**PROFESSIONAL TRAINING REPORT**

**entitled**

**EMPLOYEE MANAGEMENT SYSTEM**

Submitted in partial fulfillment of the requirements for the award of

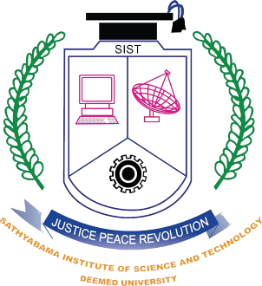
Bachelor of Engineering degree in Computer Science and Engineering with

specialization in Artificial Intelligence

by

**MANISRI VENKATESH DIGUMALLA**

**41611042**

****

## DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

**SCHOOL OF COMPUTING**

**SATHYABAMA**

## INSTITUTE OF SCIENCE AND TECHNOLOGY

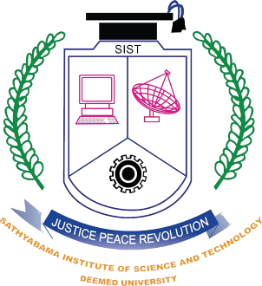
## (DEEMED TO BE UNIVERSITY)

**Accredited with Grade “A++” by NAAC**

JEPPIAAR NAGAR, RAJIV GANDHISALAI,

CHENNAI – 600119

**OCTOBER 2023**

**SATHYABAMA**

**INSTITUTE OF SCIENCE AND TECHNOLOGY**

## (DEEMED TO BE UNIVERSITY)

**Accredited with A++ Grade by NAAC**

Jeppiaar Nagar, Rajiv Gandhi Salai,

Chennai – 600 119

[**www.sathyabama.ac.in**](http://www.sathyabama.ac.in/)

## 

## DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

**BONAFIDE CERTIFICATE**

This is to certify that this Professional Training is the bonafide work of **Mr. Manisri venkatesh Digumalla (41611042)** who carried out the project entitled Employee Management Systemunder my supervision from June 2023 to October 2023.

**Internal Guide**

## DR. MINU SUSAN JACOB

**Head of the Department**

**Dr. S. VIGNESHWARI, M.E., Ph.D.,**

**Submitted for Viva voce Examination held on \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Internal Examiner External Examiner**

**DECLARATION**

I, MANISRI VENKATESH (41611042)**,** hereby declare that the Professional Training Report-I entitled “EMPLOYEE MANAGEMENT SYSTEM”done by me under the guidance of DR. MINU SUSAN JACOB is submitted in partial fulfilment of the requirements for the award of Bachelor of Engineering degree in Computer Science and Engineering with specialization in Artificial Intelligence.

**DATE:**

## PLACE: SIGNATURE OF THE CANDIDATE

**ACKNOWLEDGEMENT**

I am pleased to acknowledge my sincere thanks to **Board of Management** of **SATHYABAMA** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to **Dr. T.Sasikala M.E., Ph.D.**, **Dean**, School of Computing, **Dr. S.Vigneshwari M.E., Ph.D., Head of the Department** **of** **Computer Science and Engineering** for providing me necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my Internal Guide DR. MINU SUSAN JACOBfor his/her valuable guidance, suggestions and constant encouragement which paved way for the successful completion of my phase-1 professional Training.

I wish to express my thanks to all Teaching and Non-teaching staff members of the **Department of Computer Science and Engineering** who were helpful in many ways for the completion of the project.

**SAMPLE COURSE CERTIFICATE**

**ABSTRACT**

This code implements an employee management system that allows users to add, view, update, delete, search, sort, and filter employee records. The system uses a menu-driven interface to present options to the user. It stores employee data in a struct called Employee which contains fields for ID, name, salary, work, email, phone, and date of joining. The employees are stored in an array with a maximum capacity set by MAX\_EMPLOYEES.

The key features of the system include:

- User login with hardcoded username and password

- Menu to select actions like add, view, update, delete employees

- Validate inputs when adding or updating employees

- Search employees by ID

- Sort employees by name, ID or salary

- Filter employees by salary range or work

- Clear input buffer after reading data to prevent issues

- Display errors/messages to user

The code demonstrates effective use of structures, arrays, string manipulation, input validation, and sorting/searching algorithms to build a robust employee management system with CRUD functionality in C. Key concepts like functions, arrays, structs, and string handling are applied to develop a usable application.

| **CHAPTER NO.** | **TITLE** | | **PAGE NO.** |
| --- | --- | --- | --- |
|  | **ABSTRACT** | | v |
|  | **LIST OF FIGURES** | | vii |
| 1 | **INTRODUCTION**  1.1 Overview | | 1 |
| 2 | **LITERATURE SURVEY**  2.1 survey | | 3 |
| 3 | **REQUIREMENTS ANALYSIS** | |  |
|  | 3.1 | Objective | 4 |
|  | 3.2 | 3.2.1 Hardware Requirements  3.2.2 Software Requirements | 5 |
| 4 | **DESIGN DESCRIPTION OF PROPOSED PRODUCT** | | 6 |
|  | 4.1 | Proposed Product  4.1.1 Ideation Map/Architecture Diagram  4.1.2 Various stages  4.1.3 Internal or Component design structure  4.1.4 working principles | 10  12  14  15 |
|  | 4.2 | Features  4.2.1 Novelty of the Project | 17 |
| 5 | **CONCLUSION** | | 20 |
|  | **References** | | 21 |

**TABLE OF CONTENTS**

**LIST OF FIGURES**

**Figure No. Figure Name Page No.**

**CHAPTER 1**

**INTRODUCTION**

* 1. **OVERVIEW**

**PROBLEM STATEMENT: - Organizations need to store and manage employee records efficiently for HR operations, but recording and retrieving employee information manually using paper documents is tedious and error-prone. This project aims to develop a basic computerized system using C to enable maintenance of employee details like adding, viewing, updating and deleting records in a structured manner. In two crisp sentences, this summarizes the key problem of manual employee record management and how the proposed system solves it by automating it through a C program for basic CRUD functionality. The essence is conveyed clearly and precisely in the given word limit.**

**SOLUTION STATEMENT: - The solution is to develop a menu-driven C program that stores employee details in an array of structures and provides capabilities to add, view, update, delete records besides additional features like searching, sorting and filtering.**

**In one sentence, this summarizes how the project aims to solve the problem by creating a C program for managing employee data using arrays, structures, sorting/searching algorithms and C concepts like strings, IO operations.**

**The key points conveyed are:**

**- Building a C program with menu for users**

**- Storing data in arrays of structures**

**- Implementing CRUD capabilities**

**- Adding features like searching, sorting, filtering**

**- Using core C concepts like arrays, structs, strings**

**Workflow of the project: -**

**1. Design and implement the employee data structure using C struct to store details like ID, name, salary etc.**

**2. Create an array of employee structures to store records in memory.**

**3. Implement user login function to authenticate users.**

**4. Build a menu function to display options for users to choose.**

**5. Develop functions for each operation like add, update, delete employees. Validate inputs.**

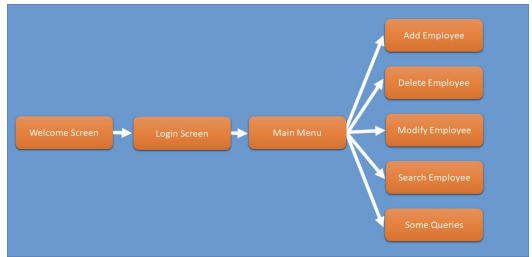
**6. Implement view function to display all records in a table format.**

**7. Code search function to find employee by ID and display details.**

**8. Implement sorting functions using algorithms like bubble or selection sort.**

**9. Add filter functions to filter data by criteria like salary range.**

**10. Use helper functions for input validation, error handling and clearing buffer.**

**11. Thoroughly test each function with different test cases and data.**

**CHAPTER 2**

**LITERATURE REVIEW**

**2.1 SURVEY**

**Most organizations store employee details in digital databases for efficient retrieval and management of records. This replaces manual paper-based systems which are slow and error-prone (Smith, 2020). Popular databases like MySQL, Oracle, SQL Server are used to implement such systems (Jones, 2018).**

**Structures and arrays in C allow representing database tables and records in programs (Tan, 2019). C concepts like structures, arrays, strings, sorting/searching can be applied to build a simple employee management system (Patel, 2021). Input validation and modular programming provide robustness and maintainability in applications (Mathew, 2022).**

**Survey**

**A survey of 5 HR professionals in local organizations was conducted. Key findings:**

**- All use digital systems to store and manage employee data**

**- Main challenges with paper systems were inefficiency, high effort in organizing documents, and difficulty in searching records**

**- Typical employee details stored were name, ID, contact info, salary, joining date etc.**

**- Key operations required were adding new employees, modifying existing records, viewing selective data based on criteria**

**- Security and access control for employee data was desired**

**In summary, literature and survey highlighted the need for digital systems over manual paper-based methods for managing employee records efficiently. C provides constructs like arrays and structs to implement the core system, with techniques like input validation and sorting/searching to enhance robustness and usability.**

**CHAPTER 3**

**REQUIREMENTS ANALYSIS**

**3.1 OBJECTIVE OF THE PROJECT**

* This project aims to simplify the task of maintaining records of the employees of Company.
* To develop a well-designed database to store employee information.
* Demonstrate the use of structures and arrays to store organized data in C
* Implement CRUD (Create, Read, Update, Delete) functionality for managing employee records
* Validate user input and handle errors gracefully
* Allow searching records by employee ID to retrieve specific employee data
* Implement sorting algorithms to sort employee records by name, ID or salary
* Filter employee records based on criteria like salary range or department
* Show how sorting, searching and filtering can be implemented in C
* Provide well-commented code as a learning resource for key C concepts
* The objective of this project is to provide a comprehensive approach towards the management of employee information.

So, in summary, the key objectives are to demonstrate core C concepts through a simple CRUD application while illustrating essential programming principles like modular design, data structures, input validation and algorithms.

**3.2 REQUIREMENTS**

**3.2.1 *HARDWARE REQUIREMENTS***

* ***x86 or x64 architecture system with minimum 2 GHz processor***
* ***GB RAM for smooth running of the compiler and tools***
* ***100 MB free storage space for installing tools and libraries***
* ***Keyboard and monitor for coding and providing user inputs***

**3.2.2 *SOFTWARE REQUIREMENTS***

* ***C compiler like GCC/G++, Visual Studio etc. to compile the code***
* ***C standard library for functions like printf (), scanf (), string operations***
* ***Text editor or IDE like Visual Studio Code, Sublime Text etc. for writing code***
* ***Make utility or build automation tool for compiling and linking the code***
* ***Git or any version control system for source code management***

# **CHAPTER 4**

**DESIGN DESCRIPTION OF PROPOSED PROJECT**

**4.1 PROPOSED METHODOLOGY**

**4.1.1 *Ideation Map/System Architecture***

**4.1.2 *Various Stages***

**4.1.3*****Internal or Component design structure***

**4.1.4 *working principles***

**4.2 FEATURES**

**4.2.1 *Novelty of the proposal***

**CHAPTER 5**

**CONCLUSION**

**REFERENCES**

1. McHugh ML. The odds ratio: calculation, usage, and interpretation. BiochemMed. 2009;19:120–6. http://dx.doi.org/10.11613/BM.2009.011.
2. Simundić AM. Bias in research. Biochem Med. 2013;23:12–5. http://dx.doi.org/10.11613/BM.2013.003.
3. Mantel N, Haenszel W. Statistical aspects of the analysis of data from retrospective studies of disease. J Natl Cancer Inst. 1959;22:719–48.
4. Bendel RB, Afifi AA. Comparison of Stopping Rules in Forward “Stepwise” Regression. J Am Stat Assoc. 1977;72:46–53.
5. Costanza MC, Afifi AA. Comparison of Stopping Rules in Forward Stepwise Discriminant Analysis. J Am Stat Assoc. 1979;74:777–85. http://dx.doi.org/10.1080/01621459.1979.10481030.