|  |
| --- |
| EMPLOYEE MANAGEMENT SYSTEM |
| **BY - TEAM TED** |
|  |
| **(Astitva Vimal, Vishnu Priya, Nilesh Bind, Padma Sree, Prudhvi)** |
|  |
|  |
| **11/21/2021** |
|  |

**ABSTRACT**

Employees are the backbone of any company therefore their management plays a major role in deciding the success of an organization. Employees Management Software makes it easy for the employer to keep track of all records. This software allows the administrator to edit employees, add new employees, and terminate employees. Each employee in the database is associated with a position can be added and edited when need arises

A flexible and easy to use Employee Management software solution for small and medium sized companies provides modules for personnel information management thereby organization and companies are able to manage the crucial organization asset – people. This system brings about an easy way of maintaining the details of employees working in any organization.

It is simple to understand and can be used by anyone who is not even familiar with simple employees system. It is user friendly and just asks the user to follow step by step operations by giving easy to follow options. It is fast and can perform many operations for a company.

The goal of this project is to design and develop an employee management system to fill existing gaps in the electronic management of employees

**1 CHAPTER ONE: INTRODUCTION TO THE RESEARCH**

**1.1 PROBLEM STATEMENT**

Manual handling of employee information poses a number of challenges. The use of paper work in handling some of these processes could lead to human error, papers may end up in the wrong hands and not forgetting the fact that this is time consuming. A number of current systems lack employee self-service meaning employees are not able to access and manage their personal information directly without having to go through their HR departments or their managers. Another challenge is that multi-national companies will have all the employee information stored at the headquarters of the company making it difficult to access the employee information from remote places when needed at short notice.

The aforementioned problems can be tackled by designing and implementing an Employee management system. This system will maintain employee information in a database by fully privacy and authority access. The project is aimed at setting up employee information system about the status of the employee, the educational background and the work experience in order to help monitor the performance and achievements of the employee through a password protected system.

**1.2 PROJECT BACKGROUND**

Employees are the backbone of any company therefore their management plays a major role in deciding the success of an organization. Employee Management Software makes it easy for the employer to keep track of all records. This software allows the administrator to edit employees, add new employees and remove an employee from the database. Employees can be managed efficiently without having to retype back their information in the database.

A flexible and easy to use Employee Management software solution for small and medium sized companies provides modules for personnel information management thereby organization and companies are able to manage the crucial organization asset – people.

This system brings about an easy way of maintaining the details of employees working in any organization.

It is simple to understand and can be used by anyone who is not even familiar with simple employees system. It is user friendly and just asks the user to follow step by step operations by giving easy to follow options. It is fast and can perform many operations for a company.

**1.3 OBJECTIVES**

In this world of growing technologies everything has been computerized. With large number of work opportunities the Human workforce has increased. Thus there is a need of a system which can handle the data of such a large number of Employees. This project simplifies the task of maintaining records because of its user friendly nature.

The objective of this project is to provide a comprehensive approach towards the management of employee information. This will be done by designing and implementing an Employee management system that will bring up a major paradigm shift in the way that employee information is handled.

The objectives of this system include:

* Design of a Employee management system to fulfill requirements of an administrator and an employee.
* Well-designed database to store employee information allowing the administrator to edit employees, add new employees, and terminate employees.

**1.4 SCOPE**

The scope of this project will be limited to the following:

* Employee profiles:
* Employees will have access to their personal profiles.
* Admin will have access to all the employee details
* Admin will be able to perform operations like adding an employee, updating the details of an employee, deleting an employee details.

**1.5 REQUIREMENTS AND CONSTRAINTS**

**1.5.1 Functional Requirements:**

**Authentication**

* Login- The user can login to the Employee system with his/her username and password.
* Logout- The user can log out from the Employee system.

Login failure- If the user does not exist in the database or the user has not yet being authorized by the admin.

**Authorization**

* User role check- After logging in, the user role will be checked from the database and the user interface will be displayed according to their role.

**Process Data**

* Display- User with defined roles can display the content of the database. Being more specific, employee can only view his/her personal information. Admin can display the personal information and all employees’ information and modify them.
* Edit- Admin can edit all information related to all employees’ including their user role type.

**Recruitment**

* Add new employee- Admin role type is able to add a new employee to the database. The new employee will have all the required information related to him/her. The new created employee will have an id.
* Add a new user- After a new employee has being created, admin role is responsible for creating a new user by the specified id assigned in the “Add a new employee” feature. The unique id will be given by the system

**1.5.2 Non-Functional Requirements:**

**Performance requirements**

There is no restriction on the number of the users to be added to the database.

**Hardware requirements**

EMS should be able to work on a computer with the following minimum hardware specifications:

OS: Windows XP/Vista/7/8 and Linux

CPU: Pentium III (700MHz) and above

Memory: 128 MB and above

Capacity: 4GB of hard drive

Others: Network interface card, mouse, keyboard, and monitor.

**Software requirements**

The EMS software personal database model will support MySQL environment as DBMS.

**1.6 SUMMARY**

This chapter began by giving a brief overview of the entire project, the background of the project and the scope. The problem definition and solution highlights the current problems faced with the use of the systems that are in place and outlines briefly the solution system to be developed. The next chapter will focus on the literature review. This is literature that relates to the project and similar systems.

**2 CHAPTER TWO –RESEARCH**

**2.1 INTRODUCTION**

This chapter summarizes the evaluation of the literature relevant to the Employee Management System. It examines theories, concepts, approaches, methods and techniques relevant to the project. Similar existing technologies relating to the development the EMS are discussed.

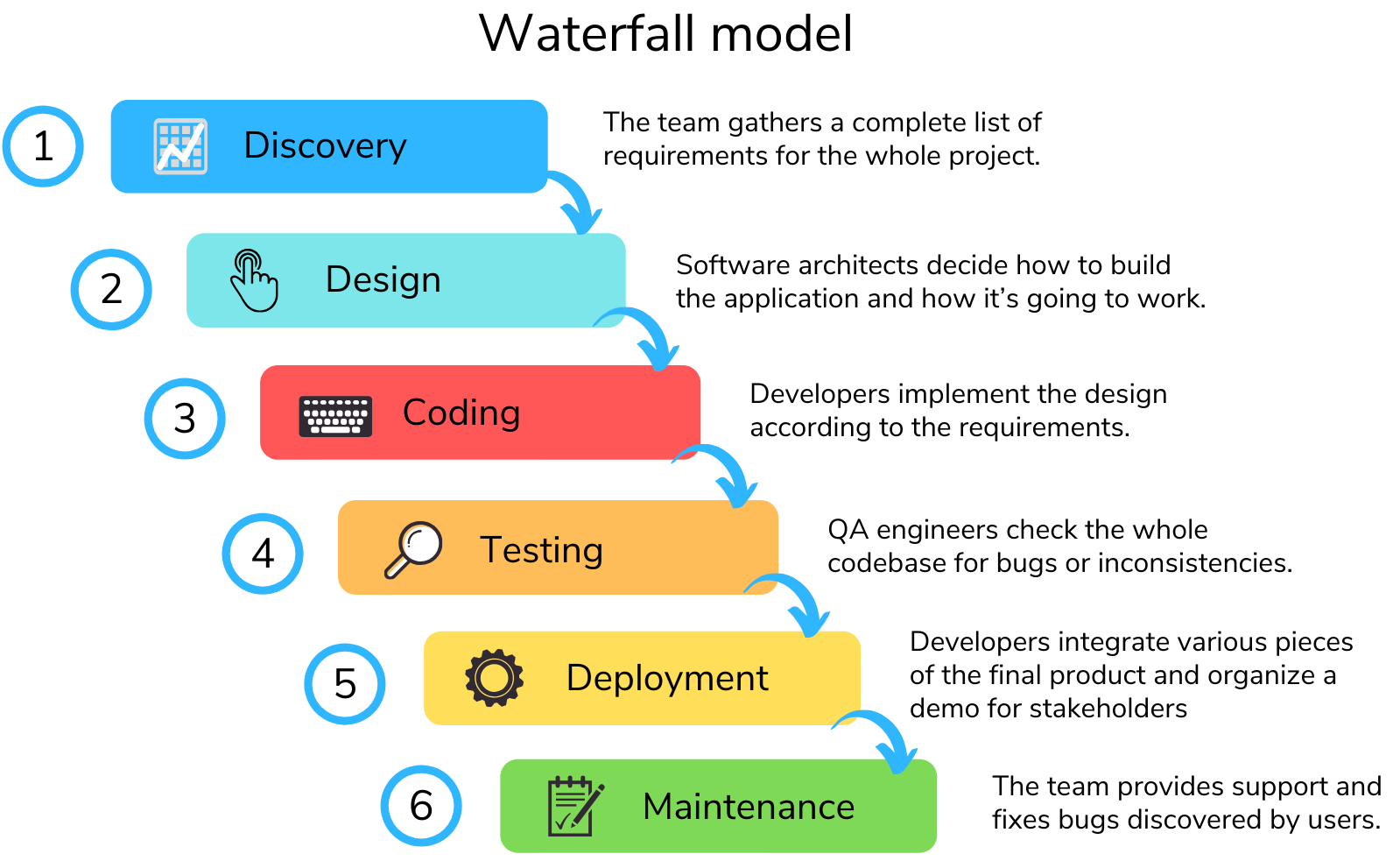
**2.2 REVIEWS ON SEVERAL SOFTWARE METHODOLOGIES**

A software development methodology is a collection of procedures, techniques, tools, and documentation aids which will help the systems developers in their efforts to implement a new information system.

There are a number of software development methodology each of which are adopted based on a number of factors relating to the project e.g. Time, cost, incorporation of requirement changes during the development process, system complexity, communication between customers and developers, software criticality, size of the development team. These generic models are not definitive descriptions of software processes. Rather, they are abstractions of the process that can be used to explain different approaches to software development. You can think of them as process frameworks that may be extended and adapted to create more specific software engineering processes. Below is the selected model:

**The Waterfall Model**

The waterfall model is a sequential design process, often used in software development processes. It takes the fundamental process activities of specification, development, validation, and evolution and represents them as separate process phases such as requirements specification, software design, implementation, testing, and so on.



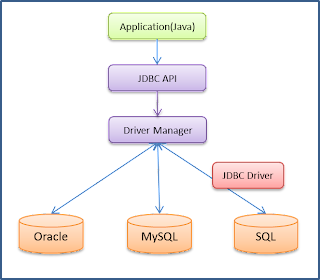
**2.4 REVIEW OF POSSIBLE DEVELOPMENT TOOLS AND SOFTWARE TO BE USED**

The following are various development tools and software that could be used for the system.

**2.4.1 Back-end Technology**

**Java**

Java is a high-level, class-based, object-oriented programming language that is designed to have as few implementation dependencies as possible. It is a general-purpose programming language intended to let programmers write once, run anywhere (WORA), meaning that compiled Java code can run on all platforms that support Java without the need for recompilation. Java applications are typically compiled to bytecode



**2.4.2 Database Management System**

**MySQL**

MySQL is an open source database that is platform independent and can easily interface with a number of scripting languages, it works best with PHP though. The number of advantages of using MySQL which include, the ability to handle stored procedures, triggers, SQL and User-Defined functions. It also offers a high-speed data load utility and support for various drivers (ODBC, JDBC, .NET, PHP).

Deploying a MySQL database has proved to be cheap and easy as it doesn’t require special hardware or software requirements, it can work well on any web server but most professionals recommend the apache web server. MySQL is an excellent database to use when developing web based applications because its platform independent and can easily interface with a number of scripting languages.

**3 CHAPTER THREE: SYSTEM ANALYSIS**

**3.1 INTRODUCTION**

This chapter gives a detailed outline of the software development methodology used in this project following up the various existing software development methodology discussed in chapter two. The strength and weaknesses of the chosen methodology have been outlined. Further, the functional and non-functional requirements of the system are explained in detail and the use cases which are a list of steps, typically defining interactions between a role and a system, to achieve a goal. Class diagrams have been given to show detailed data modeling of the system which will be translated into code.

**3.2 SOFTWARE DEVELOPMENT METHODOLOGY OF CHOICE**

Having briefly discussed a few software development methodologies in chapter two, the incremental method was favored for the following reasons:

• It allows for development of high-risk or major functions first

• Each release delivers an operational product

• Customer can respond to each build

• Uses “divide and conquer” breakdown of tasks

• Lowers initial delivery cost

• Initial product delivery is faster

• Customers get important functionality early

• Risk of changing requirements is reduced

In order to create relevant use cases for the system, the following actors for the system have been identified:

* Employee
* Admin

**3.3 SYSTEM DESIGN**

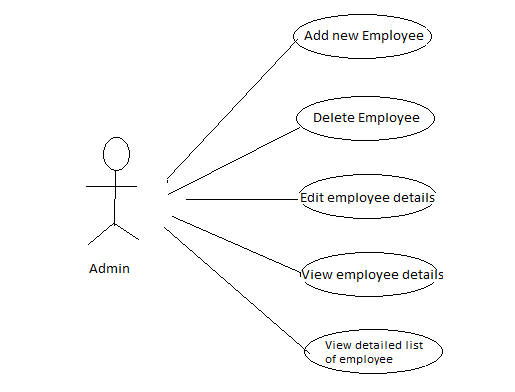
**3.3.1 Use case analysis**

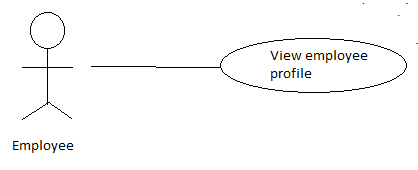
A use case defines a goal-oriented set of interactions between external users and the system under consideration or development. Thus a Use Case Scenario is a description that illustrates, step by step, how a user is intending to use a system, essentially capturing the system behavior from the user's point of view.

**Actors, Use Cases and their Description**

|  |  |  |
| --- | --- | --- |
| **Actor** | **Use case** | **Description** |
| Employee | View Profile | Employee will be able to view personal. |
| Admin | Add new employee | Admin will be able to create new employees. |
| Admin | Edit employee details | Admin will be able to edit employee details. |
| Admin | Delete employee | Admin mean will be able to delete employees. |
| Admin | View employee details | Admin will be able to view employee details. |
| Admin | View detailed list of employee | Admin will be able to view detailed employee details. |

**Use case diagrams:**





**3.4 SUMMARY**

The core and emphasis of this chapter was the analysis of the current system. The various development tools used in the project were also discussed in this chapter. The next chapter will focus on the design characteristics and aspects of the system to be developed.

**4 CHAPTER FOUR: TESTING AND VERIFICATION**

**4.1 INTRODUCTION**

Testing is very important and critical to the success of any project that aims at delivering working software. There are many types of testing that a system may be subjected to, however only the ones in the testing objectives will be carried out for this system.

**4.2 SCOPE**

The overall purpose of testing is to ensure the Employee Management System meets all of its functional and business requirements. The purpose of this chapter is to describe the overall test plan and strategy for testing the system.

**4.3 TESTING GOALS**

The goals in testing this system include validating the quality, usability, reliability and performance of the application. Testing will be performed from a black-box approach. Tests will be designed around requirements and functionality.

**4.4 CONFIRMATION TESTING**

Confirmation testing or re-testing: When a test fails because of the defect then that defect is reported and a new version of the software is expected that has had the defect fixed. In this case we need to execute the test again to confirm that whether the defect got actually fixed or not. This is known as confirmation testing and also known as re-testing. It is important to ensure that the test is executed in exactly the same way it was the first time using the same inputs, data and environments. Hence, when the change is made to the defect in order to fix it then confirmation testing or re-testing is helpful.

**4.5 REGRESSION TESTING**

During confirmation testing the defect got fixed and that part of the application started working as intended. But there might be a possibility that the fix may have introduced or uncovered a different defect elsewhere in the software. The way to detect these **‘**unexpected side-effects**’** of fixes is to do regression testing. The purpose of a regression testing is to verify that modifications in the software or the environment have not caused any unintended adverse side effects and that the system still meets its requirements. Regression testing are mostly automated because in order to fix the defect the same test is carried out again and again and it will be very tedious to do it manually. Regression tests are executed whenever the software changes, either as a result of fixes or new or changed functionality.

**4.6 SUMMARY**

The chapter discussed how the proposed system was subjected to various types of testing. This brought to light why it is very cardinal to test a new system before it is introduce on the main stream of an organization’s business.

**5 CHAPTER SEVEN: CONCLUSION**

**5.1 INTRODUCTION**

The aim of this chapter is to draw conclusions of the work done or achieved and to give an assessment of the completed system, discuss the Problems faced, limitations of the system and give future recommendations on how the system can be improved.

**5.2 RESULTS**

The software product produced was fairly good, it achieved most of the user requirements, the user interface is good and is very easy to navigate, and even novice users can find their way around the console easily. The client side validation is excellent.

**5.3 LIMITATIONS**

**5.3.1 Browser support**

The highly sleek and intuitive interface was made in order to improve Human Computer Interaction (HCI). However, this comes with challenges because lower versions of Internet Explorer (i.e. IE9 and lower) do not support certain features such as column-fill, column-span, align-self, backface-visibility etc.

**5.4 FUTURE WORK**

**5.4.1 Leave Management**

The leave management module can be improved by having all leave requests approved by the head of department before submission rather than going straight to the HR manager. This feature is important because the Supervisor should know which of his/her employees which to go on leave.

**5.4.2 Integration with payroll system**

In order for the system to be more comprehensive, I’d recommend an integration of the system to a payroll system that will enable employees view and download their pay slips on demand.

**5.4.3 Employee Performance**

The designed system provides the admin with the ability to assign tasks to project members. If further worked on, this functionality can assist in determining the performance of employees based on their ability to finish tasks on time.

**5.4.4 Information archiving**

A system holding all the employee information should have some form of archiving system so that retired, suspended or fired employees are archived rather than been completely deleted from the system. This is so because cases may occur where details of an ex-employee may be required especially in cases where an employee did several projects and there details are required for future reference.

**5.5 LEARNING EXPERIENCE**

This project assisted me to gain a practical experience and apply the knowledge assimilated from the previous courses undertook. Putting the knowledge gained earlier and applying different techniques from past courses was interesting and certain concepts, tools and techniques only made sense after seeing their application in a real world scenario. It was extremely challenging at times but it has been a great and worthwhile learning experience.

There is not at all any doubt that the employee management system would be an asset to any company, small or large.

**5.6 CONCLUSION**

In this chapter, the results were discussed, problems faced and limitations were elaborated. Future recommendations for the extension and improvement of the system have also been discussed as well as well as an assessment of achieved functionality.