

**HIMALAYAN WHITEHOUSE INTERNATIONAL COLLEGE**

**PUTALISADAK, KATHMANDU  
(Affiliated to purbanchal university)**

**A**

**Mini**

**Project Proposal**

**On**

**“EMPLOYEE MANAGEMENT SYSTEM”**

**Submitted By:**

**Samip K.C**

**Samir Balami**

**Antima Shrestha**

**Submitted To:**

**THE DEPARTMENT OF IT, COMPUTER AND ELECTRONICS**

**March, 2024**

**Kathmandu, Nepal**

# ABSTRACT

An Employee Management System is a solution designed to streamline and automate various human resources tasks within an organization. Our project will introduce an easy-to-use Employee Management System developed in JAVA. It aims to provide users with secure and efficient way to manage their daily Employee activities. This system will tackle the complexities of HR tasks like keeping track of employee data & modifying data. It will also provide an error message while entering invalid data. The system’s intuitive design and straightforward functionality will enable users to manage their employee data efficiently. The project will entirely be implemented in the JAVA & MYSQL, utilizing variables, strings, and other relevant concepts for the development. With a focus on maintaining data security, the software will incorporate a password-based login system. This will ensure that user information and data remain protected. This application will be reduced as much as possible to avoid the errors while entering the data.

**TABLE OF CONTENT**

[ABSTRACT I](#_Toc160487872)

[LIST OF FIGURES IV](#_Toc160487873)

[LIST OF TABLES V](#_Toc160487874)

[LIST OF ABBREVIATIONS VI](#_Toc160487875)

[CHAPTER 1 1](#_Toc160487876)

[INTRODUCTION 1](#_Toc160487877)

[1.1 Background 1](#_Toc160487878)

[1.2 Problem Statement 1](#_Toc160487879)

[1.3 Objective 2](#_Toc160487880)

[1.4 Scope & limitations 2](#_Toc160487881)

[CHAPTER 2 4](#_Toc160487882)

[LITERATURE REVIEW 4](#_Toc160487883)

[2.1 Study of existing system 4](#_Toc160487884)

[2.2 What’s new in our project? 5](#_Toc160487885)

[CHAPTER 3 6](#_Toc160487886)

[SYSTEM ANALYSIS 6](#_Toc160487887)

[3.1 System Analysis 6](#_Toc160487888)

[3.1.1 Requirement analysis 6](#_Toc160487889)

[3.1.2 Feasibility Analysis 7](#_Toc160487890)

[CHAPTER 4 9](#_Toc160487891)

[SYSTEM DESIGN 9](#_Toc160487892)

[4.1 SDLC model 9](#_Toc160487893)

[4.2 Selected model 10](#_Toc160487894)

[4.3 Context diagram 11](#_Toc160487895)

[4.4 DFD 12](#_Toc160487896)

[4.5 ER-Diagram 13](#_Toc160487897)

[4.6 Use Case Diagram 14](#_Toc160487898)

[CHAPTER 5 15](#_Toc160487899)

[IMPLEMENTATION & TESTING 15](#_Toc160487900)

[5.1 Tools used 15](#_Toc160487901)

[5.2 Gantt chart 15](#_Toc160487902)

[5.3 Testing 16](#_Toc160487903)

[CHAPTER 6 17](#_Toc160487904)

[EXPECTED OUTCOME 17](#_Toc160487905)

[6.1 Final System Expectation 17](#_Toc160487906)

[CHAPTER 7 19](#_Toc160487907)

[CONCLUSION & DISCUSSION 19](#_Toc160487908)

[7.1 Conclusion 19](#_Toc160487909)

[7.2 Future enhancement 19](#_Toc160487910)

[REFERENCES 20](#_Toc160487911)

# LIST OF FIGURES

Figure 1: Waterfall Model………………………………………………………………….10

Figure 2: Context Diagram ………………………………………………………………...11

Figure 3: Data Flow Diagram……………………………………………………………....12

Figure 4: ER Diagram………………………………………………………………………13

Figure 5: Use Case Diagram ……………………………………………………………….14

Figure 6: Gantt chart……….……………………………………………………………….15

# LIST OF TABLES

Table 1: Tools used…………………………………………………………………………15

# LIST OF ABBREVIATIONS

|  |  |
| --- | --- |
| ER | Entity Relationship |
| HR | Human Resource |
| EMS | Employee Management System |
| GUI | Graphical User Interface |
| IDE | Integrated Development Environment |
| MFA | Multi Factor Authentication |
| UAT | User Acceptance Testing |
| DFD | Data Flow Diagram |
| SDLC | System Development Life Cycle |

# CHAPTER 1

# INTRODUCTION

## **Background**

In today’s fast-paced and competitive business environment, effective management of human resources is essential for organizational success. Traditional manual methods of managing employee data, payroll, and performance evaluation are not only time-consuming but also prone to errors and inefficiencies. To address these challenges and streamline HR operations, the implementation of an Employee Management System (EMS) is crucial.

The conventional management system was labour-intensive and prone to mistakes since it relied on paper-based updating and human record-keeping system. However, the organizations has undergone a revolution since the development of computer technology. To conduct their management operations, organizations now rely on complex software programs, such as core employee management systems that combine multiple management services like data addition, deleting data, modification and many more.

The need for an EMS arises from the growing complexity of HR tasks, especially in organizations with a large workforce or multiple locations. Manual processes for tracking employee data and managing HR-related tasks are not scalable and often result in inaccuracies, delays, and frustration among employees and HR personnel alike. Moreover, with the advent of remote work and flexible schedules, the demand for digital solutions to manage employee-related tasks has become even more pronounced.

This project report's goal is to plan and create a thorough employee management system that will increase the efficiency of employee operations. By developing an EMS, organization can achieve several key objectives like efficiency, accuracy, transparency, compliance and many more. Overall, an EMS is a strategic investment for organizations seeking to optimize their HR processes, enhance employee satisfaction, and stay competitive in today's dynamic business landscape.

## **Problem Statement**

In today's dynamic business landscape, many organizations face significant challenges in efficiently managing their human resources. Manual processes for tracking employee data, processing payroll, and conducting performance evaluations are time-consuming, error-prone, and lack scalability. Moreover, the increasing trend towards remote work and flexible schedules exacerbates these challenges, necessitating a digital solution that can adapt to evolving workplace dynamics.

Existing HR systems often lack integration, user-friendliness, and compliance with regulatory requirements, leading to frustration among employees and HR personnel alike. There is a pressing need for an Employee Management System (EMS) that streamlines HR processes, enhances data accuracy, promotes transparency, and empowers employees to manage their own HR-related tasks. This system must be robust, secure, and customizable to meet the diverse needs of different organizations while ensuring compliance with relevant laws and regulations.

## **Objective**

The main motto of our project is to develop an EMS & to be familiar with the features of JAVA & database. The project will be mainly based on the following objectives:

**General Objective for Employee Management System:**

* To develop a user-friendly system for efficient management of employee data.
* To provide secure and safe environment to store and manage sensitive information.

**Specific Objectives for Employee Management System:**

* To implement a centralized database for storing employee information, ensuring data accuracy and accessibility.
* To provide functionality for authorized administrators to securely add, delete, modify, search & view employees records.

## **Scope & limitations**

Our system will be designed to minimize the manual work in maintaining staff details. It will aim to maximize the productivity and provide improved managed system. Some of the scope and limitations of our proposed system is listed below:

**Scope**

* The system will provide an efficient and effective method for managing employee data, including name, phone number, email address and any other relevant information.
* Implementation of authorization controls to manage access to sensitive employee information.
* Addition, modification, searching, deletion & viewing of records by authorized administrators.

**Limitations**

* The final system may contain security vulnerabilities despite authorization controls.
* The system may require some training for users to fully utilize its feature and functions.

# CHAPTER 2

# LITERATURE REVIEW

## **Study of existing system**

Earlier systems were manual where there was no way of properly storing information. Employee records were stored manually which lead to errors. There was no proper way of tracking employee records. It was very difficult and required a lot of paperwork which makes the application time consuming and not secured. There was no administrator which could handle the records. So there was the need to develop a system which could manage all these things and reduce the paperwork.

The project done by Angel Jude Suarez on Employee Management System based on Java utilizes GUI technology with SQLite as the database. The features modules like employee manager, allowance, deduction and payment providing administrative control over employee data was included in the report which enhanced record management efficiency within the application. [1]

A web based Employee Management System proposed by Uditha and the Team considers employees attendance, leaves, announcement and salary related details. They proposed a web based Employee Management System surpassing traditional manual methods and existing computerized systems. [2]

The paper reported by Geng, Xuemin, Tingmei Wang, and Qiuhui Li on Employee Management Platform Based on Java Web highlights the significance of employee efficiency in the face of intense social competition, advocating for the development of tailored employee management platforms. In their project they utilized Java web and MySQL to accurately assign employee roles, facilities attendance tracking tasks and project allocations. [3]

The other techniques that are in the market are dependent on facial recognition, biometric scan or card punching. But all of these require an external device to be installed in the working area, which is again a costly process and requires regular maintenance. Overall, the literature on employee management systems using highlights the versatility and effectiveness of Java-based solutions in addressing the diverse needs of organizations in managing their workforce efficiently.

## **What’s new in our project?**

Our Java mini-project on Employee Management System, introduces essential features tailored for educational purposes. While drawing inspiration from existing systems, our project emphasizes simplicity and accessibility. Unique aspects include user-friendly interface design with intuitive navigation, basic employee data management functionalities such as adding, searching, editing, and deleting employee records. Our project also offers learning opportunities for students to understand fundamental concepts of Java programming, database management, and user interface design. It serves as an educational tool for students to grasp the basics of software development while simulating real-world scenarios in an organizational context.

# CHAPTER 3

# SYSTEM ANALYSIS

## **System Analysis**

System analysis is the process of studying a system or a specific aspect of a system to understand its components, functions, interactions and requirements. There are various types of system analysis such as requirement analysis, feasibility analysis, system design analysis, risk analysis and others.

## **Requirement analysis**

Requirement analysis is a critical phase in system development that involves identifying, documenting, analysing and prioritizing the needs and expectations. This process typically includes gathering both functional and non-functional requirements.

1. **Functional requirement**

It described the specific functions that the system must perform.

* User authentication: the system should allow users to securely log in using their username and password.
* Employee information management: the system should provide functionality for adding, updating, editing and deleting employee records.

1. **Non-Functional Requirements**

Non-functional requirements specify the quality attributes or constraints that the system must adhere to, such as performance, reliability, usability and security. These requirements define how the system should behave or perform rather that what it should do.

* Performance: Performance refers to how well the system performs in terms of speed, responsiveness, and efficiency. It includes factors like response time, throughput, and resource utilization.
* Reliability: The system should be available whenever needed, minimizing downtime. It should ensure the integrity and consistency of employee data, preventing data loss or corruption.
* Usability: Usability focuses on the ease of use and user experience of the system. It includes factors like user interface design, navigation, accessibility, and user support features. A usable system provides intuitive interfaces, clear instructions, and efficient workflows to enhance user satisfaction and productivity.
* Security: Security encompasses measures to protect the system from unauthorized access, data breaches, and malicious attacks. It includes features like authentication, authorization, encryption, and secure communication protocols.

## **Feasibility Analysis**

A feasibility study is a systematic analysis to determine the practicality and potential success of a proposed project or venture.

1. **Technical Feasibility:** Technical feasibility is concerned with the availability of hardware and software required for the development of the system. After the study we came to conclusion that we can proceed further with the tools and development environment chosen by us.
2. **Operational Feasibility:** Operational feasibility is all about problems that may arise during operations. There are two aspects related with the issue. What is the probability that the solution developed may not be put to use or may not work? What is the inclination of the management and end users towards the solution? Though, there is very least possibility of management being averse to the solution, there is significant probability that the end users may not be interested in using the solution due to lack of training, insight, etc.
3. **Time Feasibility:** Time feasibility refers to the assessment of whether a proposed project can be completed within a reasonable timeframe. After the study of the tasks involved in completing the project including requirement gathering, frontend development, backend development, coding, testing and others, we have concluded that the project timeline appears feasible. By carefully analysing tasks, resource availability & potential risks or delays, we can assess the feasibility of meeting the projects deadlines and make adjustments to the schedule as needed to ensure timely completion.
4. **Cost Feasibility:** The cost feasibility analysis for the employee management system involves evaluating expenses related to development, maintenance, and operational aspects. This includes estimating costs for software development, infrastructure, ongoing maintenance, integration, and any other associated expenses. After the study, it was determined that the employee management system implemented in Java is cost-feasible, with expenses well within the allocated budget.
5. **Legal Feasibility:** Legal feasibility entails ensuring compliance with relevant laws and regulations governing data privacy, intellectual property, and employment practices. Specifically, for the employee management system, it involves adherence to data protection laws ensuring proper handling and security of employee data. The system was found to be legally feasible since only the authorized admin is allowed to use the system which will further enhance data privacy.

# CHAPTER 4

# SYSTEM DESIGN

## **SDLC model**

The Software Development Life Cycle (SDLC) model is a structured approach used by software development teams to design, develop, test, deploy, and maintain software systems. It encompasses a series of phases or stages, each with specific activities and deliverables. Common SDLC models include Waterfall, prototype & spiral model with its own set of principles, methodologies, and best practices tailored to different project requirements and organizational needs.

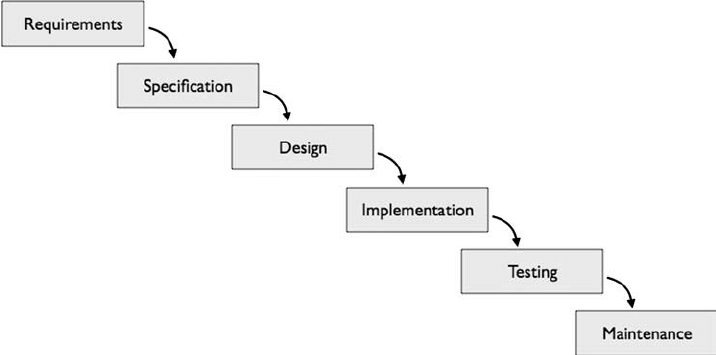
The phases in the Software Development Life Cycle (SDLC) typically include:

* 1. **Planning**: This phase involves defining the project scope, objectives, timelines, and resources required. It may also include feasibility studies and risk assessments to ensure the project's viability.
  2. **Requirement Analysis**: During this phase, the development team gathers and analyses requirements from stakeholders, such as users, customers, and business owners. The goal is to understand the needs and expectations of the software system to be developed.
  3. **Design**: In this phase, the system architecture and design specifications are created based on the requirements gathered. This includes defining the software components, data structures, interfaces, and algorithms to be used in the system.
  4. **Implementation**: Also known as coding or development, this phase involves translating the design specifications into actual code. Developers write, compile, and test the code to ensure it meets the requirements and design standards.
  5. **Testing**: Once the code is developed, it undergoes rigorous testing to identify and fix defects or bugs. Testing includes various techniques such as unit testing, integration testing, system testing, and UAT.
  6. **Deployment**: After successful testing, the software is deployed or released to the production environment. This phase involves installing the software on users' machines or servers and configuring it for use.
  7. **Maintenance**: The final phase involves maintaining and supporting the software post-deployment. This includes addressing any issues or bugs reported by users, making enhancements or updates as needed, and ensuring the software remains functional and efficient over time.

These phases may vary slightly depending on the specific SDLC model or methodology being used, but they generally encompass the key activities involved in software development projects.

## **Selected model**

Our project is aimed to develop Employee Management System using java programming language. After going through all the SDLC models we concluded on using waterfall model. The Waterfall Model is a linear and sequential approach to software development, where each phase must be completed before moving on to the next. It typically consists of distinct phases such as requirements gathering, design, implementation, testing, deployment, and maintenance. Once a phase is completed, the process moves to the next phase in a cascading manner, with minimal opportunity for revisiting earlier stages. It's characterized by its rigidity and emphasis on extensive upfront planning and documentation.

***Figure 1: Waterfall Model***

Since the requirements & objective of our project is clear to us, we selected waterfall model because it suits the best for completion of our mini project. Following the waterfall model, the project will be progressed through sequential phases including requirement gathering, designing, implementation, testing, deployment and maintenance. Extensive documentation will be maintained at each stage to ensure clarity and facilitate future maintenance. Clarity in requirements, mini project, fixed scope and objective are some of the reason behind the selection of this model.

## **Context diagram// Algorithm**

A context diagram is a high-level visual representation that illustrates the scope and boundaries of a system or process within its environment. It provides an overview of the interactions between the system being analysed and its external entities, such as users, other systems, or external stakeholders. Context diagrams help stakeholders understand the context in which the system operates and facilitate discussions about its requirements, interfaces, and dependencies.

**Log in**

**Company Admin**

**View data**

**Perform various operation**

***Figure 2: Context Diagram***

## **DFD// Flowchart**

DFD stands for Data Flow Diagram. It's a graphical representation that illustrates how data flows through a system or process. DFDs consist of processes, data stores, data flows, and external entities. Processes represent activities or transformations that occur within the system, data stores depict where data is stored, data flows show the movement of data between processes and data stores, and external entities represent sources or destinations of data outside the system. The level 1 DFD of our proposed system is as shown below:

***Figure 3: Data Flow Diagram***

**Company Admin**

**Add data**

**Delete data**

**Modify data**

**Search data**

**View data**

**Verified**

## **ER-Diagram**

An ER diagram is a visual representation that depicts the relationships among entities within a database. ER diagrams help in understanding the structure of a database and are commonly used during the database design phase to model the relationships between different entities and their attributes. The ER diagram of our purposed system which will be further modified accourding to our requirements

**Admin**

**Password**

**Manages**

**Username**

**Employee Info**

**Has**

**Department**

**Document**

**Deleted record**

**File location**

***Figure 5: ER Diagram***

## **Use Case Diagram**

A use case diagram is a type of behavioural diagram in Unified Modelling Language (UML) that illustrates the interactions between actors (users or external systems) and a system to accomplish specific goals or tasks. It shows the functionality of a system from the perspective of its users and helps to understand how users interact with the system.

**Admin**

***Figure 5: Use Case Diagram***

# CHAPTER 5

# IMPLEMENTATION & TESTING

## **Tools used**

**Table 1: Tools used**

|  |  |
| --- | --- |
| Apache Netbeans IDE 20 | IDE used for GUI design, coding & debugging, and also supports multiple programming languages. |
| XAMM | Employed as the local development environment for managing the Apache web server & MYSQL database. |
| SQL YOG | GUI tool for managing MYSQL database and relational database |

## **Gantt chart**

****We have outlined the timeline for the implementation of the EMS below using a Gantt chart. This chart illustrates the major tasks, their dependencies and the estimated duration for each task.

***Figure 6: Gantt chart***

## **Testing**

Software testing is a process of analysing an application's functionality as per the requirements. If we want to ensure that our software is bug-free or stable, we must perform the various types of software testing because testing is the only method that makes our application bug free. Here’s a brief overview of various types of software testing:

1. **Functional Testing:** In functional testing, all the components are tested by giving the value, defining the output, and validating the actual output with the expected value. Functional testing. In functional testing, all the components are tested by giving the value, defining the output, and validating the actual output with the expected value.
2. Unit Testing: Unit testing is the first level of functional testing in order to test any software. In this, the test engineer will test the module of an application independently or test all the module functionality is called unit testing.
3. Integration Testing: Once we are successfully implementing the unit testing, we will go integration testing. It is the second level of functional testing, where we test the data flow between dependent modules or interface between two features is called integration testing.
4. System Testing: Whenever we are done with the unit and integration testing, we can proceed with the system testing. In system testing, the test environment is parallel to the production environment. In this type of testing, we will undergo each attribute of the software and test if the end feature works according to the business requirement. And analysis the software product as a complete system.
5. **Non-Functional Testing:** Non-functional testing is a type of software testing that focuses on the attributes of a system that do not relate to specific behaviour’s or functions. Instead, it assesses qualities such as performance, reliability, scalability, usability, security, and compatibility. Non-functional testing helps evaluate how well a system meets its requirements in terms of these attributes and ensures that it performs satisfactorily under various conditions beyond functional correctness.

For the completion of our mini project we will be using functional testing method as it ensures that software meets specified requirements by testing individual functions or features, detecting bugs early, and ensuring user satisfaction and overall quality.

# CHAPTER 6

# EXPECTED OUTCOME

## **Final System Expectation**

By the end of this project we are expected to create an Employee Management System (EMS) with features such as add, delete, modify, view and search implemented in JAVA. The proposed system is expected to have following outcomes:

1. **Login interface: -** The proposed system will present a login interface where the admin enters their username and password.
2. **Authentication: -** Upon submission, the system verifies the entered credentials against stored admin credentials in the database.
3. **Dashboard: -** Upon successful authentication, the admin is redirected to a dashboard displaying options for various actions.
4. **Employee management options: -** The dashboard provides options like “Add Employee”, “Delete Employee”, “Modify Employee”, “View Employee” and “Search Employee”.
5. **Add Employee: -** Clicking on “Add Employee” will open a form where the admin can input details such as employee name, ID, department, designation, etc. After submitting the data will be saved in the database.
6. **Delete Employee: -** Selecting “Delete Employee” prompts the admin to enter the employee ID from a list of employees to delete. Upon confirmation, the employee record is removed from the database.
7. **Modify Employee: -** Choosing “Modify Employee” allows the admin to search for an employee by ID and then edit their details. Changes are saved to the database upon confirmation.
8. **View Employee Data: -** Clicking on “View Employee Data” displays a list of all employees along with their details.
9. **Search Employee: -** The “Search Employee” option enables the admin to search for employees based on criteria such as name, ID, department, etc. the system then displays the matching results.

The proposed system will ensure secure login, efficient management of employee data and easy access to functionalities for the admin. Additionally, it maintains data integrity and provides necessary features for managing employee records effectively.

# CHAPTER 7

# CONCLUSION & DISCUSSION

## **Conclusion**

After conducting an in-depth analysis of the project’s outcomes and requirement, the implementation an Employee Management System (EMS) would indicate significant improvements in HR processes, data integrity, security, user satisfaction, compliance, and cost-effectiveness. Efficiency gains are evident through streamlined tasks such as adding, modifying, and searching for employee records, reducing manual effort and administrative overhead. Enhanced data integrity is observed with fewer errors and inconsistencies in employee data. The implementation of user verification features strengthens security measures, ensuring only authorized access to sensitive information.

## **Future enhancement**

Looking into the future, here are some forward-thinking recommendations for employee management system:

1. **Biometric authentication:** Integrate biometric authentication methods, such as fingerprint scanning or facial recognition, to enhance the security of employee access to sensitive data and systems.
2. **Attendance tracking:** Implement a system to record employee’s attendance either through manual entry or automated methods like biometrics scanners.
3. **Leave management:** Develop a module for employee to request leaves, including vacation, sick leave or other types of absence.

# REFERENCES

[1] A. J. Suarez, “Employee Management System Project in Java with Source Code,” *Itsourcecode.com*, Oct. 17, 2023. <https://itsourcecode.com/free-projects/java-projects/employee-management-system-project-in-java-with-source-code/> [Accessed: 2nd march]

[2] Sanuji Nanayakkara, Uditha Ekanayake, Gayesha Subasinghe, Chamuditha Jayasena, D. I. De Silva, and Dulanji Cooray, “A Web Based Employee Management System,” *International Journal of Engineering and Management Research*, vol. 12, no. 5, pp. 82–89, Oct. 2022. <https://www.academia.edu/96012019/A_Web_Based_Employee_Management_System?hb-sb-sw=96799189> [Accessed: 2nd march]

[3] Geng, Xuemin, Tingmei Wang, and Qiuhui Li. "Design and Implementation of Employee Management Platform Based on Java Web." Journal of simulation, vol. 10, NO. 2, Apr. 2022. <http://www.journalofsimulation.com/d/file/previous/2022%20Volume%201/Vol%2010,%20No%202%20(2022)/2022-04-27/4f532e176364ce0a6a7c9e5cbf1352fa.pdf> [Accessed: 2nd march]