



PAY ROLL MANAGEMENT SYSTEM



A PROJECT REPORT

Submitted by

SHERWIN B (2303811710421146)

in partial fulfillment of requirements for the award of the course

CGB1221-DATABASE MANAGEMENT SYSTEMS

In

COMPUTER SCIENCE AND ENGINEERING

K. RAMAKRISHNAN COLLEGE OF TECHNOLOGY

(An Autonomous Institution, affiliated to Anna University Chennai and Approved by AICTE, New Delhi)

SAMAYAPURAM – 621 112

JUNE- 2025

**K. RAMAKRISHNAN COLLEGE OF TECHNOLOGY
(AUTONOMOUS)**

SAMAYAPURAM – 621 112

BONAFIDE CERTIFICATE

Certified that this project report on “ PAY ROLL MANAGEMENT SYSTEMS ” is the bonafide work of **SHERWIN B (2303811710421146)** who carried out the project work during the academic year 2024 - 2025 under my supervision.

SIGNATURE

Mrs.A.DELPHIN CAROLINA RANI,
M.E.,Ph.D.,

HEAD OF THE DEPARTMENT

PROFESSOR

Department of CSE

K.Ramakrishnan College of Technology
(Autonomous)

Samayapuram–621112.

SIGNATURE

Mr. R. VIGNESH KUMAR, M.E.,

SUPERVISOR

ASSISTANT PROFESSOR

Department of CSE

K.Ramakrishnan College of Technology
(Autonomous)

Samayapuram–621112.

Submitted for the viva-voce examination held on 04.06.2025

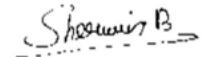
INTERNAL EXAMINER

EXTERNAL EXAMINER

DECLARATION

I declare that the project report on "**PAY ROLL MANAGEMENT SYSTEM**" is the result of original work done by me and best of my knowledge, similar work has not been submitted to "**ANNA UNIVERSITY CHENNAI**" for the requirement of Degree of **BACHELOR OF ENGINEERING**. This project report is submitted on the partial fulfilment of the requirement of the completion of the course **CGB1221 – DATABASE MANAGEMENT SYSTEMS.**

Signature



SHERWIN B

Place: Samayapuram

Date: 04.06.2025

ACKNOWLEDGEMENT

It is with great pride that I express my gratitude and in-debt to my institution “**K.Ramakrishnan College of Technology (Autonomous)**”, for providing me with the opportunity to do this project.

I glad to credit honourable chairman **Dr. K. RAMAKRISHNAN, B.E.,** for having provided for the facilities during the course of my study in college.

I would like to express my sincere thanks to my beloved Executive Director **Dr. S. KUPPUSAMY, MBA, Ph.D.,** for forwarding to my project and offering adequate duration in completing my project.

I would like to thank **Dr. N. VASUDEVAN, M.Tech., Ph.D.,** Principal, who gave opportunity to frame the project the full satisfaction.

I whole heartily thanks to **Dr. A. DELPHIN CAROLINA RANI, M.E.,Ph.D.,** Head of the department, **COMPUTER SCIENCE AND ENGINEERING** for providing her encourage pursuing this project.

I express my deep expression and sincere gratitude to my project supervisor **Mr. R. VIGNESH KUMAR, M.E.,** Department of **COMPUTER SCIENCE AND ENGINEERING**, for his incalculable suggestions, creativity, assistance and patience which motivated me to carry out this project.

I render my sincere thanks to Course Coordinator and other staff members for providing valuable information during the course.

I wish to express my special thanks to the officials and Lab Technicians of my departments who rendered their help during the period of the work progress.

VISION OF THE INSTITUTION

To serve the society by offering top-notch technical education on par with global standards

MISSION OF THE INSTITUTION

- Be a center of excellence for technical education in emerging technologies by exceeding the needs of the industry and society.
- Be an institute with world class research facilities
- Be an institute nurturing talent and enhancing the competency of students to transform them as all-round personality respecting moral and ethical values

VISION OF DEPARTMENT

To be a center of eminence in creating competent software professionals with research and innovative skills.

MISSION OF DEPARTMENT

M1: Industry Specific: To nurture students in working with various hardware and software platforms inclined with the best practices of industry.

M2: Research: To prepare students for research-oriented activities.

M3: Society: To empower students with the required skills to solve complex technological problems of society.

PROGRAM EDUCATIONAL OBJECTIVES

3. PEO1: Domain Knowledge

To produce graduates who have strong foundation of knowledge and skills in the field of Computer Science and Engineering.

2. PEO2: Employability Skills and Research

To produce graduates who are employable in industries/public sector/research organizations or work as an entrepreneur.

3. PEO3: Ethics and Values

To develop leadership skills and ethically collaborate with society to tackle real-world challenges.

PROGRAM SPECIFIC OUTCOMES (PSOs)

PSO 1: Domain Knowledge

To analyze, design and develop computing solutions by applying foundational concepts of Computer Science and Engineering.

PSO 2: Quality Software

To apply software engineering principles and practices for developing quality software for scientific and business applications.

PSO 3: Innovation Ideas

To adapt to emerging Information and Communication Technologies (ICT) to innovate ideas and solutions to existing/novel problems

PROGRAM OUTCOMES (Pos)

Engineering students will be able to:

- 1. Engineering knowledge:** Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
- 2. Problem analysis:** Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences
- 3. Design/development of solutions:** Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations
- 4. Conduct investigations of complex problems:** Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions

- 5. Modern tool usage:** Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations
- 6. The engineer and society:** Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice
- 7. Environment and sustainability:** Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development
- 8. Ethics:** Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
- 9. Individual and team work:** Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.
- 10. Communication:** Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.
- 11. Project management and finance:** Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.
- 12. Life-long learning:** Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

ABSTRACT

The Payroll Management System is a web-based application designed to streamline and automate the payroll processing tasks within an organization. It is developed using PHP for server-side scripting, MySQL for database management, and Bootstrap for creating a responsive user interface. The system provides secure login and registration features with password hashing and session management to ensure only authorized access. It allows administrators to perform CRUD operations on employee and payroll records efficiently. A dashboard displays key metrics such as the total number of employees and the total amount of payroll disbursed using SQL aggregate functions. The use of PDO (PHP Data Objects) enhances database security by preventing SQL injection attacks. This system reduces the dependency on manual payroll methods, increases accuracy, and saves time in salary disbursement processes. The application is designed to be scalable, allowing for future enhancements like leave tracking, payslip generation, email notifications, and role-based access controls. The main objective of this project is to eliminate manual payroll processing, reduce errors, and save time by providing a centralized platform for payroll operations. With features such as password hashing, employee and payroll CRUD operations, and dashboard summaries, the system demonstrates an efficient solution for small to medium-sized businesses.

ABSTRACT WITH POs AND PSOs MAPPING

CO 5: BUILD DATABASE MANAGEMENT SYSTEM APPLICATION FOR SOLVING REAL- TIME PROBLEMS

ABSTRACT	POs MAPPED	PSOs MAPPED
This project focuses on developing a Payroll Management System that automates the process of managing employee data and generating payroll information. By utilizing PHP as the backend language, MySQL as the database, and Bootstrap for styling, the system ensures both functionality and responsiveness. The application enables secure user registration and login, with features like password encryption and session control. It allows for efficient handling of employee records and salary disbursements, offering modules for data entry, updating, and deletion. Real-time dashboard statistics give users a clear overview of payroll and workforce data.	PO1 -3 PO2 -3 PO3 -3 PO4 -3 PO5 -3 PO6 -3 PO7 -3 PO8 -3 PO9 -3 PO10 -3 PO11-3 PO12 -3	PSO1 -3 PSO2 -3 PSO3 -3

Note: 1- Low, 2-Medium, 3- High

TABLE OF CONTENTS

CHAPTER NO.	TITLE	PAGE NO.
	ABSTRACT	viii
	LIST OF FIGURES	xi
	LIST OF ABBREVIATIONS	xii
1	INTRODUCTION	1
	1.1 Objective	1
	1.2 Overview	1
	1.3 SQL and Database concepts	1
2	PROJECT METHODOLOGY	4
	2.1 Proposed Work	4
	2.2 Block Diagram	5
3	MODULE DESCRIPTION	6
	3.1 User Authentication Module	6
	3.2 Employee Management Module	6
	3.3 Payroll Management Module	6
	3.4 Dashboard Module	7
	3.5 Database Connectivity and Security Module	7
4	CONCLUSION & FUTURE SCOPE	8
	4.1 Conclusion	8
	4.2 Future Scope	8
5	APPENDIX A SOURCE CODE	10
	APPENDIX B SCREENSHOTS	17
	REFERENCES	19

LIST OF FIGURES

FIG NO.	TITLE	PAGE NO.
2.1	Block Diagram	5

LIST OF ABBREVIATIONS

- PHP - Hypertext Preprocessor
RBAC - Role-Based Access Control
SQL - Structured Query Language
RDBMS - Relational Data Base Management System
1NF - First Normalization form
2NF - Second Normalization form
3NF - Third Normalization form

CHAPTER 1

INTRODUCTION

1.1 Objective

The primary objective of the Payroll Management System is to develop a robust application that automates the management of employee data, salary calculations, and payroll processing. The system ensures accuracy, consistency, and efficiency in managing employee salaries, generating payslips, tracking attendance, and handling payroll records. It reduces manual workload and minimizes the chances of errors in salary processing.

1.2 Overview

The Payroll Management System is a web-based application developed using PHP, MySQL (XAMPP), HTML, CSS, JavaScript, and Bootstrap for responsive design. It is designed to handle core payroll functions such as:

- Employee registration and details management
- Salary generation based on employee role and monthly input
- Payroll record keeping (monthly payments, history)
- User authentication for admin-level security
- Dashboard showing summaries like total employees and total payments made

The system is user-friendly, secure, and scalable for small to medium organizations that want a centralized payroll solution.

1.3 SQL and Database Concepts

◆ Basic SQL & Database Concepts Used:

1. **Database:** A structured collection of employee and payroll-related data.
2. **Tables:**

- users – for login and registration
- employees – for storing employee information
- payroll – for tracking employee payments

3. **SQL Statements Used:**

- CREATE TABLE, INSERT, SELECT, UPDATE, DELETE
- JOIN (to combine data from payroll and employees)

- COUNT(), SUM() for dashboard summaries

4. Normalization: The database is normalized to reduce redundancy—employee and payroll data are kept in separate linked tables.

5. Constraints:

- PRIMARY KEY on IDs
- FOREIGN KEY in payroll table linking to employee ID

6. Data Integrity:

- Using proper data types (INT, VARCHAR, DECIMAL)
- Validation through PHP and form controls

◆ **Project-Specific SQL Concepts:**

- **Secure User Authentication:** Passwords are hashed and stored securely using PHP's password_hash() and verified with password_verify().
- **Relational Design:** The payroll system uses relational database design principles.
- For example:
 - employee_id in the payroll table is a foreign key that references the employees table.
- **Dashboard Aggregations:**
 - SELECT COUNT(*) FROM employees — counts total employees
 - SELECT SUM(amount) FROM payroll — calculates total salary paid
- **Security Concepts:**
 - Passwords stored using password hashing.
 - User input validated before database entry.
- **Data Types:**
 - VARCHAR for names/emails
 - INT for IDs
 - DECIMAL(10,2) for salary/amount
 - DATE or VARCHAR for payroll month

- **Form to Database Interaction:**

- Frontend forms are connected to backend using PHP and PDO (PHP Data Objects) for secure DB interaction.
- Add client-side (HTML/JS) and server-side (PHP) validations.
- Keep DB connection code in a separate file (e.g., db.php) and include it
- Always use prepared statements (as shown) to prevent SQL injection.
- After successful insertion, show a confirmation message or redirect to a summary/dashboard page.
- Never trust user input; always validate both client and server side.
- Create reusable functions or classes for repetitive tasks like inserting or updating records.

- **Error Handling:**

- Try-catch blocks in PHP used to handle duplicate entries and other SQL exceptions
- Ensure all user inputs (e.g., salary amount, tax entries, leave entries) are validated before processing to prevent invalid data errors.
- Provide clear and user-friendly error messages that help users understand what went wrong and how to fix it.
- Check for empty fields, incorrect formats (e.g., numbers for salary), and invalid data types before processing
- If some form fields are optional (like bonuses), set default values (e.g., 0) to avoid NULL-related issues.

CHAPTER 2

PROJECT METHODOLOGY

2.1 Proposed Work

The proposed work focuses on designing and implementing a web-based Payroll Management System that automates the process of managing employee records and salary payments. The system aims to improve accuracy, efficiency, and transparency in payroll processing while minimizing manual errors and administrative overhead.

◇ Key Features of the Proposed System:

1. User Authentication

- Secure login and registration for administrators using hashed passwords.
- Session management to restrict unauthorized access.

2. Employee Management Module

- Add, view, and delete employee details such as name, position, and salary.
- Simple and responsive interface for easy employee record handling.

3. Payroll Management Module

- Allows entry of monthly payroll for each employee.
- Captures amount paid, employee ID, and month of payment.
- Data linked with employee details using relational database principles.

4. Dashboard Summary

- Displays the total number of employees and total payroll paid.
- Real-time stats using SQL COUNT() and SUM() functions.

5. Database Connectivity using PDO

- Secure communication between the web application and MySQL database using PHP Data Objects (PDO).

6. Responsive Design

- Use of Bootstrap 5 to ensure mobile-friendly, user-responsive interface.

7. Data Validation

- Both client-side and server-side validation to ensure only valid data is stored.

8. Error Handling

- Use of try-catch blocks for safe execution of SQL queries and handling duplicates or constraint violations.

2.2 Block Diagram

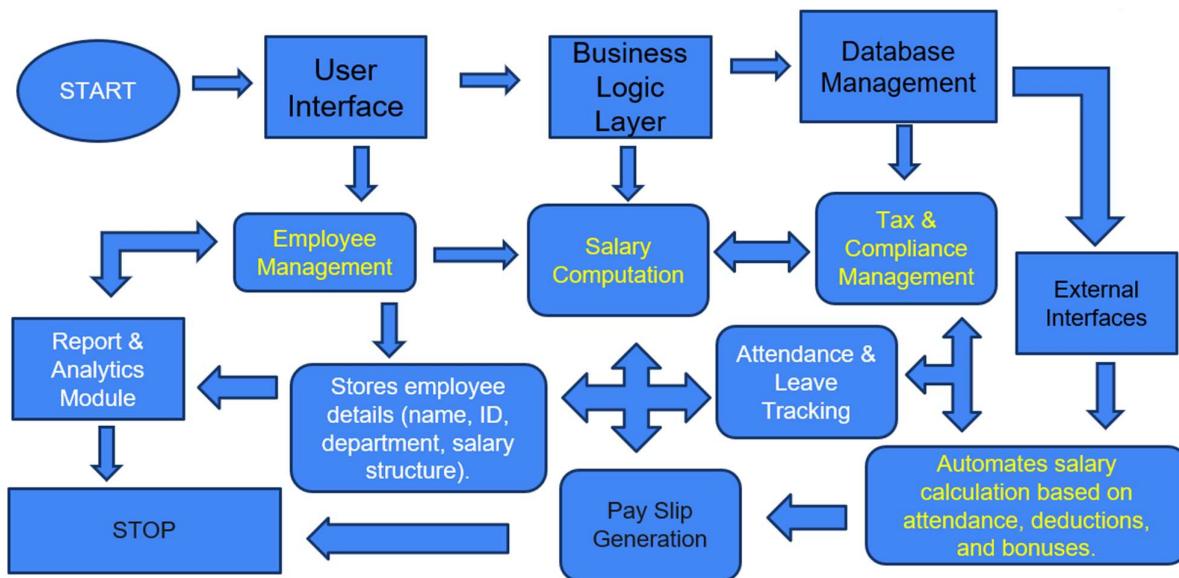


Fig 2.1

CHAPTER 3

MODULE DESCRIPTION

3.1 Module 1 : User Authentication Module

This module handles the login and registration process for the administrator. It includes:

- User registration with validation and password hashing using `password_hash()`.
- Login using `password_verify()` to authenticate the user.
- Session management to restrict access to authorized users only.
- Logout functionality to end the session securely.
- Session management.

3.2 Module 2 : Employee Management Module

This module allows the admin to manage employee data. It includes:

- Adding new employee details (name, position, salary).
- Displaying a list of all employees.
- Deleting employee records.
- Secure storage of employee data in the employees table
- Store personal details (name, DOB, contact), job details (position, department), and bank details.

3.3 Module 3 : Payroll Management Module

This module processes and stores payroll transactions. It includes:

- Assigning salary payments to employees based on employee ID.
- Input for pay month and amount.
- Viewing all payroll records with employee names using SQL JOIN.
- Deleting specific payroll entries.

3.4 Module 4 : Dashboard Module

This module provides an overview of system statistics. It displays:

- Total number of employees (COUNT() function).
- Total payroll amount paid (SUM() function).
- Helps the admin monitor payroll activity at a glance.
- Handle revisions in salary structure.
- Record daily attendance manually or integrate biometric system.

3.5 Module 5 : Database Connectivity and Security Module

This module ensures safe and secure communication between the application MySQL database. It includes:

- PDO (PHP Data Objects) for executing database queries securely.
- Prepared statements to prevent SQL injection.
- Exception handling for query errors and duplicate entries.
- Implement PDO (PHP Data Objects) or MySQLi for secure and flexible database connections (if using PHP).
- Enable Create, Read, Update, Delete functionality for employee records, payroll entries, attendance logs, etc.
- Store DB credentials in a separate config file or .env (never hardcode in main code).
- Supports prepared statements (helps prevent SQL injection).

CHAPTER 4

CONCLUSION AND FUTURE ENHANCEMENT

CONCLUSION

The Payroll Management System successfully automates the essential payroll processes such as employee registration, salary management, and monthly payroll generation. Through a secure login system and user-friendly interface, the administrator can efficiently manage employee records and salary transactions with reduced manual errors. The use of PHP with PDO, MySQL, and Bootstrap ensures smooth functionality, responsive design, and secure data handling.

This project demonstrates how digital solutions can streamline payroll operations, improve accuracy, and save administrative time and effort.

FUTURE ENHANCEMENT

To further enhance the capabilities of the system, the following improvements can be considered:

1. Role-Based Access Control

- Add multiple user roles (e.g., HR, Accountant) with different privileges.
- On profile changes (e.g., updated bank details).

2. Payroll Report Generation (PDF/Excel)

- Generate printable reports for monthly salary disbursements.
- When a new employee is added.

3. Employee Self-Service Portal

- Allow employees to log in, view payslips, and update personal information.
- On successful salary generation.

4. Email Notification System

- Send automatic emails to employees regarding salary credits or updates.
- Notify employees of salary credits, payslip availability, leave approvals, or profile updates.

5. Leave and Attendance Integration

- Include leave management to automate salary deduction based on attendance.
- Implement user roles such as Admin, HR, and Employee with different access permissions for enhanced security and functionality.
- Support for various leave types (paid leave, sick leave, casual leave, unpaid leave)

with customizable policies.

6. Cloud Deployment

- Host the system online for remote access and scalability.
- Easily scale the system up or down based on the number of employees or business needs.
- Support multiple branches or organizations from a single platform.
- Schedule regular automated backups on the cloud infrastructure.

7. Data Backup and Restore Feature

- Implement backup solutions for data safety and recovery.
- Store backups in encrypted formats to prevent unauthorized access.
- Keep logs of backup and restore operations for security and compliance tracking.
- Notify admins about backup success/failure via email or SMS.
- Ensures compliance with labor and tax regulations that require historical payroll data retention.
- Protects against data loss due to system crashes, accidental deletions, malware, or hardware failures.

APPENDIX A

SQL CODE:

```
CREATE TABLE users (
    id INT AUTO_INCREMENT PRIMARY KEY,
    username VARCHAR(50) NOT NULL UNIQUE,
    email VARCHAR(100),
    password VARCHAR(255) NOT NULL
);
```

```
CREATE TABLE employees (
    id INT AUTO_INCREMENT PRIMARY KEY,
    name VARCHAR(100),
    position VARCHAR(100),
    salary DECIMAL(10,2)
);
```

```
CREATE TABLE payroll (
    id INT AUTO_INCREMENT PRIMARY KEY,
    employee_id INT,
    pay_month VARCHAR(20),
    amount DECIMAL(10,2),
    FOREIGN KEY (employee_id) REFERENCES employees(id)
);
```

PHP CODE:

```
<?php
session_start();
$host = 'localhost';
$db = 'payroll_db';
$user = 'root';
$pass = "";
$dsn = "mysql:host=$host;dbname=$db;charset=utf8mb4";
$pdo = new PDO($dsn, $user, $pass);

// Handle Register
if (isset($_POST['register'])) {
    $uname = $_POST['username'];
    $email = $_POST['email'];
    $password = password_hash($_POST['password'], PASSWORD_DEFAULT);
    $stmt = $pdo->prepare("INSERT INTO users (username, email, password) VALUES (?, ?, ?)");
    try {
        $stmt->execute([$uname, $email, $password]);
        echo "<script>alert('Registration successful!');</script>";
    } catch (PDOException $e) {
        echo "<script>alert('Username already taken.');//";
    }
}

// Handle Login
if (isset($_POST['login'])) {
    $uname = $_POST['username'];
    $password = $_POST['password'];
    $stmt = $pdo->prepare("SELECT * FROM users WHERE username=?");
}
```

```

$stmt->execute([$uname]);
$user = $stmt->fetch();
if ($user && password_verify($password, $user['password'])) {
    $_SESSION['user'] = $user['username'];
} else {
    echo "<script>alert('Invalid login credentials.');//</script>";
}
}

// Handle Logout
if (isset($_GET['logout'])) {
    session_destroy();
    header("Location: index.php");
    exit;
}

// CRUD Operations for Employees
if (isset($_POST['add_employee'])) {
    $stmt = $pdo->prepare("INSERT INTO employees (name, position, salary) VALUES (?, ?, ?)");
    $stmt->execute($_POST['name'], $_POST['position'], $_POST['salary']);
}

if (isset($_GET['delete_emp'])) {
    $stmt = $pdo->prepare("DELETE FROM employees WHERE id=?");
    $stmt->execute($_GET['delete_emp']);
}

// CRUD Operations for Payroll
if (isset($_POST['add_payroll'])) {

```

```

$stmt = $pdo->prepare("INSERT INTO payroll (employee_id, pay_month, amount) VALUES
(?, ?, ?)");
$stmt->execute([$_POST['emp_id'], $_POST['month'], $_POST['amount']]);
}

if (isset($_GET['delete_pay'])) {
    $stmt = $pdo->prepare("DELETE FROM payroll WHERE id=?");
    $stmt->execute([$_GET['delete_pay']]);
}
?>

<!DOCTYPE html>
<html>
<head>
    <title>Payroll Management</title>
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css" rel="stylesheet">
    <style>body { padding: 30px; }</style>
</head>
<body>
<div class="container">

<?php if (!isset($_SESSION['user'])): ?>
<div class="row">
    <div class="col-md-6">
        <h3>Login</h3>
        <form method="POST">
            <input name="username" class="form-control mb-2" required placeholder="Username">
            <input type="password" name="password" class="form-control mb-2" required

```

```

placeholder="Password">

    <button name="login" class="btn btn-primary">Login</button>

</form>

</div>

<div class="col-md-6">

    <h3>Register</h3>

    <form method="POST">

        <input name="username" class="form-control mb-2" required placeholder="Username">

        <input name="email" type="email" class="form-control mb-2" required placeholder="Email">

        <input type="password" name="password" class="form-control mb-2" required placeholder="Password">

        <button name="register" class="btn btn-success">Register</button>

    </form>

</div>

</div>

<?php else: ?>

    <h3>Welcome, <?= $_SESSION['user'] ?> <a href="?logout=1" class="btn btn-danger btn-sm">Logout</a></h3>

<!-- Dashboard -->

<h4>Dashboard</h4>

<?php

$empCount = $pdo->query("SELECT COUNT(*) FROM employees")->fetchColumn();

$payTotal = $pdo->query("SELECT SUM(amount) FROM payroll")->fetchColumn();

?>

<div class="row mb-4">

    <div class="col-md-6"><div class="alert alert-info">Total Employees: <?= $empCount ?></div></div>

```

```

<div class="col-md-6"><div class="alert alert-success">Total Payroll Paid: ₹<?=
number_format($payTotal,2) ?></div></div>

</div>

<!-- Employee Management -->
<h4>Employee Management</h4>
<form method="POST" class="row g-2">
    <input name="salary" required class="form-control" type="number" step="0.01"
placeholder="Salary">
    <button name="add_employee" class="btn btn-primary mt-2">Add Employee</button>
</form>

<table class="table mt-3">
    <tr><th>ID</th><th>Name</th><th>Position</th><th>Salary</th><th>Action</th></tr>
    <?php foreach ($pdo->query("SELECT * FROM employees") as $row): ?>
        <tr>
            <td><?= $row['id'] ?></td><td><?= $row['name'] ?></td>
            <td><?= $row['position'] ?></td><td>₹<?= $row['salary'] ?></td>
            <td><a href="?delete_emp=<?= $row['id'] ?>" class="btn btn-sm btn-
danger">Delete</a></td>
        </tr>
    <?php endforeach; ?>
</table>

<!-- Payroll Management -->
<h4>Payroll Management</h4>
<form method="POST" class="row g-2">
    <select name="emp_id" class="form-control" required>
        <option value="">Select Employee</option>
    <?php foreach ($pdo->query("SELECT * FROM employees") as $emp): ?>

```

```

<option value=<?= $emp['id'] ?>><?= $emp['name'] ?></option>

<?php endforeach; ?>

</select>

<input name="month" required class="form-control" placeholder="Month">

<input name="amount" required class="form-control" type="number" step="0.01"
placeholder="Amount">

<button name="add_payroll" class="btn btn-success mt-2">Add Payroll</button>

</form>

<tr><th>ID</th><th>Employee</th><th>Month</th><th>Amount</th><th>Action</th></tr>

<td><?= $row['id'] ?></td><td><?= $row['name'] ?></td>
<td><?= $row['pay_month'] ?></td><td>₹<?= $row['amount'] ?></td>
<td><a href=?>delete_pay=<?= $row['id'] ?>" class="btn btn-sm btn-danger">Delete</a></td>
</tr>

<?php endforeach; ?>

</table>

<?php endif; ?>

</div>

</body>

</html>

```

APPENDIX B

Login

Register

DATABASE CONNECTIVITY

Extra options

				id	username	email	password
<input type="checkbox"/>				1	sherwin	sherwin.cs23@krct.ac.in	\$2y\$10\$GiROl8MMGO52/nwexasWZu9f2s5.By9GITUa8cqp4lx...
<input type="checkbox"/>				2	siva	siva@123	\$2y\$10\$4rfvNQ/q9oAHPsxeVSfvfuCwoJZf9peSzxJFEgwHn9T...
<input type="checkbox"/>				4	anbu	anbu@gmail.com	\$2y\$10\$lyZNn6aimVzfQNainezBl.A246d3dYgcv7uluzlIWaf...
<input type="checkbox"/>				5	admin	sherwin.cs23@krct.ac.in	\$2y\$10\$D/Dya6VSs03EqK2r0aWYi.uUIVRNweO5UQYvIIJA.Z...

Welcome, sherwin [Logout](#)

[Dashboard](#)

Total Employees: 5

Total Payroll Paid: ₹1,120,000.00

Employee Management

[Add Employee](#)

ID	Name	Position	Salary	Action
5	siva surya	manager	₹5000000.00	Delete
6	SHERWIN B	manager	₹2000000.00	Delete
7	SHERWIN B	manager	₹5000.00	Delete
8	Nirmal Prabhu B	manager	₹3000000.00	Delete
9	anbu carloin	homemaker	₹300000.00	Delete

Payroll Management

Select Employee				
Month				
Amount				
Add Payroll				
ID	Employee	Month	Amount	Action
2	siva surya	3	₹60000.00	<button>Delete</button>
3	siva surya	3	₹60000.00	<button>Delete</button>
4	Nirmal Prabhu B	12	₹500000.00	<button>Delete</button>
5	Nirmal Prabhu B	12	₹500000.00	<button>Delete</button>
6	anbu carloin	25	₹0.00	<button>Delete</button>

DATABASE CONNECTIVITY

			id	employee_id	pay_month	amount
<input type="checkbox"/>	Edit	Copy	Delete	2	5 3	60000.00
<input type="checkbox"/>	Edit	Copy	Delete	3	5 3	60000.00
<input type="checkbox"/>	Edit	Copy	Delete	4	8 12	500000.00
<input type="checkbox"/>	Edit	Copy	Delete	5	8 12	500000.00
<input type="checkbox"/>	Edit	Copy	Delete	6	9 25	0.00

			id	name	position	salary
<input type="checkbox"/>	Edit	Copy	Delete	5 siva surya	manager	5000000.00
<input type="checkbox"/>	Edit	Copy	Delete	6 SHERWIN B	manager	2000000.00
<input type="checkbox"/>	Edit	Copy	Delete	7 SHERWIN B	manager	5000.00
<input type="checkbox"/>	Edit	Copy	Delete	8 Nirmal Prabhu B	manager	3000000.00
<input type="checkbox"/>	Edit	Copy	Delete	9 anbu carloin	homemaker	300000.00

REFERENCES:

- Database Management Systems – Raghu Ramakrishnan, Johannes Gehrke
- Database System Concepts – Abraham Silberschatz, Henry Korth, S. Sudarshan
- <https://www.w3schools.com/php/> — PHP and MySQL tutorials
- <https://www.mysql.com/> — Official MySQL documentation
- <https://getbootstrap.com/> — Bootstrap framework for responsive design
- <https://www.tutorialspoint.com/php/> — PHP reference materials
- YouTube – "*PHP MySQL CRUD Tutorial*" by *CodeWithHarry*
- YouTube – "*PDO Tutorial for Beginners*" by *Programming with Mosh*
- Stack Overflow – <https://stackoverflow.com/> — Community for debugging and learning
- XAMPP – <https://www.apachefriends.org/> — Local server environment setup