KIGALI INDEPENDENT UNIVERSITY ULK

SCHOOL OF SCIENCE AND TECHNOLOGY

DEPARTEMENT OF COMPUTER SCIENCE

P.O BOX 2280, KIGALI

TOPIC: ONLINE EMPLOYEE MANAGEMENT SYSTEM

CASE STUDY: GLORY ACADEMY

Done by:

UWIWE JEDID JAH: 202110461

SUPERVISED BY: MR. RUTARINDWA J. PIERRE

Dissertation Submitted to The School of Science and Technology in Partial Fulfillment of the Requirement for the Award of Bachelor's Degree in Computer Science.

September 2024

Declaration

This certifies that Uwiwe Jedid Jah's project report, "**ONLINE EMPLOYEE MANAGEMENT SYSTEM**: **CASE STUDY OF GLORY ACADEMY,"** which was turned in to the Kigali Independent with roll number 202110461, is the authentic record of the original work completed.

Signature

Date:9/25/2024

Uwiwe Jedid Jah

Approval

This is to certify that the project report entitled **"ONLINE EMPLOYEE MANAGEMENT SYSTEM: CASE STUDY OF GLORY ACADEMY"** is the record of the original work done and submitted **by Uwiwe Jedid Jah** with roll number:202110461 to the Kigali Independent University in partial fulfillment of the requirements for the award of bachelor's degree in computer science.

SUPERVISOR: MR. Rutarindwa Jean Pierre

Date:12/9/2024

Signature:

Dedication

To my lovely parents,

To my brothers and sisters,

And all my other friends and colleagues.

Acknowledgement

First of all, I am very grateful to the Almighty God for his love, care and support enduring forever. He has been with me all three years ago and have helped me to succussed till this achievement of completing my bachelor degree.

Thanks to the Kigali independent university founder **Dr. RWIGAMBA BALINDA** and the vice-chancellor **Prof. NKUNDABATWARE INNOCENT.**

My gratitude goes also to my supervisor **Mr. Rutarindwa J. Pierre**, thank you for the guidance and support you provided me during this work, i thank you for reading and comment you made which helped me to do this study.

My appreciation also goes to all lecturers that have been lecturing me all these three years for their support and guidance.

Thanks to all my colleagues who have been supporting me and to my Family that supported me in different ways

May the Lord bless you abundantly!

Uwiwe Jedid Jah

Abbreviation

CSS: Cascading Style Sheets

DFD: dataflow diagram

ERD: Entity Relationship Diagram

HR: Human resource

HTML: hypertext markup language

OEMS: Online Employee Management System

PHP: Hypertext Preprocessor

SSADM: Structured System Analysis and Design

Contents

List of Figure

List of Tables

Abstract

The Employee Management System for Glory Academy is designed to automate and streamline key human resource functions such as employee record management, salary computation, leave request handling, and report generation. The system aims to enhance operational efficiency, data integrity, and employee interaction within the academy's HR processes.

Glory Academy currently faces challenges related to manual HR management, including inefficiencies in handling leave requests, inaccuracies in salary calculations, and the cumbersome nature of manual data handling. These issues result in operational delays, errors, and difficulties in managing employee records effectively.

To address these challenges, the Structured Systems Analysis and Design Method (SSADM) was employed, incorporating tools like Data Flow Diagrams (DFDs), Entity Relationship Diagrams (ERDs), and data dictionaries to model the system's functionality accurately. The development process followed the Waterfall model, which facilitated a structured, phase-by-phase approach, including requirements gathering, system design, implementation, testing, and maintenance. Data was collected through observation and documentation, and the system was built using HTML, CSS, PHP, JavaScript, Bootstrap, and MySQL.

The expected result of implementing the Employee Management System is a significant improvement in the efficiency and accuracy of HR operations at Glory Academy. By automating manual tasks, the system is projected to reduce errors, streamline processes, and provide a secure, scalable solution that enhances overall HR management. This conclusion supports the need for automated systems to replace outdated manual practices, ensuring a more efficient, reliable, and user-friendly approach to managing employee information at Glory Academy.

Keywords: online, employee, management and system.

CHAPITER I: GENERAL INTRODUCTION

1.0 Introduction of the study

The Online Employee Management System is a software application developed to assist the HR manager at Glory Academy in efficiently managing employee information and automating key HR processes. The system provides a digital platform where the HR manager can add new employees, maintain detailed records, manage salary payments, and handle leave requests. This system not only enhances the accuracy of record-keeping but also streamlines the overall management of employee data, reducing the reliance on manual processes that are prone to errors and inefficiencies. Employees can interact with the system to request leave, check their salary details, and view historical payment records, fostering transparency and communication between the HR department and staff.

This chapter discusses the study's background, problem statement, objectives, scope, methodologies, significance, limitations, and organization. The research emphasizes creating a flexible and scalable system that can be easily updated to meet future needs, ensuring the long-term effectiveness of HR operations at Glory Academy. By integrating these functionalities, the system aims to address the challenges of manual data handling, improve operational efficiency, and provide a user-friendly environment that supports both HR managers and employees.

1.2 Background of the study

Employee management in schools was primitive in the 19th century, frequently handled by headmasters in the absence of official HR procedures. Because there were usually few teachers and small schools, formal staff management methods were not really necessary. The educational system was more hierarchical during this time, and instructors' duties were tightly defined with little opportunity for growth or advancement.

As public education system Employee Management System grew and the industrial revolution occurred, schools started using more structured staff management procedures. (Bush, T., Bell, L., & Middlewood, D., 2019).

Throughout the colonial era, Belgian colonial officials and Catholic missionary organizations controlled an important part of Rwanda's educational system. The way school staff were managed

was informal and heavily centralized. The majority of education was provided by missionaries, and rather than using organized human resource management techniques, teacher selection and management were dictated by religious and colonial goals

Rwanda's education system was severely damaged by the 1994 genocide, which also claimed th e lives of numerous teachers and destroyed school buildings.
Rebuilding the educational system was a top priority in the aftermath, and hiring and preparing new instructors was imperative.
The government took immediate action to hire teachers fast, frequently with little training, with help from foreign groups.
During this time, staff management prioritized the restoration of fundamental educational servi ces, placing special emphasis on the quick hiring of teachers and the creation of short-term training programs to address pressing demands.

The Ministry of Education began instituting more structured staff management practices, including as standard hiring and training procedures, in order to stabilize the educational system.

The Rwandan government implemented a number of changes in the 2000s with the goal of reforming and professionalizing the education industry. The creation of the Rwanda Education Board (REB) signaled an important move in the direction of organized personnel administration. During this time, professional development programs, performance reviews, and organized hiring procedures were among the more formal HR management practices that were introduced. The government set up teacher training colleges and programs for ongoing professional development in an effort to improve the caliber of educators.

Furthermore, the use of technology in personnel administration began to increase with the advent of digital system Employee Management System for tracking payroll, performance metrics, and instructor records. The purpose of these projects was to improve teacher management and professional development in order to raise the standard of education.

The goal of personnel management in Rwandan schools nowadays is to conform to global norms and best practices. Enhancing teacher quality, making sure that instructors are distributed fairly across the nation, and preserving teacher motivation through rewards and possibilities for professional growth are prioritized. Despite this, certain obstacles persist, such as addressing the

scarcity of teachers, especially in remote regions, and guaranteeing that staff supervision procedures are up to date with the swift modifications in academic requirements and technology breakthroughs (Mutumbuka, R., & Ngoga, A., 2017).

The development of online employee management system will help Glory academy to effectively manage its employee and establish an effective communication between employee and human resource manager.

1.2 Problem statement

The existing employee management system at Glory Academy faces several limitations, including manual employee registration, which is time-consuming, error-prone, and burdensome for HR staff, delaying the onboarding process for new hires. Salary records are also manually maintained, increasing the risk of inaccuracies and complicating the management of salary-related information. Permission requests are handled manually, causing delays in approval and leaving employees uncertain about the status of their requests, which negatively impacts their satisfaction and productivity. Additionally, employees do not have direct access to their payment history, requiring them to contact HR for this information, which disrupts workflow and adds to the HR staff's workload. These challenges, among others, motivated the design and implementation of an Online Employee Management System for Glory Academy to address inefficiencies and improve overall HR operations.

1.3 Objectives

1.3.1 General objective.

The main objective of the research project is to design and implement a software application of Online Employee Management System in Glory Academy.

1.3.2 Specific objectives.

i. To create a database that will store information of employees in glory academy
ii. To design a user-friendly interface that the human resource management and staff may both access via the system.
iii. To create a system application which helps to generate reports about management of employees.

1.4 Research questions

i. How can a database be created to store and manage employee at Glory Academy?
ii. What features are necessary to create an interface that is easy to use for both the human resource manager and employees?
iii. How can an employee management system be created and implemented to generate precise and comprehensive reports regarding the supervision of Glory Academy employees?

1.5 Scope of the study

1.5.1 Content scope

The scope of this study encompasses the development and implementation of an online employee management system in Rwanda, aimed at improving HR and employee interactions at Glory Academy. The primary objective is to create a system that facilitates the HR manager's tasks, including registering employees, generating reports, and managing leave permissions. Additionally, the system will enable employees to communicate with the HR manager, access their historical payment information, and submit leave requests. This integrated approach aims to streamline HR processes, enhance communication, and improve overall efficiency within the academy.

1.5.0 Geographical scope

Geographically this research was carried out at Glory Academy which is in Rwanda, Kigali city province, Gasabo district, Gisozi sector,102 KG 14Ave.

1.5.2 Time scope

To improve administrative efficiency, Glory Academy at Kigali Independent University (ULK) installed an online employee management system. In order to manage payroll, staff records, leave more effectively, this system was implemented. With time, the system's integration increased overall organizational transparency, decreased human errors, and optimized workflow. Glory Academy improved its administrative capabilities and was able to concentrate more on its educational purpose by implementing digital solutions like the Employee Management System and keeping an effective and responsible staff.

1.6 Significance of the study

1.6.1 Personal interest

Building this website will allow me to apply the knowledge and skills I have acquired at ULK. It offers a practical opportunity to enhance my technical abilities in web development. Through this project, I will gain hands-on experience and deepen my understanding of design, coding, and user interface. It will also help me develop problem-solving and project management skills. Completing this website will be a physical representation of my abilities, it will help me get ready for my next career moves in the computer sector.

1.6.2. Institutional interest

The book will be added to the school library so that students can use it as a helpful tool to help them pass their practical exams and as a trustworthy reference for their future research projects.

1.6.3 Public interest

Online Employee Management System improves organizational efficiency by automating administrative operations, which lowers mistake rates and saves time. It encourages accountability and openness inside the organization by keeping up-to-date, easily available records of personnel data and payment history. An OEMS promotes improved lines of communication between employee and the manager, guaranteeing that issues and requests like those for leave permission are addressed are handled fairly and promptly. By giving them self-service access to their historical payment and personal data, the technology empowers workers. Overall, the OEMS helps to create a more transparent, efficient, and well-organized work environment, which benefits management, employee, and the community at large by facilitating the smooth operation of the educational institution.

1.7 Project methodology

Project methodologies encompass the tools and guidelines employed to achieve research goals. In this project, data collection involved two primary methods: observation and documentation. Observation entailed gathering data through direct, real-time analysis of events and interactions, utilizing both structured and unstructured approaches to capture detailed and natural behaviors. Documentation involved the collection and analysis of existing records, including employee data, historical payments, and leave requests, to assess the current system's performance. For software development, the Waterfall model was chosen, a linear and sequential approach requiring the

completion of each phase—requirements analysis, design, implementation, testing, and maintenance—before moving to the next. The Structured Systems Analysis and Design Method (SSADM) guided the project, organizing it into clear, manageable phases. SSADM's tools, such as Data Flow Diagrams (DFDs), Entity-Relationship Diagrams (ERDs), and a data dictionary, were utilized to design the Online Employee Management System. DFDs provided visual representations of processes and data flow, with different levels detailing system functionalities. ERDs illustrated database entities and their relationships, while the data dictionary specified attributes, names, data types, and constraints, ensuring a well-defined system structure.

1.8 Limitation of the study

Sample size: This has restricted my research because Glory Academy is the exclusive topic of the study.

Limitation time: research has been limited due to the deadline of submission this have led me to not collect other useful information.

1.9. Organization of the project

My study consists of four chapters:

The first chapter, **"General Introduction"** gives an overview of the research project and explains the reasons behind selecting it, as well as the problem statement, objectives, and project methodology used for data collection. It also describes the study's scope and expected results.

Second Chapter titled **“Literature Review”** where we define the concepts and keys words, describe the environment of the system, the technics that will be used by the system solution and of how a new system will operating and review of related literature.

The third chapter called **“System Analysis and Design”** focuses on analyzing the current system and identifying its limitations, gathering system requirements, and designing the architecture for the proposed solution. It includes data flow diagrams (DFDs), entity relationship diagrams (ERDs), and a detailed description of the new system's structure and functionality, ensuring it meets the project's objectives.

The fourth chapter **"System Implementation"** outlines the process of converting the system design into a functional application. It covers the coding, testing, and integration of system

components, as well as the deployment of the system. Additionally, it addresses any challenges encountered during implementation and provides documentation for users and administrators.

The conclusion and recommendation will be discussed at the end of chapter four.

CHAPTER II: LITERATURE REVIEW

2.0 Introduction

This chapter's goal is to review the many authors' presentations of theoretical and empirical facts regarding the topic under study. It includes insights from magazines, journals, textbooks, websites, and electronic materials relevant to the topic. This review aims to offer a thorough overview of the issue by synthesizing and critically analyzing the existing literature. It will identify key themes, highlight significant findings, and pinpoint gaps in the current research.

2.1 Definition of key terms

2.1.0. Online

Online as used in computer technology and telecommunications, denotes a connected condition. Typically, this refers to an Internet connection, but it can also refer to any device or functional unit that is a part of a larger system, particularly when stated as "on line" or "on the line." Being online indicates that the equipment or subsystem is linked or operational (Dymoke, Arabella, 2024).

2.1.1 Employee

An employee is someone who performs services for an employer where the employer has the right to control what will be done and how it will be done (Hackerearth, 2024).

2.1.2 Management

Management is the process of planning, organizing, leading, and controlling resources, including human, financial, and material assets, to achieve organizational goals efficiently and effectively. It involves coordinating the efforts of people to accomplish desired objectives and ensuring optimal use of resources (Robbins, S. P., & Coulter, M., 2018).

2.1.3. System

A system is defined as an organized set of interrelated components or elements that work together to achieve a specific purpose or function. It encompasses inputs, processes, outputs, and feedback mechanisms that collectively ensure the system operates effectively and adapts to changing conditions. Systems are designed to process data or information in a structured manner to meet predefined objectives (Stair, R. M., & Reynolds, G. W., 2024).

2.1.4. Salary

Salary is defined as a fixed, regular payment made by an employer to an employee, typically on a monthly or biweekly basis, in exchange for the employee's services or work. It is often expressed as an annual sum but paid periodically throughout the year (Armstrong, M, 2020).

2.1.5 Human resource manager

A Human Resources Manager is a professional responsible within an organization of supervising and managing various HR operations. This role includes recruiting and hiring employees, managing employee relations, administering benefits and compensation, ensuring compliance with labor laws, and developing policies and procedures. The HR manager also plays a key role in employee development, training, and performance management, working to align the workforce with the organization's strategic goals (Armstrong, M., 2020).

2.1.6 Department

A department is a distinct division within an organization that focuses on a specific function, task, or area of expertise, such as human resources, finance, or marketing. Each department is responsible for carrying out specialized duties to contribute to the overall goals of the organization (Armstrong, M., 2020).

2.1.6. Leave

leave is the period during which an employee is permitted to be absent from work for personal reasons, such as sickness, vacation, or emergencies, with prior approval from management. The system helps in managing, tracking, and automating the leave request process to ensure efficiency and transparency in handling employee absences (Dessler, G., 2020).

2.2. Review of related literature

2.2.1. Database

A database is an organized collection of structured information or data, typically stored electronically in a computer system. Databases are designed to efficiently manage, retrieve, and manipulate data, providing a systematic way to store and access large amounts of information. They are managed by Database Management Systems (DBMS), which facilitate the creation, querying, updating, and administration of the database (Harrington, J. L., 2024).

A Database Management System (DBMS) is software that provides an interface for users to create, retrieve, update, and manage data in databases. It ensures data consistency, security, and integrity by controlling access to the database, managing transactions, and providing tools for data backup and recovery. DBMS supports the organization of data into structured formats like tables, making it easy to manage large amounts of information efficiently (Elmasri, R., & Navathe, S. B., 2022).

2.2.1.1 Types of databases

A relational database is a database that arranges data into rows and columns to create a table where the data points are connected to one another.

Structured Query Language (SQL) is the most common language for reading, creating, updating and deleting data.

Object-oriented databases: An Object-Oriented Database (OODB) is a type of database that stores data in the form of objects, as used in object-oriented programming. In an object-oriented database, data is represented as objects, which include both data (attributes) and behaviors (methods). This approach integrates database capabilities with object-oriented programming principles, allowing for better handling of complex data types and relationships (Harrington, J. L., 2024).

Distributed databases: Two or more files dispersed over multiple sites comprise a distributed database. The database may be stored on multiple computers, distributed over numerous networks, or hosted in a single physical location.

Data warehouses: is a type of database designed specifically for fast query and analysis. It serves as a central data repository.

NoSQL databases: Unstructured and semi structured data can be stored and manipulated in a nonrelational database, which is different from relational databases in that it specifies the structure of all data added into the database.

Graph databases is a type of database designed to handle and store information in the form of nodes which stand in for entities, and edges, which show relationships between entities. It works well for managing complex relationships and related data.

A cloud database: is a collection of data, either structured or unstructured, that resides on a private, public, or hybrid cloud computing platform. There are two types of cloud database models: traditional and database as a service (DBaaS). With DBaaS, administrative tasks and maintenance are performed by a service provider (Harrington, J. L., 2024).

2.2.1.2 Advantages and disadvantages of databases

2.2.1.2.1 Advantages of database

- **Data Consistency and Integrity:** Databases ensure that data is consistently updated and maintained, preventing inconsistencies and ensuring data accuracy across various users and applications.
- **Improved Data Security:** Databases provide access control mechanisms to protect sensitive information, ensuring only authorized users can access, modify, or delete data.
- **Efficient Data Management:** With centralized data storage, databases allow for efficient data retrieval, management, and backup, improving overall productivity.
- **Data Sharing and Accessibility**: Multiple users can access and share data concurrently, improving collaboration within an organization without compromising data integrity.
- **Redundancy elimination:** databases reduce data redundancy by organizing data, removing duplication and optimizing storage capacity (Silberschatz, A., Korth, H. F., & Sudarshan, S., 2019).

2.2.1.2.2 Disadvantages of databases

- **Complexity and Cost:** Database systems can be complex to design and implement, requiring skilled personnel and significant financial investment for setup and maintenance.
- **Security Risks:** Although databases provide security measures, they are still vulnerable to data breaches, unauthorized access, and cyber-attacks if not properly managed.
- **Performance Issues:** Large databases with numerous concurrent users may experience performance slowdowns, especially if not optimized or if hardware resources are insufficient.
- **Maintenance Overhead:** Regular maintenance, updates, and backups are necessary, requiring ongoing time and resource investment to ensure the system functions optimally (Connolly, T., & Begg, C., 2020).

2.2.1.3. Importance of database

- Databases enable businesses to efficiently and methodically store and handle massive volumes of data.
- Using databases, organizations can improve efficiency and simplify procedures.
  Automating routine tasks, such as generating reports using a database or sending emails, can reduce errors and save time.
- Databases have security features that help businesses protect their data, such as user acc ess limits and backup and recovery capabilities.
  This could reduce the possibility that private information would be misplaced or accesse d by uninvited individuals.
- Databases can make it easier for organizations to collaborate and share data. For example, allowing multiple people to view and change data in a database at the same time could promote collaboration and teamwork (Connolly, T., & Begg, C., 2020).

2.2.2. User interface and experience

2.2.2.1 User-friendly interfaces

A user-friendly interface is designed to be intuitive, easy to use, and accessible, allowing users to interact with software or systems efficiently without requiring extensive training or technical expertise. Key characteristics of a user-friendly interface include clear navigation, consistent design elements, responsive feedback, and minimal complexity, ensuring that users can achieve their desired tasks with ease and satisfaction (Shneiderman, B., & Plaisant, C., 2010).

Characteristics and User-friendly interfaces

- **Simplicity**: Minimalistic design that avoids clutter and confusion.
- **Consistency:** the uniformity of design elements and interface behavior.
  Accessibility: Features that ensure the interface is usable by users of various abilities.
  Responsiveness: Quick and appropriate responses to user inputs.
- **Intuitive Navigation**: Easy to understand and use menus, buttons, and links. (Shneiderman, B., & Plaisant, C., 2020)

2.2.2.2 User-friendly experiences

A user-friendly experience encompasses the overall interaction a user has with a product or service, including emotional, cognitive, and physical aspects. It goes beyond the interface to include how the product or service meets the user's needs and expectations. (Nielsen, J., & Norman, D. A., 2020)

Characteristics of User-Friendly Experiences

- **Usability:** Efficiency in completing tasks and user-friendliness
- **Desirability:** Emotional satisfaction and appeal of the product
- **Accessibility:** Ease of access for users of all abilities
- **Value:** The perceived benefit or usefulness of the product or service
- **Engagement:** The ability to attract and retain users' interest and participation (Garrett, J. J., 2020).

2.2.3. Report

A **report** is a structured document that presents information on a specific topic or issue, often including data, analysis, and findings. Reports are used to communicate results, provide insights, and support decision-making processes. They can vary in format, ranging from formal research reports to informal status updates, and typically include an introduction, methodology, results, and conclusions or recommendations (Hargreaves, A., 2018).

2.2.3.1 Types of reports

- **A formal report** is an official report that contains detailed information, research, and data necessary to make business decisions.
- **An informal report** is a brief document distributed among employees to apprise them of pertinent policies and facts.
- **An analytical report** is a type of business document where information is displayed based on facts and leads to a specific conclusion.
- **A proposal report** is a unique kind of professional report that serves two distinct purposes: to convince the reader of your suggested course of action for resolving a problem or issue. to persuade the reader to follow your advice**.**

- **Financial reporting** is the process of giving statistics and financial information about a company's financial performance and condition to external stakeholders**,** such as creditors, investors and regulatory bodies.
- **An annual report** is a document that public companies are required to give their shareholders each year that details their financial situation and operational results**.**
- **An evaluation report** is a paper that examines whether a product, service, or process is working, according to a set of standards.
- **A project report** is a document providing detail on the project's overall status or specific aspects of its performance (Kolin, P. C., 2016).

2.2.3.2. Importance of a report

- **Instrument for Making Decisions:** In today's complex business organizations, thousands of bits of information are required. Reports provide the essential information, and many important business and other decisions are made on the basis of the information they offer.
- **Investigation:** Every problem is investigated by a committee, commission, or study group to find out what's causing it. After that, they present their results in a report that may or may not include recommendations.
- **Evaluation:** Big organizations handle numerous activities. It is not possible for one
- top executive to personally oversee every activity taken by everyone else.
  As a result, the executive reviews report to determine how various departments or divisions are doing.
- **Neutral fact presentation:** It is necessary for facts to be presented in a neutral manner, which is ensured by a report's independent investigation, explanation, and evaluation of all relevant facts.
- **A management tool:** A variety of reports simplify tasks for managers. A report that serves as a source of information is necessary for the management to plan, organize, coordinate, motivate, and control (Robbins, S. P., & Coulter, M. , 2016).

Chapter III: SYSTEM ANALYSIS AND DESIGN

3.1 Introduction

Online Employee Management System is a system designed to manage employee and establish effective communication between employee and the human resource manager in Glory Academy. This chapter outlines the process involved in creating the Online Employee Management System, detailing the steps taken during its design and development. After analyzing the current employee management, I have found the existing challenges such as manual record of employee and their salary, delayed approval of leave permission for employee and restriction from payment history unless they go to meet the human resource manager. This chapter suggest a new system in order to solve these problems, outline functional and non-functional requirement. I used data flow visualization techniques to demonstrate how information moves through the system and discuss methodological approaches for effective system development, this chapter also discuss different data collection techniques used, the structured system analysis and design method to ensure that the system will be user friendly for both employee and human resource manager.

3.2 Analysis of current system

Manual Record-Keeping: The Glory Academy's present existing employee information management relies on manual management on manual procedures. This manual process takes a lot of time; therefore, it is easy to make mistakes. and makes data retrieval and updates inefficient. The lack of a centralized database also makes it difficult to maintain consistency and accuracy in employee records.

Problem with Leave Management: Currently, leave requests are handled manually, often through paper forms or emails. This process is not only slow but also lacks transparency, as employees do not have a clear way to track the status of their leave requests. HR managers face challenges in tracking leave balances and ensuring that all leave requests are properly documented and approved, leading to potential conflicts and misunderstandings.

Problem with Payroll and Salary Management: Employee salaries are kept in physical files or spreadsheets, which can lead to mistakes and delays. There is a greater chance of error when handling salary components like bonuses, deductions, and overtime since it involves a lot of

human labor. It is also challenging to keep an accurate record of compensation payments and modifications when there is no automated system in place for salary administration.

3.3 Analysis of the new system

3.3.1. Introduction

The new online employee management system plays a big role in any organization because it directly impacts productivity, employee satisfaction, and overall academy efficiency. In educational institutions like Glory Academy, effective employee management is essential to maintain a motivated and well-organized workforce. This project suggests creating an Online Employee Management System that is especially designed to meet Glory Academy's demands in order to address these problems. The new system aims to automate and streamline certain HR tasks, including processing leaves, administering payments, keeping track of employees, and reporting. The OEMS will offer a single system for managing personnel information, enhancing communication, and enabling data-driven decision-making.

3.3.2 System requirement

System requirements are specifications that outline the necessary features, behaviors, and constraints a software system must satisfy to achieve its intended objectives. These include functional requirements, which specify the system's tasks and processes, and non-functional requirements, which address performance, reliability, security, and user experience aspects. System requirements provide a comprehensive guideline for developers, ensuring that the system meets user expectations and complies with predefined standards throughout the design, implementation, and testing phases. (Sommerville, I., 2020).

These specifications set forth the features that the Online Employee Management System must have in order to properly handle HR activities and employee interactions. Functional requirements outline what the system should do, while non-functional requirements describe the manner in which the system should operate. these requirements will be clearly defined an understanding of these specifications helped to develop a strong, effective, and user-friendly OEMS that improves both operational effectiveness and employee satisfaction.

3.3.2.1 Functional requirement

Functional requirements outline the specific actions, tasks, and behaviors that a system must perform, defining the core functionalities it needs to provide. They describe how the system interacts with users, handles data, and responds to different inputs to produce the desired outputs. These requirements are crucial for guiding the development process, ensuring that the system delivers the expected features and meets user needs effectively (Van Vliet, H. , 2020).

The functional requirement includes the following:

- The system will allow the HR manager to login and to logout
- The system will allow the HR manager to add and manage departments
- The system will allow the HR manager to record and manage employees
- The system will allow the HR manager to approve or reject leave permission and check leave history for each employee
- The system will allow employee to login and to logout
- The system will allow employee to check for their salary
- The system will allow employee to apply for leave permission and check for his/her leave history.

3.3.2.2 Non-functional requirement

Non-functional requirements describe the quality attributes, system constraints, and operational characteristics that a system must meet. These include performance metrics, usability, reliability, security, and scalability, focusing on how the system performs its functions rather than what functions it performs. (Pohl, K., & Rupp, C., 2018)

Scalability: The system should be able to handle an increase in users and data without any problems. Going forward, support should be provided for both the volume of data and the number of employees. Regular backups and the implementation of a disaster recovery plan are necessary for data restoration.

Performance: The system needs to be able to complete tasks given to it quickly.

Reliability: The system must always be able to finish the tasks that are assigned to it.

Data protection: Sensitive information should be encrypted by the system while it is in transit and at rest to guarantee data integrity and confidentiality.

User Interface: The user interface should be easy to use and intuitive so that users can operate the system and complete activities with little to no training.

Compatibility: To guarantee accessibility on many platforms, the system should work with the most popular web browsers (Chrome, Firefox, Safari, and Edge).

Availability: Every user should have at all times access to the system.

Maintainability: Regular system updates are necessary to fix bugs, resolve security flaws and enhance system performance.

3.3.3 Functional diagram

A **functional diagram** is a diagram that shows different functions or processes of a system and how they interact with each other. It is beneficial to understand how different system components interact to accomplish overall objectives.

Functional diagram would illustrate the different functions of the system, such as managing employee records, processing salary payments, handling leave requests, and generating reports. It would demonstrate how these roles are related to one another and how they cooperate to give the HR manager and employees a smooth management experience.

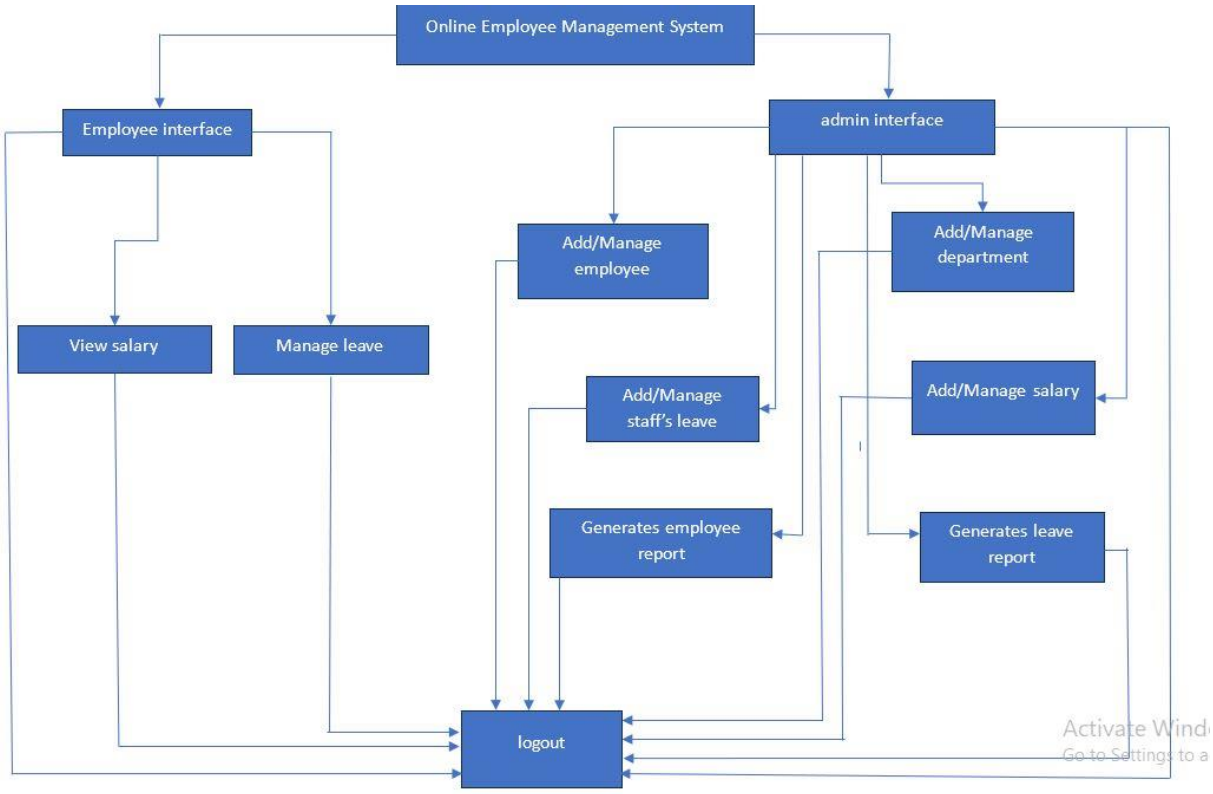

Figure 1 Functional diagram visually represents the relationships and interactions between different functions within a system, showing how inputs are processed into outputs.

3.4. Methodological approach

3.4.1. Data collection

3.4.1.1. Observation

In my research i have used observation, a qualitative data gathering method, to methodically observe and document processes, interactions, and behaviors as they occur in natural environments. During my research observation has provided important information on how employee write letter to apply a permission to leave and the HR manager have to sign on the paper in order to grant them a permission also employee request for historical payment face to face with the HR manager in her office.

3.4.1.2. Documentation

Documentation is a data collection method that involved collecting and analyzing existing records and documents relevant to my research. This method included collecting how Glory academy keep employee record, historical payment and leave request.

3.4.2. Software development methodology

Software methodology refers to a structured approach or set of practices used to plan, design, develop, test, and manage the software development process. It encompasses a set of rules, techniques, and procedures that guide the team in creating high-quality software that meets user requirements. Software methodologies help organize tasks, ensure effective collaboration, and manage the complexity of large-scale software projects (Sommerville, I., 2020).

The project moves through a number of clearly defined phases in the traditional software development process known as the Waterfall model. The model is called "Waterfall" because the process flows in one direction, like water cascading down a waterfall, from one stage to the next. Each phase must be completed before the following phase begins, making it a structured and sequential approach. The key steps in the Waterfall model are:

- **Requirement Gathering and Analysis**: In this initial phase, all system requirements are gathered from stakeholders. To guarantee that the requirements for the system are understood clearly, this calls for in-depth analysis. The goal is to document all the functional and non-functional requirements, which serve as the foundation for the entire project.
- **System Design:** Once the requirements are understood, the next phase involves designing the system architecture. This includes both high-level design (breaking down the system into modules) and low-level design (defining how each module will function). The outcome is carefully documenting the design process for the system.
- **Implementation (or Coding):** After the design phase, the actual development or coding begins. Developers write the code based on the previously defined designs. Each module or component is developed and unit-tested individually.
- **Integration and Testing:** Once all modules are implemented, they constitute the entire system when combined. The integrated system is entirely tested for defects, bugs, or errors. This phase ensures that the system meets the requirements and functions correctly.
- **Deployment:** After successful testing, the system is deployed into the production environment where it becomes accessible to end users. This phase includes setting up the system in real-world conditions, ensuring proper configurations, and making it operational.

- **Maintenance:** Once the system is live, the final phase involves ongoing maintenance. This involves fixing any problems that arise during practical use, making updates, and improving the system over time (Sommerville, 2020).

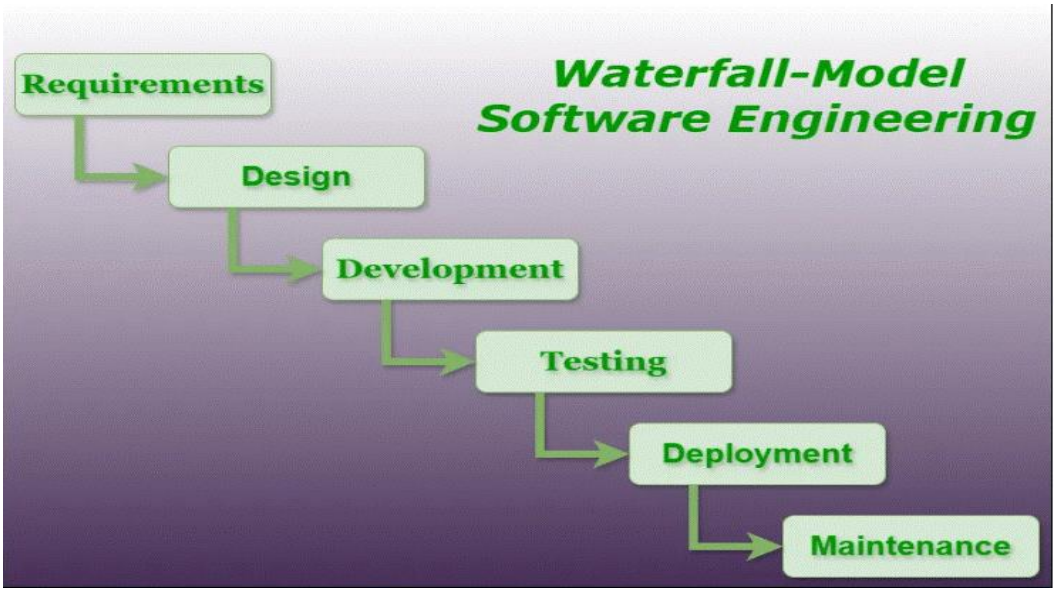

Figure 2 Waterfall model illustrates a linear, sequential software development process where each phase must be completed before the next begins.

3.4.3 System design methodology

Structured System Analysis and Design Method (SSADM): This is a traditional method to system development that plans, analyzes, designs, implements, and maintains information system through a set of stages called the waterfall model. Important methods consist of: **Data Flow Diagrams (DFDs):** Illustrative charts that illustrate the flow of data within an information system and the steps the system takes to process it or is a visual representation that illustrates how data moves through a system. It shows how data is transmitted through various processes, entities, and data stores, highlighting how inputs are transformed into outputs. DFDs are used in system analysis and design to model and document system functionality and data pathways.

Entity-Relationship Diagrams (ERDs): diagrammatic methods for representing the relationships between data entities and the logical structure of databases.

3.4.3.1 Level 0

Level 0 DFD is another term for Context Diagram. It represents the entire system as a single process with external entities interacting with it. It provides a high-level summary of the system's functionality without going into detailed processes, showing the system as one process box and its relationships with external entities.

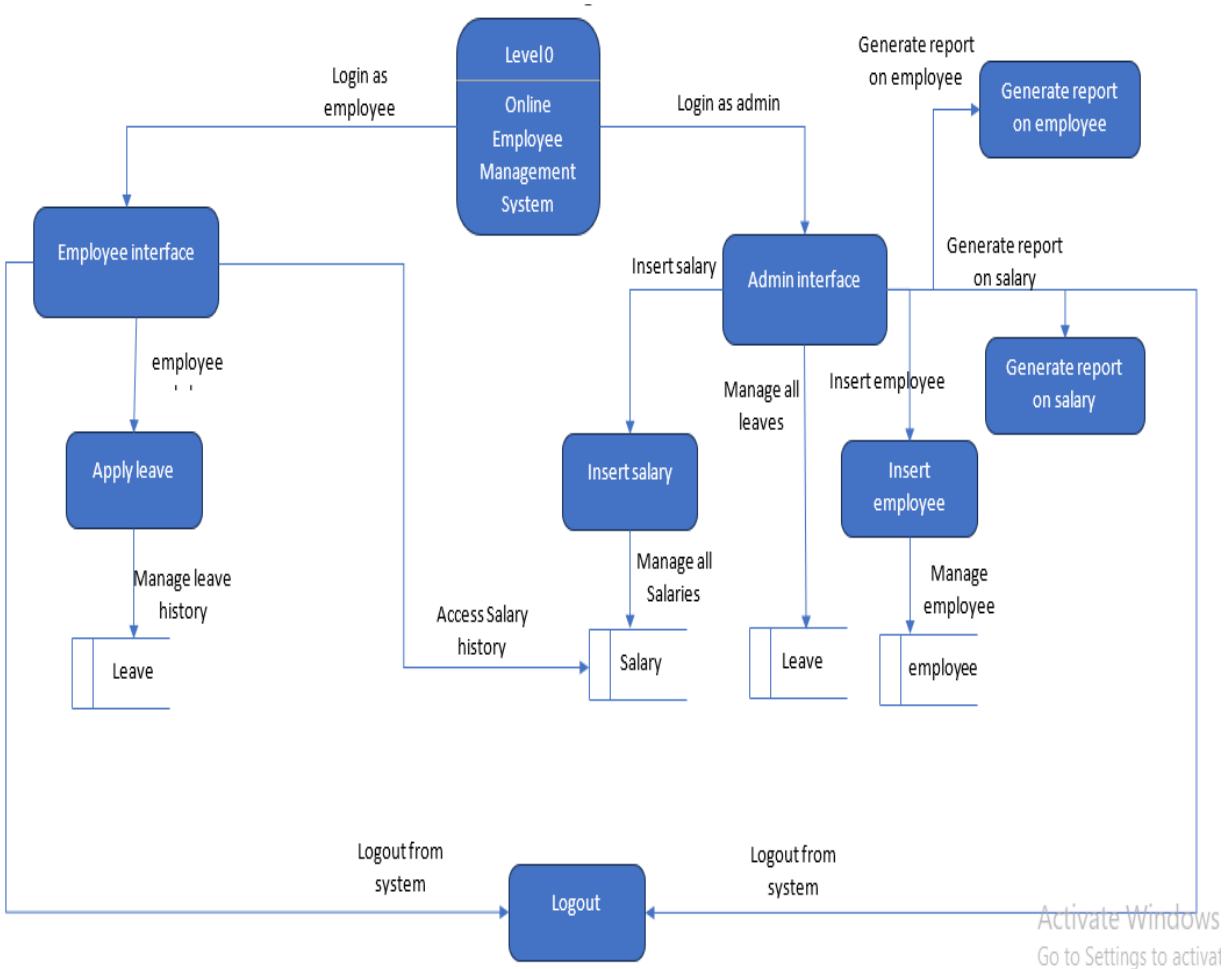

Figure 3 Level 0 represents the entire system as a single process with external entities interacting through inputs and outputs.

3.4.3.2. Level 1

A Level 1 Data Flow Diagram (DFD) provides a detailed view of the system's processes by breaking them down into sub-processes. It shows how individual functions or modules within the system interact with external entities and how data flows between them.

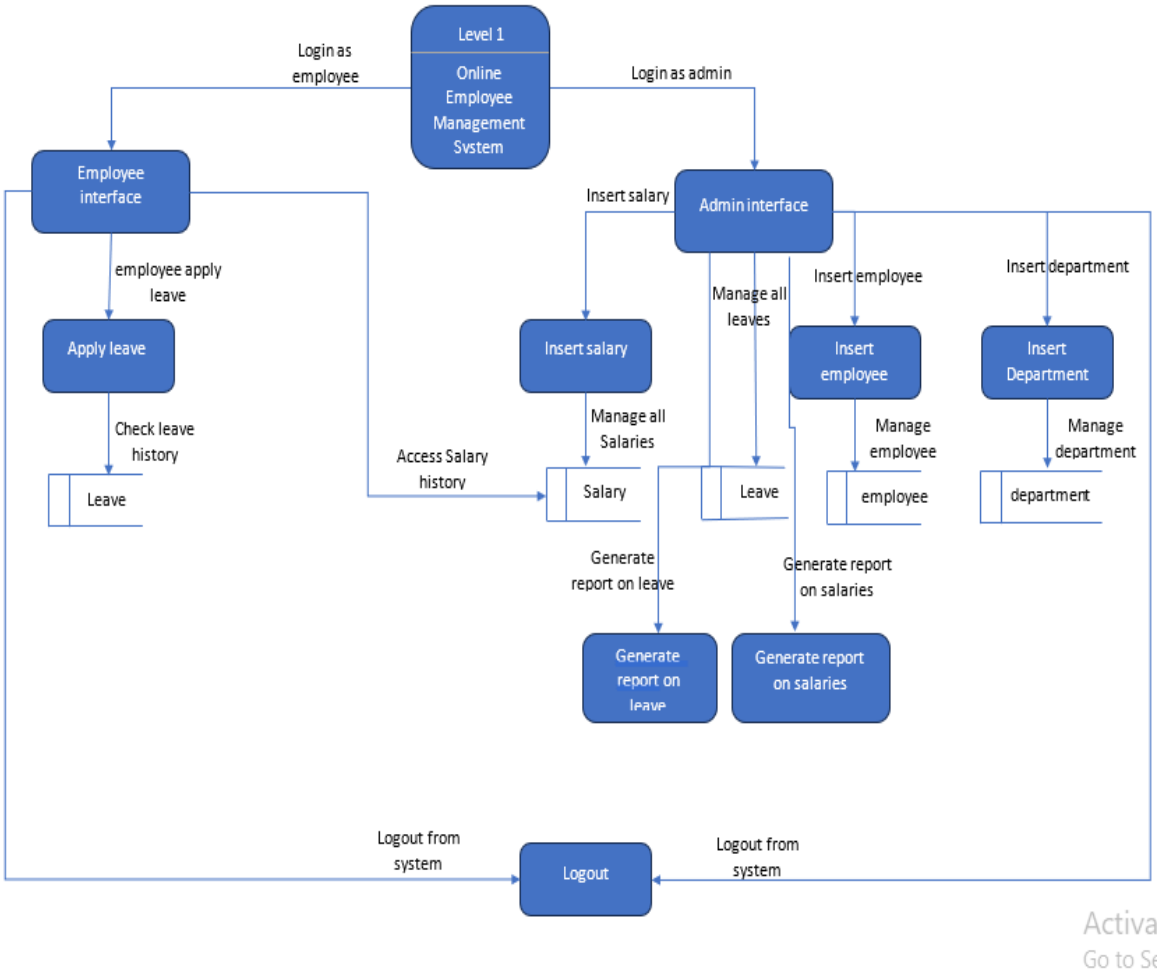

Figure 4 Level 1 breaks down the main process from Level 0 into sub-processes, showing more detailed interactions between data flows and external entities.

3.4.3.3 Entity Relationship Diagram (ERD)

An Entity Relationship Diagram (ERD) is a visual illustration of the relationships between entities in a database system. It demonstrates the structure of data, showing how entities (such as people, objects, or concepts) are related to each other. Entities are typically represented as boxes, and relationships between them are shown using lines. ERDs also include attributes (properties or characteristics of entities) and may define key constraints that dictate how entities interact, making them essential for database design and system modeling.

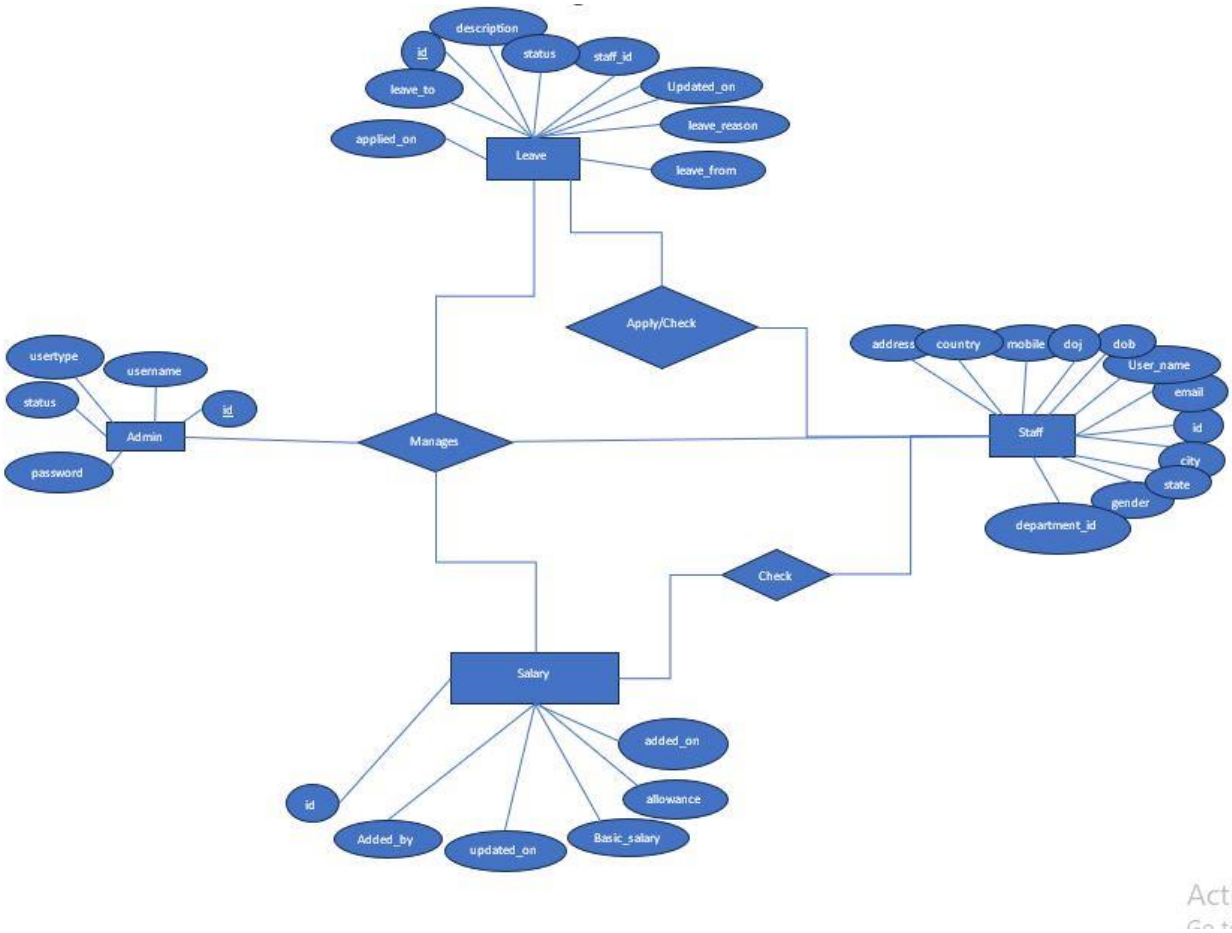

Figure 5 Entity Relationship Diagram visually represents the structure of a database, illustrating entities, their attributes, and the relationships between them.

3.4.3.4 Data dictionary

A data dictionary is a centralized repository that contains definitions and details about the data elements used in a system or database. It describes the structure, relationships, formats, and constraints of the data, including information about each field, such as data type, length, allowed values, and meaning. The data dictionary ensures consistency in data usage and helps developers, analysts, and users understand how data is organized and managed within a system. It's an essential tool for system design, development, and documentation.

Table 1 login_tbl

Table	Field	Data type	Constraint
	id	int (11)	Primary key
	username	varchar (80)	Null
	password	varchar (80)	Null
	User_type	int (11)	Null
	status	int (11)	Null

Table 2 staff_tbl

Table	Field	Data type	Constraint
	id	int (11)	Primary key
	staff_name	varchar (150)	Null
	gender	varchar (11)	Null
	email	varchar (150)	Null
	mobile	bigint (20)	Null
	dob	date	Null
	doj	date	Null
	address	text	Null
	city	varchar (100)	Null
	state	varchar (100)	Null
	country	varchar (50)	Null
	department_id	int (11)	Null

	pic	varchar (150)	Null
	added_by	Int (11)	Null
	updated_on	date	Null
	added_on	time_stamp	Null

Table 3 salary_tbl

Table	Field	Data type	Constraint
	Id	Int (11)	Primary key
	staff_id	Int (11)	Null
	basic_salary	bigint (20)	Null
	allowance	bigint (20)	Null
	total	bigint (20)	Null
	added_by	Int (11)	Null
	updated_on	date	Null
	added_on	timestamp	Null

Table 4 leave_tbl

Table	Field	Data type	Constraint
	id	int (11)	Primary key
	staff_id	int (11)	Null
	leave reason	varchar (90)	Null
	description	text	Null
	status	int (11)	Null
	leave_from	date	Null
	leave_to	date	Null
	update_on	date	Null
	applied_on	date	Null

Table 5 department_tbl

Table	Field	Data type	Constraint
	id	int (11)	Primary key
	department_name	varchar (100)	Null
	Added_on	timestamp	Null

Table 6 country_tbl

Table	Field	Data type	Constraint
	id	int (11)	Primary key
	country_code	varchar (2)	Null
	country_name	varchar (100)	Null

CHAPTER 4: SYSTEM IMPLEMENTATION

4.1 Implementation and coding

4.1.1. Introduction

system implementation is also known as the software development lifecycle it is when the planned system is constructed, tested, and made available for usage. This step involves transforming the system design for Glory Academy's Online Employee Management System into a functional application that handles leave management, salaries management, and employee records management. Writing code, integrating different system components, and setting up the system to satisfy Glory Academy's unique requirements are all part of this process. Thorough testing is also a part of system implementation to make sure the OEMS performs as planned and satisfies all functional and non-functional requirements. For the new automated system to seamlessly replace manual procedures and improve productivity, accuracy, and communication within the company, proper implementation is essential. Careful planning, execution, and support during the implementation phase are essential to the OEMS's success.

Throughout the development and design process different tools have been used those tools are HTML, CSS, JAVASCRIPT, PHP, MYSQL, XAMPP**,** BOOTSTRAP and IDE.

4.1.2. Description of implementation tools

HTML: HyperText Markup Language is the standard markup language used to create web pages. It is a combination of Hypertext, and Markup language. The Hypertext defines the link between web pages, and Markup is used to define the text document within tags to structure the web pages. This markup language is used to annotate text so that machines can understand and manipulate it accordingly. HTML is human-readable and uses tags to define what manipulation has to be done on the text (GeeksforGeeks, 2024).

CSS: Cascading Style Sheets (CSS) is a stylesheet language used for describing the presentation and formatting of web pages written in HTML or XML. CSS controls the layout, colors, fonts, and overall visual appearance of web elements, allowing developers to create visually appealing and responsive designs that adapt to different screen sizes and devices. It separates content from design, enabling easier maintenance and consistent styling across multiple web pages (Meyer, E. A., & Weyl, R. , 2018).

JavaScript: JavaScript is a high-level, dynamic programming language commonly used to create interactive effects within web browsers. It enables developers to build complex web applications by allowing client-side script to interact with the user, control the browser, and asynchronously communicate with servers. JavaScript enhances the user experience by adding interactivity, such as form validations, animations, and dynamic content updates, making it a core technology alongside HTML and CSS for web development (Flanagan, D. , 2022).

PHP: PHP (Hypertext Preprocessor) is a popular open-source server-side scripting language primarily used for web development. It is designed to create dynamic and interactive web pages, allowing developers to embed scripts within HTML, manage databases, handle sessions, and build entire web applications. PHP's flexibility, ease of use, and strong database connectivity make it a preferred language for backend development, enabling efficient interaction between the server and the user interface (Welling, L., & Thomson,, 2020).

MySQL: MySQL is an open-source relational database management system (RDBMS) that uses Structured Query Language (SQL) for managing and manipulating data. It is known for its speed, reliability, and ease of use, making it one of the most popular database systems for web applications. MySQL supports large-scale data handling, allows for complex queries, and is commonly used in conjunction with programming languages like PHP to build dynamic websites and applications (Welling, L., & Thomson, L., 2020).

XAAMP: Apache Friends developed XAMPP, a cross-platform web server solution stack package that is available for free and open source. By combining a number of necessary parts, it makes the process of creating a local development environment easier: the Apache HTTP Server (used for web page serving), MySQL (used for database administration), PHP (used for server-side scripting), and Perl (used for scripting). Before publishing their web apps to a live server, developers can test and refine them locally using XAMPP. It is especially helpful for creating PHP-based apps and has an easy-to-use control panel to facilitate controlling the different parts.

Bootstrap: Bootstrap is a popular open-source front-end framework developed by Twitter for designing responsive and mobile-first websites. It provides developers with a collection of pre-designed HTML, CSS, and JavaScript components such as navigation bars, buttons, forms, and grids, which significantly streamline the web development process. Bootstrap's responsive grid system, extensive documentation, and customizable components enable developers to quickly

create aesthetically pleasing and consistent user interfaces across different devices and screen sizes (Otto, M., & Thornton, J., 2018).

IDE: or Integrated Development Environment is a software program that offers a wide range of capabilities and tools to make software development easier. IDEs are made to make coding easier by combining multiple features into one interface. These features include:

Code Editor: An editor that helps to write and modify code more effectively by providing features like syntax highlighting, code completion, and error detection. **Debuggers**: are tools for finding, analyzing, and resolving coding faults. It enables code step-through, breakpoint setting, and variable inspection for developers.

Compiler/Interpreter: Depending on the programming language being used, this software converts source code into executable programs or bytecode.

Popular IDEs examples:

Visual studio code

Eclipses

PyCharm

4.1.3 Screenshot

4.1.3.1 Admin dashboard

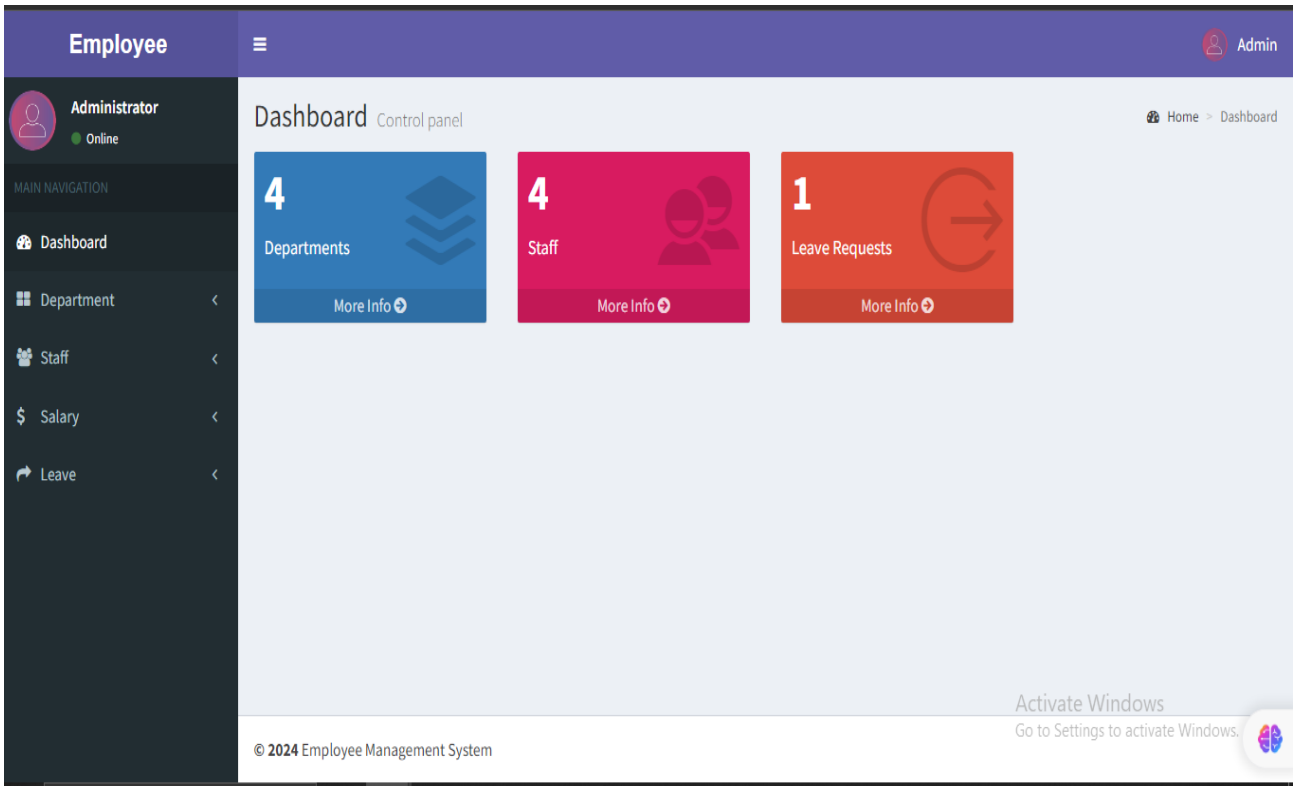

Figure 6 admin dashboard: An admin dashboard provides a centralized interface to efficiently manage and oversee system operations such as add department and manage department under department, add staff and manage staff under staff and manage staff's leave and leave history under leave.

4.1.3.2 Admin manage salary

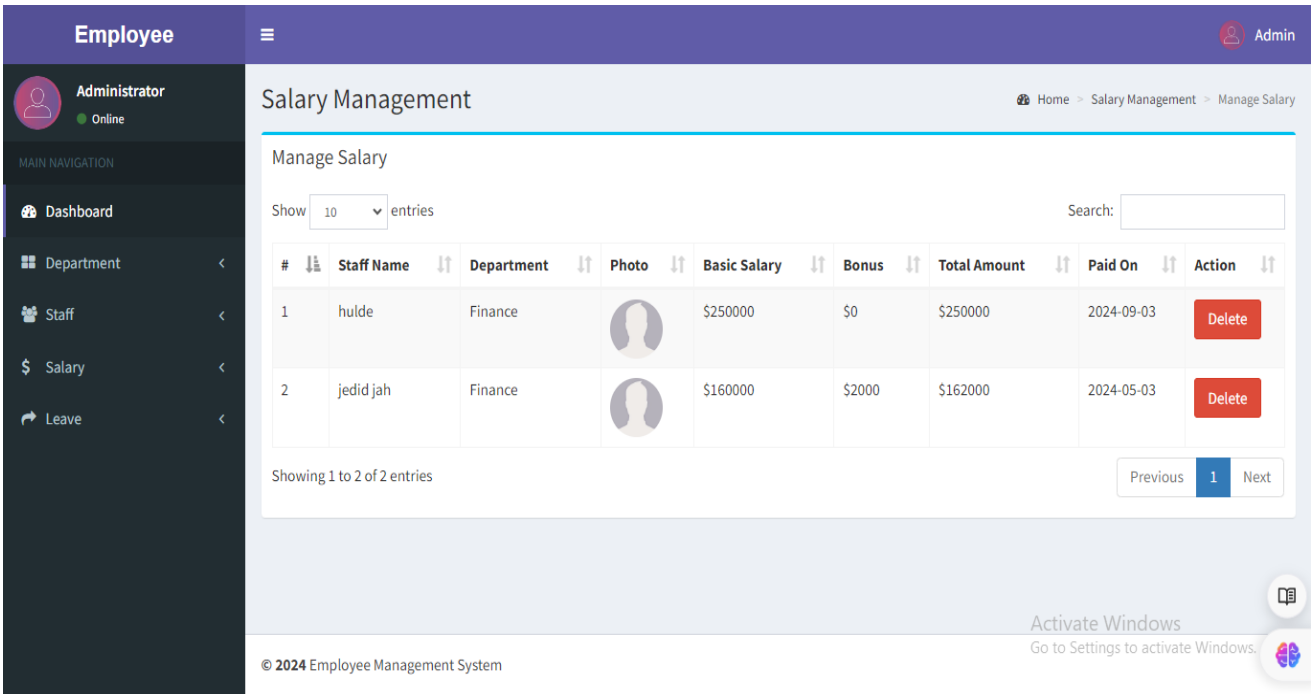

Figure 7 Admin manage salary: through this interface the admin can manage salary of each employee by delete or editing any field within the interface.

4.1.3.3. Admin manage staff's leave

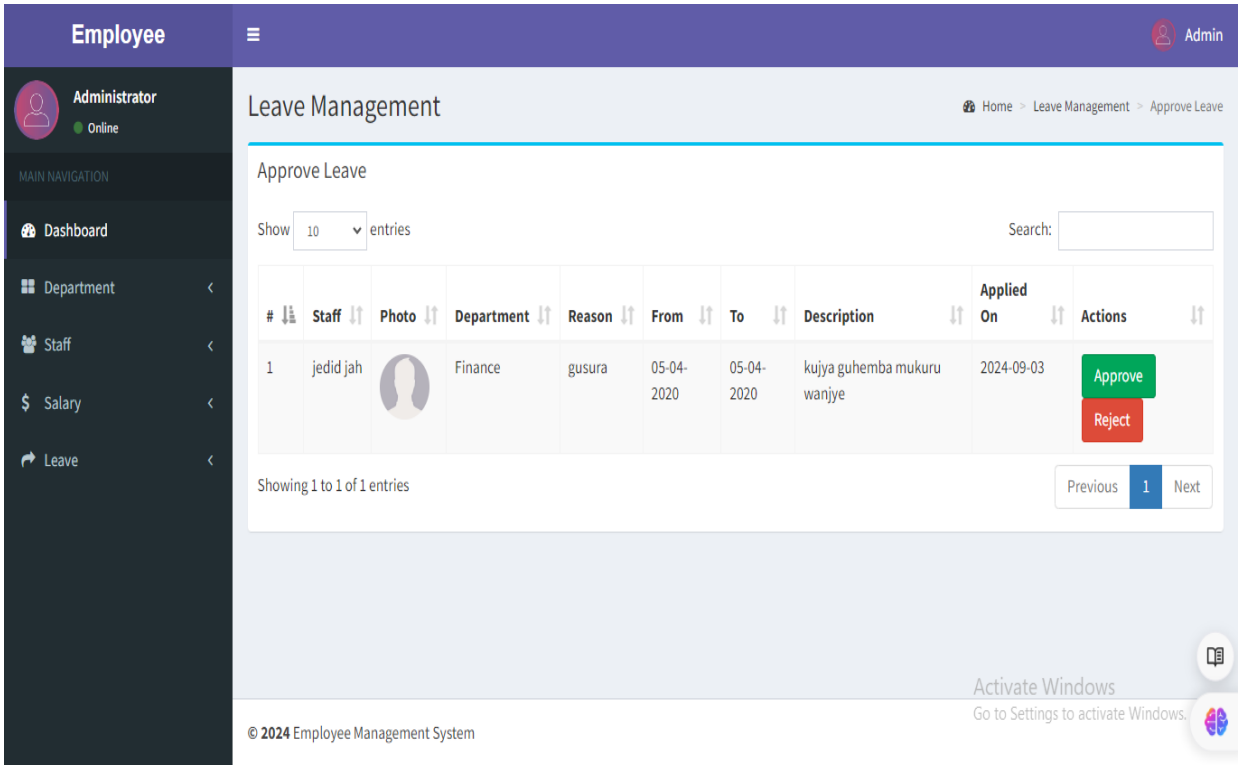

Figure 8 admin manage staff's leave: through this interface the admin manages leave by approving or rejecting the permission of employee who have applied to leave.

4.1.3.4 Employee dashboard

Figure 9 Employee dashboard or staff dashboard displays an organized interface with quick access to salary and leave where employee can apply a permission to leave and can view leave history.

4.1.3.5 Employee salary

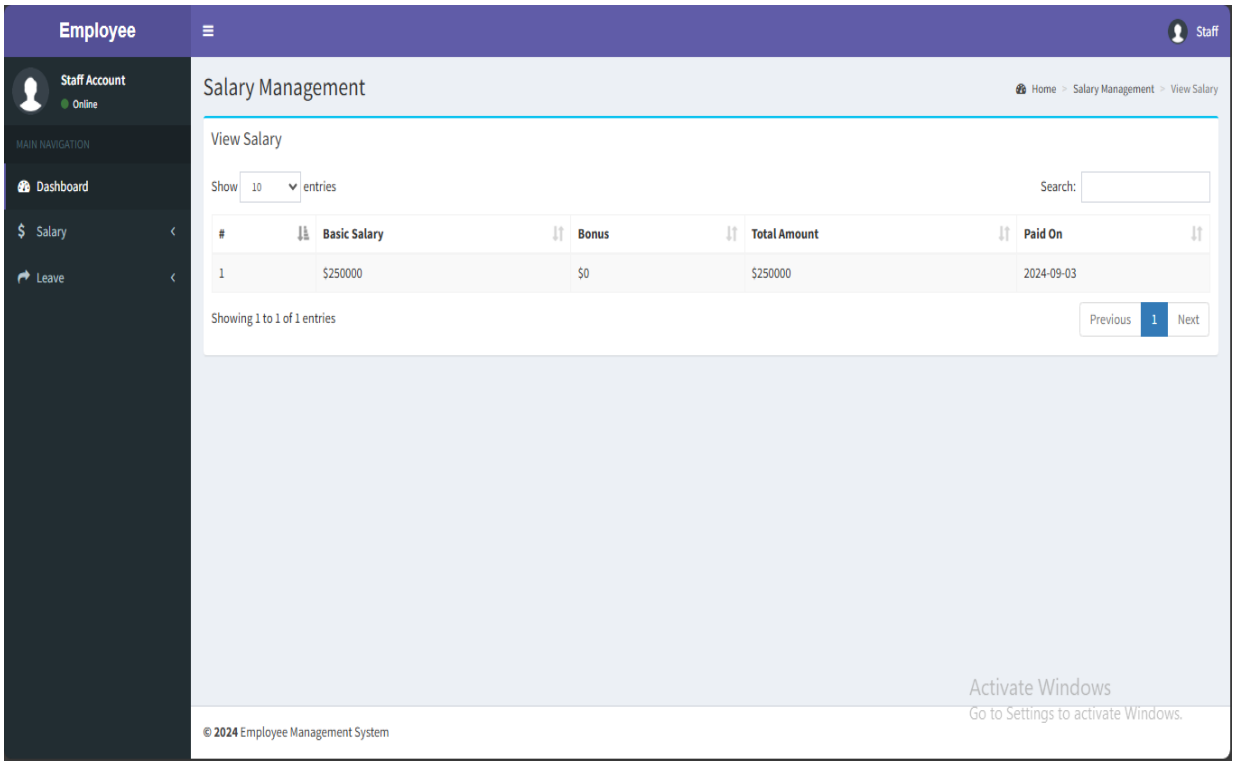

Figure 10 Employee salary: through this interface of salary management an employee can view his/her salary.

4.1.3.6 Employee apply leave

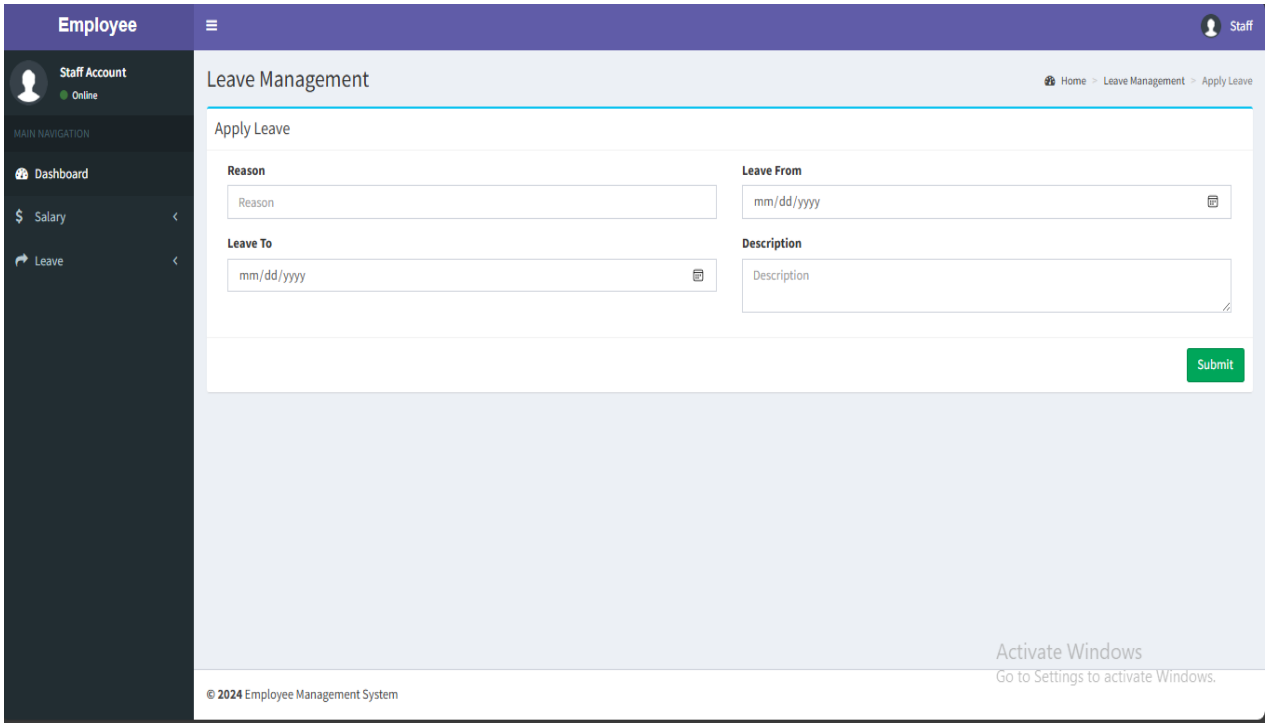

Figure 11 Employee apply leave: through this interface an employee can apply for a permission to leave.

4.2. Testing

4.2.1 Introduction

In order to make sure the Glory Academy Employee Management System works as planned and satisfies all requirements; testing is a crucial stage of the development process. Before the system is put into service, this phase entails methodically assessing the OEMS to find and address any problem. To ensure that every part of the system, including leave management, personnel records management, and salary payment functions accurately and consistently, a variety of testing methods, including unit testing output, validation testing output, integration testing, functional and system testing and acceptability testing, will be used. Testing also makes sure the system is safe, user-friendly, and satisfies performance requirements. Ensuring a strong and dependable OEMS that improves HR accuracy and efficiency. Delivering a solid and dependable OEMS that improves the efficacy and precision of HR operations at Glory Academy requires a strict testing procedure.

4.2.2 Unit testing output

Case study: I'm going to test the OEMS to make sure it computes the total salary correctly when an employee receives a bonus. This entails verifying that the bonus amount is appropriately added to the basic salary by the system in order to ensure that the total amount of salary shown is right. The test will confirm that the system can manage salary computations with bonus inputs, guaranteeing accuracy and consistency in the processing of salaries.

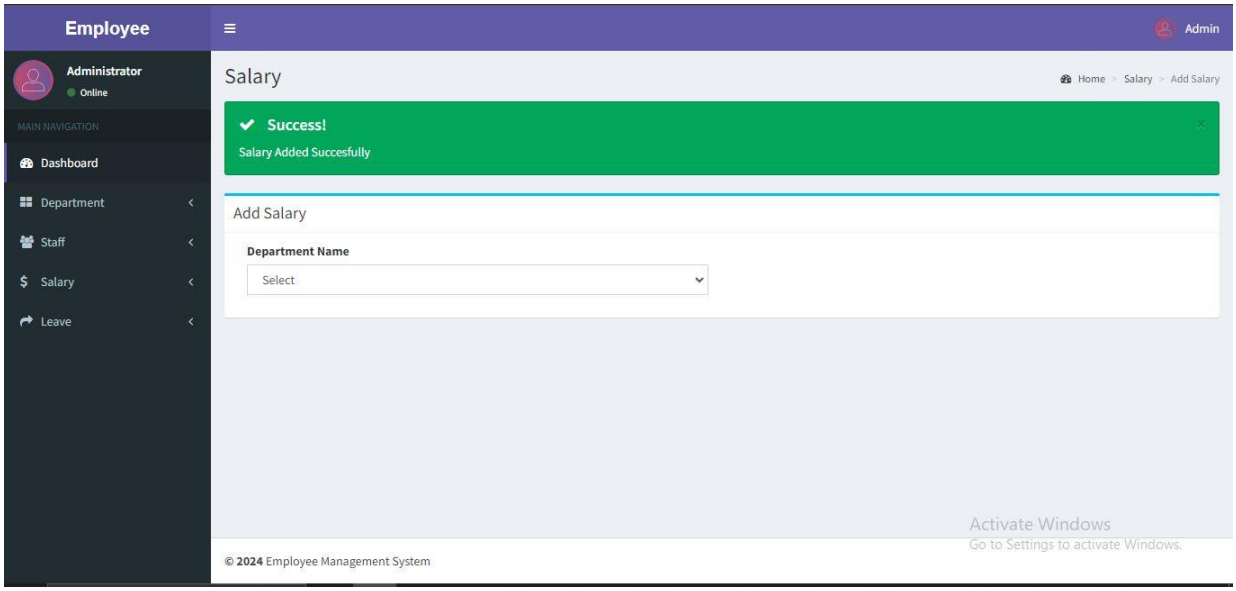

Figure 12 Admin add salary

4.2.3. Validation testing output

The purpose of this test is to confirm that the system prevents inserting blank fields, to be certain that all necessary information is correctly entered before submission.

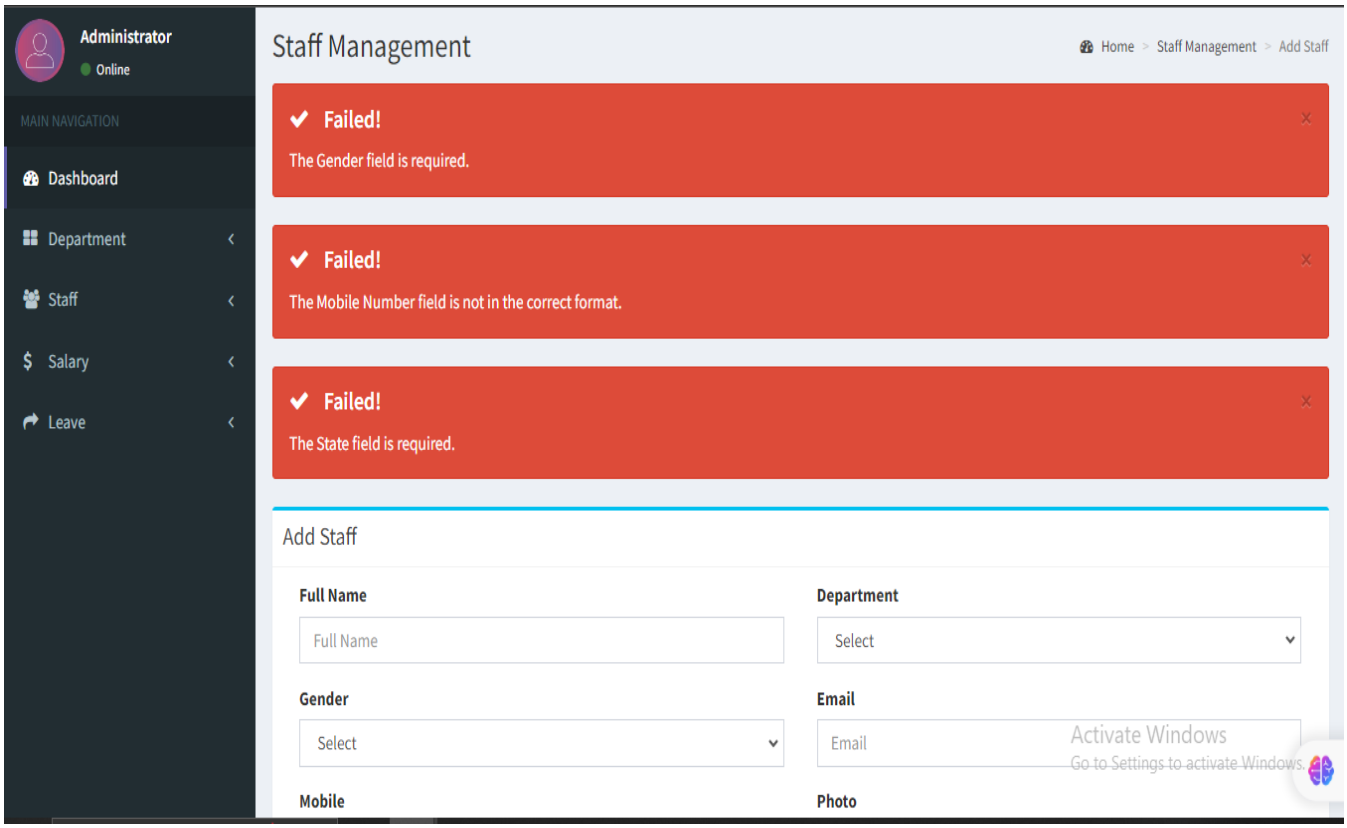

Figure 13 Employee management

4.2.3. Integration

An integration test is performed in order to confirm that various system components work together effectively and efficiently, guaranteeing smooth interaction and appropriate data flow between all components of the system.

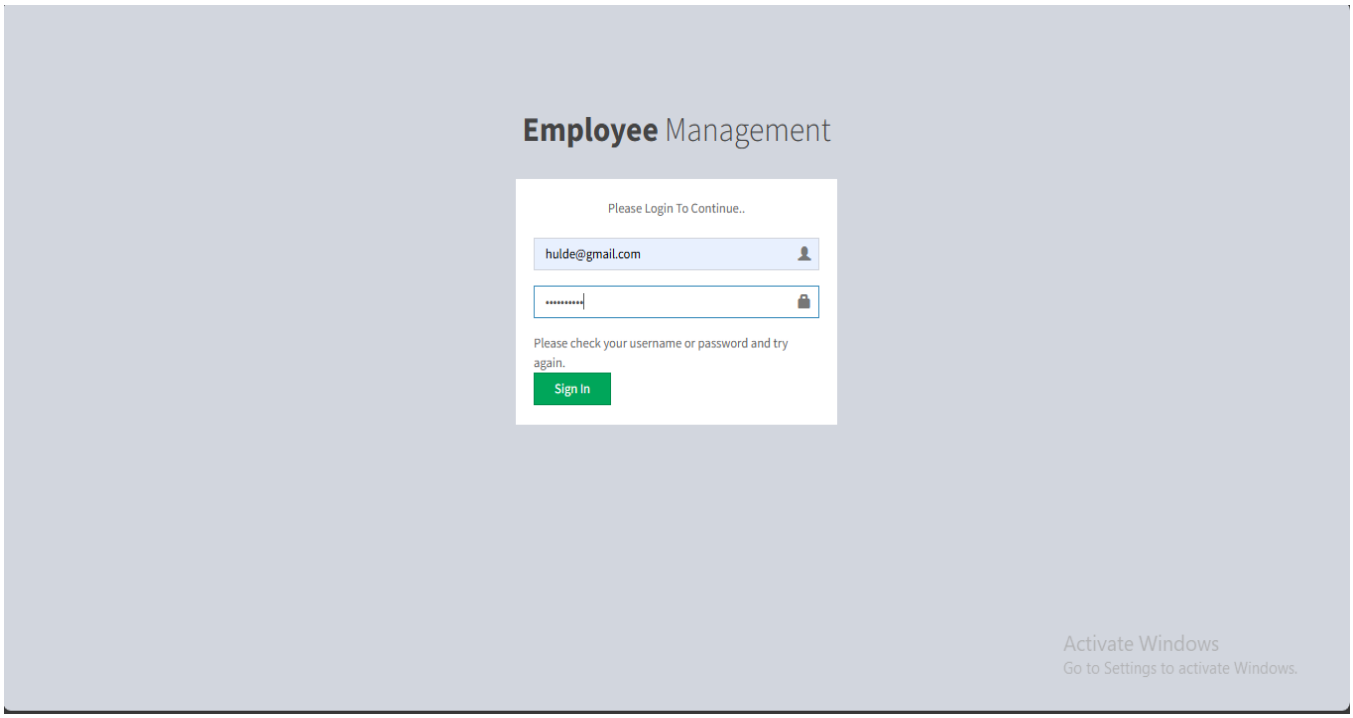

Figure 14 employee login

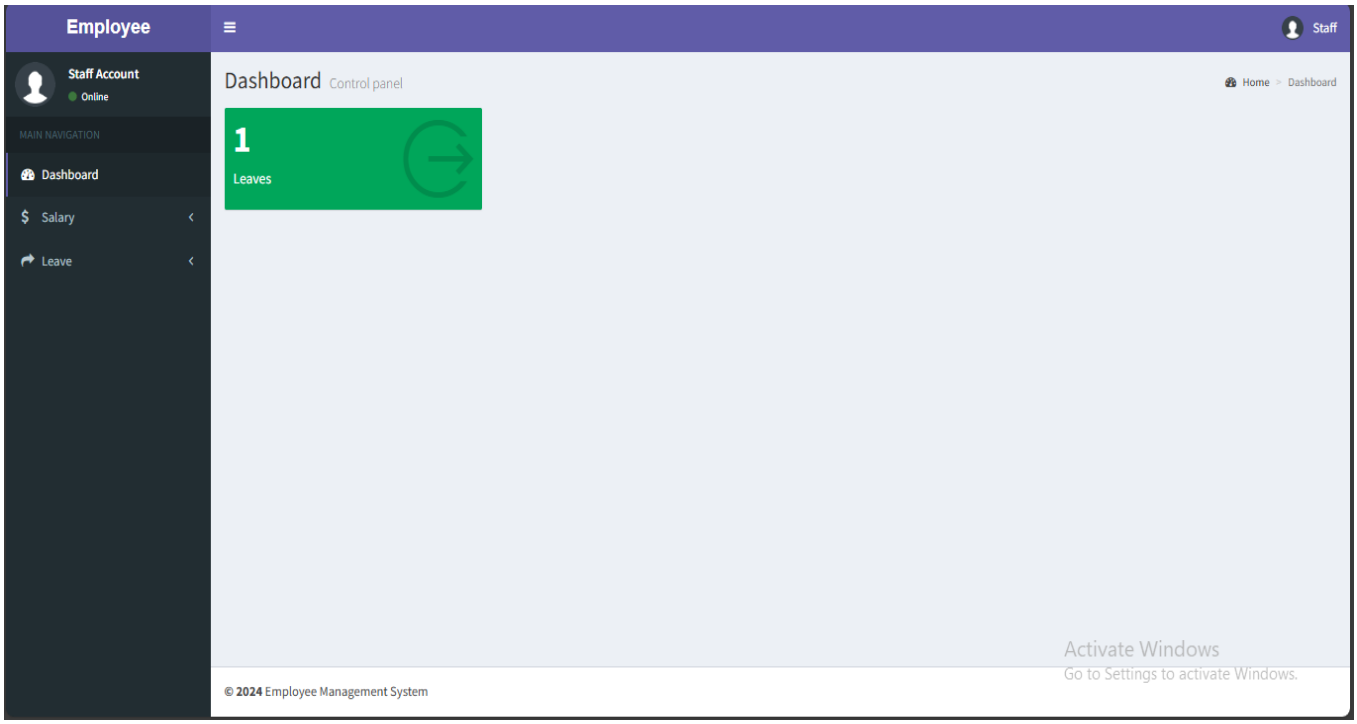

Figure 15 employee dashboard

4.2.4. Functional system testing

The purpose of this test is to confirm that all data entered into the system is successfully stored in the database. It guarantees that all input, including employee salaries, department and leave are correctly saved and kept without errors. The test verifies that the system can manage data insertion effectively while preserving platform-wide data integrity.

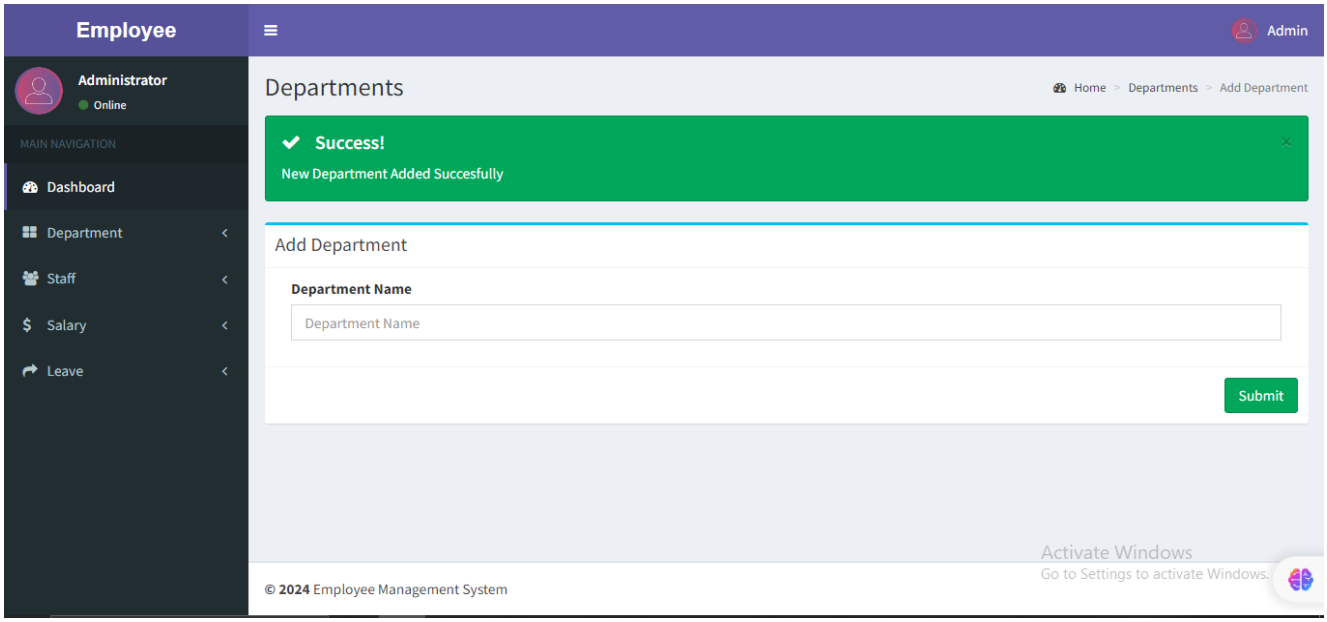

Figure 16 admin add employee

4.2.6 Acceptance testing report

Acceptance testing, involves assessing the system to make sure it satisfies user and business objectives. End users or other stakeholders carry it out to confirm that the system performs as planned in in real-world scenarios. the aim is to verify that the system can perform without errors to confirm deployment. The system is authorized for usage in production if acceptance testing is successful. This testing guarantees the system's usability and functionality.

4.2.1: The purpose of the unit test was to confirm that the system computes the final payment amount correctly and that it correctly determines an employee's total income, including any bonuses that may be awarded.

4.2.2: The validation testing effectively verified that the system forbids the entry of blank fields, guaranteeing that all necessary data is accurately recorded. This feature improves data integrity by mandating that all required fields be filled out before submitting. The system guarantees the

correctness and dependability of employee records by avoiding incomplete entries, which is essential for preserving operational efficiency.

4.2.3: the integration test was successful performed in confirming that the system's various components function efficiently and effectively. The test verified that all of the components interacted without problem guaranteeing accurate data flow and seamless operation of the System.

4.2.4: The functional and system testing successful in verifying that every piece of data entered into the system is stored without error in the database. This validation guarantees the safe storage and retrieval of vital data, including department, salaries, leave and employee records. Data integrity is preserved and the system's dependability is increased by its capacity to manage data entry effectively. The system's ability to handle data insertion efficiently enhances its reliability and ensures data integrity is maintained.

The overall assessment of my system was successful in verifying that every piece of data entered into the system is precisely and error-free recorded in the database. This validation guarantees the safe storage and retrieval of vital data, including employee records and payment records. Data integrity is preserved and the system's dependability is increased by its capacity to manage data entry effectively. The system has proven its efficacy in organizing and preserving crucial personnel data by passing this test.

CONCLUSION AND RECOMMENDATION

Conclusion

The Online Employee Management System (OEMS) designed for Glory Academy has been developed to address key challenges in managing employee records, salaries, and leave requests. The current manual system is prone to errors, time-consuming, and lacks the transparency needed for efficient operations. This OEMS will automate tasks like employee registration, salary management, leave processing, and communication between employees and the HR department. By implementing this system, Glory Academy can streamline human resource processes, leading to better time management and improved data accuracy.

The system is built using the Waterfall development model, which provides a structured and sequential approach, ensuring each phase is completed before the next begins. Additionally, the Structured System Analysis and Design Method (SSADM) has been instrumental in mapping out the detailed requirements and design of the system. Features such as data flow diagrams (DFDs), entity-relationship diagrams (ERDs), and a data dictionary were used to guide the development process.

Overall, the OEMS will significantly improve employee management by providing better access to records, enhancing communication, and offering data-driven insights for management. This system is a step toward more efficient, transparent, and accountable HR management at Glory Academy.

Recommendation: I recommend that this system will be essential in Glory Academy. I also recommend that this system to the ULK's students that will be useful in the future for research purposes.

References

Armstrong, M. (2020). *Armstrong's Handbook of Human Resource Management Practice.* London: Kogan Page.

Bush, T., Bell, L., & Middlewood, D. (2019). *Principles of Educational Leadership and Management.* SAGE.

Connolly, T., & Begg, C. (2020). *Database Systems: A Practical Approach to Design, Implementation, and Management.*

Dessler, G. (2020). *Human Resource Management.* Hoboken: Pearson. Retrieved from cambridge dictionary.

Dymoke, Arabella. (2024). *an a to z of internet terms.* The Good Web Guide Ltd.

Elmasri, R., & Navathe, S. B. (2022, 5 21). *Fundamentals of Database Systems.* Boston: Pearson.

Flanagan, D. . (2022). *JavaScript: The Definitive Guide.* O'Reilly Media.

Garrett, J. J. (2020). *The Elements of User Experience: User-Centered Design for the Web and Application.*

GeeksforGeeks. (2024, August 27). *https://www.geeksforgeeks.org/html-introduction/*. Retrieved from GeeksforGeeks: https://www.geeksforgeeks.org/html-introduction/

Hackerearth. (2024, 5 2). *Hackerearth*. Retrieved from Hackerearth Website: https://www.hackerearth.com/recruit/glossary/common-law-employee/

Hargreaves, A. (2018). *The name of the book is Reports and Report Writing: A Practical Guide.* London: Routledge.

Harrington, J. L. (2024, 11 24). *Harrington's Relational Database Design and Implementation.* San Fransisco: Morgan Kaufmann Publishers.

Kolin, P. C. (2016). *Successful Writing at Work (11th ed.).* Cengage Learning.

Meyer, E. A., & Weyl, R. . (2018). *CSS: The Definitive Guide. .* O'Reilly Media.

Mutumbuka, R., & Ngoga, A. (2017). *Rebuilding the Education System in Post-Conflict Rwanda.* Palgrave Macmillan.

Nielsen, J., & Norman, D. A. (2020). *The Design of Everyday Things.* Basic Books: New York.

Otto, M., & Thornton, J. (2018). *Bootstrap Documentation.* Bootstrap team.

Pohl, K., & Rupp, C. (2018). *Requirements Engineering: Fundamentals, Principles, and Techniques.* Cham: Springer.

Robbins, S. P., & Coulter, M. . (2016). *Management (13th ed.).* Pearson.

Robbins, S. P., & Coulter, M. (2018, 6 10). *Management.* Harlow, England: Pearson.

Shneiderman, B., & Plaisant, C. (2010). *Designing the User Interface: Strategies for Effective Human-Computer Interaction.* Boston: Addison-Wesley.

Shneiderman, B., & Plaisant, C. (2020). *Designing the User Interface: Strategies for Effective HumanComputer Interaction.* Pearson.

Silberschatz, A., Korth, H. F., & Sudarshan, S. (2019, August 7). *Database System Concepts.* New York: McGraw-Hill.

Sommerville, I. (2020). *Software Engineering.* Boston: Pearson.

Stair, R. M., & Reynolds, G. W. (2024). *Principles of Information Systems., Cengage Learning,.* Boston: Cengage Learning.

Van Vliet, H. . (2020). *Software Engineering: Principles and Practice.* Hoboken: Wiley.

Welling, L., & Thomson,. (2020). *L. PHP and MySQL Web Development.* Addison-Wesley.

Appendixes

Source code

manage-salary.php

C: > xampp > htdocs > EMS-CI > application > views > admin > manage-salary.php

```
65  ?>
66      <tr>
67          <td><?php echo $i; ?></td>
68          <td><?php echo $cnt['staff_name']; ?></td>
69          <td><?php echo $cnt['department_name']; ?></td>
70          <td><img src="<?php echo base_url(); ?>uploads/profile-pic/<?php
            echo $cnt['pic'] ?>" class="img-circle" width="50px" alt="User
            Image"></td>
71          <td>$<?php echo $cnt['basic_salary']; ?></td>
72          <td>$<?php echo $cnt['allowance']; ?></td>
73          <td>$<?php echo $cnt['total']; ?></td>
74          <td><?php echo date('Y-m-d', strtotime($cnt['added_on'])); ?></td>
75          <!-- <td><a href="<?php echo base_url(); ?>salary-invoice/<?php
            echo $cnt['id']; ?>" class="btn btn-warning">Invoice</a></td> -->
76          <td>
77              <a href="<?php echo base_url(); ?>delete-salary/<?php echo $cnt
                ['id']; ?>" class="btn btn-danger">Delete</a>
78          </td>
79      </tr>
80  <?php
```


Figure 17 admin manage salary: these are source code for managing salary by either deleting or editing salary.

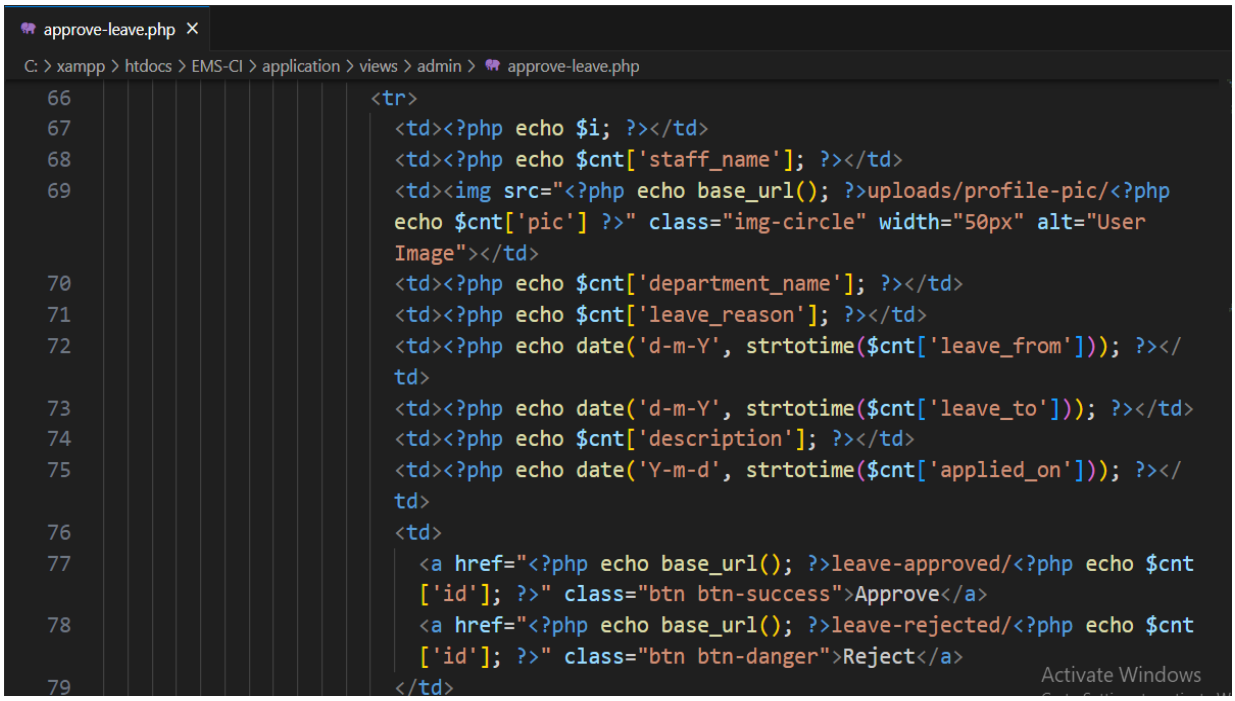

Figure 18 admin manage staff's leave: these are source code for admin approving or rejecting the request to leave

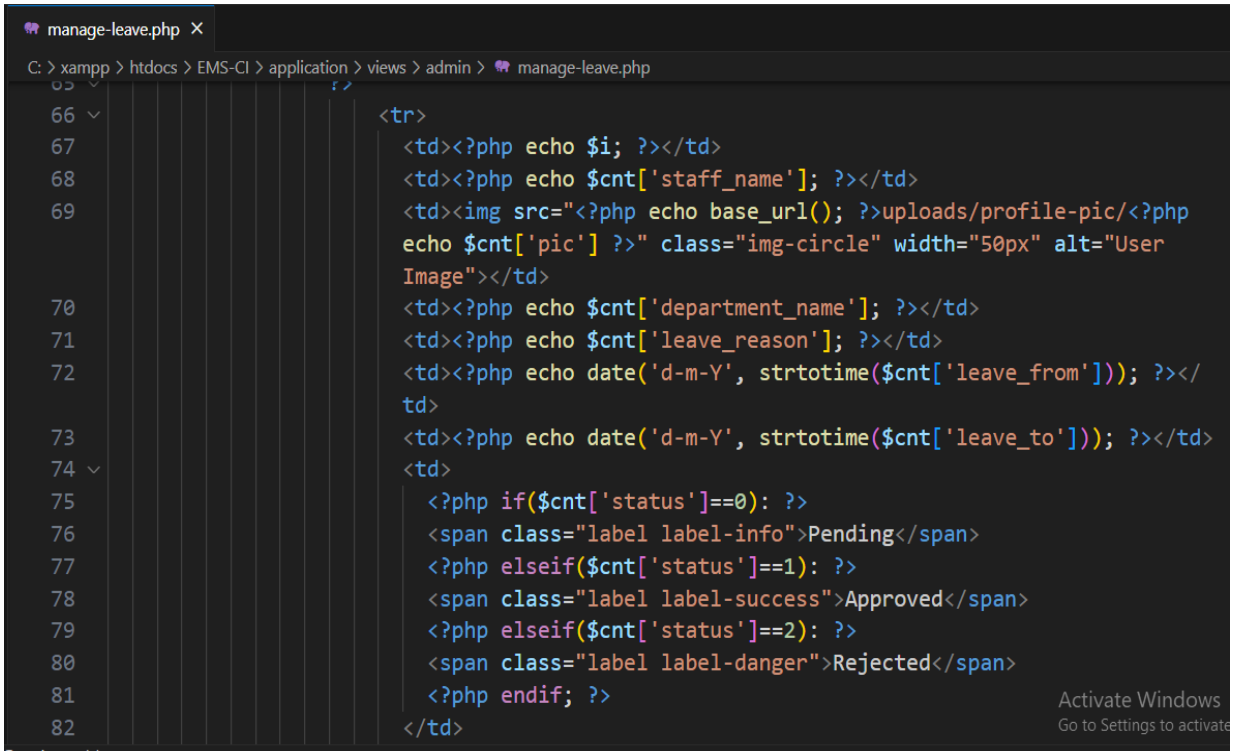

Figure 19 admin manage leave: these are source code for view leave interface and activities.

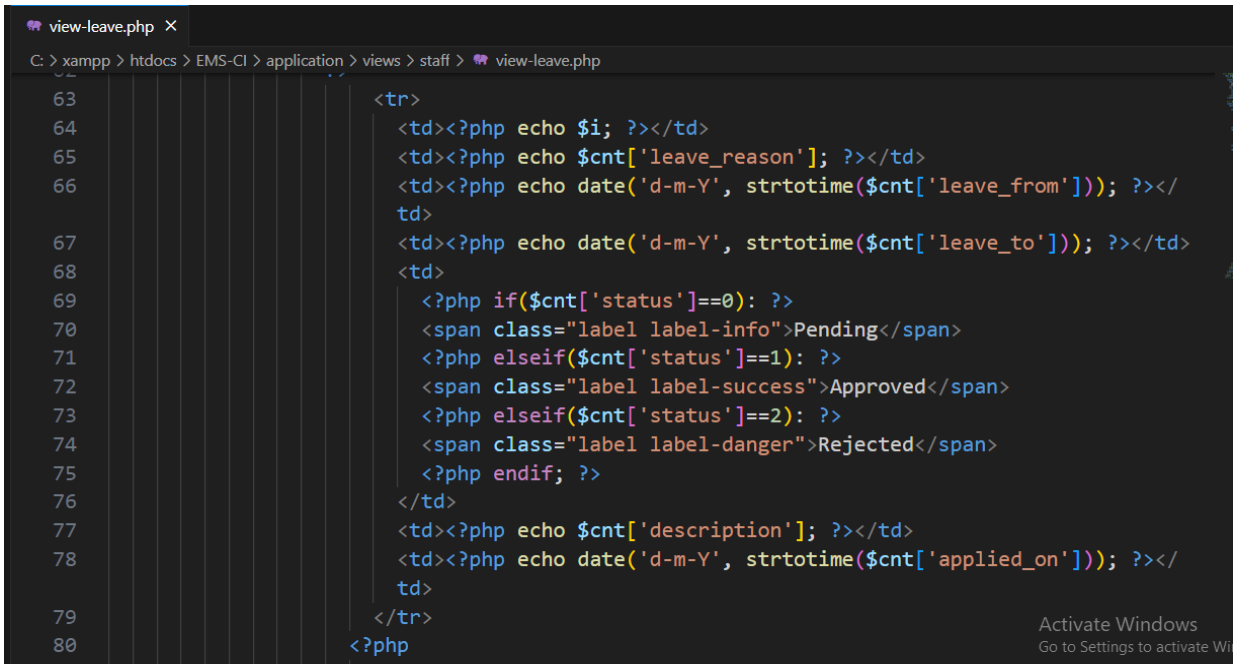

Figure 20 Employee view leave: these are source code for viewing leave history for an employee

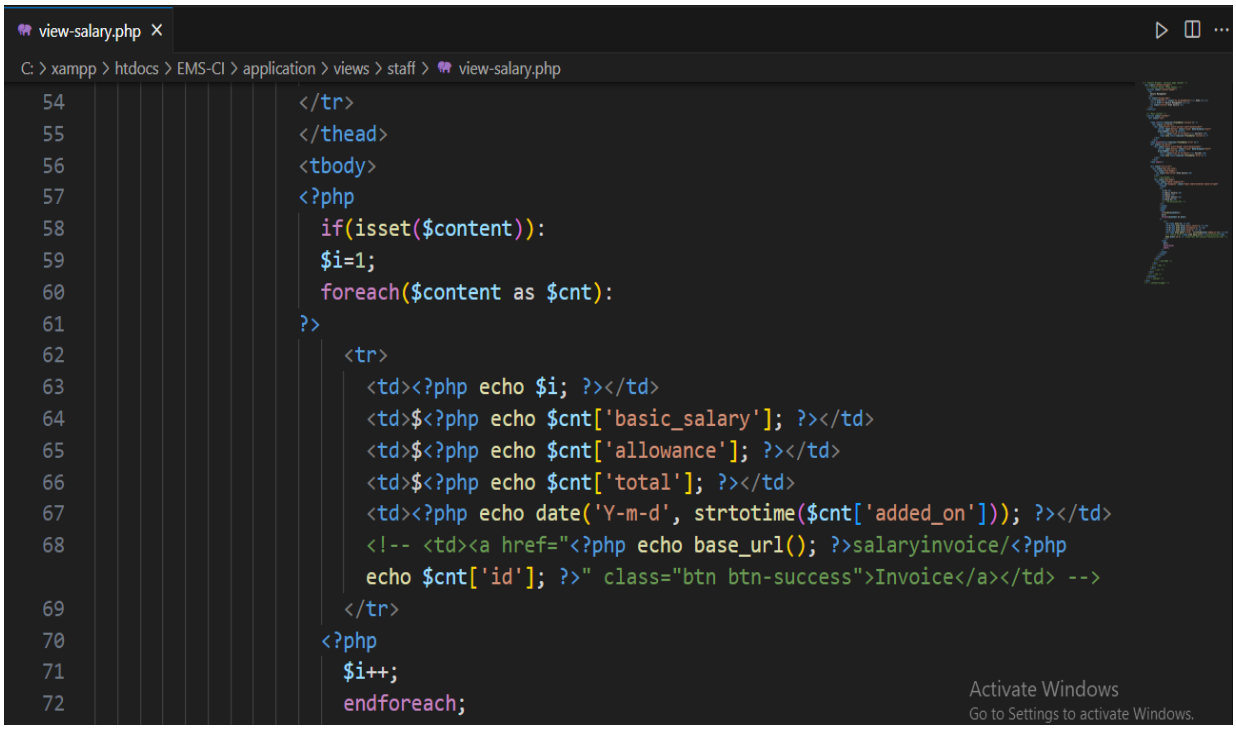

Figure 21 employee view salary: these are source code for view salary for an employee