

Automatic Time Table Generator -For Faculty

An Application-I Development Project Report Submitted
In partial fulfillment of the requirement for the award of the degree of

Bachelor of Technology In Computer Science and Engineering -Artificial Intelligence and Machine Learning

by

R. Koushik Maharushi	-	22N31A66F3
V. Sreeja	-	22N31A66J0
P. Tharun	-	22N31A66E2
S. Vennela	-	22N31A66J4

Under the Guidance of
Dr. P. Hari Krishna M.Tech PhD
Associate Professor
Computational Intelligence Department
MRCET



**DEPARTMENT OF COMPUTATIONAL INTELLIGENCE
MALLA REDDY COLLEGE OF ENGINEERING AND TECHNOLOGY**
(Affiliated to JNTU, Hyderabad)
ACCREDITED by AICTE-NBA
Maisammaguda, Dhulapally post, Secunderabad-500014.
2024-2025

DECLARATION

I hereby declare that the project entitled “**Automatic Time Table Generator - For Faculty**” submitted to **Malla Reddy College of Engineering and Technology**, affiliated to **Jawaharlal Nehru Technological University Hyderabad (JNTUH)** for the award of the degree of **Bachelor of Technology in Computer Science and Engineering- Artificial Intelligence and Machine Learning** is a result of original research work done by me.

It is further declared that the project report or any part thereof has not been previously submitted to any University or Institute for the award of degree or diploma.

R. Koushik Maharushi (22N31A66F3)

P. Tharun (22N31A66E2)

S. Vennela (22N31A66J4)

V. Sreeja (22N31A66J0)



MALLA REDDY COLLEGE OF ENGINEERING & TECHNOLOGY

(AUTONOMOUS INSTITUTION - UGC, GOVT. OF INDIA)

Affiliated to JNTUH; Approved by AICTE, NBA-Tier 1 & NAAC with A-GRADE | ISO 9001:2015

CERTIFICATE

This is to certify that this is the bonafide record of the project titled “**Automatic Time Table Generator - For Faculty**” submitted by **R. Koushik Maharushi (22N31A66F3), V. Sreeja (22N31A66J0), S. Vennela (22N31A66J4), P. Tharun (22N31A66E2)** of B.Tech in the partial fulfillment of the requirements for the degree of **Bachelor of Technology in Computer Science and Engineering- Artificial Intelligence and Machine Learning**, Dept. of CI during the year 2024-2025. The results embodied in this project report have not been submitted to any other university or institute for the award of any degree or diploma.

Dr. P. Hari Krishna M.Tech, PhD
Associate Professor

INTERNAL GUIDE

Dr. D. Sujatha M.Tech, PhD
Professor

HEAD OF THE DEPARTMENT

EXTERNAL EXAMINER

ACKNOWLEDGEMENT

We feel honored and privileged to place our warm salutation to our college Malla Reddy College of Engineering and technology (UGC-Autonomous), our Director ***Dr. VSK Reddy*** who gave us the opportunity to have experience in engineering and profound technical knowledge.

We are indebted to our Principal ***Dr. S. Srinivasa Rao*** for providing us with facilities to do our project and his constant encouragement and moral support which motivated us to move forward with the project.

We would like to express our gratitude to our Head of the Department ***Dr. D. Sujatha***, Professor for encouraging us in every aspect of our system development and helping us realize our full potential.

We would like to thank our Application Development-I coordinator as well as our internal guide ***Dr. P. Hari Krishna***, Associate Professor, for his structured guidance and never-ending encouragement. We are extremely grateful for their valuable suggestions and unflinching co-operation throughout Application Development-I work.

We would also like to thank all supporting staff of department of CI and all other departments who have been helpful directly or indirectly in making our Application Development-I a success.

We would like to thank our parents and friends who have helped us with their valuable suggestions and support has been very helpful in various phases of the completion of the Application Development-I.

By:

R. Koushik Maharushi (22N31A66F3)

P. Tharun (22N31A66E2)

V. Sreeja (22N31A66J0)

S. Vennela (22N31A66J4)

ABSTRACT

The *Automatic Time Table Generator - For Faculty* is a software tool developed to simplify and improve scheduling within academic institutions. By automating timetable generation based on specific constraints and preferences, it reduces manual effort and minimizes common scheduling errors, allowing institutions to create balanced, conflict-free schedules more efficiently. A user-friendly interface enables faculty members to view their schedules and adapt to changes as needed. The streamlined approval process ensures that requests are managed promptly, supporting both academic requirements and individual faculty needs. Administrators, like Heads of Departments (HODs), have hierarchical access, allowing them to oversee and manage the schedules of all faculty within their departments. This feature helps optimize resource allocation and ensures comprehensive class coverage. The system also generates reports on past scheduling data, enabling HODs to analyse trends and make informed decisions for future planning. For students, a consistent, well-organized timetable means fewer disruptions and improved course accessibility, enhancing their educational experience. Overall, the *Automatic Time Table Generator* not only reduces administrative workload but also creates a flexible, responsive academic environment. This tool supports improved operational efficiency, transparency, and adaptability, benefiting faculty, administrators, and students alike.

TABLE OF CONTENTS

S.No.	Topic	Page No.
1	INTRODUCTION.....	1
	1.1 Purpose.....	1
	1.2 Background of project.....	1
	1.3 Scope of project.....	2
	1.4 Project features.....	2
2	SYSTEM REQUIREMENTS.....	4
	2.1 Hardware Requirements.....	4
	2.2 Software Requirements.....	4
	2.3 Existing System.....	4
	2.4 Proposed System.....	6
3	SYSTEM DESIGN.....	7
	3.1 System Architecture.....	7
	3.2 UML Diagram.....	7
4	IMPLEMENTATION.....	11
	4.1 Source Code.....	11
	4.2 Output Screens.....	21
5	CONCLUSION & FUTURE SCOPE.....	28
	5.1 Conclusion.....	28
	5.2 Future Scope.....	28
6	BIBLIOGRAPHY.....	29

CHAPTER 1

INTRODUCTION

1.1 Purpose:

The "Automatic Time Table Generator - For Faculty" aims to streamline academic scheduling by automating timetable creation, reducing manual workload, and minimizing conflicts. It provides a user-friendly interface that enables faculty to view, manage, and request schedule swaps, adding flexibility to accommodate unexpected changes. Additionally, it grants Heads of Departments (HODs) oversight to manage schedules across the department, ensuring optimal class coverage and effective resource allocation. By simplifying and centralizing scheduling, the system enhances productivity, organization, and adaptability, benefiting both faculty and administrators and promoting a more efficient academic environment.

1.2 Background of project:

The "Automatic Time Table Generator - For Faculty" project was created to address the inefficiencies and challenges of traditional scheduling methods in academic institutions. Manual processes, often using spreadsheets or basic software, are time-consuming, prone to errors, and challenging to adjust for faculty needs. This leads to scheduling conflicts, limited flexibility, and an increased administrative burden. Faculty members struggle with managing their schedules, while Heads of Departments (HODs) have limited oversight for ensuring fair workload distribution and effective resource allocation. The project aims to automate timetable generation, using predefined constraints and preferences to create balanced, conflict-free schedules. Faculty can access their schedules, enhancing adaptability, while HODs can oversee department schedules to ensure optimal class coverage. Additionally, the system's data analytics provide insights into historical scheduling patterns, supporting data-driven decision-making and continuous improvement in resource management and scheduling practices.

1.3 Scope of project:

The scope of the Automatic Time Table Generator project is to automate and optimize the process of generating class schedules for faculty members in academic institutions. It is designed to address the inefficiencies of traditional manual scheduling methods by providing a digital solution that minimizes errors and reduces administrative workload. The system is intended to serve faculty members, department heads, and administrative staff by offering an intuitive interface for timetable management, including class assignments, and approval workflows. Additionally, it supports hierarchical access, allowing Heads of Departments (HODs) to oversee and manage faculty schedules across departments. The tool also provides historical data reports to analyze past scheduling trends for better decision-making. The project focuses on ensuring that the generated timetables are conflict-free, well-balanced, and optimized for resource utilization, ultimately enhancing the operational efficiency and flexibility of educational institutions. Future extensions may include integration with other academic systems.

1.4 Project Features:

The system features are as follows:

1. **User-Friendly Interfaces:** A simple and intuitive interface allows faculty, HODs, and administrators to easily navigate and manage schedules without technical complexity.
2. **Automated Timetable Generation:** Automatically creates class schedules for faculty based on predefined rules, constraints, and preferences, minimizing human errors and reducing manual effort.
3. **Hierarchical Access for HODs:** HODs can view, monitor, and manage schedules across their departments, ensuring effective resource allocation and class coverage.
4. **Conflict Resolution:** The system ensures generated schedules are conflict-free by automatically identifying and avoiding overlapping classes or faculty assignments.

CHAPTER 2

SYSTEM REQUIREMENTS

2.1 Hardware Requirements:

- HDD 512 GB
- RAM 4 GB
- Processor i5
- Keyboard
- Mouse

2.2 Software requirements:

Frontend:

- HTML5
- CSS3
- JavaScript ES6
- php 7.4

Backend:

- MySQL 8.0

Operating System:

- Windows 11

2.3 Existing System:

The existing system for timetable scheduling in many academic institutions is largely manual, relying on staff members to allocate class times, rooms, and faculty according to availability. Typically, this process involves using spreadsheets or basic scheduling software with limited automation. However, manual scheduling is highly prone to errors, such as overlapping classes, conflicts in faculty assignments, and inefficient use of resources. The process is often time-consuming, requiring frequent adjustments when unexpected changes arise, such as faculty unavailability or class cancellations. Moreover, manual timetabling lacks flexibility and is challenging to modify without risking errors, resulting in schedules that are static and difficult to adapt to dynamic academic needs. The absence of data insights or historical analysis also means that

institutions struggle to identify trends or improve scheduling efficiency over time. These drawbacks highlight the need for a more streamlined, automated solution to enhance scheduling accuracy and operational efficiency.

2.3.1 Drawbacks of existing system:

1. **Manual Processes:** Reliance on manual methods such as spreadsheets or paper-based forms leads to inefficiencies and errors in data entry and scheduling, consuming significant time and effort from administrative staff.
2. **Limited Flexibility:** Lack of flexibility in accommodating faculty preferences or handling scheduling changes results in difficulties in adjusting schedules, leading to scheduling conflicts and disruptions.
3. **Limited Oversight:** Department heads often have limited visibility and control over faculty schedules, making it challenging to ensure equitable distribution of teaching responsibilities and optimal resource allocation.
4. **Minimal Data Analysis:** The existing system typically lacks robust data analysis capabilities, hindering the identification of scheduling trends, optimization of resource allocation, and informed decision-making for scheduling improvements.
5. **Communication Gaps:** Inefficient communication channels for schedule changes and updates can lead to miscommunication among faculty, staff, and students, further complicating the scheduling process and increasing the potential for errors.

2.4 Proposed System:

The proposed **Automatic Time Table Generator-For Faculty** system is a software solution designed to streamline and automate the timetable creation process in academic institutions, focusing on faculty scheduling. This system minimizes manual effort and the errors commonly associated with scheduling by using a rules-based approach to assign classes based on faculty availability, predefined constraints, and institutional preferences. One of the main features of this system is its user-friendly interface, which allows faculty members to view their schedules, request changes, and adapt to new assignments efficiently. Faculty can submit in case of scheduling conflicts, and these requests are routed to the Heads of Departments (HODs) for quick approval, ensuring both academic needs and faculty preferences are met. For administrators, the system provides hierarchical access, enabling HODs and other staff to manage department-wide schedules. They can monitor faculty assignments, ensuring optimal coverage for all classes and efficient resource utilization. Additionally, the system generates historical data reports, which allow departments to analyze past trends, optimize future schedules, and make data-driven decisions. By automating scheduling, the proposed system aims to reduce administrative workload, improve flexibility, and create balanced, conflict-free timetables, ultimately supporting a more efficient and adaptable academic environment for faculty and students alike.

CHAPTER 3

SYSTEM DESIGN

3.1 System Architecture

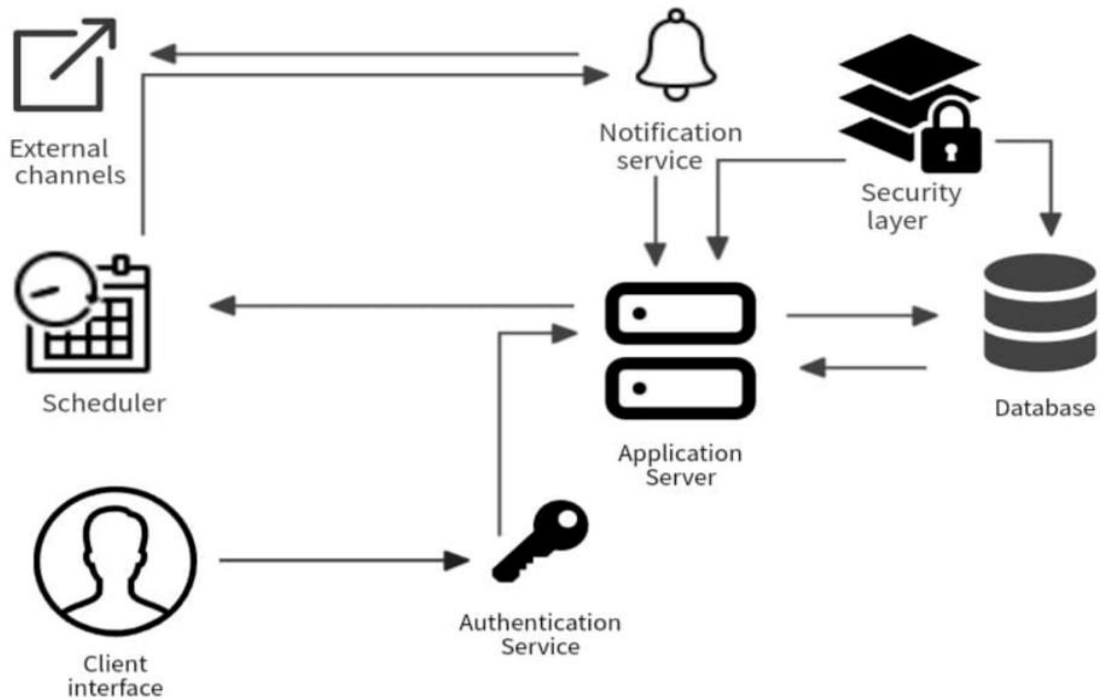


Fig 3.1 System Architecture of Automatic Time Table Generator - For Faculty

3.2 UML Diagrams

3.2.1 Use case diagram

Use Case during requirement elicitation and analysis to represent the functionality of the system. Use case describes a function by the system that yields a visible result for an actor. The identification of actors and use cases result in the definitions of the boundary of the system i.e., differentiating the tasks accomplished by the system and the tasks accomplished by its environment.

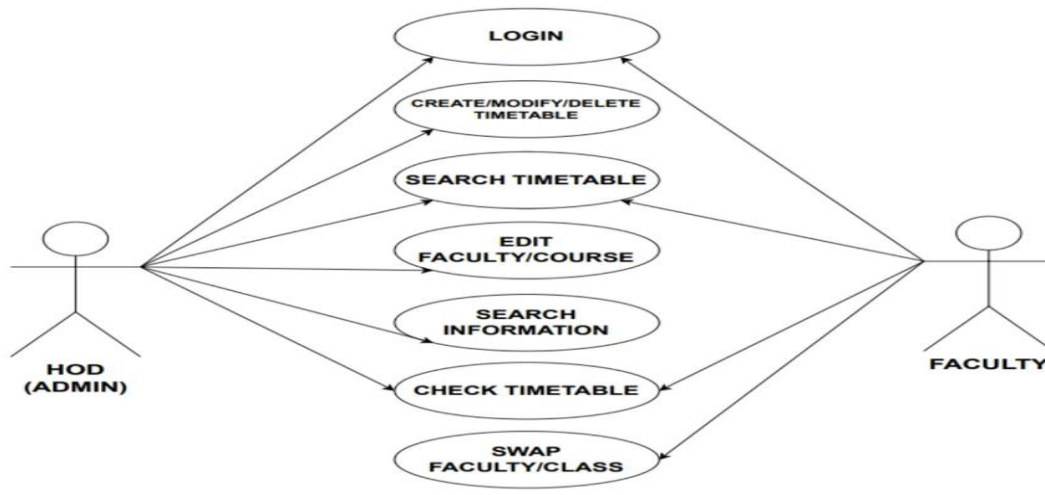


Fig 3.2 Use Case Diagram of Automatic Time Table Generator - For Faculty

3.2.2 Class Diagram

Class diagrams model class structure and contents using design elements such as classes, packages and objects. Class diagram describe the different perspective when designing a system-conceptual, specification and implementation. Classes are composed of three things: name, attributes, and operations. Class diagram also display relationships such as containment, inheritance, association etc. The association relationship is most common relationship in a class diagram. The association shows the relationship between instances of classes.

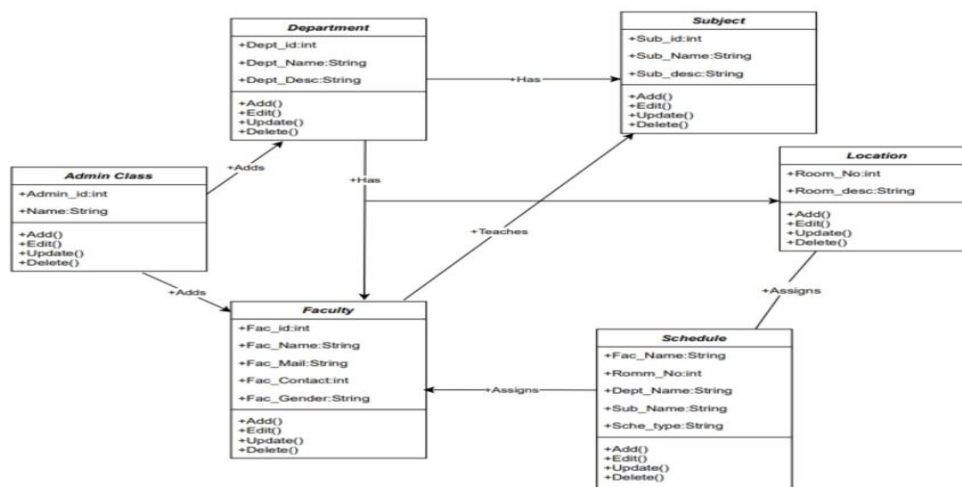


Fig 3.3 Class Diagram of Automatic Time Table Generator - For Faculty

3.2.3 Sequence Diagram

Sequence diagram displays the time sequence of the objects participating in the interaction. This consists of the vertical dimension (time) and horizontal dimension (different objects).

Objects: An object can be thought of as an entity that exists at a specified time and has a definite value, as well as a holder of identity. A sequence diagram depicts item interactions in chronological order. It illustrates the scenario's objects and classes, as well as the sequence of messages sent between them in order to carry out the scenario's functionality. In the Logical View of the system under development, sequence diagrams are often related with use case realizations. Event diagrams and event scenarios are other names for sequence diagrams. A sequence diagram depicts multiple processes or things that exist simultaneously as parallel vertical lines (lifelines), and the messages passed between them as horizontal arrows, in the order in which they occur. This enables for the graphical specification of simple runtime scenarios.

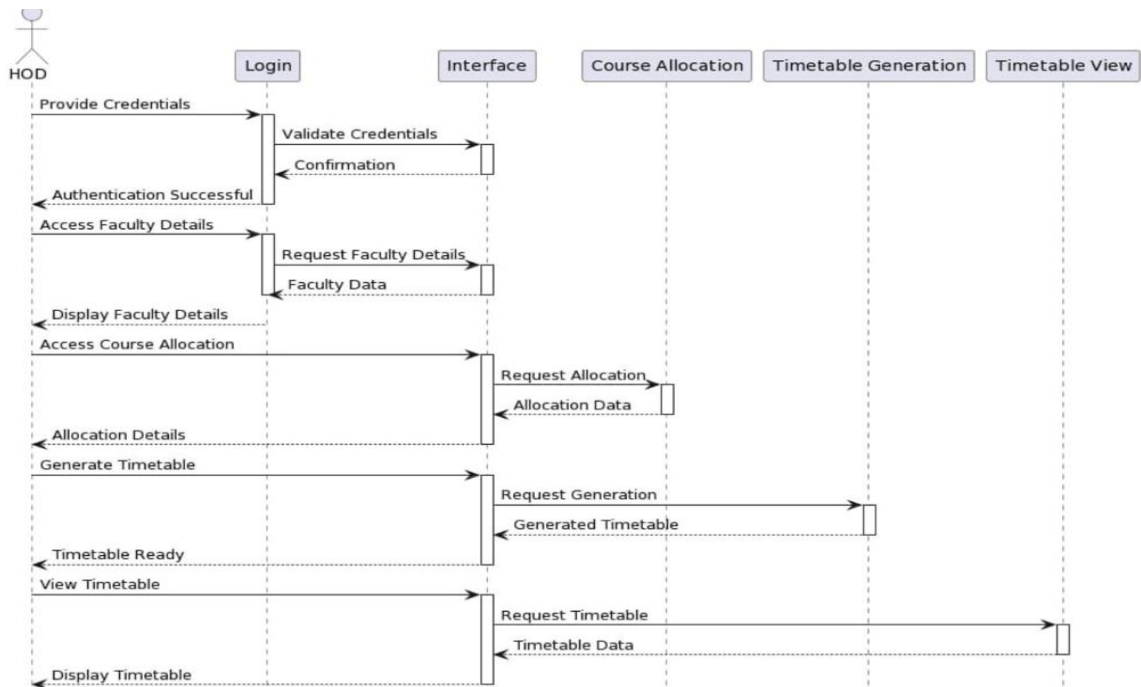


Fig 3.4 Sequence Diagram of Automatic Time Table Generator - For Faculty

3.2.4 Activity Diagram

The process flows in the system are captured in the activity diagram. Similar to a state diagram, an activity diagram also consists of activities, actions, transitions, initial and final states, and guard conditions



Fig 3.5 Activity Diagram of Automatic Time Table Generator - For Faculty

CHAPTER 4

IMPLEMENTATION

4.1 Code

```
<?php
// Start the session
session_start();
if (isset($_GET['success'])) {
    echo "<script type='text/javascript'>alert('Time Table
Generated');</script>";
}
?>
<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
    <meta charset="utf-8"/>
    <meta name="viewport" content="width=device-width, initial-scale=1,
maximum-scale=1"/>
    <meta name="description" content=""/>
    <meta name="author" content=""/>
    <title>TimeTable Management System</title>
    <script type="text/javascript"
src="assets/jsPDF/dist/jspdf.min.js"></script>
    <script type="text/javascript" src="assets/js/html2canvas.js"></script>
    <link href="assets/css/bootstrap.css" rel="stylesheet"/>
    <link href="assets/css/font-awesome.min.css" rel="stylesheet"/>
    <link href="assets/css/style.css" rel="stylesheet"/>
    <script
src="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js
" integrity="sha384-
B4gt1jrGC7Jh4AgTPSdUtOBvfO8shuf57BaghqFfPIYxofvL8/KUEfYiJO
MMV+rV" crossorigin="anonymous"></script>
    <link
href='http://fonts.googleapis.com/css?family=Open+Sans:400,700,300'
rel='stylesheet' type='text/css'/>
</head>
```



```

<body>

<div class="navbar navbar-inverse navbar-fixed-top " id="menu">
  <div class="container">
    <div class="navbar-header">
      <button type="button" class="navbar-toggle" data-
toggle="collapse" data-target=".navbar-collapse">
        <span class="icon-bar"></span>
        <span class="icon-bar"></span>
        <span class="icon-bar"></span>
      </button>

    </div>
    <div class="navbar-collapse collapse move-me">
      <ul class="nav navbar-nav navbar-left">
        <li><a href="addteachers.php">ADD FACULTY</a></li>
        <li><a href="addsubjects.php">ADD SUBJECTS</a></li>
        <li><a href="addclassrooms.php">ADD
CLASSROOMS</a></li>
        <li class="dropdown"><a class="dropdown-toggle" data-
toggle="dropdown" aria-expanded="false">ALLOTMENT
          <span class="caret"></span></a>
          <ul class="dropdown-menu">
            <li>
              <a href=allotsubjects.php>THEORY COURSES</a>
            </li>
            <li>
              <a href=allotpracticals.php>PRACTICAL
COURSES</a>
            </li>
            <li>
              <a href=allotclasses.php>CLASSROOMS</a>
            </li>
          </ul>
        </li>
        <li><a href="generatetimetable.php">GENERATE
TIMETABLE</a></li>

      </ul>
      <ul class="nav navbar-nav navbar-right">

```

```

        <li><a href="index.php">LOGOUT <svg
xmlns="http://www.w3.org/2000/svg" width="13px" height="13"
fill="currentColor" class="bi bi-box-arrow-right " viewBox="0 0 16 16">
        <path fill-rule="evenodd" d="M10 12.5a.5.5 0 0 1-.5.5h-8a.5.5 0
0 1-.5-.5v-9a.5.5 0 0 1 .5-.5h8a.5.5 0 0 1 .5.5v2a.5.5 0 0 1 0v-2A1.5
1.5 0 0 0 9.5 2h-8A1.5 1.5 0 0 0 0 3.5v9A1.5 1.5 0 0 1 1.5 14h8a1.5 1.5 0 0
0 1.5-1.5v-2a.5.5 0 0 0-1 0z"/>
        <path fill-rule="evenodd" d="M15.854 8.354a.5.5 0 0 0-.708l-
3-3a.5.5 0 0 0-.708.708L14.293 7.5H5.5a.5.5 0 0 0 0 1h8.793l-2.147
2.146a.5.5 0 0 0 .708.708z"/>
        </svg></a></li>
</ul>

```

```

    </div>
</div>
</div>

```

```

<br>

```

```

<div id="myModal" class="modal">
  <div class="modal-content">
    <div class="modal-header">
      <span class="close">&times</span>
      <h2 id="popupHead">Assign Substitute</h2>
    </div>
    <div class="modal-body" id="AssignSubstitute">
      <div style="display:block" id="assignSubstituteForm">
        <form method="post"
action="assignSubstituteFormValidation.php">
          <div class="form-group">
            <label for="substitute">Substitute</label>
            <select class="form-control" id="substitute" name="SB">
            </select>
            <input type="hidden" id="cell_number" class="btn btn-default"
name="CN">
          </div>
          <div align="right" class="form-group">

```

```

<input type="submit" id="submit" class="btn btn-default" name="ADD"
value="CHECK">
    </div>
</form>
</div>
</div>
</div>
</div>
<script>
    var assignsubstutueForm =
document.getElementById("assignSubstitueForm");
    // Get the <span> element that closes the modal
    var modal = document.getElementById('myModal');
    var span = document.getElementsByClassName("close")[0];
    span.onclick = function () {
        modal.style.display = "none";
        assignsubstutueForm.style.display = "none";
    }

    // When the user clicks anywhere outside of the modal, close it
    window.onclick = function (event) {
        if (event.target == modal) {
            modal.style.display = "none";
            assignsubstutueForm.style.display = "none";
        }
    }
</script>
<form action="algo.php" method="post">
    <div align="center" style="margin-top: 50px">
        <button type="submit"
            id="generatebutton" class="btn btn-success btn-lg">GENERATE
        </button>
    </div>
</form>
<form action="generatetimetable.php" method="post">
    <div align="center" style="margin-top: 30px">
        <select name="select_teacher" class="list-group-item">
            <option selected disabled>Select Faculty</option>
        <?php
            $q = mysqli_query(mysqli_connect("localhost", "root", "", "ttms"),

```

```

        "SELECT * FROM teachers ");
        while ($row = mysqli_fetch_assoc($q)) {
            echo "<option
value=\"{" $row['faculty_number']}\">{" $row['name']}</option>\"";
        }
    ?>

</select>
<button type="submit" id="viewteacher" class="btn btn-success btn-
lg" style="margin-top: 5px">VIEW TIMETABLE
</button>
</div>
</form>
<form action="generatetimetable.php" method="post">
    <div align="center" style="margin-top: 20px">
        <select name="select_semester" class="list-group-item">
            <option selected disabled>Select Course</option>
            <option value="3">CSE-AIML</option>
            <option value="5">BTech-AIML</option>
            <option value="7">AI & DS</option>
        </select>
        <button type="submit" id="viewsemester" class="btn btn-success btn-
lg" style="margin-top: 5px">VIEW TIMETABLE
        </button>
    </div>
</form>

<div>
    <br>
    <style>
        table {
            margin-top: 20px;
            font-family: arial, sans-serif;
            border-collapse: collapse;
            width: 100%;
        }

        td, th {
            border: 1px solid #000000;
            text-align: left;

```

```

padding: 8px;
}

</style>
<div id="TT" style="background-color: #FFFFFF">
  <table border="2" cellspacing="3" align="center" id="timetable">
    <caption><strong><br><br>
      <?php
        if (isset($_POST['select_semester'])) {
echo "COMPUTATIONAL INTELLIGENCE DEPARTMENT COURSE
" . $_POST['select_semester'] . " ";
$year = (int)($_POST['select_semester'] / 2) + $_POST['select_semester']
% 2;
$r = mysqli_fetch_assoc(mysqli_query(mysqli_connect("localhost", "root",
"", "ttms"), "SELECT * from classrooms
WHERE status = '$year'"));
echo " ( " . $r['name'], " ) ";
} else if (isset($_POST['select_teacher'])) {
$id = $_POST['select_teacher'];
$r =
mysqli_fetch_assoc(mysqli_query(mysqli_connect("localhost", "root", "",
"ttms"), "SELECT * from teachers
WHERE faculty_number = '$id'"));
echo $r['name'];
} else if (isset($_GET['display'])) {
$id = $_GET['display'];
$r =
mysqli_fetch_assoc(mysqli_query(mysqli_connect("localhost", "root", "",
"ttms"), "SELECT * from teachers
WHERE faculty_number = '$id'"));
echo $r['name'];
}
?>
    </strong></caption>
    <tr>
      <td style="text-align:center"><b>WEEKDAYS</b></td>
      <td style="text-align:center"><b>9:20-10:20 </b></td>
      <td style="text-align:center"><b>10:20-11:20</b></td>
      <td style="text-align:center"><b>11:20-11:30</b></td>

```

```

<td style="text-align:center"><b>11:30-12:30</b></td>
<td style="text-align:center"><b>12:30-01:30</b></td>
<td style="text-align:center"><b>01:30-02:30</b></td>
<td style="text-align:center"><b>02:30-03:30</b></td>
</tr>
<tr>
<?php
$stable = null;
if (isset($_POST['select_semester'])) {
    $stable = " semester" . $_POST['select_semester'] . " ";
} else if (isset($_POST['select_teacher'])) {
    $stable = " " . $_POST['select_teacher'] . " ";
} else if (isset($_GET['display'])) {
    $stable = " " . $_GET['display'] . " ";
} else
    echo '</table>';
if (isset($_POST['select_semester']) || isset($_POST['select_teacher']) ||
isset($_GET['display'])) {
$q = mysqli_query(mysqli_connect("localhost", "root", "",
"ttms"),"SELECT * FROM" . $stable);
$qqq = mysqli_query(mysqli_connect("localhost", "root", "",
"ttms"),"SELECT * FROM subjects");
$days = array('<b>MONDAY</b>','<b>TUESDAY</b>',
'<b>WEDNESDAY</b>','<b>THURSDAY</b>','<b>FRIDAY</b>',
'<b>SATURDAY</b>');
$i = -1;
$str = "<br>";
$id = "";
if (isset($_POST['select_semester'])) {
while ($r = mysqli_fetch_assoc($qq)) {
if ($r['isAlloted'] == 1 && $r['semester'] == $_POST['select_semester']) {
$str .= $r['subject_code'] . ": " . $r['subject_name'] . " ";
if (isset($r['allotedto'])) {
$id = $r['allotedto'];
$qqq = mysqli_query(mysqli_connect("localhost", "root", "", "ttms"),
"SELECT * FROM teachers WHERE faculty_number = '$id'");
$r = mysqli_fetch_assoc($qqq);
$str .= " " . $r['alias'] . ": " . $r['name'] . " ";}
if ($r['course_type'] != "LAB") {
$str .= "<br>";

```

```

continue;
} else {
$str .= ", ";
}
if (isset($r['allotedto2'])) {
$id = $r['allotedto2'];
$qqq = mysqli_query(mysqli_connect("localhost", "root", "", "ttms"),
    "SELECT * FROM teachers WHERE faculty_number = '$id'");
$rr = mysqli_fetch_assoc($qqq);
$str .= " " . $rr['alias'] . ": " . $rr['name'] . ", ";
if (isset($r['allotedto3'])) {
$id = $r['allotedto3'];
$qqq = mysqli_query(mysqli_connect("localhost", "root", "", "ttms"),
    "SELECT * FROM teachers WHERE faculty_number = '$id'");
$rr = mysqli_fetch_assoc($qqq);
$str .= " " . $rr['alias'] . ": " . $rr['name'] . "<br>";
}
}
} else if (isset($_POST['select_teacher']) || isset($_GET['display'])) {
    if (isset($_POST['select_teacher'])) {
        $tid = $_POST['select_teacher'];
        } else if (isset($_GET['display'])) {
            $tid = $_GET['display'];
            $tid = strtoupper($tid);
        }
        while ($r = mysqli_fetch_assoc($qq)) {
            if ($r['isAlloted'] == 1 && $r['allotedto'] == $tid) {
                $str .= $r['subject_code'] . ": " . $r['subject_name'] . "
<br>";
            } else if ($r['isAlloted'] == 1 && isset($r['allotedto2'])
&& $r['allotedto2'] == $tid) {
                $str .= $r['subject_code'] . ": " . $r['subject_name'] . "
<br>";
            } else if ($r['isAlloted'] == 1 && isset($r['allotedto3'])
&& $r['allotedto3'] == $tid) {
                $str .= $r['subject_code'] . ": " . $r['subject_name'] . "
<br>";
            }
        }
    }
}

```

```

    }
    while ($row = mysqli_fetch_assoc($q)) {
        $i++;

        echo "
        <tr><td style=\"text-align:center\">$days[$i]</td>
        <td style=\"text-align:center\">{$row['period1']}</td>
        <td style=\"text-align:center\">{$row['period2']}</td>
        <td style=\"text-align:center\">{$row['period3']}</td>
        <td style=\"text-align:center\">{$row['period4']}</td>
        <td style=\"text-align:center\">{$row['period5']}</td>
        <td style=\"text-align:center\">LUNCH</td>
        <td style=\"text-align:center\">{$row['period6']}</td>
        </tr>\n";
    }

    echo '</table>';
    $sign = "MALLA REDDY COLLEGE OF ENGINEERING
    AND TECHNOLOGY (MRCET)";
    echo "<div style='margin-left: 10px' align='center'>" . "<br>" .
    $str . "<br></div>" .
    "<div style='margin-left: 10px' align='center'>" . "<strong>"
    . $sign . "<br></strong></div>";
    }
    if (isset($_POST['select_teacher'])) {
        echo "<script>Substitute();</script>";
        $_SESSION['shown_id'] = $_POST['select_teacher'];
    }
    if (isset($_GET['display'])) {
        echo "<script>Substitute();</script>";
        $_SESSION['shown_id'] = $_GET['display'];
    }
    ?>

</div>
</div>
<script type="text/javascript">
function gendf() {
    var doc = new jsPDF();

    doc.addHTML(document.getElementById('TT'), function () {

```



```

doc.save('<?php
    if (isset($_POST["select_semester"])) {
        echo "ttms semester " . $_POST["select_semester"];
    } else if (isset($_POST["select_teacher"])) {
        echo "ttms " . $_POST["select_teacher"];
    } else if (isset($_GET["display"])) {
        echo "ttms " . $_GET["display"];
    }
    ?>' + '.pdf');
,alert("Downloaded!"));

});
}

</script>
<div align="center" style="margin-top: 10px">
    <button id="saveaspdf" class="btn btn-info btn-lg"
onclick="gendf()">SAVE AS PDF</button>
</div>

<script src="assets/js/jquery-1.10.2.js"></script>
<script src="assets/js/bootstrap.js"></script>
<script src="assets/js/jquery.flexslider.js"></script>
<script src="assets/js/scrollReveal.js"></script>
<script src="assets/js/jquery.easing.min.js"></script>
<script src="assets/js/custom.js"></script>
</body>
</html>

```

4.2 Output Screens:

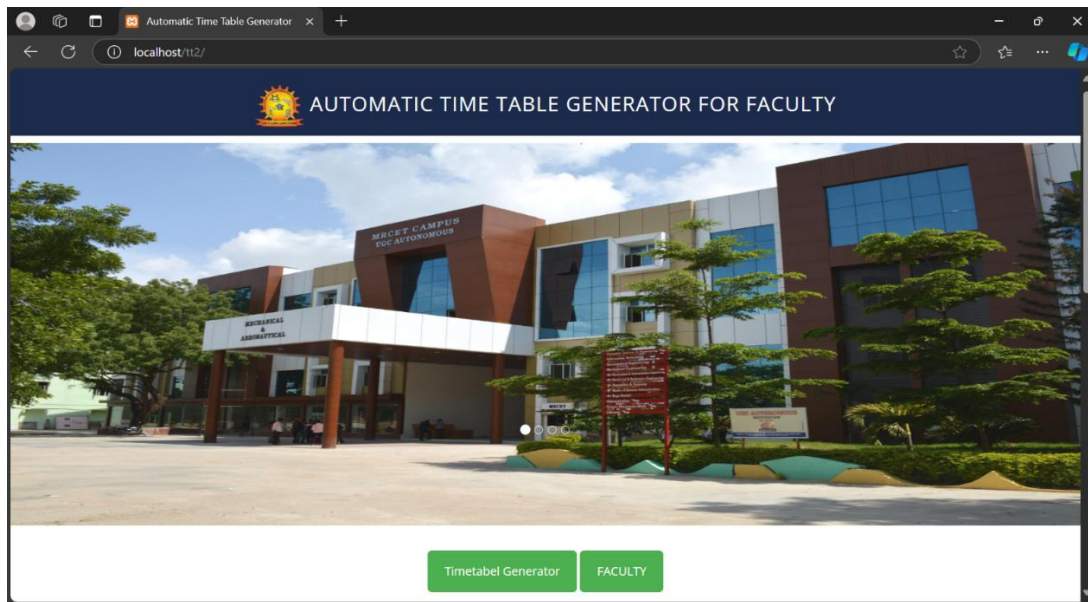


Fig: ATTG Home Page with Hod & Faculty logins

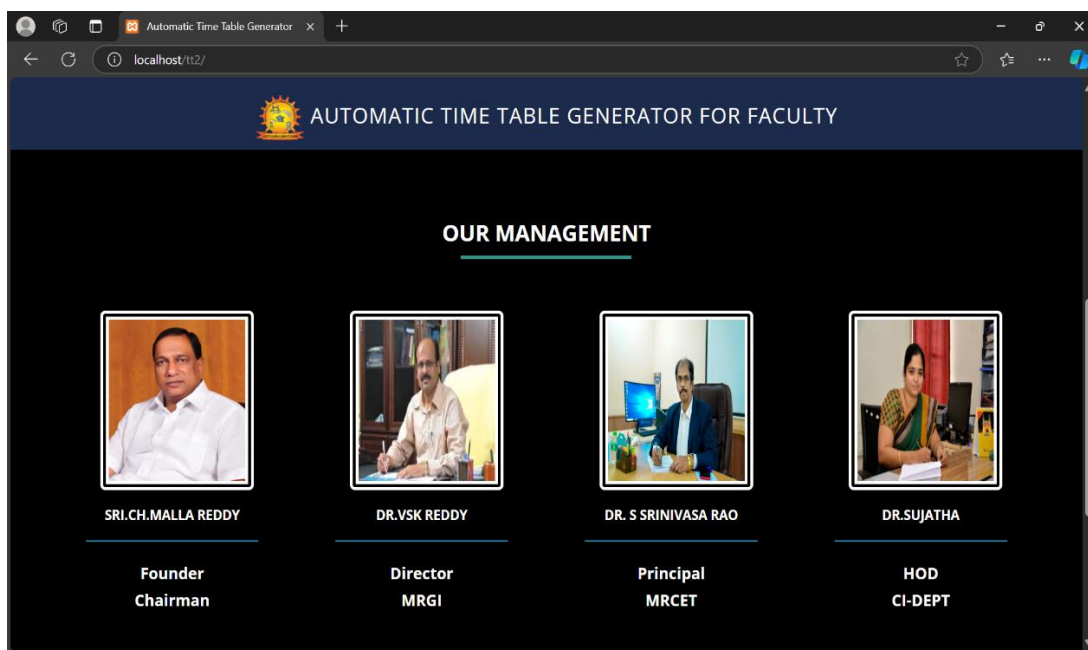


Fig: ATTG Home Page showing Our Management details

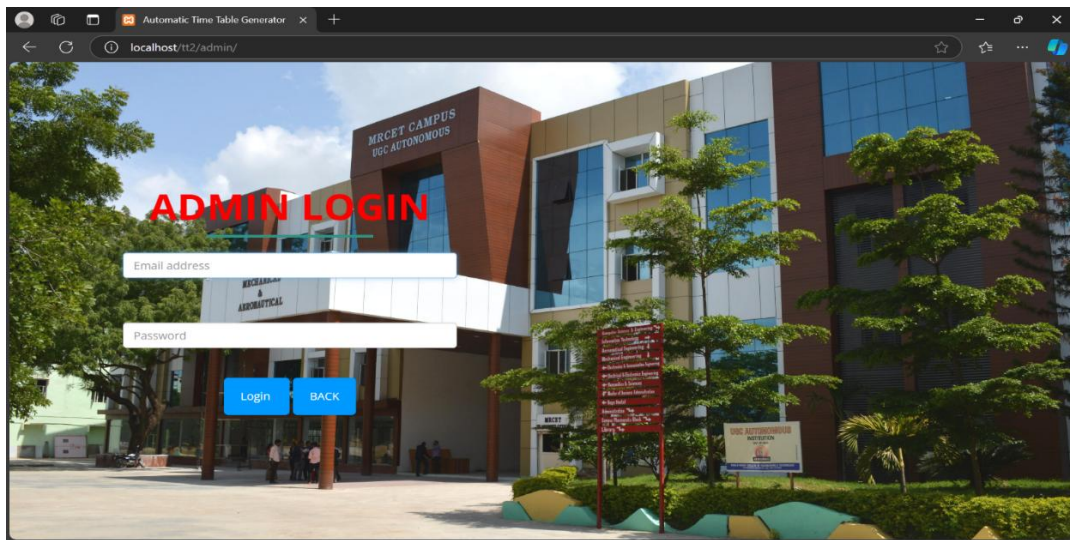


Fig: Admin's (HODs)Login page

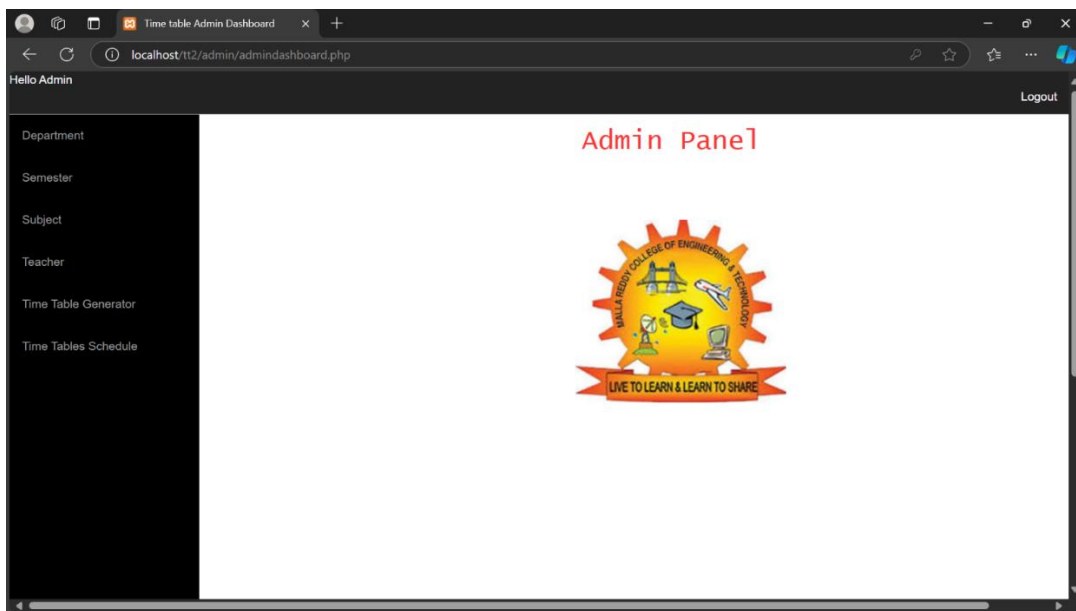


Fig: Admin's Home Page

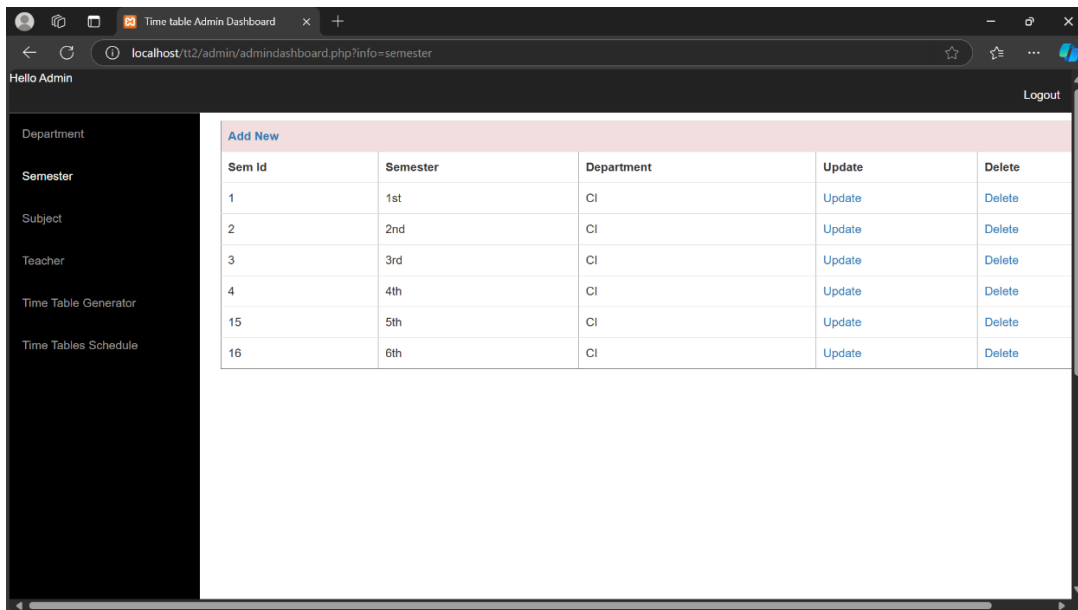


Fig: Adding New Semester page

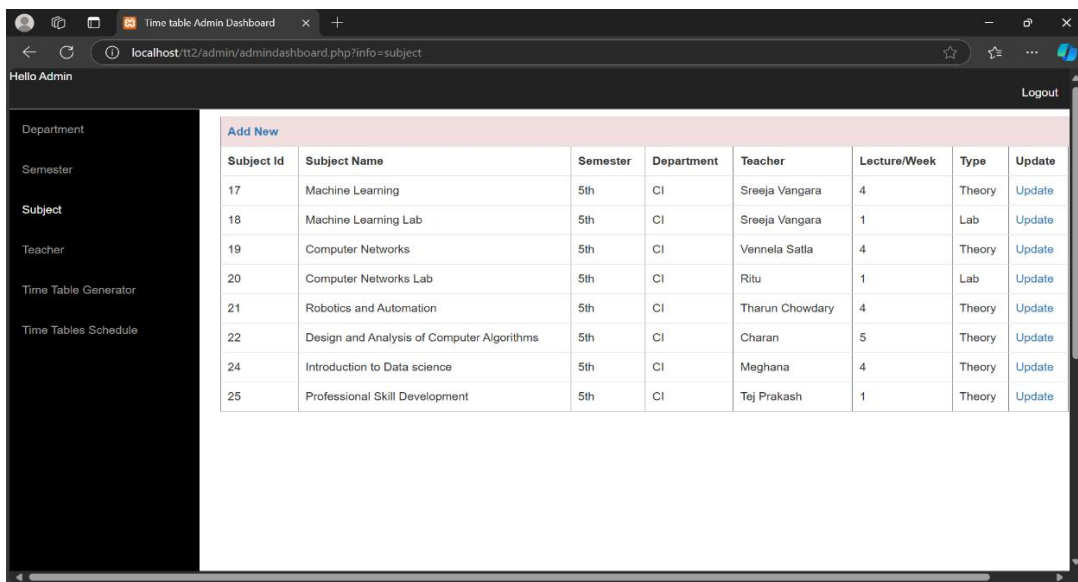


Fig: Adding New Subject page

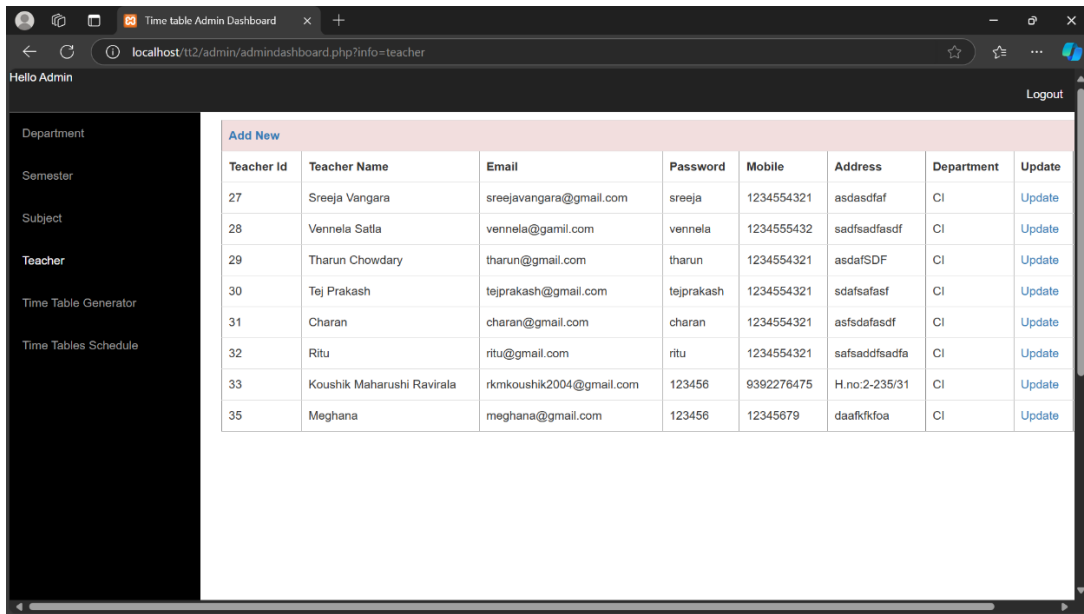


Fig: Adding New Teacher Page

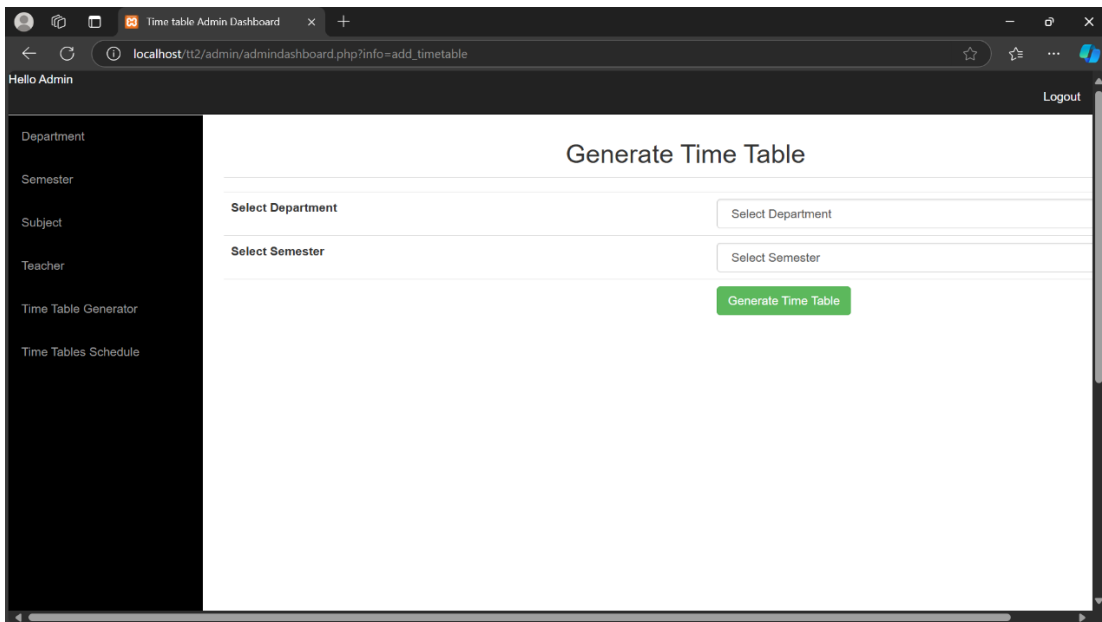


Fig: Generating Time Table Page

The screenshot shows a web browser window at localhost/tt2/admin/admindashboard.php?info=add_timetable. The page has a dark sidebar with navigation links: Department, Semester, Subject, Teacher, Time Table Generator (active), and Time Tables Schedule. The main area displays the "Generate Time Table" form. It includes input fields for "Select Department" (CI) and "Select Semester" (5th), a "Generate Time Table" button, and a "Download Timetable" button.

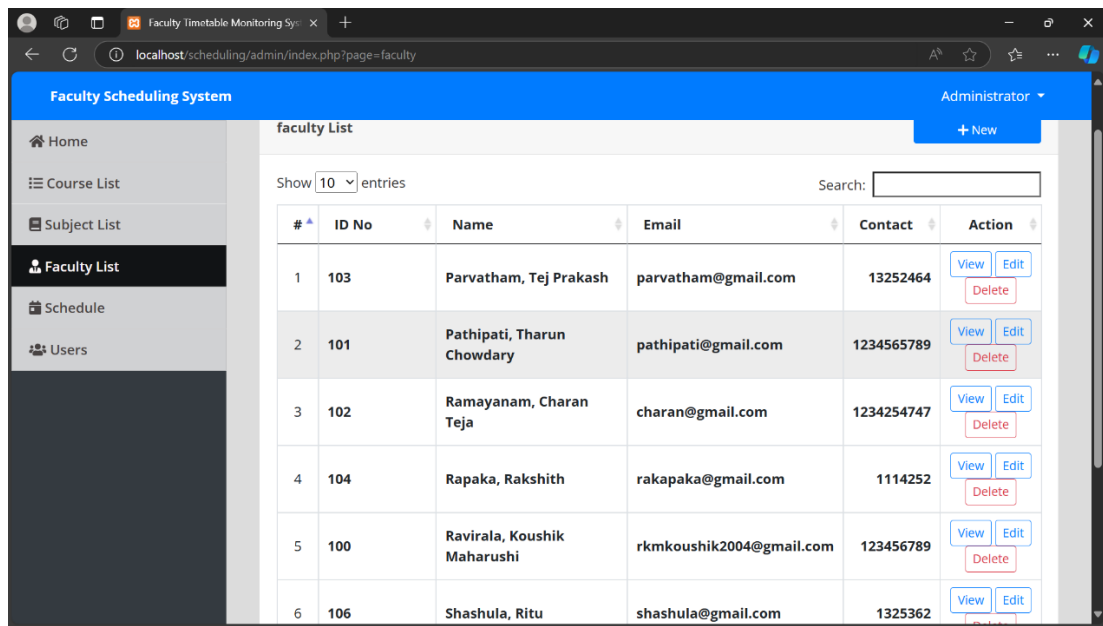


Fig: Entering Faculty Details page

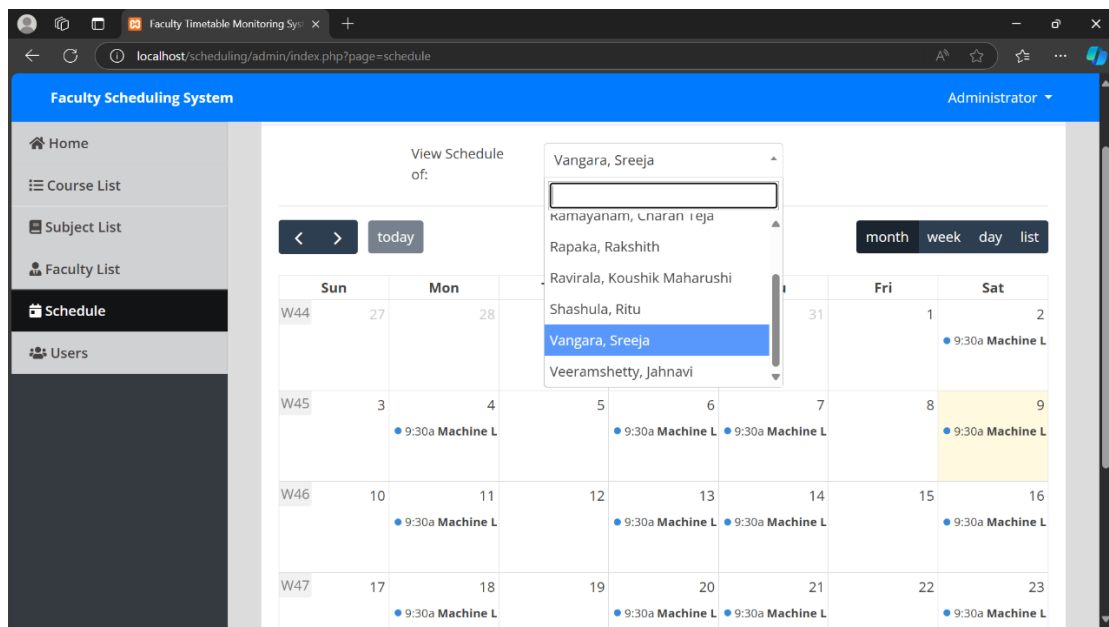


Fig: Scheduling the Faculties Page

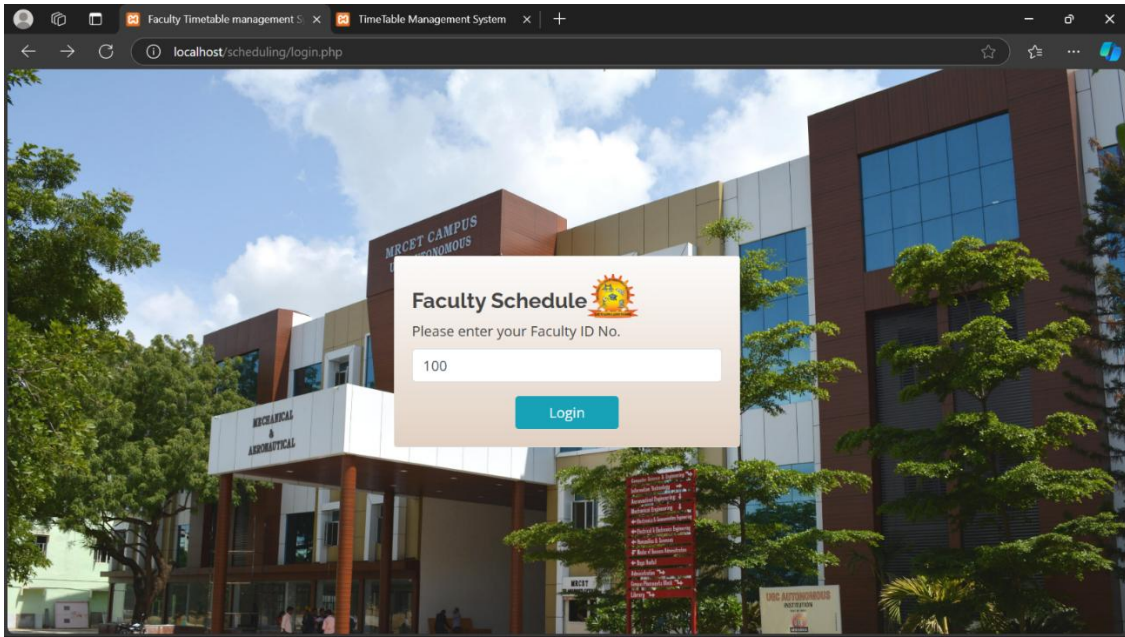


Fig: Login page for Faculties

November 2024							month	week	day	list
Sun	Mon	Tue	Wed	Thu	Fri	Sat				
W44 27	28	29	30	31	1	2 9:30a Machine Lear				
W45 3 9:30a Machine Lear	4	5 9:30a Machine Lear	6 9:30a Machine Lear	7 9:30a Machine Lear	8	9 9:30a Machine Lear				
W46 10 9:30a Machine Lear	11	12 9:30a Machine Lear	13 9:30a Machine Lear	14	15	16 9:30a Machine Lear				
W47 17 9:30a Machine Lear	18	19 9:30a Machine Lear	20 9:30a Machine Lear	21	22	23 9:30a Machine Lear				

Fig: Displaying their respective schedule

4.3: Testing

Software testing is a critical element of software quality assurance and represents the ultimate review of specification, design and code generation.

CHAPTER 5

CONCLUSION & FUTURE SCOPE

5.1 Conclusion

The Automatic Time Table Generator simplifies scheduling for academic institutions by automating the creation of accurate, conflict-free timetables. It addresses challenges like class overlaps, faculty conflicts, and inefficient resource allocation, saving time and reducing manual effort. Faculty benefit from a user-friendly interface to view schedules and adapt to changes seamlessly, fostering greater satisfaction and flexibility. Administrators and HODs can monitor department-wide schedules, approve swaps, and manage resources effectively through a hierarchical access system. By incorporating specific constraints and faculty preferences, the tool generates balanced timetables with minimal manual intervention. This system enhances scheduling accuracy, efficiency, and adaptability, supporting a well-organized academic environment that benefits faculty, administrators, and students alike.

5.2: Future Scope

The Future Scope of the Automatic Time Table Generator includes several key enhancements. Integrating AI and machine learning could optimize schedules by analyzing historical data and predicting optimal faculty and resource assignments. Adding student scheduling would create a unified system to manage both faculty and student timetables, minimizing class conflicts. Incorporating room and resource management would ensure efficient use of classrooms and facilities. Cloud and mobile access would provide real-time updates and greater flexibility for users. Additionally, advanced analytics and reporting would offer deeper insights to help institutions refine and optimize scheduling processes.

CHAPTER 6

BIBLIOGRAPHY

- [1] Krithi, Dr M Ramakrishna, student management system a survey Computer Science and Engineering, Vemana Institute of Technology, Bangalore-34 International Research Journal of Computer Science (IRJCS) Issue 05, Volume: 04 May 2017.
- [2] Lalit Mohan Joshi M.Tech schola BTKIT Dwarahat, Almora, Uttarakhand "A Research Paper on College Management System" International Journal of Computer Applications (0975-8887) Volume 122-No.11, July 2015.
- [3] S. W. Liew, N. F. M. Sani, M. T. Abdullah, R. Yaakob, and M. Y. Sharum, “A Research Paper on Faculty Management System in Educational Institutions’’, vol. 83, pp. 201–207, Jun. 2019.
- [4] Srikant Patnaik, Khushboo kumari Singh, Rashmi Ranjan, Niki Kumari "College management system", International Research Journal of Engineering and Technology, IRJ Volume:03 Issue: 05/May-2016.