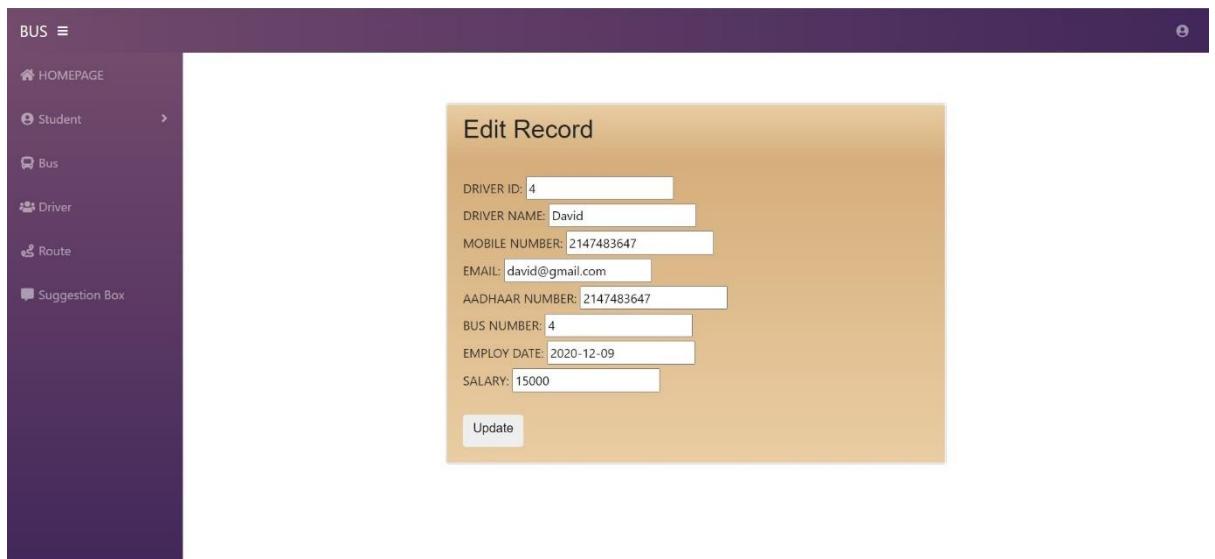
Fig.4.91: Driver Add page

2. Driver Update: The figures 4.92 and 4.93 shows update records, the updated database is stored in this webpage.



The screenshot shows a dark purple header with the word 'BUS' and a menu bar on the left containing 'HOMEPAGE', 'Student', 'Bus', 'Driver', 'Route', and 'Suggestion Box'. The main content area has a light orange background and a title 'Update Record'. It contains a single input field labeled 'DRIVER ID' with the value '3' and a 'Update' button below it.

Fig.4.92: Driver Update page



The screenshot shows a dark purple header with the word 'BUS' and a menu bar on the left containing 'HOMEPAGE', 'Student', 'Bus', 'Driver', 'Route', and 'Suggestion Box'. The main content area has a light orange background and a title 'Edit Record'. It contains several input fields for updating a driver record, including 'DRIVER ID' (4), 'DRIVER NAME' (David), 'MOBILE NUMBER' (2147483647), 'EMAIL' (david@gmail.com), 'AADHAAR NUMBER' (2147483647), 'BUS NUMBER' (4), 'EMPLOY DATE' (2020-12-09), and 'SALARY' (15000). There is also an 'Update' button at the bottom.

Fig.4.93: Driver Update page1

3. Driver Delete: The figures 4.94 shows driver delete records, the driver database is deleted in this webpage.

Fig.4.94: Driver Delete page

- **Route:** The figures 4.95 shows Route records, the route database is stored in this webpage.

Route Number	Route Name	Enroute Stages	Start Time	Departure
1	Suchitra	Jeedimetla,suchitra,mmr gardens	07:30:00	06:30:00
2	Alwal	Alwal,Luthkunta	07:30:00	06:30:00
3	Kukatpally	Kukatpally,Hitech city,Shilparamam	07:30:00	06:30:00

Fig.4.95: Route Front page

1. Route Add: The figures 4.96 shows route records, the route database is stored in this webpage.

ROUTE NUMBER: 2  
 ROUTE NAME: Alwal  
 ENROUTE STAGES: Alwal,Luthkunta  
 START TIME: 7:30  
 DEPARTURE: 6:30

Submit

Fig.4.96: Route Add page

2. Route Update: The figures 4.97 and 4.98 shows route records, the route database is stored in this webpage.

ROUTE NUMBER: 1

Update

Fig.4.97: Route Update page

ROUTE NUMBER: 1  
 ROUTE NAME: Suchitra  
 ENROUTE STAGES: Jeedimetla,suchitra,mmr  
 START TIME: 07:30:00  
 DEPARTURE: 06:30:00

Update

Fig.4.98: Route Update page1

3. Route Delete: The figures 4.99 shows route records, the route database is deleted in this webpage.

Fig.4.99: Route Delete page

- **Complaint/Suggestion Box:** The figures 4.100 shows complaints and suggestion records, the complaints and suggestion database is stored in this webpage.

Fig.4.100: Complaint/Suggestion page

## STUDENT:

- **Homepage:** The front page of the student module contains a sidebar, drop down box and footer consisting of transport contact details. Figure 4.101 shows us the frontpage

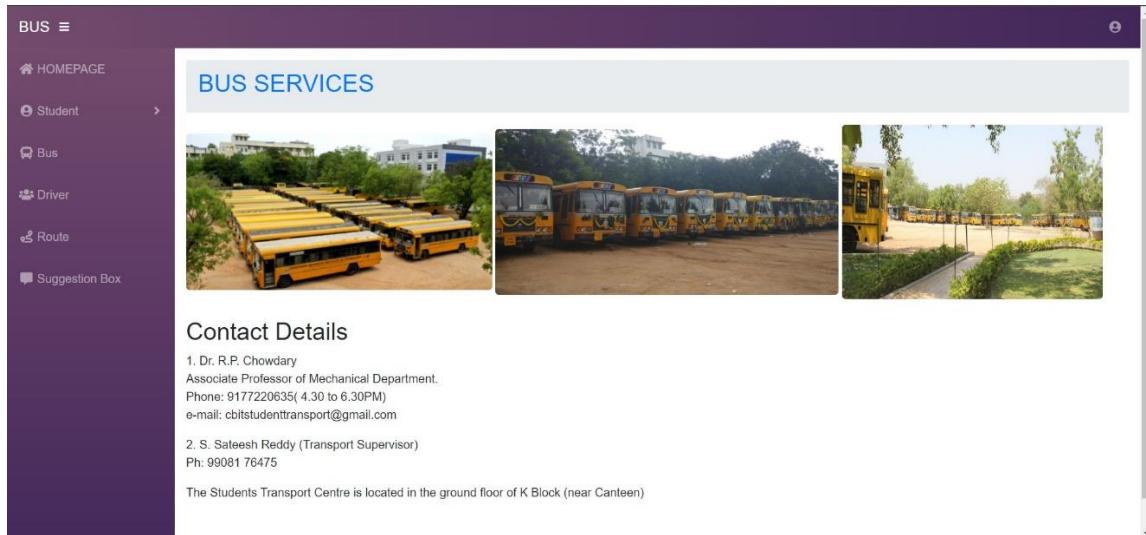


Fig.4.101: Homepage

- **Student:** The figures 4.102 and 4.103 shows student records and student fee records.

The screenshot shows the 'Student Record' page. The sidebar on the left includes links for HOMEPAGE, Student (selected), Bus, Driver, Route, and Suggestion Box. A sub-menu for 'Student' is open, showing 'My Info' and 'Fee Details'. The main content area displays a table titled 'Student Record' with the following data:

Roll No.	Name	Gender	Branch	Sem	Mobile No.	Email	Bus No.	Seat No.	Pick-Up Point	Guardian Name	Guardian No.
160120747100	Harry	BE	AIDS	3	9594001257	harry@gmail.com	1	12	Suchitra	William	9638229347

Fig.4.102: Student page

Roll No.	Name	Mobile No.	Bus No.	Fee	Amount Paid	Payment Method
160120747100	Harry	9594001257	1	paid	32000	Gpay

Fig.4.103: Student Fee page

- **Bus:** The figures 4.104 shows bus records

Route Number	Route Name	Driver ID	Seating Capacity	Occupancy
1	SUCHITRAA	1	60	10
2	Alwal	2	60	10
3	Kukatpally	3	60	20
4	Paradise	4	60	30

Fig.4.104: Bus page

- **Driver:** The figures 4.105 shows driver records.

Driver ID	Driver Name	Bus Number
1	Ravi	1
2	Faraz	2
3	Harsha	3
4	David	4
5	Kiran	4

Fig.4.105: Driver page

- **Route:** The figures 4.106 shows route records

CBIT BUS TRANSPORT 

 HOMEPAGE

 Student >

 Bus

 Driver

 Route

 Suggestion Box

## Route Records

Show: 10 entries Search:

Route Number	Route Name	Enroute Stages	Start Time	Departure
1	Suchitra	Jeedimetla,suchitra,mmr gardens	07:30:00	06:30:00
2	Alwal	Alwal,Luthkunta	07:30:00	06:30:00
3	Kukatpally	Kukatpally,Hitech city,Shilparamam	07:30:00	06:30:00

Showing 1 to 3 of 3 entries

Previous 1 Next

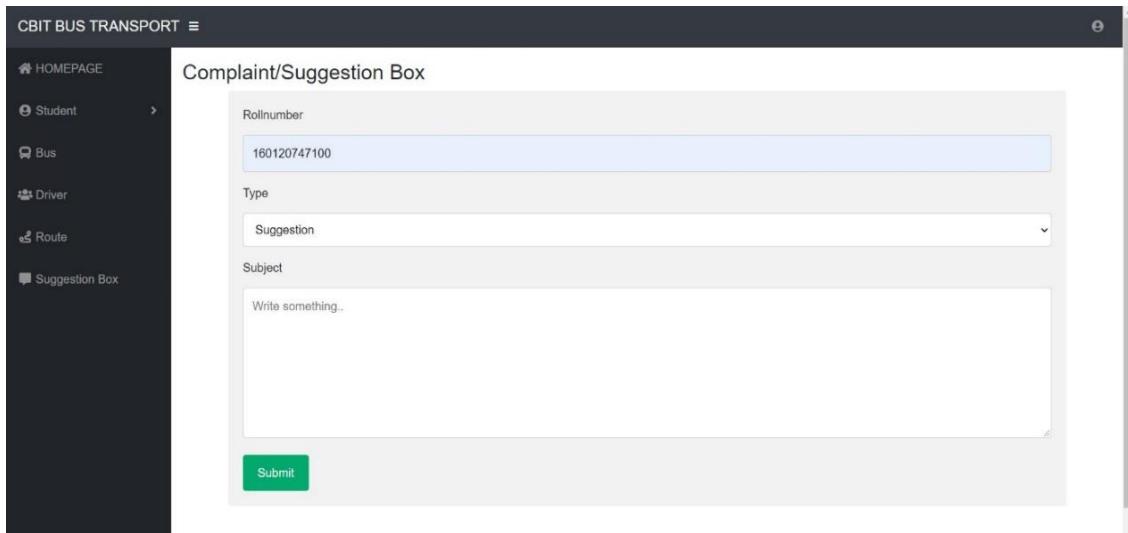
Fig.4.106: Route page

If a User clicks on Route Records (which is highlighted in blue), they are redirected to route pdf

TIME-TABLE			
Route-14(BOD UPPAL)		Route – 19 (UPPAL)	
1	BOD Uppal	Uppal Dapor	7.10AM
2	Uppal x roads	RTC bus stop	7.25AM
3	Survey of India	Opp. to Survey of India	7.28
4	Habiguda	Street no. 8 x roads	7.30
5	10 x roads	Opp. to 10 x roads	7.35
6	Vidyasagar (SVES)	Bus stop (Andhra Mahilaashabha)	7.45
7	Narayana	Vidyal book stall	7.50
8	Fever hospital	Chrysotis fever hospital	7.52
9	Barkampura	Raghavendra swamy temple road	7.57
10	Narayanguda	Agrawali	8.00
11	Himayath nagar	Food world	8.05
12	Lakdi ka pool	Dwarka hotel	8.10
13	Masab Tank	Near Arrempur bus stop	8.13
14	NMDC	After Madhu sweets	8.15
15	Mehdipatnam	Pillar No.21 & 22	8.20
16	Golconda Nagar	Olive hospital	8.25
17	CHIT		8.50
1		Uppal	7.55AM
2		Modi n Bakery	Modi Bakery Bus stop
3		Church	Church Gate
4		Ramnathapuram	Opp. Public School
5		Amberpet	Opp. to 10 x Studio
6		Amberpet	Iron Hotel
7		No.	Domino's
8		Syndicate bank	at the bank
9		Vidyasagar (SVES)	Bus stop
10		10 x roads	Vidyal book stall
11		Fever hospital	Chrysotis fever hospital
12		Barkampura	Raghavendra swamy temple road
13		Narayanguda	Agrawali sweets
14		Himayath nagar	Food world
15		Lakdi ka pool	Dwarka hotel
16		Masab Tank	Near Arrempur bus stop
17		NMDC	RTC bus stop
		Mehdipatnam	Pillar No.21 & 22
		CHIT	8.50

Fig.4.107: Route pdf

- **Complaint/Suggestion box:** The figures 4.108 shows complaint/suggestion record



CBIT BUS TRANSPORT

Complaint/Suggestion Box

Rollnumber

160120747100

Type

Suggestion

Subject

Write something...

Submit

Fig.4.108: Complaint/Suggestion page

## 5. CONCLUSION AND FUTURE SCOPE

It is a known fact that works like student bus transport management when handled manually cause a lot of complications while using it. With the use of this website, these complications can be handled and tasks related to bus transport can be done easily and efficiently. Information of students, buses, drivers, bus routes can be viewed with ease by admins and students. Admins are also able to perform tasks like adding a record or deleting, updating a record easily. Data stored is consistent. Students can also bring up complaints and suggestions regarding the transport facility. We've aimed at this and have been successful in achieving it.

This project is only a humble venture satisfying few needs of student bus transport management. There are various other features that can be included in this project to make it a more efficient and user-friendly website.

- Bus reports can be generated.
- Bus passes can be generated for each student.
- Bus fee transactions can be done online.
- Bus attendance can be recorded.
- Announcements can be declared in the site itself.
- Alerts can be given for when the admin receives a new suggestion/complaint or when students receives a new announcement.

## BIBLIOGRAPHY

1. [https://youtu.be/aGF\\_gtOJMuE](https://youtu.be/aGF_gtOJMuE)
2. [https://stackoverflow.com/questions/39044793/php-displaying-user-info-when-loggedin#:~:text=if%20\(isset\(%24\\_POST%5B,\)%3B%20%2F%2FChecking%20is%20user](https://stackoverflow.com/questions/39044793/php-displaying-user-info-when-loggedin#:~:text=if%20(isset(%24_POST%5B,)%3B%20%2F%2FChecking%20is%20user)
3. [https://www.w3schools.com/sql/sql\\_autoincrement.asp](https://www.w3schools.com/sql/sql_autoincrement.asp)
4. <https://uigradients.com/#Purplin>
5. <https://stackoverflow.com/questions/15786937/add-image-in-title-bar>
6. <https://www.cluemediator.com/how-to-validate-password-strength-in-php>
7. <https://www.tutorialspoint.com/php/index.htm>
8. <https://www.w3schools.com/css/>
9. <https://www.wdb24.com/how-to-store-textarea-value-in-database-in-php/>
10. [https://www.w3schools.com/html/html\\_table\\_styling.asp](https://www.w3schools.com/html/html_table_styling.asp)
11. [https://www.w3schools.com/howto/howto\\_css\\_contact\\_form.asp](https://www.w3schools.com/howto/howto_css_contact_form.asp)
12. [https://www.w3schools.com/bootstrap/bootstrap\\_buttons.asp](https://www.w3schools.com/bootstrap/bootstrap_buttons.asp)