PREMIER UNIVERSITY
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Final Project Report On
AUTOMATED PAYROLL SYSTEM

Course Title: Software Development
Course Code: CSE 364

Submitted To:
Dhrubajyoti Das
Assistant Professor
Department of Computer Science and Engineering

Submitted By:
Aurpita Barua (2104010202160)
Nusrat Warida Saima (2104010202163)
Bobi Barua (2104010202165)
6th Semester A Section

September, 2024

TABLE OF CONTENTS

List of Figures

1. Introduction

In today's rapidly evolving digital landscape, organizations are continuously striving to automate and streamline their operations for improved efficiency. Payroll processing, a critical component of business administration, is often a complex and time-consuming task, especially when handled manually. Nowadays all establishments are becoming modernized, modern technologies now make business transactions fast, easy, and accurate in order to avoid waste of time and for the sake of safety and security. Studies have shown that automated payroll systems significantly reduce human errors, enhance data security, and lead to better resource management in business operations. This project seeks to address the challenges associated with manual payroll management through the development of an Automated Payroll System using Laravel 11.

Payroll systems play an essential role in managing employee compensation, calculating deductions, and ensuring compliance with tax regulations. In a manual system, these processes are prone to errors, can be inefficient, and might lead to delayed payments, legal complications, or employee dissatisfaction. Therefore, the automation of these tasks is crucial for enhancing accuracy, saving time, and reducing operational costs. This project aims to develop a comprehensive system that enables administrators to manage employees, track attendance, and generate accurate payroll records effortlessly.

The scope of the project involves creating a web-based solution where administrators can efficiently manage employee information, track and monitor attendance, and automatically generate payroll, while employees can submit their attendance and view payroll slips in real time. This system is intended to increase productivity, reduce administrative overhead, and improve overall employee satisfaction. The broader goal is to demonstrate how automation in payroll management can directly enhance the performance of an organization. [1]

2.   Problem Statement

In many organizations, managing employee attendance and payroll can be a time-consuming and error-prone task when done manually. The traditional methods of tracking attendance, calculating wages, and managing payroll records often result in inefficiencies, errors in data entry, and delayed payments. Furthermore, with manual systems, there is a lack of transparency for employees to view their payroll details and attendance records in real-time, leading to potential disputes and dissatisfaction.

The problem becomes even more pronounced in larger organizations where the volume of employees and data management increases the complexity. Without a streamlined system, administrators face challenges in maintaining accurate records, ensuring compliance with labor laws, and generating timely payroll slips.

This project addresses the critical need for an automated system that simplifies and enhances payroll management by allowing for more accurate, secure, and transparent handling of payroll operations. Through automation, the proposed solution eliminates the errors and delays associated with manual systems and provides a user-friendly platform for both administrators and employees.

3. Objectives

Employee Management

- Create a module where administrators can add, update, and manage employee details including personal information, salary, and department.

Attendance Tracking

- Develop a feature to track employee attendance. Attendance data will be recorded and stored in the database for automatic payroll calculation.

Automated Payroll Calculation

- Automate the calculation of gross pay, deductions, and net pay based on attendance and pre-defined salary structures.

Payslip Generation

- Generate detailed payslips that employees can view and download from a self-service portal.

Secure Data Management

- Ensure that sensitive payroll and employee data is securely stored with proper encryption and access control measures.

User-Friendly Admin and Employee Portals

- Create intuitive and easy-to-navigate portals for both administrators (to manage payroll and attendance) and employees (to view attendance and payslips).

By achieving these objectives, the project is developed to deliver an efficient and automated payroll system that minimizes errors, saves time, and provides a secure and transparent platform for both administrators and employees.

4. Related Works

Mahajan et al. [1] provides a detailed examination of cloud-based payroll systems designed to automate payroll calculations and record-keeping for large organizations. The proposed system operates on a cloud platform, enabling multiple users, including employees, HR personnel, and administrators, to access and manage payroll data with distinct roles and privileges. It features real-time updates to salary structures, automatic net pay calculations, and the generation of pay slips. The system is structured using a 3-tier architecture with a presentation layer (HTML, JavaScript, AJAX, JSON, CSS), a business layer (built with .NET 4.5), and a data access layer (handling SQL queries). The paper outlines the Software Development Life Cycle (SDLC), emphasizing system analysis, design, and implementation stages. It contrasts manual payroll processes with computerized systems, highlighting benefits such as improved accuracy, reduced error rates, and enhanced productivity. Strengths of the system include its cloud-based architecture, which supports real-time updates and access flexibility, and its robust 3-tier structure that enhances security and functionality. The use of technologies like ASP.NET, and SQL Server 2012 ensures efficient backend development and data management. The system also benefits from automation, reducing manual effort and improving accuracy and security in payroll management. Limitations include the lack of detailed discussion on how the system addresses specific security threats beyond basic protections, and the absence of network and web-based functionalities which could further enhance flexibility and integration. Additionally, while the system is designed to accommodate various payroll operations, future updates will be needed to adapt to evolving requirements and improve its capabilities. Overall, the review emphasizes the advantages of transitioning to computerized payroll systems and the potential for further advancements in cloud-based payroll management.

William et al. [2] outlines the creation of a web-based system at Le Moyne College to simplify and automate the payroll process for student employees in the IT department. The system, built using FileMaker Pro and a custom web interface with Claris Data Markup Language (CDML), allows students to log in and out using their ID cards, automating attendance tracking and payroll calculation. It replaces the manual paper-based process, reducing transcription errors, and minimizing the risk of fraud through IP address verification of login locations. The system also automatically generates payroll reports, significantly reducing the time required to process payroll from 3.5 hours to about 15 minutes.Strengths of the system include its automation, which cuts down manual work, improves accuracy by eliminating errors from handwritten sign-in sheets, and provides security through the use of ID cards and restricted database access. The custom-built interface also provides greater control and functionality tailored to the college's needs. However, there are limitations,

such as reliance on ID cards, which can cause issues if students lose them or fail to update them, requiring manual intervention. The system also requires regular maintenance, such as updating student employment status or correcting missed logins. While effective for a small group of students, scaling the system for larger populations or more complex payroll structures may pose challenges. The development process using FileMaker Pro and CDML was complex and required significant time and effort, indicating a steep learning curve for future updates. Furthermore, the system is not yet fully integrated with the college's main payroll system, requiring manual submission of reports, with plans for full electronic integration still in progress.

Rachman et al. [3] presents a system designed to streamline the payroll process for part-time employees by automating attendance tracking and salary calculations. This automated approach addresses the inefficiencies and error-prone nature of manual processes. The system features a login-based attendance tracker that logs working hours, automatically calculates payroll, and generates relevant reports, simplifying data management for finance and HR departments. Developed using Visual Studio and MySQL, the system focuses on part-time employees, integrating attendance data directly with payroll calculations to ensure accurate and timely compensation. However, its scope is limited to part-time staff, and it lacks cloud or web-based functionalities, which may affect scalability and accessibility. Basic security measures like user authentication are included, but advanced features such as data encryption or multi-factor authentication are not discussed. Additionally, scalability concerns and a small survey sample size of 17 respondents could affect the system's broader applicability and effectiveness.

Singh et al. [4] presents a web-based solution designed to automate leave and payroll management for organizations. The system aims to replace outdated paper-based processes with a modern web portal that supports multiple user accounts, including admin, employee, HOD, and principal, for efficient management of leave applications, approvals, and payroll computations. By integrating with a biometric attendance system, it enhances accuracy in tracking employee attendance and calculating payroll, including salary computation, tax deductions, and allowances. Developed using technologies such as Bootstrap, MongoDB, Blaze, and Meteor.js, the system offers a user-friendly interface and transparency, allowing employees to view their leave and payroll details. It is also noted for its flexibility, enabling easy adaptation and integration with other systems. However, the paper does not thoroughly address scalability issues for larger organizations, lacks in-depth discussion of advanced security measures, and relies on biometric systems, which may not be feasible for all organizations. Additionally, it does not explore cloud integration, which could enhance accessibility and data backup, and it lacks detailed analysis on system performance and handling large datasets.

Udo et al. [5] offers a practical examination of automating payroll processes in a real-world context, showcasing its relevance to similar organizations. Its strengths include a clear focus on reducing manual errors and inaccuracies in salary computations, leveraging a secure database with user authentication and access control, and following a structured Waterfall methodology that provides clarity on the project phases. The system's design aims to enhance efficiency and reduce the costs associated with manual payroll processes. However, the paper has limitations, such as the use of Visual Basic 6.0, an outdated programming language that limits modern scalability and flexibility. It lacks web or cloud integration, which restricts remote access and multi-user support, and does not delve into advanced security measures like data encryption or multi-factor authentication. Additionally, there is insufficient discussion on the system's scalability to accommodate growing data and employee counts, and the paper does not provide detailed performance evaluation or real-world testing. The absence of network capabilities further limits the system's applicability in distributed or multi-location environments.

Ahmed et al. [6] explores the development of a web-based payroll management system (WPMS) aimed at addressing the inefficiencies of manual payroll processes in the Kurdistan Region of Iraq. The WPMS automates key functions such as salary calculations, allowances, deductions, payslip generation, and detailed report production, significantly reducing errors and increasing processing speed. It boasts a high usability score of 87.8% based on the System Usability Scale (SUS), indicating a user-friendly design. The system is flexible and scalable, allowing dynamic updates like adding new employees and allowances without altering the core code. It also incorporates modern security features, including password encryption with the Bcrypt algorithm, and stores historical data in a data mart for easy retrieval and auditing. Despite its strengths, the system's focus on the specific needs of universities in the Kurdistan Region may limit its applicability elsewhere without further customization. Additionally, its dependency on robust technical infrastructure could be challenging in regions with limited technological resources. While the system flags errors during audits, it requires human intervention for resolution rather than automatic correction. Scalability concerns remain as the paper does not provide a detailed analysis of system performance under high data loads or in larger organizations. Overall, while the WPMS offers significant improvements over manual processes, there is room for enhancing its broader applicability and automation in error handling.

Soegoto et al. [7] details the development of an integrated payroll information system for CV. Bandung, focusing on automating payroll processes and integrating attendance tracking using Radio Frequency Identification (RFID) technology. The system aims to enhance efficiency by linking attendance data directly with payroll calculations, including overtime and leave management, thus reducing manual efforts and improving accuracy. The use of RFID accelerates data input and minimizes errors, while the integration of attendance and payroll into a unified system ensures timely and precise payroll generation. The system employs an object-oriented design and prototype development method, facilitating modularity and future expansion. However, the paper highlights several limitations, including the system's desktop-based nature, which may restrict scalability and accessibility compared to web-based solutions. Implementation challenges include potential resistance from users transitioning from manual methods, necessitating training and adaptation. Additionally, there is limited discussion on security measures such as data encryption, which are crucial for protecting sensitive payroll information. Overall, while the system significantly enhances efficiency and accuracy, it could benefit from improved scalability and security considerations for larger and more diverse implementations.

5. Methodology

The development of the Automated Payroll System followed a structured approach to address the specific challenges and objectives outlined in the problem statement. This section presents the steps taken, from requirement identification to system design, with a focus on feasibility analysis and the technical framework adopted.

5.1. Requirement Identification

The first step was to conduct a comprehensive review of existing systems and relevant literature to identify gaps and limitations in current payroll management processes. Research papers such as "Web-based payroll management system: design, implementation, and evaluation" by Ahmed et al., and "Designing Payroll Information System: Case Study on CV. Bandung ID card" by Y. Soegoto were reviewed to better understand current practices and innovations in payroll systems. These papers highlighted both the strengths and limitations of various systems, especially regarding automation, efficiency, scalability, and security.

5.1.1. Study of Existing Systems

The need for real-time automation and error-free calculations. The lack of scalability in desktop-based systems, as outlined in Soegoto's study. Limited security features in older payroll systems, like those discussed by Ikechukwu in his case study of the Nigerian Communications Commission. While these systems addressed many challenges in manual payroll processing, they often lacked advanced security protocols and web-based access. This project aimed to incorporate the lessons learned from these studies, emphasizing cloud-based deployment, improved security, and system scalability.

5.2. Requirement Analysis

After studying existing systems, the specific requirements for the Automated Payroll System were defined based on the limitations found in manual processes and gaps in current systems.

5.2.1. Technical Requirements

The system must support employee management, attendance tracking, payroll generation, and secure data storage. It must use Laravel 11, providing a robust backend framework for managing data and processes. A relational database must be used to store and manage employee records, attendance logs, and payroll data securely.

5.2.2. Operational Requirements

Admins should be able to perform key functions, such as adding employees, tracking attendance, and generating payroll. Employees should be able to log attendance and access their payroll slips via a web interface. The system must be easy to use with minimal training required.

5.2.3. User Requirements

Admin users should have full control over employee data, attendance logs, and payroll management. Employee users should be able to give attendance and view payslips securely through the system.

5.3. Feasibility Study

5.3.1. Technical Feasibility

The technical feasibility of the project was assessed by considering the tools, frameworks, and expertise required to build the system. The choice of Laravel 11 as the backend framework ensures that the system can handle large amounts of data while maintaining security and efficiency. A relational database, such as MySQL, was selected to manage employee information, attendance logs, and payroll data. The use of encryption for sensitive data ensures compliance with security protocols. The development team had the necessary expertise in Laravel, PHP, and MySQL, ensuring the system's technical soundness.

5.3.2. Operational Feasibility

The operational feasibility focused on how easily the system could be integrated into the organization's daily workflows.

The user interface was designed to be intuitive and straightforward for both admin users and employees. This minimizes the need for extensive training. The system was designed to integrate seamlessly into the organization's existing infrastructure, requiring minimal changes to current processes.

5.3.3. Economic Feasibility

A cost-benefit analysis was conducted to determine whether the project was financially viable. The use of open-source technologies, such as Laravel and MySQL, significantly reduced development costs. The system's automated features, such as attendance tracking and payroll generation, reduce the time and labor required for manual payroll processes, offering long-term savings. The automation of payroll processes is expected to reduce manual errors and the time spent on payroll, leading to financial savings over time.

5.3.4. Schedule
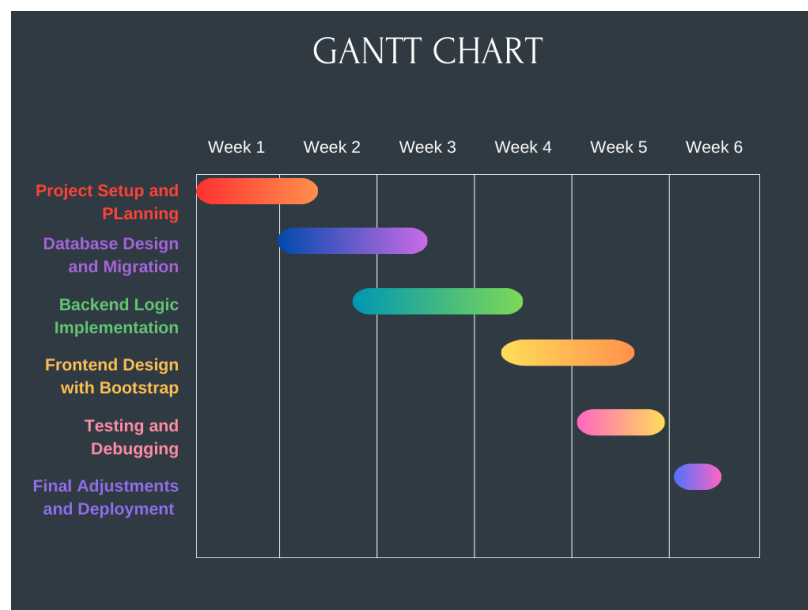
Figure 5.1: Gantt Chart demonstrating schedule feasibility

5.4.   High-Level Design of System

5.4.1.   Methodology of the proposed system

The Automated Payroll System is designed to streamline the process of employee management, attendance tracking, and payroll generation through a well-structured workflow. The system begins with a secure login interface where users, including administrators and employees, are required to authenticate their credentials. If the credentials are valid, the user gains access to the system's dashboard. Administrators have full access to manage employees, track attendance, and process payroll, whereas employees can view their attendance and payroll details.

Once logged in, the administrator can add, edit, or remove employee records. This ensures that the system maintains an up-to-date database of all employees, capturing essential details such as personal information, salary structure, and roles. After managing employees, the next step involves tracking attendance. The system records each employee's attendance, including working hours, leaves, and overtime. This real-time attendance data is directly linked to the payroll module.

The payroll management component then automatically calculates salaries based on the recorded attendance. It computes gross pay, applies deductions (such as taxes, benefits, and allowances), and determines the net salary. This automation ensures precision and eliminates manual errors that could occur in traditional payroll systems. Following the payroll calculation, the system generates comprehensive payroll reports for the administrator to review, providing insights into the overall payroll processing. Employees can also access their individual payslips, ensuring transparency and allowing them to review their earnings and deductions. This systematic approach not only reduces administrative overhead but also ensures timely, accurate payroll management.

5.4.2. Flow Charts/Working Mechanism of Proposed System

Include flowcharts or diagrams that illustrate how the system will function.
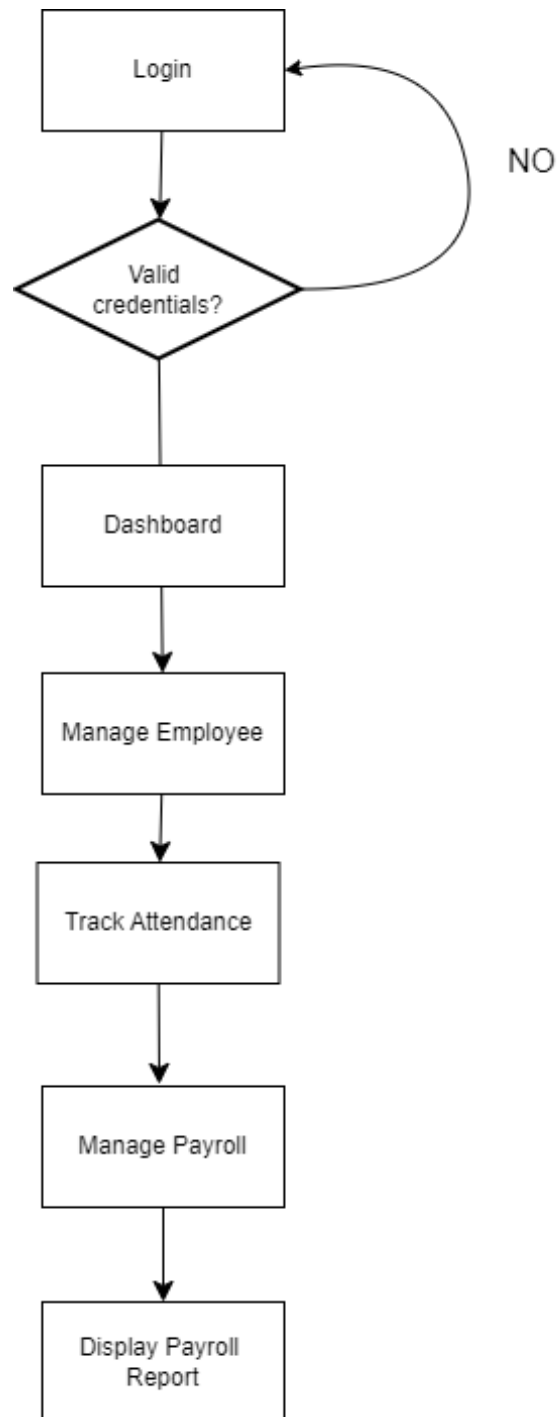

Figure 5.2: Sample flowchart

5.4.3. Entity-Relationship Diagram

This ER Diagram (Entity-Relationship Diagram) illustrates the relationships between key entities in an employee management system:

Admin: Manages the system and controls employee records.

Attributes: ID, name, email, password.

Employee: The central entity representing employees in the system.

Attributes: ID, name, email, designation, phone, salary.

Attendance: Tracks employee attendance records.

Attributes: ID, date, present status.

Payroll: Handles employee payroll details.

Attributes: ID, basic pay, overtime pay, tax deductions, net pay.
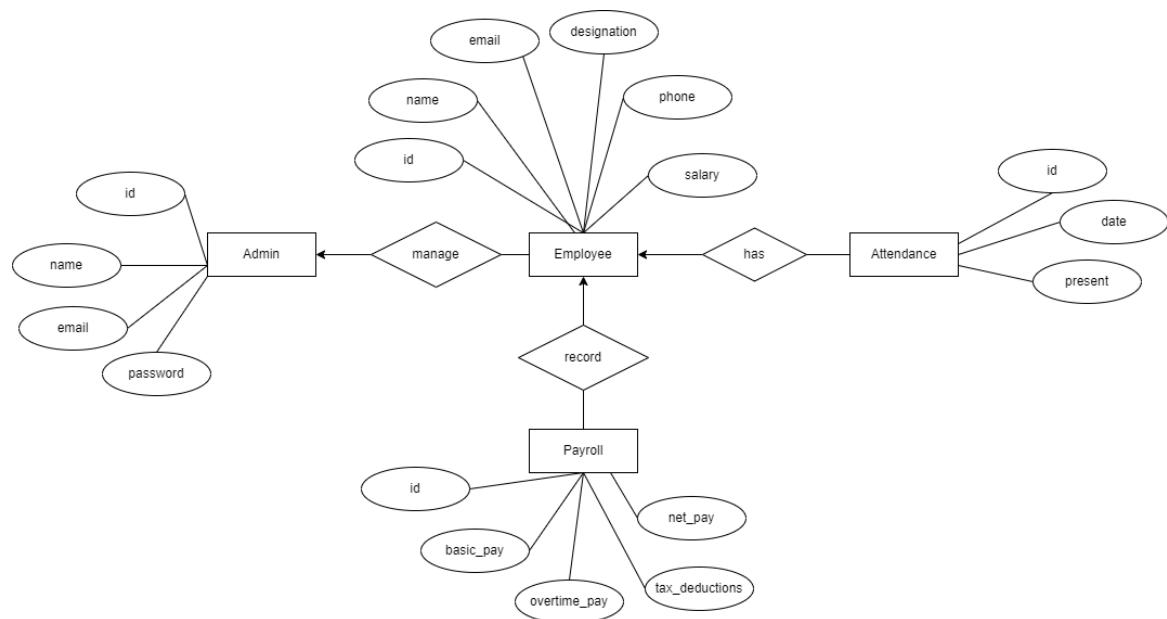
Figure 5.3: ER Diagram

5.4.4. Use Case Diagram

This Use Case Diagram outlines the interactions between two key roles: Admin and Employee, with the system.

Admin Manage employee records, payroll, and attendance. Handle user management and registration tasks.

Employee View their pay slips and perform basic actions like logging in and out.
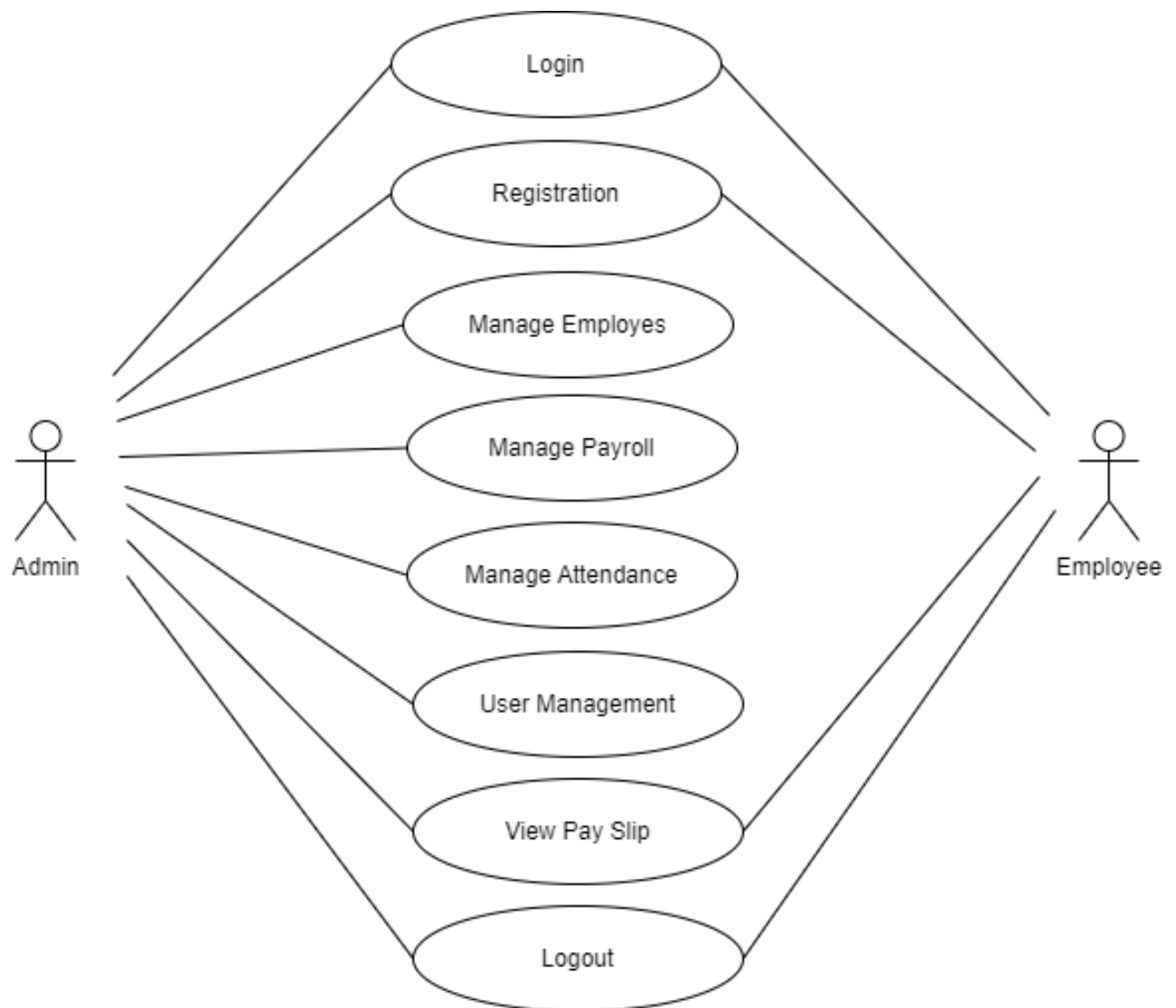
Figure 5.4: Use Case Diagram

5.4.5. Class Diagram

This class diagram represents a the payroll and attendance management system, consisting of four main entities: Users, Employee, Attendance, and Payroll.

Admin manage employees and have attributes like name, email, and password, with methods for authentication and employee management.

Employee stores personal and financial details, including salary, designation, and overtime rates, with methods to calculate salary, update details, and mark attendance.

Attendance tracks employee presence with attributes like date and status (present/absent), and methods to retrieve attendance records.

Payroll calculates employee payments, including basic pay, overtime, and deductions, with methods to generate payslips and calculate net pay.
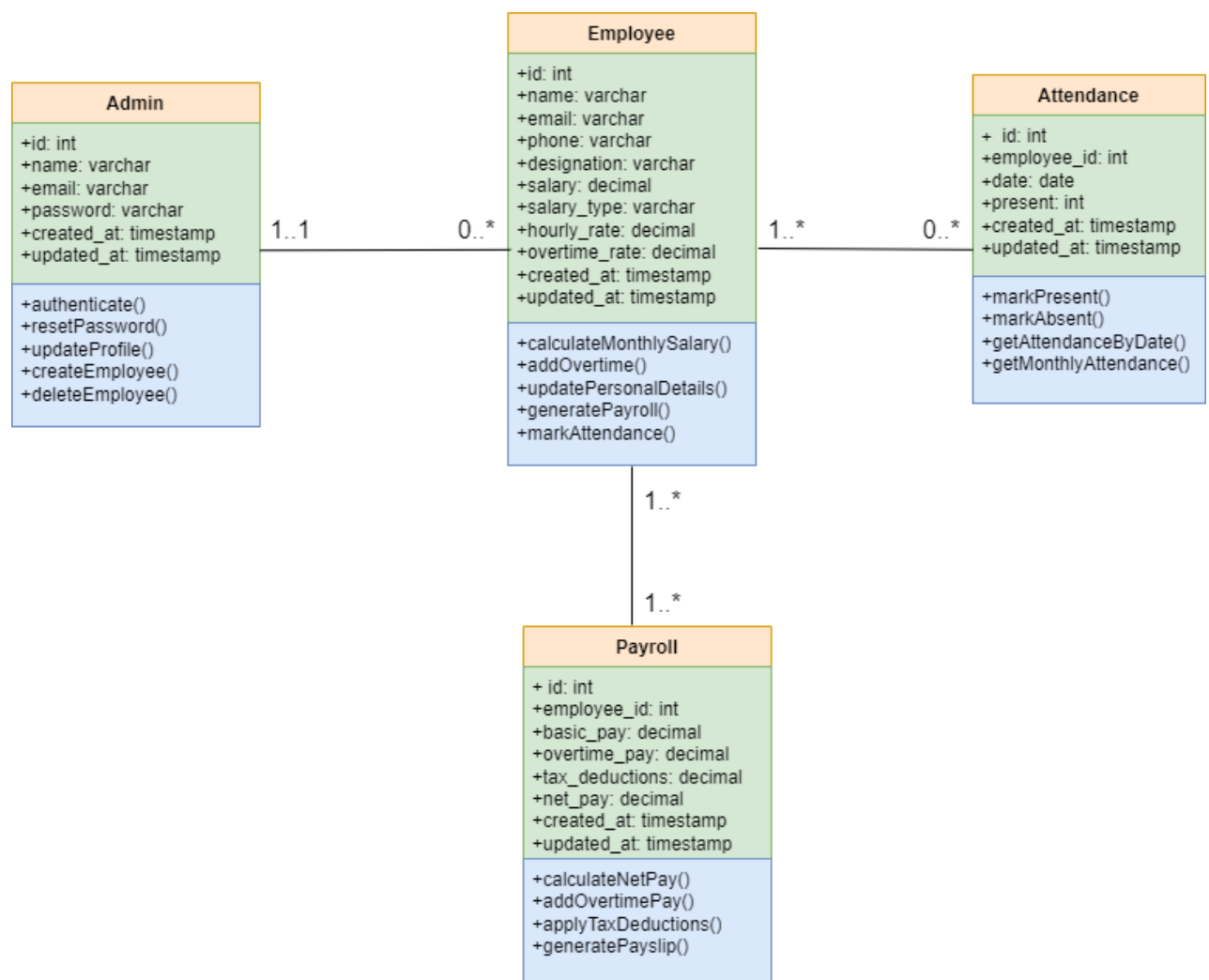
Figure 5.5: Class Diagram

6. Expected Output

The expected output of this project is a comprehensive Payroll and Employee Management System built with Laravel, designed to streamline the management of employee payroll, attendance, and details. The system allows administrators to view, add, edit, and delete payroll records for employees, which includes basic pay, overtime pay, tax deductions, and automatically calculated net pay. Additionally, it provides an interface for managing employee information such as name, email, phone number, designation, salary, and salary type (e.g., hourly or monthly). Through the attendance management feature, admins can track employee attendance by marking them present or absent for specific dates. The project ensures efficient record-keeping through a responsive and user-friendly interface, enabling administrators to manage all critical employee-related information in one place, while ensuring secure access to the system through authentication.
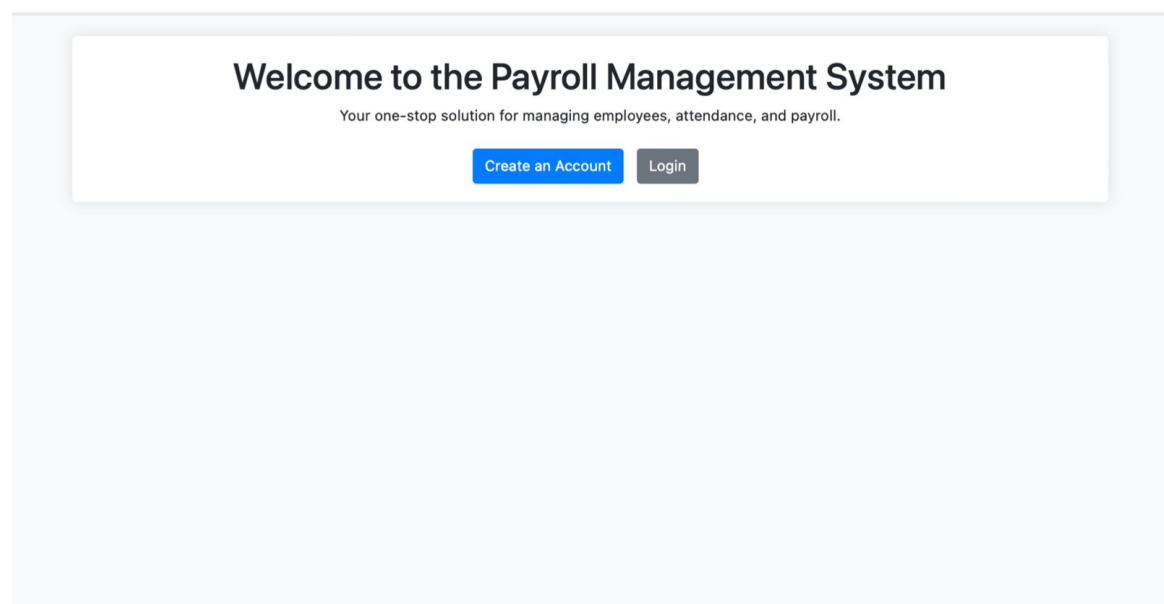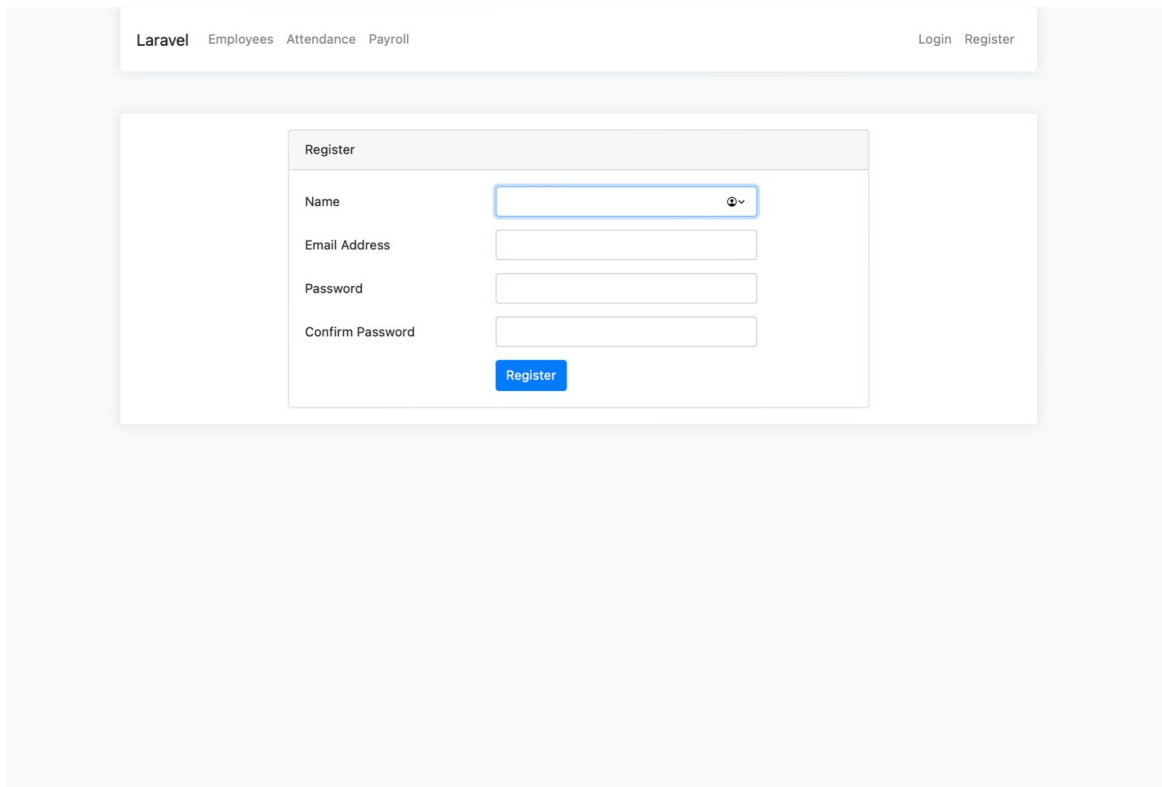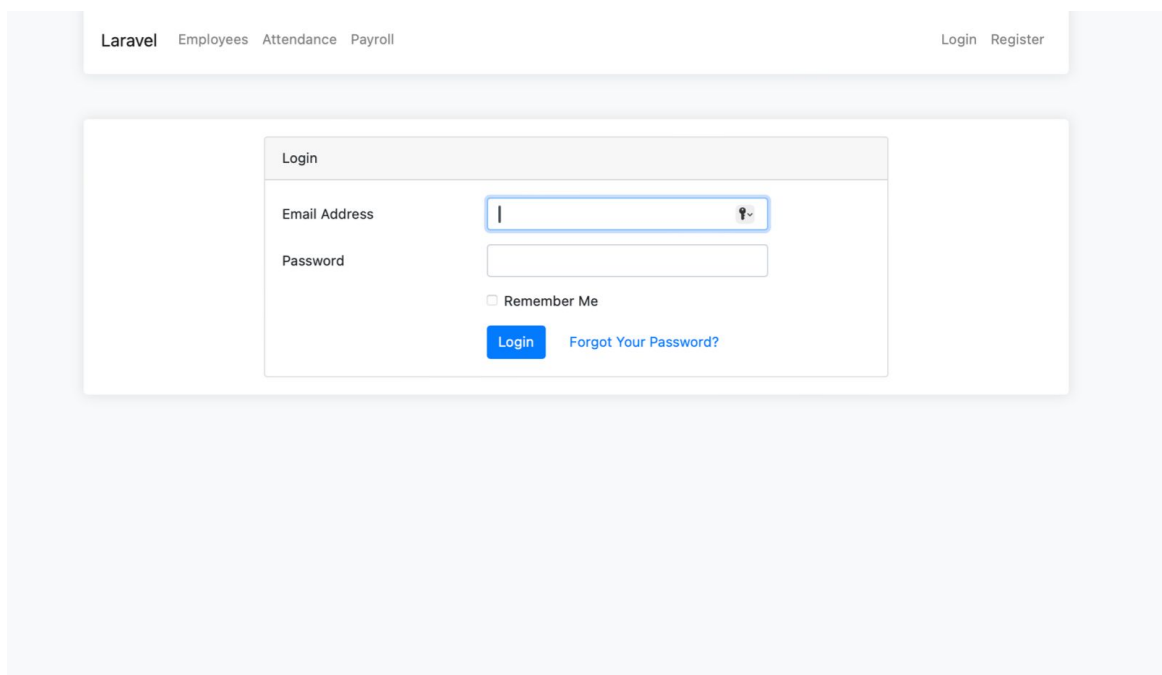
Figure 6.1: Interface of Payroll Management

Figure 6.2: Registration page

Figure 6.3: Login Page
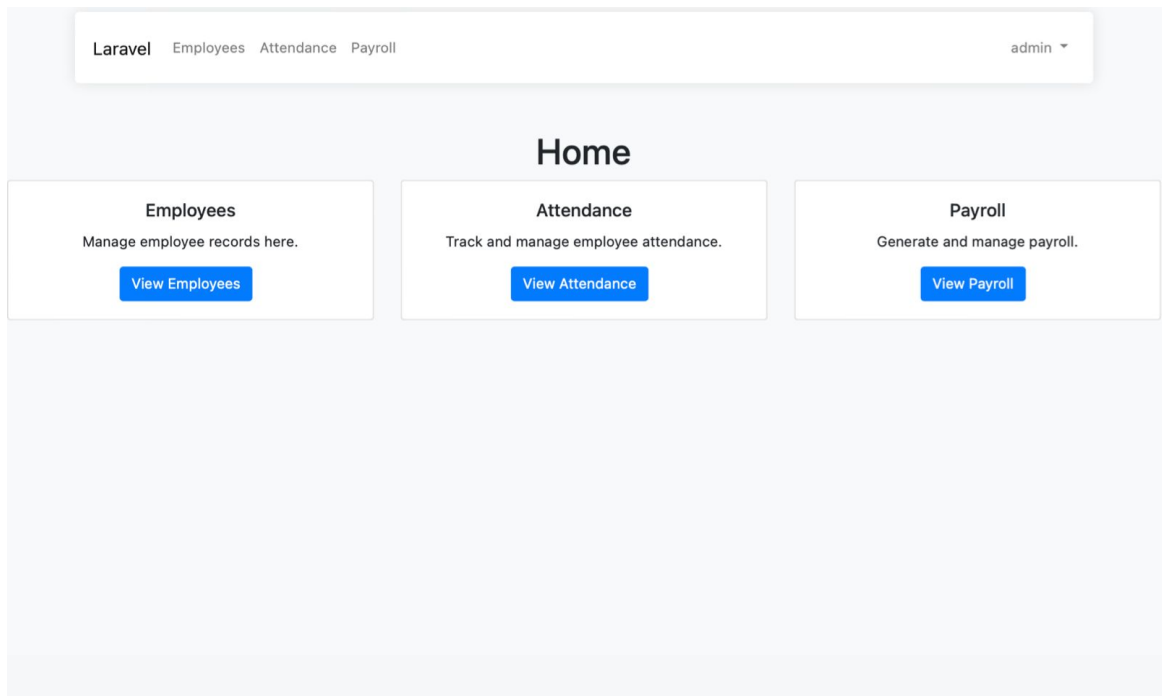
Figure 6.4: Home Page
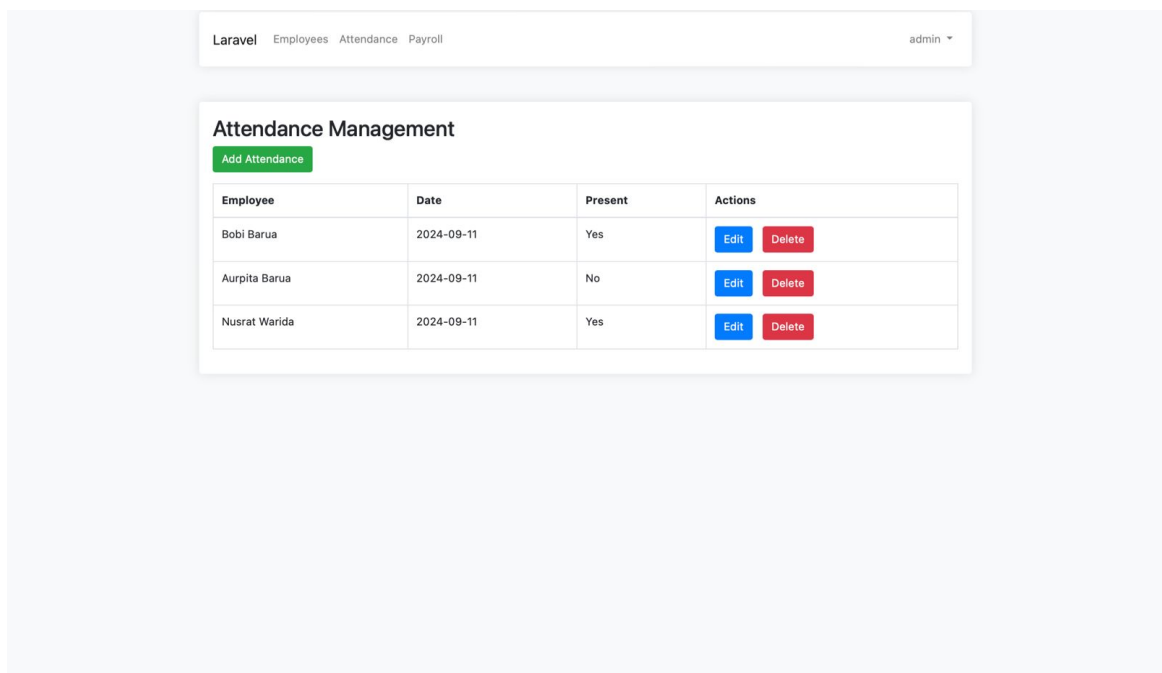
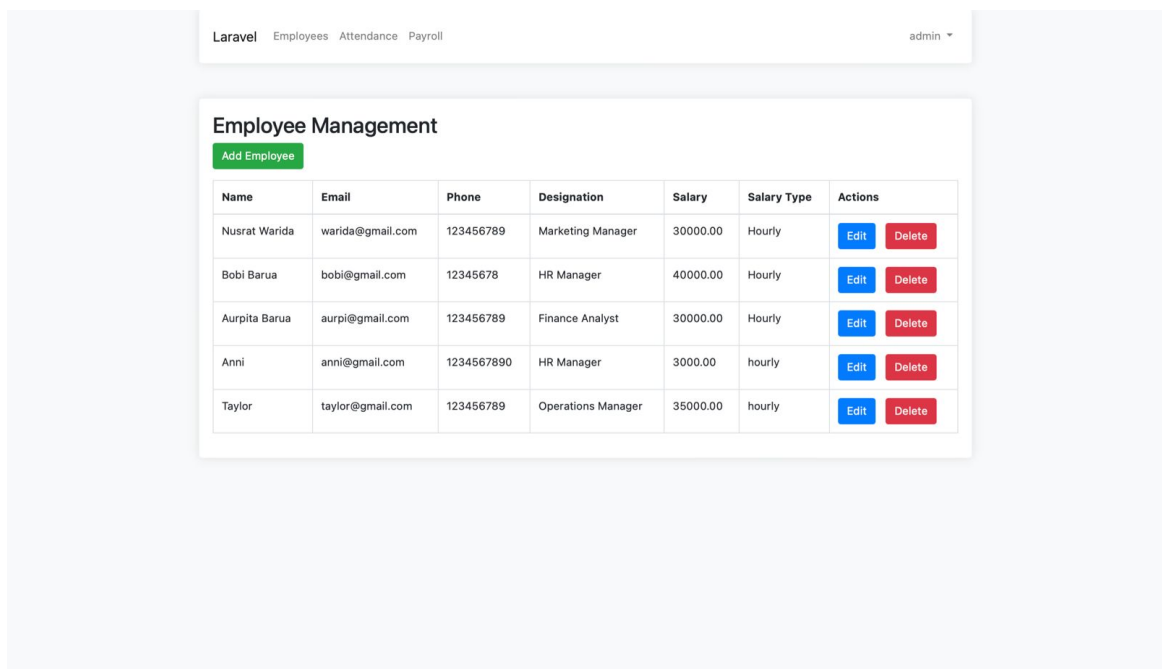Figure 6.5: Attendance Management
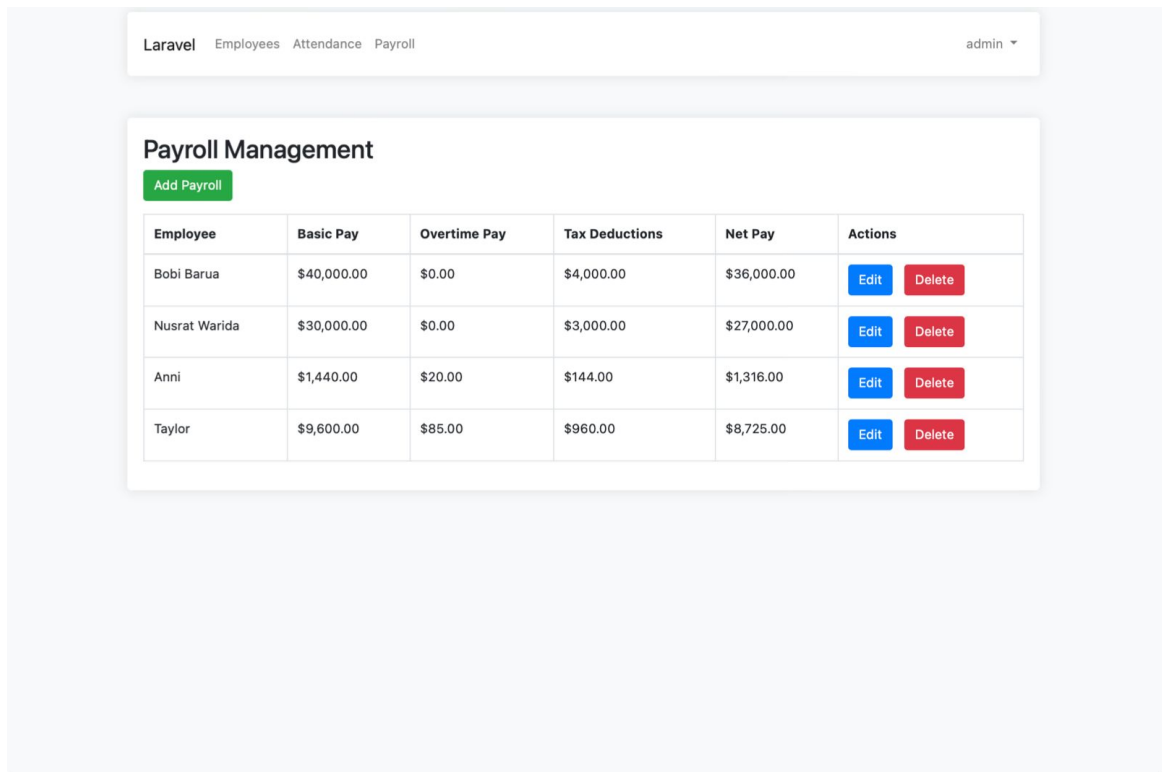
Figure 6.6: Payroll Management

Figure 6.7: Employee Management

Figure 6.8: Add Employee

Figure 6.9: Add Payroll

Figure 6.10: Add Attendance

Figure 6.11: Payroll Generation Constraint

7. Conclusion

The Automated Payroll System developed using Laravel 11 successfully addresses the inefficiencies and challenges of manual payroll processes. By automating employee management, attendance tracking, and payroll generation, the system significantly reduces human errors, enhances accuracy, and improves processing speed. The system also provides easy access for employees to view their payroll slips and manage their attendance records, thereby promoting transparency and trust within the organization.

In terms of practical application, the system demonstrates its potential to streamline administrative operations, reduce labor costs, and ensure compliance with tax regulations and labor laws. With features like real-time payroll calculations, automated deduction handling, and secure data management, it offers a comprehensive solution for organizations seeking to modernize their payroll systems.

The project has proven to be both technically and operationally feasible, ensuring that the system is scalable, user-friendly, and secure. Future enhancements could include extending the system to cloud-based architecture for broader accessibility, integrating advanced analytics for payroll forecasting, and enhancing security measures to safeguard sensitive payroll data. Ultimately, this project has laid a solid foundation for more efficient, accurate, and secure payroll management processes.

References

[1] K. Mahajan, S. Shukla, and N. Soni, "A review of computerized payroll system," *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*, vol. 5, no. 2, pp. 142–147, 2019.

[2] W. S. Thieke, "Payroll made easy: Developing a web-based system for student employee payroll," *Journal of Information Technology Education*, vol. 9, pp. 245–253, 2010.

[3] R. Wintarto, "Automated part-time payroll system based on attendance," *Journal of Software Engineering and Applications*, vol. 13, pp. 131–140, 2020.

[4] M. Singh, P. Singh, R. Singh, S. Singh, and S. Gupta, "Leave and payroll management system," *International Journal of Computer Science and Information Technology*, vol. 9, pp. 44–52, 2018.

[5] U. Ikechukwu, "Design and implementation of an automated payroll system: A case study of the nigerian communications commission," *Journal of Applied Computer Science Mathematics*, vol. 12, no. 4, pp. 57–65, 2019.

[6] Ahmed *et al.*, "Web-based payroll management system: Design, implementation, and evaluation," *Kurdistan Journal of Applied Research*, vol. 5, no. 1, pp. 123–132, 2020.

[7] Y. Soegoto, "Designing payroll information system: Case study on cv. bandung id card," *International Journal of Information Systems and Computer Sciences*, vol. 8, pp. 58–65, 2019.