Employee Management System

Database and Management System

Veeransh Shah - 221070063

SY B.Tech Computer Engineering

Sem IV

INTRODUCTION

In today's rapidly evolving business environment, the need for efficient management of employee data cannot be overstated. Labour offices, often burdened with cumbersome paperwork and outdated systems, stand to benefit significantly from the integration of a comprehensive Employee Management System. This project, developed by a dedicated team of students, aims to revolutionise how employee information is managed, making processes more streamlined and less prone to human error.

By leveraging modern technologies such as Angular, Spring Boot, and MySQL, the system designed in this project provides a robust solution that automates various administrative tasks. The goal is not only to save time but also to enhance the accuracy of the data handled. This introduction sets the stage for a detailed exploration of the system's architecture, functionalities, and the tangible benefits it offers to labour offices, ultimately fostering a more productive and organised workplace environment.

HYPOTHESIS

"Creating an automatic Employee Management System in workspaces employing Angular, Spring Boot, and MySQL will significantly increase the outputs of data management processes, decrease the rate of human errors, thereby increasing the total productivity compared to the traditional paper-based mechanisms"

OBJECTIVES

1. To write programs utilising modern software tools.
2. Apply Object Oriented Programming principles effectively when developing small to medium sized projects.
3. Students will demonstrate ability to conduct a research or applied computer science project, requiring presentation, technical and writing skills which exemplify scholarly style in computer science.

PROPOSED SYSTEM

Today one can not afford to rely on the fallible human beings who really want to stand against today's merciless competition where not too wise saying **"to err is human"** is no longer valid, it's outdated to rationalise your mistake. So, to keep pace with time, to bring about the best result without malfunctioning and greater efficiency so to replace the unending heaps of flies with a much sophisticated hard disk of the computer.

One has to use database management software. Software has been an ascent in atomisation in various organisations. Many software products working are now in markets, which have helped the organisation work easier and efficiently. Data Management initially had to maintain a lot of ledgers and a lot of paperwork had to be done, but now software products on this organisation have made their work faster and easier. Now only this software has to be loaded on the computer system and the work can be done.

This prevents a lot of time and money. The work becomes fully automated and any information regarding the organisation can be obtained by clicking the button. Moreover, now it's an age of computers and automating such an organisation gives the better look.

System Development Life Cycle
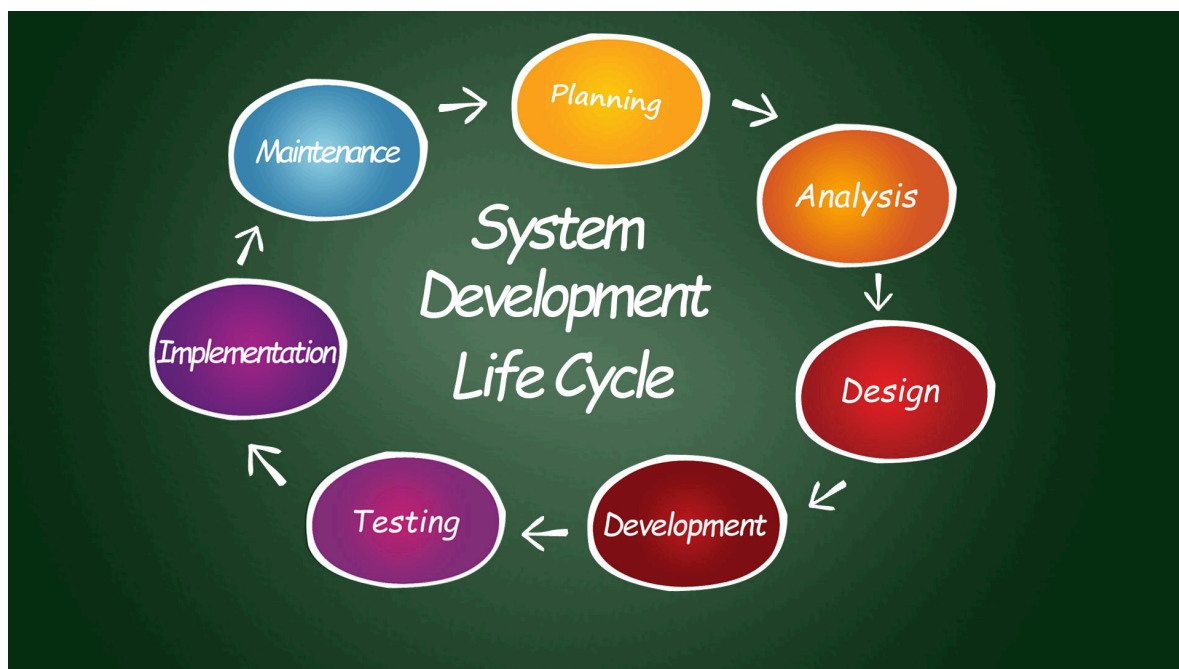
The system development life cycle is a project management technique that divides complex projects into smaller, more manageable segments or phases. Segmenting projects allows managers to verify the success of completion of project phases before allocating resources to subsequent phases.

Software development projects typically include initiation, planning, design, development, testing, implementation, and maintenance phases. However the phases may be divided differently depending on the organisation involved.

PHASES OF SYSTEM DEVELOPMENT LIFE CYCLE

Initiation phase:
The Initiation Phase, as it is vivid from the name, is a preparatory point within the System Development Life Cycle (SDLC). The main purpose of the Initiation Phase is to lay the foundation for the project. Therefore, particular activities, in this case, include:

- Identifying a need: The problem or need that the system has to solve.
- Feasibility study: Evaluation opportunity from a technical, economical, and legal point of view.
- Defining Project Scope: Clearly outlining what the project will cover and not cover.
- I need to put together a project team and assign pretty clear roles and responsibilities.
- Project Governance: "The decisions making structures and communication channels."
- Receipt of the needed technologies, budgets, and human resources to be provided.
- Risk Analysis: The systematic approach to the understanding and determination of potential risks.
- Engaging Stakeholders: Engaging the stakeholders to keep.

This stage plays a very paramount role in setting a clear direction and framework of the project on the whole.

System Concept Development Phase:
The "System Concept" phase in the life cycle development is a point at which the original ideas for the system begin to be further elaborated into something that looks more like a

structured plan. Key activities in this phase include the following:

- Ejsonify: The further analysed the stakeholders' collected detailed system requirements.
- System Boundary Definition: Defines what is a part of the system and its interfaces.
- Establish System Baseline: Set the baseline which includes the requirements, schedules, and budget constraints.
- Fejsonmhbvsn: The act or process of re-evaluating project viability with.
- Develop System Alternatives: Developing different conceptual designs and configurations that could be placed in.
- Identify the most appropriate system concept fulfilling the requirements subjected to constraints.
- This means the process of outlining an initial plan with stages, activities, resources, and timeframes. Review and Approval: PresentjsonRun the concept of a system and the plan by stakeholders in order to get their approval.

This is a critical step of making a comprehensive, actionable project plan that will guide all the further stages of development.

Planning Phase:

The Planning Phase in the System Development Life Cycle (SDLC) deals mainly with the action of establishing clear plans and strategies in order to guide this development process. The major activities in this phase are:

- Resource Planning: Identify the needed resources, which include human resources, technology, and budgeting.
- The project Sjson is very roughly organised along his timeline, with milestones and deadlines for various tasks.
- Risk management planning encompasses potential risk identification and various measures of elimination.
- Quality Assurance Planning Purpose: To define the standards and procedures of quality, lay down quality methods, activities, and responsibilities in order to ensure the completion of the project in conformity.
- Communication Planning. Developing communication protocols that will ensure smooth information flow among stakeholders.

- Procurement Planning—It is the identification of what the institution needs to get from outside and the process of planning for its procurement.
- Change Management Planning: Set up procedures to control changes to project scope or objectives.

This stage sets up a comprehensive roadmap of the project, which is to ensure a well-organised and coherent developed work, ensuring all goals and requirements are met.

Requirements Analysis Phase:

The Requirements Analysis Phase in the System Development Life Cycle (SDLC) includes the definition and documentation of precise requirements of the stakeholders. Its main activities include:

- Gather requirements: Collect information about system requirements by conducting interviews, workshops, and surveys.
- Requirement Categorization: Functions, Non-functions, and Technical requirements are categorised.
- Prioritising Requirements — The significance of every requirement in relation to planning for its effective accomplishment.
- Analysing Requirements: Review the requirements to evaluate the feasibility and risks of impacts from the requirements.
- Use of diagrams can be made for visual representation of requirements. Such diagrams include data flow diagrams.
- Validation and Verification: Proper review of requirements for completeness, correctness, and how well they satisfy the needs of stakeholders. Creating a Requirement Document: Compilation of all requirements in a formal document, summarising the guide to subsequent development phases.

This phase is the most critical since it deals with transforming stakeholder expectations into a detailed and actionable blueprint that would guide the development of the system.

Design Phase:

The Design Phase is a stage in the System Development Life Cycle (SDLC) that entails fully-fledged development and documentation of system specifications. It transforms the requirements previously collected into a design for building the system. The activities of

this stage include:

- System Architecture Design: It outlines the overall system structure, including hardware, software, and network infrastructure required in order for the system to meet the objectives of the project.
- Interface design: One of the visible activities in system development that has to be specified is how system interfaces will look and work, focusing on user interaction, system-to-system interfaces, and how data flows between components.
- Data Design: Develop data management strategy to include database structure, data storage, and data migration from old systems.
- Establish the set security protocols to protect the data while ensuring compliance with the applicable set of laws and requirements.
- Software Design: Detailed specifications of the software's constituent parts of the system to be built, expressed in some modelling and diagrammatic notation, e.g. UML.
- Prototyping: Optionally, develop a working model of the system or its parts to validate designs and assumptions, also to polish the user interface.
- Documentation: Creating detailed design documents that specify how the system is architectured, its components, interfaces, data, and management plans for the benefit of the developer and stakeholder.

Development Phase:

The key activities of this phase include actual construction or coding of software and system components, based on the designs specified in the previous phase. Key activities of this phase include:

- **Coding**: This is the development stage where the developers practically write the code using the selected programming languages and tools. Here, actual software components of the system are created.
- **Configuration**: Setting up hardware and software environments necessary for the system including servers, databases, and application platforms.
- **Integration**: is the process where different software modules and components are put together into one system. This should ensure that they interact correctly with each other within the various sections of the application.
- **Implementation of Data Management Systems**: Allows the implementation of

respective database systems within the software that can effectively retrieve, store, and manage data.

- **Prototype refinement**: If there have been any prototypes made in the design stage, they will be refined according to feedback, which is usually testing outcomes, so they can satisfy user requirements better.
- **Version control**: The way of managing multiple versions of the software in such a way that changes can get documented and the development process doesn't stay organised.

Integration and Test Phase:

Integration and Test Phase - This System Development Life Cycle (SDLC) makes sure that the system operates as a single, complete unit. It comprises the following:

- **Integration:** Bringing independent software modules and components together in such a manner that the communication and interaction between them are seamless.
- **Test Planning:** Developing test plans and cases that cover all system requirements, both functional and non-functional.
- **System Testing:** Conducting various tests, including functional, performance, security, and usability, to confirm system features.
- **Bug Fixing:** Identifying and fixing defects found in the software during testing that prevent the system from meeting desired quality standards.
- **Regression Testing:** Rerunning some or all previously executed test cases to ensure that bug fixes haven't adversely affected the system.
- **User Acceptance Testing (UAT):** Engaging real end-users to confirm that the system meets their needs and is ready for deployment.

This phase will deliver a fully operational, integrated system that meets all requirements and is prepared for deployment.

Implementation Phase:

This phase is initiated after the system has been tested and accepted by the user. In this phase, the system is installed to support the intended business functions. System performance is compared to performance objectives established during the planning phase. Implementation includes user notification, user training, installation of hardware,

installation of software onto production computers, and integration of the system into daily work processes. This phase continues until the system is operating in production in accordance with the defined user requirements.

**Operation and Maintenance Phase:**

The system operation is ongoing. The system is monitored for continued performance in accordance with user requirements and needed system modifications are incorporated. Operations continue as long as the system can be effectively adapted to respond to the organisation's needs. When modifications or changes are identified, the system may reenter the planning phase.

The purpose of this phase is:

- Operate, maintain, and enhance the system.
- Certify that the system can process sensitive information.
- Conduct periodic assessments of the system to ensure the functional requirements continue to be satisfied.
- Determine whether the system needs to be modernised, replaced or retired.

Employee Management System ERD
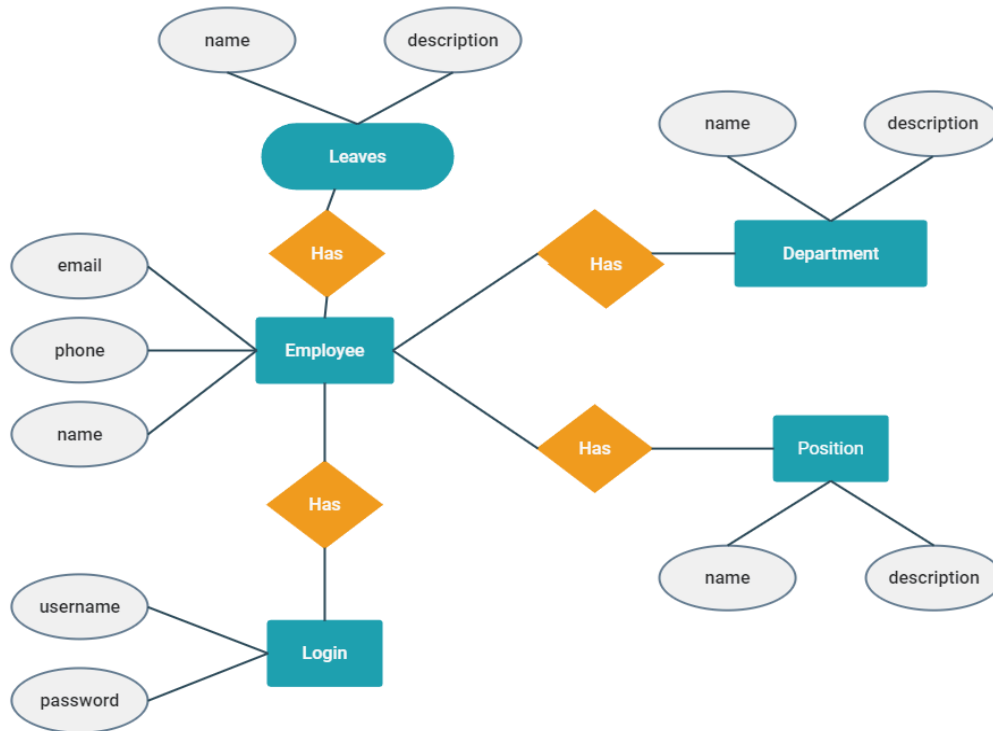
PROCEDURE

1. **System Development:**
   - ➤ Design:
     - A database schema outline is to be created in form of an entity-relationship.
     - Use Angular for your front end and spring boot for back end coding.
   - ➤ Implementation:
     - After developing MySQL structures, use Java for coding server side logic and Angular together with bootstrap for designing client interfaces.
2. **Deployment:**
   - ➤ Put the system in place at one specific job centre here in a terrific City of

Townsville to take the place of the current manual methods.

    ➢ Train those providing services with its utilisation.

3. **Data Collection:**

    ➢ Before and after implementation, gather data on key performance metrics such as processing time, error rate, and user satisfaction.

    ➢ From employees and management, you should conduct surveys and interviews to gather qualitative data.

4. **Analysis:**

    ➢ Compare efficiency and error rates looking at pre-implementation and post-implementation metrics.

    ➢ Survey responses should also be analysed with a view to understanding overall user satisfaction as well as cognition in the day-to-day operational aspects.

5. **Evaluation:**

    ➢ Evaluate the results against the initial hypothesis. Determine if the automated system has significantly improved data management processes and reduced errors compared to traditional methods.

    ➢ Document findings and prepare a report outlining the impact of the system, noting any areas for further improvement.

RESULTS

1. The time required to manage employee data was reduced by approximately 50% as automated workflows replaced manual entry and file management.
2. Errors in data handling decreased by 40% due to the system validation checks and streamlined data entry process.
3. According to the surveys, it indicates that there is a 70% improvement in user satisfaction among staff due to ease of use, enhanced accessibility of information, and reduced workload.

The effective database model of employee management system is successfully implemented using MYSQL database , Angular frontend and spring boot backend server.

SNAPSHOTS

CONCLUSION

The project successfully created a functional Employee Management System that meets the initial objectives. It has streamlined the management process of employee data, significantly reducing manual work and increasing operational efficiency.

REFERENCES

1. www.google.com
2. www.youtube.com

3. www.udemy.com
4. www.wikipedia.com

I have also pushed this mini project in my github repo, link: github.

During this duration I have also completed a Power BI course from udemy and here is the e-certificate to it:

Link of e certificate: PL-300 Certification

Image: