CHAPTER 1

INTRODUCTION

1.1 INTRODUCTION

The project "Payroll management system" has been developed to overcome the problems faced in the practicing of manual system. This software out built to eliminate and, in some cases, reduce the hardships faced by the existing system. Moreover, this is designed for particular need of the company to carry out its operations in a smooth and effective manner.

This web application is reduced as much as possible to avoid errors while entering data. It also provides error message while entering invalid data. It is user-friendly as no formal knowledge is required to use the system.

Human resource challenges are faced by every organization which has to be overcome by the organization. Even organization has different employee and payroll management needs. Therefore, I have design exclusive payroll management system that are adapted to the organization's managerial requirements.

A user-friendly GUI (Graphical User interface) based software "Payroll management system" has been developed for organization, keeping in view the requirements of employees to prepare salary. The objective of the project was to computerize the payroll involved in the complex task of salary preparation.

By computerization it implies that all process handling is entirely being done by the computer. All the records necessary for the processes involved in the salary calculation are stored in a computer-based record keeping system using a back-end tool.

All the processes handling is done at the front-end by using a front-end. The project provides sample facilities to its users. The most important of the facilities are the report generation and printing which provides user with the hard copies of the data also.

The Payroll Management System deals with the financial aspects of employee's salary, allowances, deductions, gross pay, net pay etc. and generation of pay-slips for a specific period.

The outstanding benefit of Payroll Management System is its easy implementation. Other advantages of Payroll Management System are its extensive features and reports.

1.2 OBJECTIVE

The main objective of the analysis is to provide better transperancy on payroll data of commenting and accuracy.

- Designing a relational database schema for payroll management.
- Implementing create, read, delete, update operations.
- Ensuring data integrity and security.

CHAPTER 2

SYSTEM STUDY

2.1 EXISISTING SYSTEM

- In the existing system the user has to search in a search engine for the data, has to see through various website. Even the searched website may not contain required data. If the desired website does not contain, the user has to visit another website. It is more time consuming.
- The data assurance is less because the data is scattered in various websites.
- Here all data required has to be entered manually to get a desired output of the data. There are lot of spelling mistakes.
- This system will not allow the user to save his results for future reference as one will use search engine to get the results.

Disadvantages of Existing System

- It is time consuming
- It involves more man power

2.2 PROPOSED SYSTEM

The manual system of searching data in search engine has been around for many years. If paper work is made, maintain all the data is a huge burden. Here retrieving the searched data is not nearly impossible. This requires lot of man power.

Limitations of the existing system area time consuming, error prone and inaccuracy.

Hence the data analysis is proposed, where not only the above-mentioned limitations can overcome, but also it can analyze the data with various patterns to give better result.To overcome the above said problem designing system entitled as payroll management system.

CHAPTER 3

HARDWARE AND SOFTWARE REQIUREMENTS

3.1 HARDWARE REQIUREMENTS

Processor	: Intel Core i3 or higher
RAM	: 4GB
Hard disk	: 50GB

3.2 SOFTWARE REQIUREMENTS

Operating System	: Windows 10
Front End	: HTML, CSS, JavaScript
Back End	: MySQL
Programming Language	: R
API	: Plumber
Web Server	: Apache 2

CHAPTER 4

SOFTWARE REQUIREMENT SPECIFICATION

4.1 INTRODUCTION

A software requirements specification (SRS) is a detailed description of a software system to be developed with its functional and non-functional requirements. The SRS is developed based the agreement between customer and contractors. It may include the use cases of how user is going to interact with software system

The software requirement specification document consistent of all necessary requirements required for project development. To develop the software system, we should have clear understanding of Software system. To achieve this, we need to continuous communication with customers to gather all requirements.

4.2 TECHNOLOGIES USED

4.2.1 HTML

Hypertext Markup Language (HTML) is the standard markup language for documents designed to be displayed in a web browser.

Web browser receive HTML documents from a web server or from local storage and render the documents into multimedia web pages. HTML describes the structure of a web page semantically and originally included cues for the appearance of the document.

HTML elements are the building blocks of HTML pages. With HTML constructs, images and other objects such as interactive forms may be embedded into the rendered page. HTML provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items. HTML elements are delineated by tags,

written using angle brackets. Tags such as <img>and <input> directly introduce content into the page. Other tags such as <p>surround and provide information about document text and may include other tags as sub-elements. Browsers do not display the HTML tags, but use them to interpret the content of the page.

HTML can embed programs written in a scripting language such as Java Script, which affects the behaviour and content of web pages. Inclusion of CSS defines the look and layout of content. The World Wide Web consortium (W3C), former maintainer of the HTML and current maintainer of the CSS standards, has encouraged the use of CSS over explicit presentational HTML since 1997.

ATTRIBUTES

<!DOCTYPE>	Defines the document type
<html>	Defines an HTML document
<head>	Contains metadata/information for the document
<title>	Defines a title for the document
<body>	Defines the document's body
<h1> to <h6>	Defines HTML headings

<p>	Defines a paragraph
 	Inserts a single line break
<hr>	Defines a thematic change in the content
<!--...-->	Defines a comment

4.2.2 PHP

PHP is an open-source, server-side programming language that can be used to create websites, applications, customer relationship management systems and more. It is a widely-used general-purpose language that can be embedded into HTML. This functionality with HTML means that the PHP language has remained popular with developers as it helps to simplify HTML code.

PHP stands for ‘PHP: Hypertext Preprocessor’, with the original PHP within this standing for ‘Personal Home Page’. The acronym has changed as the language developed since its launch in 1994 to more accurately reflect its nature.

Since its release, there have been 8 versions of PHP, as of 2022, with version 8.1 currently a popular choice among those using the language on their websites.

PHP USED FOR

PHP programming can be used to create most things that a software developer needs. However, there are three main areas in which it thrives.

- Server-side scripting

Server-side Script is PHP’s main strength. If you are just learning to code and want to explore server-side scripting, PHP is a great language to learn. To get cracking with PHP server-side scripting you’ll need to have a PHP parser, web server and web browser.

- Command-line scripting

Command-line scripting is ideal for scripts made using cron (Linux) or Task Scheduler (Windows). It is also great for simple text processing.

- Writing desktop applications

PHP is probably not the best language to use to create desktop applications but for the advanced web developer, it provides you with many more options than its competitors.

Advantages of PHP

- PHP has endured as a popular programming language for almost three decades owing to the number of benefits it offers users and developers. The most significant of these are:
- It is cross-platform: As mentioned above, PHP can be run on any major operating system, so you don’t have to worry about compatibility. This also means that you can work on a project team and not have to worry about members being able to access code!
- It’s open-source: Open source means that PHP is a free programming language, so you don’t have to worry about additional costs when building your project
- Many developers understand it: The longevity of PHP means there’s already a large community out there to support you, and a large amount of legacy code to help you get started with projects.
- It works brilliantly with HTML: PHP can help simplify your projects and works seamlessly with this dominant programming language.
- As a language, it is relatively easy to learn: Because of its simplicity, PHP is quicker and easier to pick up than some other alternatives.
- There are plenty of tools available: There are tools to help you with just about anything for PHP, from integration, code hinting, syntax highlighting and more, there is a lot of support for you.
- It offers great load times for websites: With website performance becoming more and more tied to speed, PHP’s quick load can really help you succeed.

4.2.3 MY SQL

MySQL Workbench is a visual database design tool that integrates SQL development, administration, database design, creation and maintenance into a single integrated development environment for the MySQL database system.

Basic queries

- SELECT. The SELECT query is the standard means by which to retrieve information from the database.
- INSERT. The INSERT statement is used to insert data into a particular table of a database
- UPDATE. The UPDATE query is used to update a particular table of a database
- DELETE. The DELTE query is used to delete a particular table from a database
- Select * from table_name;
- Insert into table table_name(column_name);
- Update table table_name;
- Delete table table_name;

4.3 NON -FUNCTIONAL REQIUREMENTS

Non-functional requirements is a requirement that specifies criteria that can be used to judge the operation of a system, rather than specific behaviors. Functional requirements define

what a system is supposed to do whereas Non-functional requirements defines how a system Is supposed to be. The Non-functional requirements are the constraints or the environment in which the software is developed

- Reliability
- Security
- Usability
- Interoperability

4.4 FUNCTIONAL REQIUREMENTS

User

- Login through employee payroll webpage.
- Add the employee details.
- View the employee details.
- Sign out.

CHAPTER 5

SYSTEM DESIGN

Designing the system for a payroll management system involves creating a robust database schema and defining the functionalities it will support.

5.1 DATABASE SCHEMA DESIGN

1. Employees Table

- Fields: employee name, salary.
- Description: Stores information about employees such as their name and salary details.

2. Payroll Table

- Fields: views employee name, salary.
- Description: Holds information about various employees within the organization.

5.2 Functionalities

Employee Management

- Add Employee: Allow users to add new employee records by entering their details through a form.
- Delete Employee: Provide the ability to delete employee records from the system.

5.3 User Interface

1. Frontend Design

Design a simple user interface with forms for adding and deleting employee records. Include input fields for entering employee details and buttons for submitting the form and deleting records. Display a list of existing employees with an option to select and delete them.

2. Backend Logic

Implement server-side logic to handle requests from the frontend. Create endpoints for adding and deleting employee records. Validate user input to ensure data integrity and prevent errors.

5.4 Data Integrity and Security

Implement basic data validation to ensure that required fields are not empty and data types are correct. Use parameterized queries or ORM (Object-Relational Mapping) libraries to prevent SQL injection attacks. Implement basic authentication to restrict access to authorized users only./

5.5 Testing

1. Unit Testing

Write unit tests to validate database operations such as inserting and deleting records. Test frontend components to ensure that forms are rendered correctly and user input is handled properly.

2. Integration Testing

Perform integration tests to verify the interaction between the frontend and backend components. Test the entire workflow of adding and deleting employee records to ensure seamless functionality.

5.6 Deployment

Deploy the application to a suitable hosting environment. Ensure that the deployment environment is configured securely and can handle expected traffic.

5.7 Maintenance and Support

Provide ongoing maintenance and support to address any issues or bugs that arise. Regularly update dependencies and apply security patches to keep the system secure.

By following this system design, you can create a basic payroll management system with functionalities to add and delete employee records. This simplified version can serve as a foundation for further development and expansion of the system with additional features.

CHAPTER 6

IMPLEMENTATION

Implementation is the realization of an application, or execution of a plan, idea, model, design, specification, standard, algorithm, or policy

In computer science, an implementation is a realization of a technical specification or algorithm as a program, software component, or other computer system through computer programming and deployment. Many implementations may exist for a given specification or standard. For example, web browsers contain implementations of World Wide Web Consortium-recommended specifications, and software development tools contain implementations of programming languages.

A special case occurs in object-oriented programming, when a concrete class implements an interface; in this case the concrete class is an implementation of the interface and it includes methods which are implementations of those methods specified by the interface.

6.1 MODULES

1. **ADMIN:**

This module is controlled by the admin, where he can add new employee records, edit these records, update the salary of the employees, and provide any deductions or allowance according to the need. These things are managed through an Admin menu window where he gets access to these operations and also there are some of the admin settings provided.

- Admin Menu
- Add Emp
- Edit Emp
- Update Emp Salary
- Delete employement details

2. **EMPLOYEE:**

This module is used by the employee to edit or view his/her individual information. These include viewing the profile of the employee, settings regarding the employee details . From the same window you came logout and move to the login screen for new login-id to log in and view their details. The employees cannot edit their information.

- Emp Menu
- View Details
- Emp Settings
- Emp details

6.2 FEASIBILITY STUDY

After identifying the scope of the project, the feasibility study is needed to be carried out. It is basically keeping the following points in mind.

Building the software for meeting the scope

This software has met the scope. As there is no data involved in the system, processing on the file, and the behavior of this project is already identified and bundled in quantitative manner. The processing of this software is very simple as it has been designed in PHP and it has been well divided into several functions according to the need.

Technically feasible

This software is very much technically feasible. This software is very much concerned with specifying equipment and the software will successfully satisfy almost all the admin’s requirements. The technical need for this system may vary considerably but might include:

- The facility to produce output in a given time.
- Response time under certain conditions.
- Ability to process data at a particular speed.

Therefore, the basic input/output of data is identified. So, the project can easily be built up and it will also be technically feasible.

State of Art

The project is very much within the state of art since the project is a WINDOWS based; it uses very modern and common technique. Beside it is very much modern and user friendly. It also works as middleware i.e. only in between the user and the file. So, it is completely a state of art project.

Financially Feasible

The project is very much financially feasible. The implementation and development cost of this software under the reach of any college. Moreover, it requires some training for the use. So, training cost can be neglected and the resources of this software are very much available. It also reduces the labor and extra cost to be paid for labor. So indeed, it is financially feasible.

Resources

As motioned earlier that the resources are easily available and the cost of training is almost negligible. Sometimes situations may arise when it may not be so much easy. For a person completely unaware of using a computer system could result in a training cost or for a very small organization the purchase of a computer, instalment of the system and other charges may lead to a difficult matter.

6.3 ENTITY RELATIONSHIP DIAGRAM

An Entity–relationship model (ER model) describes the structure of a database with the help of a diagram, which is known as Entity Relationship Diagram (ER Diagram). An ER model is a design or blueprint of a database that can later be implemented as a database. The main components of E-R model are: entity set and relationship set.

An ER diagram shows the relationship among entity sets. An entity set is a group of similar entities and these entities can have attributes. In terms of DBMS, an entity is a table or attribute of a table in database, so by showing relationship among tables and their attributes, ER diagram shows the complete logical structure of a database. Let’s have a look at a simple ER diagram to understand this concept.

CHAPTER 7

SOURCE CODE

7.1 ADD EMPLOYEE

```
<?php
    include_once 'db_connection.php';
    if ($_SERVER['REQUEST_METHOD'] == 'POST') {
        $name = $_POST['name'];
        $salary = $_POST['salary'];
        $sql = "INSERT INTO employees (name, salary) VALUES ('$name', '$salary')";
        mysqli_query($conn, $sql);
        header("Location: view_employees.php");
        exit();
    }
?>
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Add Employee</title>
    <link rel="stylesheet" href="styles.css">
```

```
</head>
<body>
    <nav>
        <ul>
            <li><a href="index.php">Home</a></li>
            <li><a href="add_employee.php">Add Employee</a></li>
            <li><a href="view_employees.php">View Employees</a></li>
        </ul>
    </nav>
    <div class="container">
        <h1>Add Employee</h1>
        <form action="<?php echo $_SERVER['PHP_SELF']; ?>" method="POST">
            <label for="name">Name:</label><br>
            <input type="text" id="name" name="name" required><br>
            <label for="salary">Salary:</label><br>
            <input type="number" id="salary" name="salary" step="0.01" required><br><br>
            <input type="submit" value="Submit">
        </form>
    </div>
</body>
</html>
```

7.2 VIEW EMPLOYEE

```
<?php
    include_once 'db_connection.php';
    // Check if delete button is clicked
    if(isset($_POST['delete_id'])) {
        $delete_id = $_POST['delete_id'];
        // Delete employee from database
        $sql_delete = "DELETE FROM employees WHERE id = $delete_id";
        mysqli_query($conn, $sql_delete);
    }
    // Retrieve employees data
    $sql = "SELECT * FROM employees";
    $result = mysqli_query($conn, $sql);
?>
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>View Employees</title>
    <link rel="stylesheet" href="styles.css">
```

```
</head>
<body>
    <nav>
        <ul>
            <li><a href="index.php">Home</a></li>
            <li><a href="add_employee.php">Add Employee</a></li>
            <li><a href="view_employees.php">View Employees</a></li>
        </ul>
    </nav>
    <div class="container">
        <h1>View Employees</h1>
        <table>
            <tr>
                <th>ID</th>
                <th>Name</th>
                <th>Salary</th>
                <th>Action</th> <!-- Add this column for delete button -->
            </tr>
            <?php while ($row = mysqli_fetch_assoc($result)) { ?>
                <tr>
                    <td><?php echo $row['id']; ?></td>
```

```
            <td><?php echo $row['name']; ?></td>
            <td><?php echo $row['salary']; ?></td>
            <td>
              <form action="<?php echo $_SERVER['PHP_SELF']; ?>" method="POST">
                <input type="hidden" name="delete_id" value="<?php echo $row['id']; ?>">
                <button type="submit">Delete</button>
              </form>
            </td>
          </tr>
        <?php } ?>
      </table>
    </div>
</body>
</html>
```

7.3 INDEXED CODE

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
```

```
    <title>Employee Payroll System</title>
    <link rel="stylesheet" href="styles.css">
</head>
<body>
    <nav>
        <ul>
            <li><a href="index.php">Home</a></li>
            <li><a href="add_employee.php">Add Employee</a></li>
            <li><a href="view_employees.php">View Employees</a></li>
        </ul>
    </nav>
    <div class="container">
        <h1>Welcome to Employee Payroll System</h1>
        <p>This is a simple employee payroll management system.</p>
    </div>
</body>
</html>
```

7.4 DATA BASE CONNECTION

```
<?php
    $conn = mysqli_connect('localhost', 'root', '', 'employee_payroll');
```

```
    if (!$conn) {
        die("Connection failed: " . mysqli_connect_error());
    }
?>
```

7.5 CREATE DATA BASE USING MYSQL

```
CREATE DATABASE IF NOT EXISTS employee_payroll;
USE employee_payroll;
CREATE TABLE IF NOT EXISTS employees (
    id INT AUTO_INCREMENT PRIMARY KEY,
    name VARCHAR(100) NOT NULL,
    salary DECIMAL(10, 2) NOT NULL
    );
```

CHAPTER 8

SCREENSHOTS

8.1 HOME PAGE

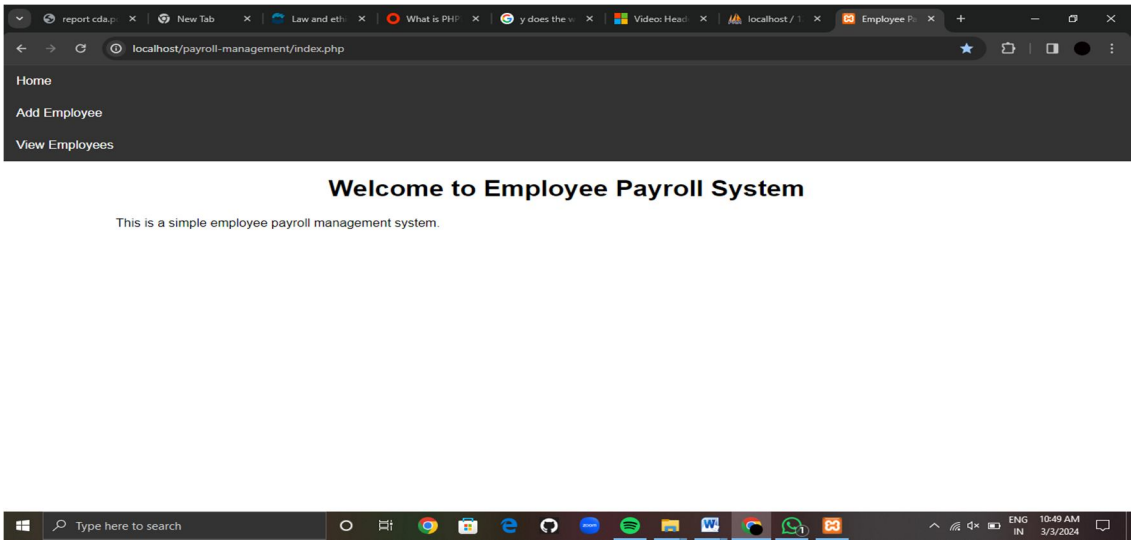

Fig8.1: Homepage

8.2 ADD EMPLOYEE

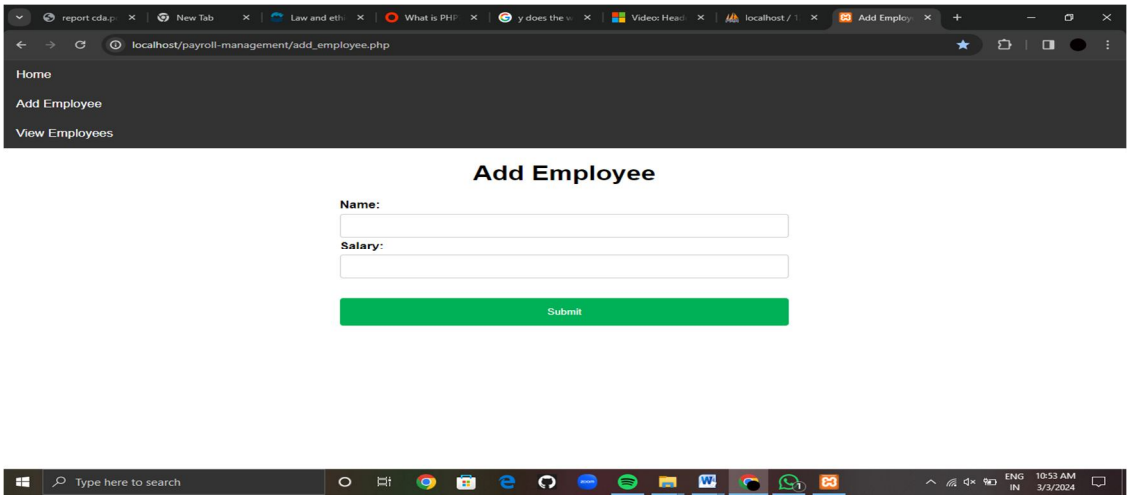

Fig8.2: Add Employee

8.3 VIEW EMPLOYEE

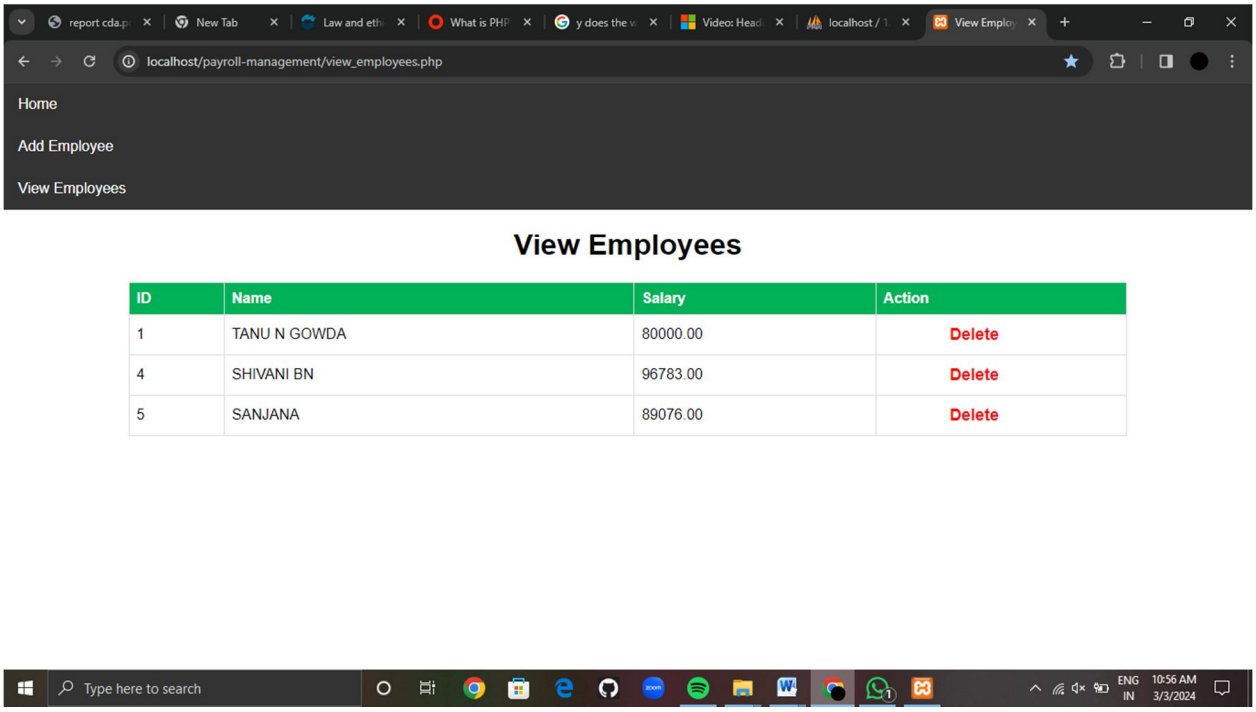

Fig 8.3 : View Employee

CHAPTER 9

ADVANTAGES AND DISADVANTAGES

9.1 ADVANTAGES

- **Time-saving process**

It takes a very long time to process payroll. If you invest in a payroll service provider, the staff members will be able to concentrate on other crucial areas of the organization, such as creating financial plans that boost the firm's competitive edge by freeing up the payroll department from these time-consuming payroll duties.

- **Reduces errors**

You reduce the likelihood of errors, missed deadlines, omissions, or late payroll tax filings by employing experts whose primary job and area of expertise is payroll.

- **Reduces expenditure**

Payroll outsourcing helps small and big firms in cutting costs and estimating the direct and indirect costs associated with managing payroll. In addition, the flexibility you need to change your fixed overhead costs into a variable cost structure is provided by managed payroll services. For instance, if the workload does not warrant the fixed cost, you won't be required to keep paying a payroll clerk a hefty compensation and other company benefits.

- **Maintaining employee records**

The majority of organizations also store additional information about their employees, such as records of their annual leave. You can avoid the requirement for an additional software package by hiring a company which provides payroll services and can easily keep track of these other types of data.

- **Increases the speed of responsiveness**

Payroll service providers can spend time addressing any concerns or questions your employees may have, which reduces response time and speeds up the overall process.

- **Ensures compliance**

Payroll outsourcing companies keep up with labor laws and regulations, which simplifies the process for employers to maintain compliance. You can maintain compliance with rules and ensure that your data is protected from the risk of payroll theft and fraud by working with a payroll service provider who will provide services in an effective, safe, precise, and confidential manner.

9.2 DISADVANTAGES

- **Delays in the payment of salaries**

The failure to provide employees with their salaries on time is a significant issue when outsourcing payroll. This happens when a company employs people from all over the world. This issue occurs when employees do not get the necessary information to process their paychecks. Additionally, you should plan ahead and prepare for disruptions like holidays, unpredictable weather, and technical difficulties.

- **Increase in expenses**

Payroll services are an additional expense to your budget. Paying for the service can not fit into the budget if your company is small, has few employees, and has a limited budget. A payroll service will be more expensive for a small firm than for a bigger company. You need to compare the expense of the service against the time it takes to complete the payroll in-house and determine whether that time would be worth accomplishing other business activities.

- **Data accessibility**

If you employ a payroll service provider, the computer system of another company will have the information about your employee payroll. This can be a problem if you want to make a modification or if an employee is experiencing trouble receiving their paycheck. Depending on how much access the service offers, you can choose as to how much of the payment has to be made. You will need to compare the payroll service providers and their costs while keeping this requirement in mind.

- **Loss of Control**

You lose control over procedures and information accessibility to an extent when you outsource your payroll operations to a third party.

- **Risks to confidential information**

It is crucial to develop confidentiality and data protection rules and regulations in order to guarantee the security of all corporate and employee information. Otherwise, confidentiality threats may create issues for your business.

- **Coordination challenges**

Although it can take time, businesses still need to set out time to organise specific tasks and activities with their payroll providers. Moreover, they need to give continuous guidance to ensure that the third party obtains the targeted objectives.

CHAPTER 10

CONCLUSION

Payroll management system is a critical component of any organization, facilitating the efficient and accurate calculation, processing, and disbursement of employee salaries and benefits. In conclusion, such a system offers several significant advantages:

Efficiency: By automating payroll processes, including salary calculation, tax deductions, and payment generation, a payroll management system streamlines operations, reducing the time and effort required for payroll administration tasks.

Accuracy: Automation minimizes the risk of human error inherent in manual payroll processing, ensuring that employees are paid accurately and on time, thereby enhancing trust and morale within the workforce.

Compliance: Payroll systems help ensure compliance with relevant labor laws, tax regulations, and reporting requirements, reducing the likelihood of costly penalties or legal issues arising from non-compliance.

Cost-Effectiveness: While initial investment may be required to implement a payroll management system, the long-term cost savings from increased efficiency, reduced errors, and minimized compliance risks often outweigh these upfront expenses.

Data Management: Payroll systems centralize employee data, facilitating easy access and retrieval of information for reporting, analysis, and decision-making purposes, thereby supporting strategic workforce management initiatives.

Security: Payroll systems incorporate security features to protect sensitive employee information, such as salary details and personal data, from unauthorized access or breaches, ensuring data privacy and confidentiality.

Scalability: As organizations grow or change, payroll systems can scale to accommodate increasing numbers of employees, additional pay structures, or evolving regulatory requirements, providing flexibility and adaptability to meet changing business needs.

In essence, a well-designed and effectively implemented payroll management system offers numerous benefits, ranging from operational efficiency and accuracy to compliance and data security, ultimately contributing to the overall success and sustainability of an organization.

CHAPTER 11

REFERENCES

Reference Books

- "Payroll Management: 2022 Edition" by Steven M. Bragg
- "Payroll Best Practices" by Steven M. Bragg
- "The Payroll Source" by American Payroll Association

Websites

- "The Benefits of Using a Payroll System" by Investopedia
- "How to Choose the Right Payroll Management System" by ADP
- "Top Payroll Software Features to Look for in 2022" by Paychex