

A

PROJECT REPORT

ON

ATTENDANCE MANAGEMENT SYSTEM

Submitted to:

KURUKSHETRA UNIVERSITY,

KURUKSHETRA

In the partial fulfill requirement for

B.VOC(SOFTWARE DEVELOPMENT) SEM-II

SESSION : 2022-2023



Under the guidance of:
Mrs. Shaina
Assistant Processor
Department of Computer and Science

Submitted By:
Hemant Dhiman
B.VOC(SD) SEM II
University Roll. No. 220119617
College Roll. No. 1221673042001

SANATAN DHARMA COLLEGE,
AMBALA CANTT

DECLARATION

I hereby declare that this submission is my own work and that to the best of my knowledge and belief. It contains no material previously published or written by another person nor material which to a substantial extent has been accepted for the Award of any other degree or diploma of the university or other institute of higher learning, except where acknowledgement has been made in the next.

Hemant Dhiman

CERTIFICATE-I

It is to validate that **Hemant Dhiman S/O Darshan Dhiman** of class **B.Voc(Software Development) 1st Year** under **University Roll. No. 220119617 College Roll No. 1221673042001** has completed the project titled "**Attendance Management System**" in languages PHP, HTML and CSS for the degree of Bachelors in Vocation (Software Development) Semester-II under my supervision. The work done in project is a result of the candidate's own efforts and report maintains is satisfied as per requirement.

I wish him good success in his future.

Ms. Shaina

Project Guide

Department Of Computer Science

CERTIFICATE-II

It is to validate that **Hemant Dhiman s/o Sh. Darshan Dhiman** of class **B.Voc(Software Development) 1st Year** under University Roll No **220119617** College Roll No. **1221673042001** has completed the project titled "**Attendance Management System**" in languages PHP, HTML and CSS for the degree of Bachelors in Vocation (Software Development)- Semester-II under the supervision of Ms. Shaina. The work done in project is a result of the candidate's own efforts and report maintains is satisfied as per requirement.

We wish him good success in his future.

Dr. Girdhar Gopal

H.O.D.

Department Of Computer Science

Dr. Rajinder Singh

Principal

Sanatan Dharm College

Ambala Cantt

ACKNOWLEDGMENT

I take this opportunity to express my profound gratitude and deep regard to my guide Ms. Shaina, Department of Computer Science for her exemplary guidance, monitoring and constant encouragement throughout the course of this project. The blessing, help and guidance given by her time to time shall carry me a long way in the journey of life on which I am about to embark.

I am oblige to staff members of **SANATAN DHARMA COLLEGE, AMBALA CANTT** for the valuable information provided by them in their respective fields. I am grateful for their cooperation during the period of my assignment.

Lastly, I thank almighty, my parents and my classmates for their constant encouragement without which this assignment would not have been possible.

Hemant Dhiman

INDEX

TOPIC	PAGE NO.
1. INTRODUCTION TO PROJECT.....	7
2. SOFTWARE AND HARDWARE REQUIREMENTS.....	8
3. INTRODUCTION.....	9
4. SOURCE CODE.....	21
5. SCREENSHOTS OF PROJECT.....	43
6. CONCLUSION.....	46
7. BIBLIOGRAPHY/REFERENCES.....	47
8. PROJECT DATA DISK.....	48

INTRODUCTION TO PROJECT

Attendance Management System is a software-based solution designed to automate and streamline attendance tracking and management processes in educational institutions and businesses. The system uses digital tools to monitor attendance, record data, and generate reports, eliminating the need for manual data entry and time-consuming administrative tasks. With this system, institutions can easily track employee and student attendance, monitor tardiness and absenteeism, and generate reports that can help them make informed decisions. The Attendance Management System can also be integrated with other HR or educational systems to create a comprehensive solution that supports efficient workforce management. This project aims to develop a user-friendly and efficient Attendance Management System that will help educational institutions and businesses save time and resources while ensuring accurate and reliable attendance data.

HARDWARE AND SOFTWARE REQUIREMENT

1. HARDWARE REQUIREMENTS

- Intel i3 processor
- 4 GB RAM
- Hard Drive 256 GB
- Compatible Keyboard and Mouse

2. SOFTWARE REQUIREMENTS

- Operating System: Windows7/8/9/10 or Linux based OS
- Web-Technology: PHP
- Front-End: HTML and CSS
- Back-End: MySQL
- Web Server: Apache Server

INTRODUCTION

1. INTRODUCTION TO HTML

- ➔ HTML stands for Hyper Text Markup Language, and it is the standard markup language used to create and structure content for the web. It provides a set of predefined tags and attributes that are used to define the structure and content of a web page, including headings, paragraphs, images, links, forms, and more.

1.1 What is HTML?

- ➔ HTML stands for HyperText Markup Language.
- ➔ HTML code is written in plain text format and can be easily understood and edited by humans.
- ➔ HTML files can be viewed in a web browser, which interprets the code and displays the web page as a graphical interface.
- ➔ HTML is an essential language for web development and forms the foundation of all web pages, along with other web technologies like CSS and JavaScript.
- ➔ HTML is a crucial language for web development and plays a vital role in the creation of websites, web applications, and digital content.

1.2 A Example of HTML Document

- ➔ **Example**

```
<!DOCTYPE html>  
<html>
```

```
<head>  
    <title>My Web Page</title>  
</head>  
  
<body>  
    <h1>Welcome to my Web Page!</h1>  
    <p>This is a paragraph of text.</p>  
      
</body>  
</html>
```

1.3 Explanation of example code

- ➔ In this example, the HTML code begins with a `<!DOCTYPE html>` declaration that specifies the version of HTML being used.
- ➔ The `html` tag encloses the entire document, while the `head` tag contains information about the web page, such as the title.
- ➔ The `html` tag encloses the entire document, while the `head` tag contains information about the web page, such as the title.
- ➔ The `body` tag contains the visible content of the web page, including the `h1` tag for the heading, `p` tag for the paragraph, and `img` tag for the image.
- ➔ The `src` attribute of the `img` tag specifies the location of the image file, while the `alt` attribute provides a text description of the image for accessibility purposes.

1.4 What is an HTML Element?

- ➔ HTML (Hypertext Markup Language) elements are the building blocks of web pages. An HTML element is defined by a start tag, some content, and an end tag. The start tag and end tag surround the content, which is the text or other elements that make up the content of the element.

- For example, the HTML element for a heading is defined by the `<h1>` start tag, followed by the text of the heading, and then the `</h1>` end tag. The `<h1>` and `</h1>` tags together define the heading element, and the text between them is the content of the element.

Start Tag	Element content	End Tag
<code><h1></code>	My First Heading	<code></h1></code>
<code><p></code>	My First Paragraph	<code></p></code>
<code>
</code>	None	None

1.5 Web Browser

- The purpose of a web browser (Chrome, Edge, Firefox, Safari) is to read HTML documents and display them correctly.
- A browser does not display the HTML tags, but uses them to determine how to display the document.

1.6 HTML History

- Since the early days of the World Wide Web, there have been many versions of HTML:

Year	Version
1989	Tim Berners-Lee invented www
1991	Tim Berners-Lee Invented HTML
1993	Dave Raggett drafted HTML+
1995	W3C Recommendation : HTML 2.0
1997	W3C Recommendation : HTML 3.0

1999	W3C Recommendation : HTML 4.01
2000	W3C Recommendation : XHTML 1.0
2008	WHATWG HTML5 First public Draft
2012	WHATWG HTML5 Living Standards
2014	W3C Recommendation : HTML5
2016	W3C Candidate Recommendation : HTML 5.1
2017	W3C Recommendation : HTML5.1 2 nd Edition
2017	W3C Recommendation : HTML5.2

2. INTRODUCTION TO CSS

→ CSS (Cascading Style Sheets) is a stylesheet language used to describe the presentation and styling of HTML (Hypertext Markup Language) and XML (Extensible Markup Language) documents. It allows web developers to separate the presentation of a document from its structure and content, making it easier to create visually appealing and well-organized web pages.

2.1 What is CSS?

- CSS works by defining rules that apply to specific HTML elements or groups of elements. These rules specify the properties of the elements, such as their size, color, font, background, and positioning. CSS rules can be defined in an external style sheet file, which can be shared by multiple pages, or in the HTML document itself using the <style> tag.
- CSS uses a selector to target specific elements to apply styles to. Selectors can be based on the element type, class, or ID, as well as other attributes. For example, to target all paragraphs in

an HTML document, the selector would be "p". To target a specific paragraph with a class of "highlight", the selector would be ".highlight".

- ➔ CSS also supports inheritance and cascading, which allows styles to be applied to elements based on their relationship to other elements in the document. For example, a style applied to a parent element can be inherited by its child elements, and styles defined in an external style sheet can be overridden by styles defined in the HTML document itself.
- ➔ CSS has gone through several versions and updates, with CSS3 being the most recent version. CSS3 introduced many new features and capabilities, such as transformations, transitions, animations, and more advanced layout and grid systems.
- ➔ Overall, CSS is an essential tool for web developers to create attractive and functional web pages, and to separate presentation from structure and content.

2.2 A Example of CSS

```
body {  
background-color: lightblue;  
}  
  
h1 {  
color: white; text-align: center;  
}  
  
P{  
font-family: verdana; font-size: 20px;  
}
```

2.3 CSS Solved a Big Problem

- ➔ HTML was NEVER intended to contain tags for formatting a web page! HTML was created to describe the content of a web page, like:

```
<h1>This is a heading</h1> <p>This is a paragraph.</p>
```

When tags like ``, and color attributes were added to the HTML 3.2 specification, it started a nightmare for web developers. Development of large websites, where fonts and color information were added to every single page, became a long and expensive process.

To solve this problem, the World Wide Web Consortium (W3C) created CSS. CSS removed the style formatting from the HTML page!

2.4 Using CSS

- ➔ CSS can be added to HTML documents in 3 ways:

- 1) **Inline CSS**
- 2) **Internal CSS**
- 3) **External CSS**

2.4.1 Inline CSS

- ➔ Inline CSS is applied directly to individual HTML elements using the "style" attribute. Inline styles are defined within the element itself and take precedence over external and internal styles.
- Example of Inline CSS:
`<p style="color: red; font-size: 18px;">This is a paragraph with inline CSS.</p>`

2.4.2 Internal CSS

- ➔ Internal CSS is defined within the `<style>` tag in the head section of an HTML document. Internal styles apply to the entire document or to specific sections or elements using selectors.
- Example of Internal CSS:

```
<!DOCTYPE html>

<html>

<head>

    <title>My Web Page</title>

    <style>

        p {

            color: blue;

            font-size: 16px;

        }

    </style>
```

```
</head>

<body>

    <p>This is a paragraph with internal CSS.</p>

</body>

</html>
```

2.4.3 External CSS

- External CSS is defined in a separate .css file and linked to the HTML document using the `<link>` tag in the head section. External styles apply to the entire document or to specific sections or elements using selectors.

- Example of External CSS

HTML file: index.html

```
<!DOCTYPE html>

<html>

<head>

    <title>My Web Page</title>

    <link rel="stylesheet" type="text/css" href="styles.css">

</head>

<body>

    <h1>Hello world</h1>

    <p>This is a paragraph with external CSS.</p>

</body>

</html>
```

CSS file: style.css

```
body {

background-color: lightblue;
```

```
}

h1 {

color: white; text-align: center;

}

P{

font-family: verdana; font-size: 20px;

}

*****
```

3. INTRODUCTION TO MYSQL

- ➔ MySQL is an open-source relational database management system (RDBMS) that is widely used to store and manage large amounts of data.
- ➔ It was developed by Oracle Corporation and first released in 1995.
- ➔ MySQL uses a client-server architecture and supports a variety of operating systems, including Windows, Linux, and macOS. It is known for its scalability, high performance, and reliability, and is used by many popular websites and applications, including Facebook, Twitter, and YouTube.

3.1 What is SQL?

- ➔ SQL (Structured Query Language) is a standard language used to manage and manipulate relational databases. It is used to create, modify, and query databases, as well as to manage the data stored in those databases.
- ➔ SQL is a declarative language, meaning that users specify what they want the database to do, rather than how to do it. SQL is designed to be easy to learn and use, and it is supported by a wide variety of database management systems, including MySQL, Oracle, SQL Server, PostgreSQL, and many others.

3.2 What Can SQL do?

Here are the some common tasks performed using SQL include:

- ➔ Creating databases and tables: SQL can be used to create new databases and tables, and to specify their structure and data types.
- ➔ Inserting and updating data: SQL can be used to add, modify, and delete data from databases.

- ➔ Querying data: SQL can be used to retrieve data from databases based on specified criteria, such as filtering by specific values or sorting data in a particular way.
- ➔ Joining tables: SQL can be used to join two or more tables together based on a common column or set of columns.
- ➔ Creating views: SQL can be used to create views, which are virtual tables that can be used to retrieve specific data from the database.

3.3 SQL Standard

- ➔ The SQL (Structured Query Language) standard is a set of guidelines and specifications that define the syntax, semantics, and functionality of the SQL language. It is maintained by the International Organization for Standardization (ISO) and the American National Standards Institute (ANSI).

The SQL standard is divided into several parts, each covering a different aspect of the language. Some of the key parts of the standard include:

- SQL/Foundation: This defines the basic syntax and semantics of SQL, including data types, expressions, and basic query constructs.
- SQL/CLI: This defines a set of APIs (Application Programming Interfaces) for accessing SQL databases from programming languages.
- SQL/PSM: This defines a procedural language for SQL, allowing users to create stored procedures, functions, and triggers.
- SQL/OLB: This defines a set of commands for managing online transaction processing (OLTP) systems.
- SQL/XML: This defines a set of commands for working with XML data in SQL databases.

3.4 Using SQL in a Website

- ➔ MySQL can be used in websites in several ways, depending on the specific needs of the website and the resources available.

Here are some common ways to use MySQL in a website:

- Content Management Systems (CMS)
- Custom-built websites
- E-commerce websites

➔ To use MySQL in a website, you will need to first install MySQL on a web server or a local development environment. Once installed, you will need to create a database and tables to store your website data. You can then use a programming language such as PHP or Python to interact with the MySQL database and retrieve or modify data as needed.

3.5 RDBMS

➔ RDBMS stands for Relational Database Management System. RDBMS is the basis for SQL, and for all modern database systems such as MS SQL

Server, IBM DB2, Oracle, MySQL, and Microsoft Access. The data in RDBMS is stored in database objects called tables. A table is a collection of related data entries and it consists of columns and rows. Look at the "Customers" table:

- **Example**

➔ `SELECT * FROM Customers;`

Every table is broken up into smaller entities called fields. The fields in the Customers table consist of CustomerID, CustomerName, ContactName, Address, City, PostalCode and Country. A field is a column in a table that is designed to maintain specific information about every record in the table.

A record, also called a row, is each individual entry that exists in a table. For example, there are 91 records in the above Customers table. A record is a horizontal entity in a table. A column is a vertical entity in a table that contains all information associated with a specific field in a table.

4. INTRODUCTION TO PHP

➔ PHP (Hypertext Preprocessor) is a server-side scripting language used to build dynamic web applications and websites. It was created by Rasmus Lerdorf in 1994 and has since become one of the most widely used programming languages on the web.

4.1 What is PHP?

- ➔ PHP is an open-source language, meaning that it is free to use and can be modified by developers to suit their needs. It is also cross-platform, meaning that it can run on a variety of operating systems, including Windows, Linux, and macOS.

Some of the key features of PHP include:

- Easy integration with HTML: PHP can be embedded directly into HTML code, allowing developers to mix PHP code and HTML markup in the same file.
- Support for a wide range of databases: PHP can be used to connect to many different types of databases, including MySQL, Oracle, and PostgreSQL.
- Object-oriented programming (OOP) support: PHP supports OOP concepts such as classes, objects, and inheritance, allowing developers to write modular and reusable code.
- Extensive library of functions: PHP comes with a large standard library of functions that can be used to perform common tasks, such as working with strings, arrays, and files.

4.2 What is a PHP File?

- ➔ A PHP file is a file that contains PHP code and is used to build dynamic web applications and websites. PHP files typically have a ".php" file extension and are stored on a web server.

4.3 What Can PHP do?

- ➔ PHP is a versatile server-side scripting language that can do a wide range of tasks to build dynamic and interactive web applications. Here are some of the things that PHP can do:

- Generate dynamic content.
- Process form data
- Manage sessions and user authentication.
- Interact with databases.
- Build APIs.
- Process files and images.
- Build content management systems (CMS).

4.4 Why PHP?

- ➔ There are several reasons why developers choose PHP over other languages for building web applications:

- Easy to learn.
- Large community and resources.
- Cross-platform compatibility.
- Integrates well with databases.
- Powerful frameworks.
- Server-side scripting.
- Widely used.

4.5 A example of PHP code

→ Here's an example of a PHP script that outputs an HTML page with a greeting:

```
<!DOCTYPE html>

<html>
  <head>
    <title>Greeting</title>
  </head>
  <body>
    <?php
      $name = "John"; // set a variable for the name
      echo "<h1>Hello, $name!</h1>"; // output the greeting
    ?>
  </body>
</html>
```

- In this script, the PHP code sets a variable \$name to "John" and then outputs an HTML heading element (<h1>) with the greeting "Hello, John!" using string interpolation to insert the value of the \$name variable. When this script is executed on the server, it generates an HTML page that displays the greeting.

SOURCE CODE

1. LOGIN PAGE

➔ FILE NAME: login5.php

The source code is copy below:-

```
<?php  
include("header.php");  
  
// Initialize variables  
  
$username = "";  
$password = "";  
$role = "";  
$error = "";  
  
// Check if the form has been submitted  
if ($_SERVER["REQUEST_METHOD"] == "POST") {  
    $username = $_POST["username"];  
    $password = $_POST["password"];  
    $role = $_POST["role"];  
  
    // Perform validation  
    if (empty($username) || empty($password) || empty($role)) {
```

```

$error = "All fields are required.";

} else {

// Connect to the database

$host = "localhost";

$user = "phpmyadmin";

$pass = "root";

$db = "attd_system";

$conn = new mysqli($host, $user, $pass, $db);

if ($conn->connect_error) {

die("Connection failed: " . $conn->connect_error);

}

// Perform the query to retrieve the user from the database

$sql = "SELECT * FROM login WHERE username='$username' AND password='$password' AND role='$role'";

$result = $conn->query($sql);

// Check if a user was found

if ($result->num_rows > 0) {

// Check the user's role

$user = $result->fetch_assoc();

if ($user['role'] == 'admin') {

// Redirect to the home page

header("Location: home.php");

exit;

} else {

```

```
header("Location: userhome.php");
exit;
//echo "User logged in successfully!";
}

} else {

$error = "Invalid username or password.";
}

// Close the database connection
$conn->close();
}
}

?>

<html>
<head>
<link rel="stylesheet" tyle="text/css" href="style.css">

<style>
.login-form {
margin: 0 auto;
max-width: 400px;
padding: 20px;
border: 1px solid #ccc;
border-radius: 5px;
```

```
background-image:linear-gradient(to right, rgba(135,135,135,1),rgba(135,135,135,0.1));  
}  
</style>  
</head>  
<body>  
<div class="container">  
  
<div class="container">  
<form action="login5.php" method="post" class="login-form">  
<?php if ($error != "") { ?>  
<div class="alert alert-danger">  
<?php echo $error; ?>  
</div>  
<?php } ?>  
<div class="form-group">  
<label for="username">USERNAME</label>  
<input name="username" id="username" class="form-control" required>  
</div>  
<div class="form-group">  
<label for="password">PASSWORD</label>  
<input type="password" name="password" id="password" class="form-control" required>  
</div>  
<div class="form-group">  
<label for="role">ROLE</label>  
<select name="role" id="role" class="form-control" required>
```

```

<option value="">Select Role</option>
<option value="admin">Admin</option>
<option value="user">User</option>
</select>
</div>
<div class="form-group">
<input type="submit" name="submit" value="Submit" class="btn btn-primary" required>
</div>
</form>
</div>
</div>
</body>
</html>

```

2. HOME PAGE

→ FILE NAME: home.php

The source code is copy below:-

```

<?php include("header.php");?>

<html>
<title>
Welcome to Attandance Management System
</title>
<head>

```

```
<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">
<link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css" integrity="sha384-ggOyR0iXCbMQv3Xipma34MD+dH/1fQ784/j6cY/iJTQUOhcWr7x9JvoRxT2MZw1T" crossorigin="anonymous">
<link rel="stylesheet" type="text/css" href="style.css">
<style>
body {
font-size: 30px;
}
.container {
max-width: 1000px;
margin: 0 auto;
}
.btn {
font-size: 2.5rem;
background-image: linear-gradient(to right,rgba(135,135,135,1),rgba(135,135,135,0));
}
</style>

</head>

<body>

<div class="container">
<div class="row">
<div class="col-md-6">
```

```

<a href="add3.php" class="btn btn-primary btn-lg btn-block">Add Student</a>
</div>

<div class="col-md-6">
<a href="index.php" class="btn btn-primary btn-lg btn-block">Update Attendance</a>
</div>
</div>

<div class="row mt-5">
<div class="col-md-6">
<a href="view2.php" class="btn btn-primary btn-lg btn-block">View Records</a>
</div>
<div class="col-md-6">
<a href="logout.php" class="btn btn-danger btn-lg btn-block">Logout</a>
</div>
</div>
</div>
</body>
</html>

```

3. ADD STUDENT

➔ FILE NAME: add3.php

The source code is copy below:-

```
<?php
```

```
include("header.php");

include("db.php");

$flag = false;

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

$class = mysqli_real_escape_string($con, $_POST['class']);

$sem = mysqli_real_escape_string($con, $_POST['sem']);

$name = mysqli_real_escape_string($con, $_POST['name']);

$roll = mysqli_real_escape_string($con, $_POST['roll']);

$result = mysqli_query($con, "INSERT INTO attendance (class, sem, student_name, roll_number)
VALUES ('$class','$sem','$name','$roll')");

if ($result) {

$flag = true;

}

}

?>

<html>
<head>
<title>Attendance System - Add Record</title>
<link rel="stylesheet" type="text/css" href="style.css">
28
```

```
<style>

.panel {
width: 100%;

max-width: 500px;

height: auto;

margin: 0 auto;

padding: 20px;

border: 1px solid #ccc;

border-radius: 5px;

background-color: #f0f0f0;

}

</style>

<style>

.logout-btn {

position: absolute;

top: 10px;

right: 10px;

}

</style>

</head>

<body>

<div class="container">

<div class="panel panel-default">
```

```

<div class="panel panel-heading">

<h2>
<a class="btn btn-info" href="home.php">BACK</a>
<a class="btn btn-danger pull-right" href="logout.php">Logout</a>
</h2>

<?php if ($flag) { ?>
<div class="alert alert-success">
<strong>Success!</strong> Record added successfully!
</div>
<?php } ?>

<form action="add3.php" method="post" class="login-form">
<div class="form-group">
<label for="class">CLASS</label>
<select name="class" id="class" class="form-control" required>
<option value="">Select Class</option>
<option value="B.VOC(SD)">B.VOC(SD)</option>
</select>
</div>

<div class="form-group">
<label for="sem">SEMESTER</label>
<select name="sem" id="sem" class="form-control" required>
<option value="">Select Semester</option>
<option value="1st SEM">1st SEM</option>

```

```
<option value="2nd SEM">2nd SEM</option>
<option value="3rd SEM">3rd SEM</option>
</select>
</div>

<div class="form-group">
<label for="name">STUDENT NAME</label>
<input type="text" name="name" id="name" class="form-control" required>
</div>

<div class="form-group">
<label for="roll">ROLL NUMBER</label>
<input type="text" name="roll" id="roll" class="form-control" required>
</div>

<div class="form-group">
<input type="submit" name="submit" value="submit" class="btn btn-primary">
</div>
</form>

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

</body>
```

```
</html>
```

4. UPDATE ATTENDANCE

→ FILE NAME: index.php

The source code is copy below:-

```
<?php
```

```
include("db.php");
```

```
include("header.php");
```

```
$flag = 0;
```

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

```
foreach ($_POST['attendance_status'] as $id => $attendance_status) {
```

```
    $class = $_POST['class'][$id];
```

```
    $class = $_POST['sem'][$id];
```

```
    $student_name = $_POST['student_name'][$id];
```

```
    $roll_number = $_POST['roll_number'][$id];
```

```
    $date = date("Y-m-d H:i:s");
```

```
// Check if the data already exists in the database
```

```
$check_query = mysqli_query($con, "SELECT * FROM attendance_record WHERE  
student_name='$student_name' AND roll_number='$roll_number' AND date='$date'");
```

```
if (mysqli_num_rows($check_query) > 0) {
```



```
margin: 0 auto;  
max-width: 400px;  
padding: 20px;  
border: 1px solid #ccc;  
border-radius: 5px;  
background-color: #f0f0f0; /* set the background color to light gray */  
}  
</style>  
  
</head>  
  
<body>  
<div class="container">  
<div class="panel panel-default">  
<div class="panel panel-heading">  
<h2>  
<a class="btn btn-primary" href="home.php">HOME</a>  
<a class="btn btn-success" href="add3.php">ADD STUDENT</a>  
<a class="btn btn-info" href="view2.php">CHECK RECORD</a>  
<a class="btn btn-danger pull-right" href="logout.php">Logout</a>  
</h2>  
<?php if ($flag) { ?>  
<div class="alert alert-success">  
Attendance Data Inserted Successfully
```

```

</div>

<?php } ?>

<h3><div class="well test-center">Date: <?php echo date("Y-m-d"); ?></div></h3>

</div>

<div class="panel panel-body">

<form action="index.php" method="post">

<table class="table table-striped">

<tr>

<th>Serial Number</th>

<th>Class</th>

<th>Semester</th>

<th>Student Name</th>

<th>Roll Number</th>

<th>Attendance Status</th>

</tr>

<?php while ($row = mysqli_fetch_array($result)) { ?>

<tr>

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

<td><?php echo $row['class']; ?><input type="hidden" name="class[]" value="<?php echo
$row['class']; ?>"></td>

<td><?php echo $row['sem']; ?><input type="hidden" name="sem[]" value="<?php echo $row['sem'];
?>"></td>

<td><?php echo $row['student_name']; ?><input type="hidden" name="student_name[]" value="<?
php echo $row['student_name']; ?>"></td>

```

```

<td><?php echo $row['roll_number']; ?><input type="hidden" name="roll_number[]" value="<?php
echo $row['roll_number']; ?>"></td>

<td>

<input type="radio" name="attendance_status[<?php echo $counter; ?>]" value="Present">Present
<input type="radio" name="attendance_status[<?php echo $counter; ?>]" value="Absent">Absent

</td>
</tr>

<?php $counter++; ?>

<?php } ?>

</table>

<input type="submit" name="submit" value="Submit" class="btn btn-primary">

</form>

</div>

</div>

</div>

</body>

</html>

<?php mysqli_close($con); ?>

```

5. VIEW RECORDS

→ FILE NAME: view2.php

The source code is copy below:-

```
<?php
```

```

include("db.php");

include("header.php");

if(isset($_POST['date']) && !empty($_POST['date'])) {

$date = $_POST['date'];

$result = mysqli_query($con, "SELECT * FROM attendance_record WHERE date LIKE '%$date%'" );

}

?>

<html>

<head>

<title>Attendance Management System</title>

<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">

<link rel="stylesheet" type="text/css" href="style.css">

<style>

.login-form {

margin: 0 auto;

max-width: 400px;

padding: 20px;

border: 1px solid #ccc;

border-radius: 5px;

background-color: #f0f0f0; /* set the background color to light gray */

}

```

```
</style>

</head>

<body>
<div class="container">
<div class="panel panel-default">
<div class="panel panel-heading">
<h2>
<a class="btn btn-success" href="home.php">BACK</a>
<a class="btn btn-danger pull-right" href="logout.php">Logout</a>
</h2>
</div>

<div class="panel panel-body">
<form action="view2.php" method="post">
<div class="form-group">
<label for="date">Select Date:</label>
<input type="date" class="form-control" name="date" required>
</div>
<input type="submit" name="submit" value="Show Record" class="btn btn-primary">
</form>
<hr>
<table class="table table-striped">
```

```

<tr>
<th>Serial Number</th>
<th>Student Name</th>
<th>Roll Number</th>
<th>Attendance Status</th>
<th>Date</th>
</tr>

<?php
$serialnumber = 0;
while ($row = mysqli_fetch_array($result)) {
?>
<tr>
<td><?php echo ++$serialnumber; ?></td>
<td><?php echo $row['student_name']; ?></td>
<td><?php echo $row['roll_number']; ?></td>
<td><?php echo $row['attendance_status']; ?></td>
<td><?php echo $row['date']; ?></td>
</tr>
<?php } ?>
</table>
</div>
</div>
</div>
</body>

```

```
</html>

<?php mysqli_close($con); ?>

*****
```

6. LOGOUT

➔ FILE NAME: logout.php

The source code is copy below:-

```
<?php

session_start();

session_unset();

session_destroy();

header("Location: login5.php");

exit;

?>
```

7. DATABASE CONNECTION

➔ FILE NAME: db.php

The source code is copy below:-

```
<?php

$con=mysqli_connect("localhost","phpmyadmin","root","attd_system") ;

?>
```

8. PAGE HEADER

→ FILE NAME: header.php

The source code is copy below:-

```
<html>

<head>

<!-- Latest compiled and minified CSS -->

<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/css/bootstrap.min.css">

<!-- jQuery library -->

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.3/jquery.min.js"></script>

<!-- Latest compiled JavaScript -->

<script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/js/bootstrap.min.js"></script>

<style>

.header-text {
    font-size: 48px;
    font-weight: bold;
    text-shadow: 2px 2px #ccc;
    margin: 50px 0;
    text-align: center;
    background-image: linear-gradient(to right, rgba(135,135,135,1),rgba(135,135,135,0));
}

</style>

</head>
```

```
<body>

<div class="container">
<h2><div class="well header-text"> ATTENDANCE MANAGEMENT SYSTEM </div></h2>
</div>

</body>
</html>
```

9. EXTERNAL CSS FILE

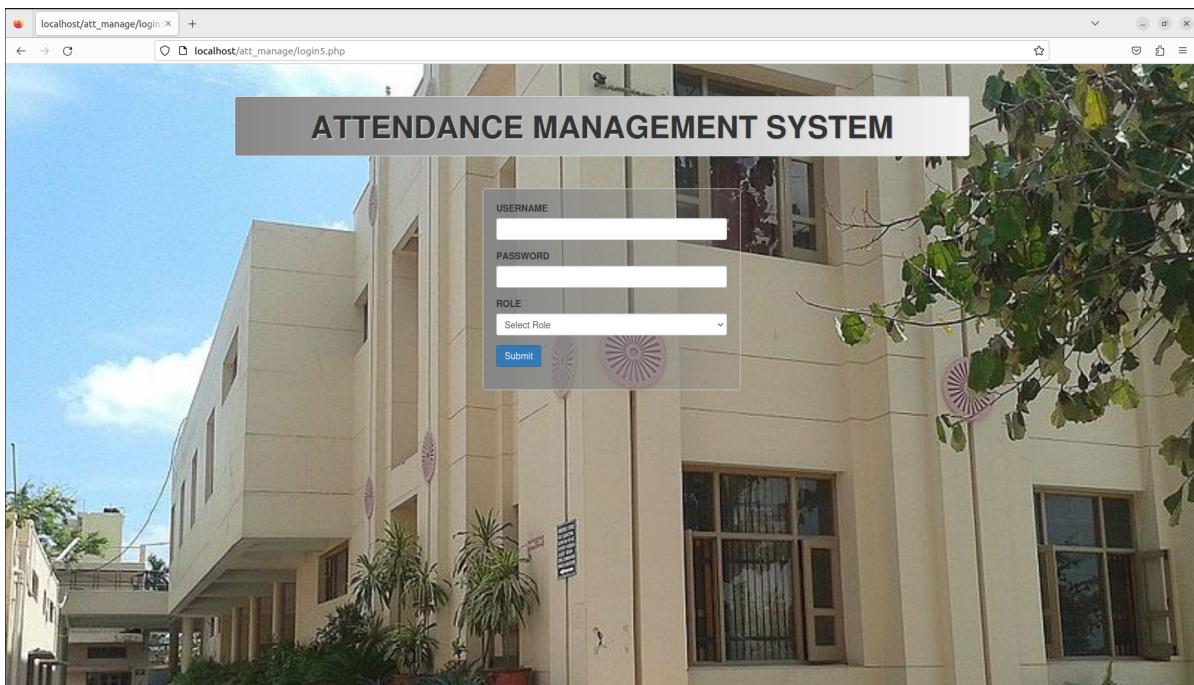
➔ FILE NAME: style.css

The source code is copy below:-

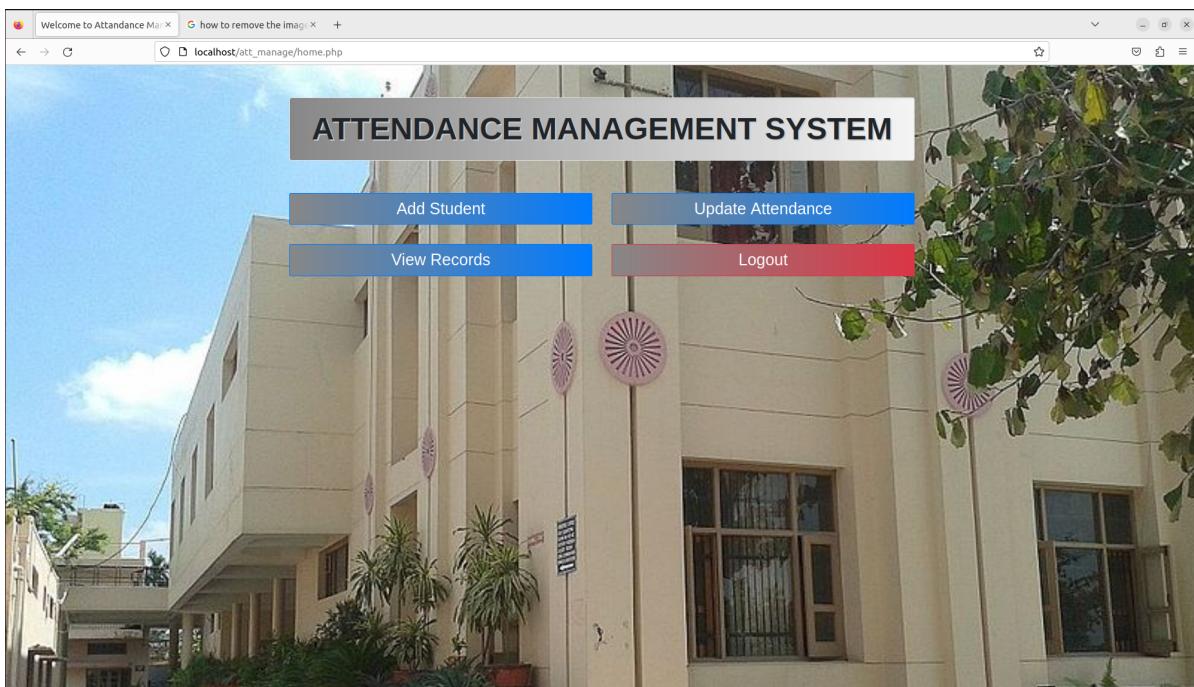
```
body{
background-image: url("pic/sd.jpg");
background-repeat: no-repeat;
background-position: center 5%;
background-size: cover;
}
```

SCREENSHOTS OF PROJECT

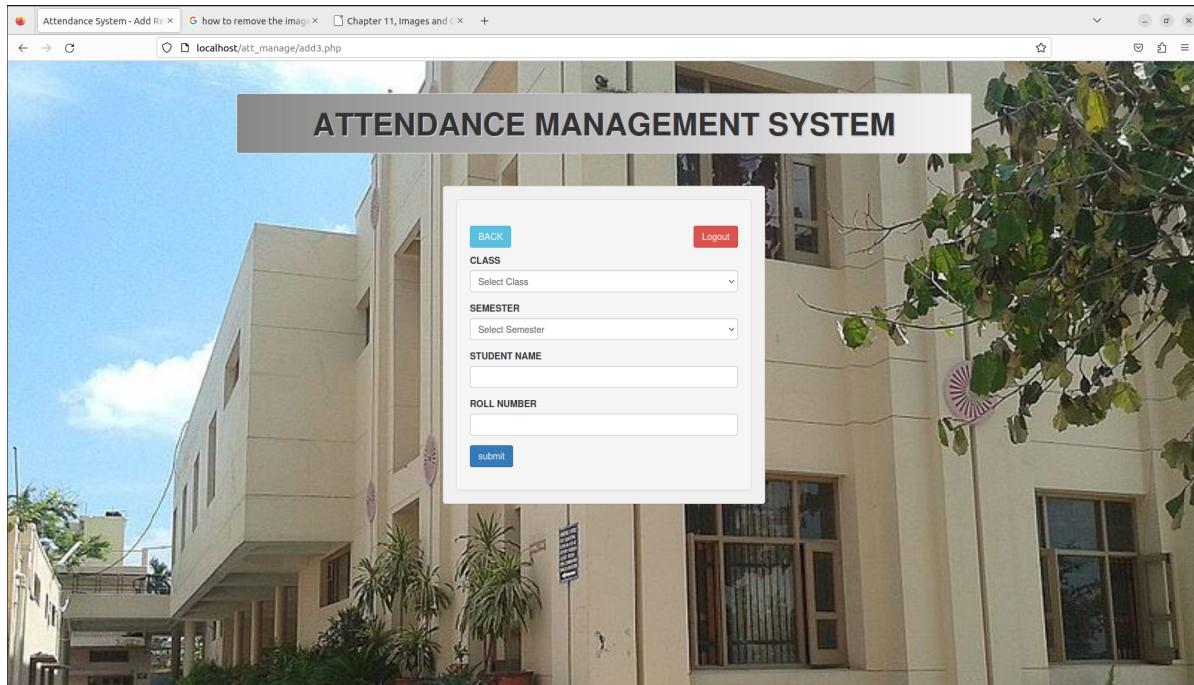
1. LOGIN PAGE



2. HOME PAGE



3. ADD STUDENT PAGE



4. UPDATE ATTENDANCE PAGE

A screenshot of a web browser displaying the 'Attendance Management System'. The main header reads 'ATTENDANCE MANAGEMENT SYSTEM'. Below it, there are three navigation buttons: 'HOME' (blue), 'ADD STUDENT' (green, currently active), and 'CHECK RECORD' (white). A date field shows 'Date: 2023-04-29'. To the right is a 'Logout' button. The main content area displays a table of student attendance records:

5. VIEW RECORDS PAGE

The screenshot shows a web browser window titled "Attendance Management System" with the URL "localhost/att_manage/view2.php". The page has a header with "ATTENDANCE MANAGEMENT SYSTEM" and navigation buttons for "BACK" and "Logout". Below the header is a form with a date input field labeled "Select Date: dd/mm/yyyy" and a "Show Record" button. A table displays 12 student records:

Serial Number	Student Name	Roll Number	Attendance Status	Date
1	Hemant	2001	Present	2023-04-29
2	Manish	2002	Absent	2023-04-29
3	Abhishek	2003	Present	2023-04-29
4	Harsh	2004	Absent	2023-04-29
5	Anuj	2005	Absent	2023-04-29
6	Sorubh	2006	Absent	2023-04-29
7	Sumit	2007	Absent	2023-04-29
8	Harman	2008	Absent	2023-04-29
9	Harshit	2009	Absent	2023-04-29
10	Harshit	2010	Absent	2023-04-29
11	Ankit	2011	Absent	2023-04-29
12	Sahab	2012	Present	2023-04-29

CONCLUSION

The attendance management system is a vital tool for organizations to keep track of their employees' attendance and manage their work schedules effectively. By using this system, organizations can improve their overall productivity, reduce administrative workload, and ensure compliance with labor laws and regulations. The web-based attendance management system that we have developed provides a user-friendly interface for both employees and managers, allowing them to access the system from any location and at any time. It offers features such as real-time attendance tracking, leave management, and employee performance analytics, all of which can be customized to meet the specific needs of the organization. Overall, our attendance management system is a powerful tool that can help organizations streamline their operations, save time and resources, and ensure a more efficient and productive workforce.

BIBLIOGRAPHY/REFERENCE

1. WEBSITES

- ➔ <https://www.w3schools.com/>
- ➔ <https://www.javapoint.com/>
- ➔ <https://www.geeksforgeeks.com/>
- ➔ <https://www.googlemaps.com/>

2. BOOKS

- ➔ Learning PHP, MySQL & Javascript with Jquery, CSS & HTML5
Author: Robin Nixon
- ➔ HTML and CSS: Design and build websites
Author: Jon Duckett

PROJECT DATA DISK