**Project Report on**

**Employee Management System**



Bangladesh University of Business and Technology Dhaka-1216, Bangladesh

Department of Computer Science and Technology

A project report submitted in partial fulfillment of the requirements

For the course of Software Development 1

|  |  |
| --- | --- |
| **Project Supervisor** | **Submitted By** |
| Md.. Kawsher Mahbub  **Lecturer**  Department of Computer Science & Technology | Md.Firoz Abdullah, intake-43(2),  Roll: 22231203059  Abir Hossain, intake-43(2),  Roll: 22231203082  Md.Firoz Abdullah, intake-32,  Roll: 18192203034 |

01

**ABSTRACT**

Employee Management System Project is essential software designed to keep track of employee information in any company. It stores data such as their employees’ personal information. The goal of the “Employee Management System” is to create a work center scheduling system. Scheduling is a technology that makes the process of informing activities and notifications in the company where it is implemented simply and even online.

The employee management system project gives managers a better idea of their employees and helps them plan and manage their work hours in order to cut costs and boost productivity. It gives appropriate directions and supervision for employees. It also secures and manages information that is important to the employees including personal and work-related information.

.

**Table of Contents**

[Acknowledgements v](#_TOC_250042)

1. [Introduction 1](#_TOC_250041)
   1. [Goal 1](#_TOC_250040)
   2. [Need of the application 1](#_TOC_250039)
   3. [Scope 2](#_TOC_250038)
   4. [Platform Specifications – Deployment 2](#_TOC_250037)
      1. [Hardware Specification 2](#_TOC_250036)
      2. [Software Specification 3](#_TOC_250035)
2. [System Requirement Analysis 3](#_TOC_250034)
   1. [Information Gathering 3](#_TOC_250033)
   2. [System Feasibility 4](#_TOC_250032)
      1. [Economic Feasibility 4](#_TOC_250031)
      2. [Technical Feasibility 4](#_TOC_250030)
      3. [Behavioral Feasibility 4](#_TOC_250029)
3. [System Analysis 5](#_TOC_250028)
   1. [ER Diagram 5](#_TOC_250027)
   2. [Data Flow Diagram 6](#_TOC_250026)
   3. [Use case Diagram 10](#_TOC_250025)
   4. [Class Diagram 11](#_TOC_250024)

4.[Architectural Design 12](#_TOC_250021)

5.[Implementation 14](#_TOC_250015)

6.[Testing. 19](#_TOC_250011)

7.[Results & Challenges 24](#_TOC_250005)

* 1. [Challenges 25](#_TOC_250004)

8.[Conclusions 25](#_TOC_250003)

# Acknowledgements

I would like to thank my major professor Dr. Daniel Andresen for his constant guidance and help throughout the project. I would also like to thank Dr. Torben Amtoft and Dr. Mitchell Neilsen for graciously accepting to be on my committee.

Finally, I would like to thank my family and my friends for all the support and encouragement.

# Introduction

## Goal

Employee management systems are important because a business’s workforce is its greatest asset. Yet, despite this intrinsic value, employee engagement is sometimes overlooked because HR professionals are either too busy with administrative work or lack the integrations necessary to use their people data effectively. Technology can alleviate such burdens and afford employers more time to connect with workers and create strategic initiatives that will attract and retain talent.

* + - The automation and machine learning capabilities that are common with most employee management systems can help HR departments accomplish more with less effort.
    - Mobile self-service features, online training and up skilling programs and flexible pay options are just a few of the ways technology can enrich the employee experience.

Some employee management system providers offer global and/or local regulatory monitoring services that can help decrease the risk of fines or penalties for inadvertent non-compliance.

* + - Detect hackers and prevent security breaches, employee management systems typically use multi-factor authentication, data encryption and fraud detection.

.

#### Employee Management System Project Report: Modules

#### Here are the modules present in Employee Management System Project Report.

#### • Employee Information Management: This module provides a consolidated database for

#### Application tracking, employee demographics, compensation and benefit options, time

#### Tracking and more. It's also the central location for all of your personnel data, including

#### The most up-to-date statistics on your organization's hiring and retention patterns.

#### • Employee Schedule Monitoring: Employee schedule monitoring entails the use of a

#### Variety of workplace surveillance techniques to obtain information on employees' activities

#### And whereabouts. Businesses keep tabs on employees in order to boost efficiency and

#### Safeguard company assets.

#### • Monitor Working Days and Holidays: This is the practice of employers watching over

#### Their employees' activities in the workplace and working days in order to ensure that they

#### are productive. It will also help them to be aware when should their customer be active or

#### Take a break.

#### • Set Leave Processing: Employee leave request, approval, and monitoring will be aided by

#### Leave management in an organization's administrative processes and controls. With the help

#### Of this module, this task would be much lighter and manageable for the admin.

#### • Generate Leave Reports: It's useful for keeping track and securing the leave information

#### Of the employees. This can also can assist in problem solving and avoid conflicts.

#### Generating these reports will assist the admin in updating details regarding important

#### Information.

#### Benefit:

* + - Clear communication of expectations and goals is paramount in effective employee management. Managers should collaborate with employees to establish well-defined, measurable objectives that align with the company’s strategic goals.
    - Regular feedback and recognition of accomplishments are vital in keeping employees motivated and engaged. Managers should provide timely, constructive feedback on performance, acknowledging employees’ achievements and identifying areas for improvement.
    - Encouraging collaboration and teamwork is another essential aspect of successful employee management. Managers should create opportunities for employees to work together on projects, share ideas, and learn from one another.
    - Offering opportunities for growth and development is crucial in retaining top talent and nurturing a high-performing workforce. By identifying skill gaps and providing relevant training, managers can support employees in advancing their careers and acquiring new competencies.

## Scope

* + - First and foremost, employee management software saves time and reduces the administrative burden on managers and HR professionals.
    - Another important benefit of employee management software is the improved accuracy and compliance it offers.
    - Employee management software can also enhance communication and collaboration within teams
    - Another benefit of using employee management software is the valuable insights it provides into employee performance and workforce trends

## Platform Specifications – Deployment

### Hardware Specification

Processor P IV RAM 250 MB

Minimum Space Required 100 MB Display 16 bit color

### Software Specification

Operating Environment Win 2000/XP

Platform .Net Framework & IIS Visual Studio 2008 Database SQL Server 2005

# System Requirement Analysis

## Information Gathering

At the very commencement, I proceeded to a decision to carry out the development of my task into the following steps:

1. Exploring the available development environments and techniques.

2. Database Analyzing.

3. Database design and Implementation.

4. Program’s Structure Analyzing.

5. GUI (Graphical User Interface) constructing.

6. Bringing all the stuff together (controls data binding and functions implementation).

7. Tests. Each one of these steps could be explained in some brief details as follows:

1. Exploring the available development environments and techniques there is a lot of programming environments available to be used for such kind of elaborations. The point is to choose such an environment that we will be able to operate within a convenient and easy way. This is more or less optional and individual process that depends on the developer’s experience as well.

2. Database analyzing it concerns all of the demands, put upon the database content and its functionality. The database should be designed and implemented in a way that the user would expect it to be.

3. Database design and Implementation this step is tightly related with the previous one as it is completely determined by the requirements, analyzed and discussed in step2.

4. Program’s Structure Analyzing The application program as an interface between the users and the database should be an accurate “reflection” of the database on the screen; hence a well analyzed and defined structure is needed.

5. GUI Constructing After analyzing the program’s structure and defining what it should consist of, a graphical representation of this stuff is needed in order to enable the user to interact with the data.

6. Bringing all the stuff together the next step that should be taken is connecting the program with the database and performing the necessary functionality upon all of the controls.

has to be done upon the system’s functionality.

## System Feasibility

The system feasibility can be divided into the following sections:

### Economic Feasibility

The project is economically feasible as the only cost involved is having a computer with the minimum requirements mentioned earlier. For the users to access the application, the only cost involved will be in getting access to the Internet.

### Technical Feasibility

To deploy the application, the only technical aspects needed are mentioned below: Operating Environment Win 2000/XP/7/10

Platform .Net Framework & Code Block IDE

#### For Users:

IDE Code-Block

### Behavioral Feasibility

The application requires no special technical guidance and all the views available in the application are self explanatory. The users are well guided with warning and failure messages for all the actions taken.

# System Analysis

After carefully analyzing the requirements and functionality of the web application, I had two important diagrams by the end of the analysis phase. They are the ER diagram and data flow diagram which were the basis for finding out entities and relationships between them, the flow of information.

## ER Diagram:



**Information System**

U S E R S

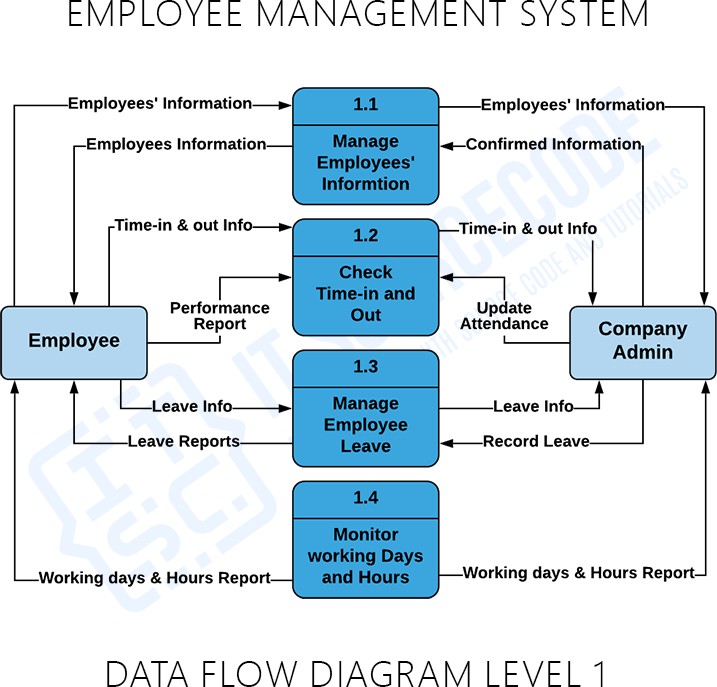
**Console**

**i**

Application Program

Figure 3.1 Entity Relation Ship Diagram

## Data Flow Diagram :

******

Data Flow Diagram Level : 2

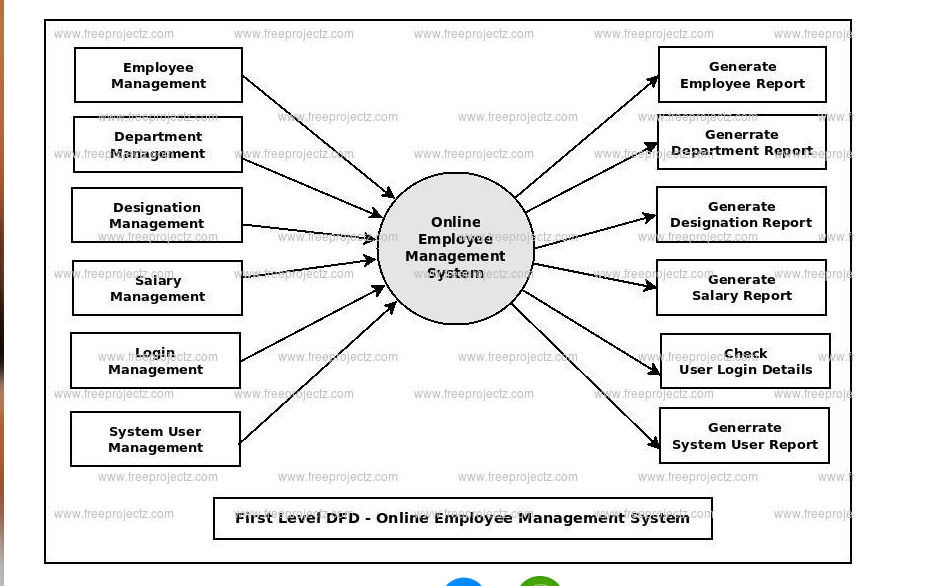


Figure 3.3: A Second Level Diagram

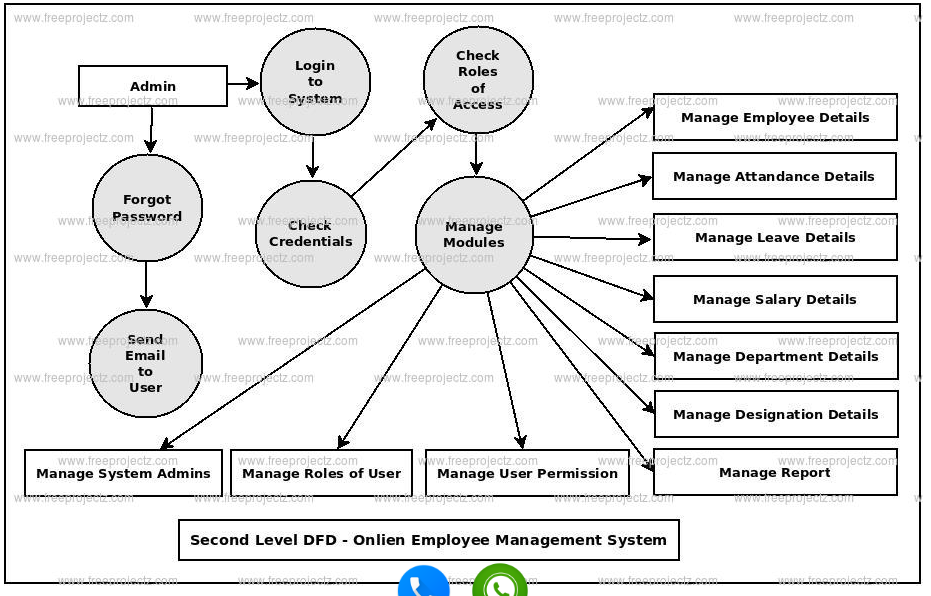


Figure 3.4: A Third Level Diagram

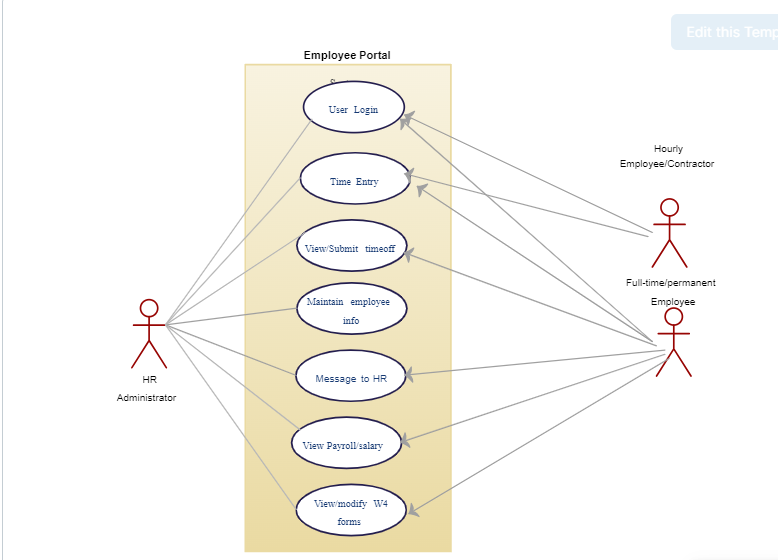
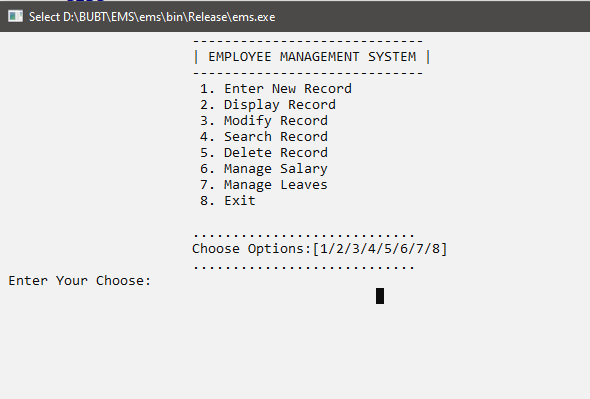


Figure 3.6: Use Case Diagram

### Database Analyzing, design and implementation

The database for the system should include information of company’s staff, respectively of its employees. The data is subdivided into the following groups:

|  |  |  |
| --- | --- | --- |
| Employees’ Basic Details | Working History | Time\_Information |
| Employee\_ID\_Number | Employee\_ID\_Number | Employee\_ID\_Number |
| Personal\_ID\_Number | Company\_Name | Wroked\_Hours |
| First\_Name | Employer\_Name | Off\_Hours |
| Middle\_Name | Company\_Employer\_Address | Days\_off |
| Last\_Name | Company\_Employer\_Cellular\_Phone | Over\_Time |
| Day\_of\_Birth | Company\_Employer\_Office\_Phone | Extra\_Days |
| Month\_of\_Birth | Previous\_Qualification | w\_From\_Date\_Day |
| Year\_of\_Birth | Previous\_Experience | w\_From\_Date\_Month |
| Cellular\_Phone | p\_Start\_Date\_Day | w\_From\_Date\_Year |
| Home\_Phone | p\_Start\_Date\_Month | w\_To\_Date\_Day |
| City | p\_Start\_Date\_Year | w\_To\_Date\_Month |
| Address | p\_End\_Date\_Day | w\_To\_Date\_Year |
| Postal\_Code | p\_End\_Date\_Month |  |
| Qualification | p\_End\_Date\_Year |  |
| Current\_Experience |  |  |
| Start\_Date\_Day |  |  |
| Start\_Date\_Month |  |  |
| Start\_Date\_Year |  |  |
| End\_Date\_Day |  |  |
| End\_Date\_Month |  |  |
| End\_Date\_Year |  |  |
| Type\_of\_Employee |  |  |
| Gender |  |  |
| Marital\_Status |  |  |

**Code-Implementation: **

#include <iostream>

#include <fstream>

#include <cstdlib>

#include <string>

#include <regex>

using namespace std;

bool Emailcheck(string email)

{

const regex pattern("(\\w+)(\\.|\_)?(\\w\*)@(\\w+)(\\.(\\w+))+");

return regex\_match(email, pattern);

}

class employee

{

private:

string name, emp\_id, position, address, email\_id;

double salary;

int leaves;

public:

void menu();

void insert();

void display();

void modify();

void search();

void deleteRecord();

void manageSalary();

void manageLeaves();

};

void employee::menu()

{

menustart:

int choice;

char x;

system("cls");

cout << "\t\t\t-----------------------------" << endl;

cout << "\t\t\t| EMPLOYEE MANAGEMENT SYSTEM |" << endl;

cout << "\t\t\t-----------------------------" << endl;

cout << "\t\t\t 1. Enter New Record" << endl;

cout << "\t\t\t 2. Display Record" << endl;

cout << "\t\t\t 3. Modify Record" << endl;

cout << "\t\t\t 4. Search Record" << endl;

cout << "\t\t\t 5. Delete Record" << endl;

cout << "\t\t\t 6. Manage Salary" << endl;

cout << "\t\t\t 7. Manage Leaves" << endl;

cout << "\t\t\t 8. Exit\n"

<< endl;

cout << "\t\t\t............................" << endl;

cout << "\t\t\tChoose Options:[1/2/3/4/5/6/7/8]" << endl;

cout << "\t\t\t............................" << endl;

cout << " Enter Your Choose: ";

cin >> choice;

switch (choice)

{

case 1:

do

{

insert();

cout << "\n\n\t\t\t Add Another Employee Record (Y, N) : ";

cin >> x;

} while (x == 'y' || x == 'Y');

break;

case 2:

display();

break;

case 3:

modify();

break;

case 4:

search();

break;

case 5:

deleteRecord();

break;

case 6:

manageSalary();

break;

case 7:

manageLeaves();

break;

case 8:

cout << "\n\t\t\t Program Is Exit";

exit(0);

default:

cout << "\n\n\t\t\t Invalid Choice... Please Try Again...";

}

cin.ignore();

cin.get();

goto menustart;

}

void employee::insert()

{

system("cls");

fstream file;

cout << "\n----------------------------------------------------" << endl;

cout << "------------ Add Employee Details ------------" << endl;

cout << "\t\t\tEnter Name: ";

cin.ignore();

getline(cin, name);

cout << "\t\t\tEnter Employee ID: ";

getline(cin, emp\_id);

cout << "\t\t\tEnter Position: ";

getline(cin, position);

cout << "\t\t\tEnter Email Id(name@gmail.com): ";

cin >> email\_id;

if (Emailcheck(email\_id))

{

cout << "\t\t\t Your Email-Id is Valid" << endl;

}

else

{

cout << "\t\t\t Your Email-Id is Invalid" << endl;

return;

}

cin.ignore();

cout << "\t\t\tEnter Address: ";

getline(cin, address);

cout << "\t\t\tEnter Salary: ";

cin >> salary;

cout << "\t\t\tEnter Leaves: ";

cin >> leaves;

file.open("employeeRecord.txt", ios::app | ios::out);

file << name << " " << emp\_id << " " << position << " " << email\_id << " " << address << " " << salary << " " << leaves << "\n";

file.close();

}

void employee::display()

{

system("cls");

fstream file;

int total = 0;

cout << "\n----------------------------------------------------" << endl;

cout << "------------ Employee Record Table ------------" << endl;

file.open("employeeRecord.txt", ios::in);

if (!file)

{

cout << "\n\t\t\tNo Data is Present... ";

}

else

{

while (file >> name >> emp\_id >> position >> email\_id >> address >> salary >> leaves)

{

cout << "\n\n\t\t\tEmployee No.: " << ++total << endl;

cout << "\t\t\tName: " << name << "\n";

cout << "\t\t\tEmployee ID: " << emp\_id << "\n";

cout << "\t\t\tPosition: " << position << "\n";

cout << "\t\t\tEmail Id: " << email\_id << "\n";

cout << "\t\t\tAddress: " << address << "\n";

cout << "\t\t\tSalary: " << salary << "\n";

cout << "\t\t\tLeaves: " << leaves << "\n";

}

if (total == 0)

{

cout << "\n\t\t\tNo Data is Present..." << endl;

}

}

file.close();

}

void employee::modify()

{

system("cls");

fstream file, temp;

bool found = false;

string modify\_emp\_id;

cout << "\n----------------------------------------------------" << endl;

cout << "------------ Modify Employee Details ------------" << endl;

file.open("employeeRecord.txt", ios::in);

temp.open("temp.txt", ios::out);

if (!file)

{

cout << "\n\t\t\tNo Data is Present..";

}

else

{

cout << "\nEnter Employee ID to Modify: ";

cin >> modify\_emp\_id;

while (file >> name >> emp\_id >> position >> email\_id >> address >> salary >> leaves)

{

if (modify\_emp\_id != emp\_id)

{

temp << name << " " << emp\_id << " " << position << " " << email\_id << " " << address << " " << salary << " " << leaves << "\n";

}

else

{

cout << "\n\t\t\tEnter New Name: ";

cin.ignore();

getline(cin, name);

cout << "\t\t\tEnter New Position: ";

getline(cin, position);

cout << "\t\t\tEnter New Email Id(name@gmail.com): ";

cin >> email\_id;

if (Emailcheck(email\_id))

{

cout << "\t\t\t Your Email-Id is Valid" << endl;

}

else

{

cout << "\t\t\t Your Email-Id is Invalid" << endl;

return;

}

cin.ignore();

cout << "\t\t\tEnter New Address: ";

getline(cin, address);

cout << "\t\t\tEnter New Salary: ";

cin >> salary;

cout << "\t\t\tEnter New Leaves: ";

cin >> leaves;

temp << name << " " << emp\_id << " " << position << " " << email\_id << " " << address << " " << salary << " " << leaves << "\n";

found = true;

}

}

if (!found)

{

cout << "\n\t\t\t Employee ID Not Found...";

}

}

file.close();

temp.close();

remove("employeeRecord.txt");

rename("temp.txt", "employeeRecord.txt");

}

void employee::search()

{

system("cls");

fstream file;

bool found = false;

string search\_emp\_id;

cout << "\n----------------------------------------------------" << endl;

cout << "------------ Search Employee Record ------------" << endl;

file.open("employeeRecord.txt", ios::in);

if (!file)

{

cout << "\n\t\t\tNo Data is Present... ";

}

else

{

cout << "\nEnter Employee ID to Search: ";

cin >> search\_emp\_id;

while (file >> name >> emp\_id >> position >> email\_id >> address >> salary >> leaves)

{

if (search\_emp\_id == emp\_id)

{

cout << "\n\n\t\t\tName: " << name << "\n";

cout << "\t\t\tEmployee ID: " << emp\_id << "\n";

cout << "\t\t\tPosition: " << position << "\n";

cout << "\t\t\tEmail Id: " << email\_id << "\n";

cout << "\t\t\tAddress: " << address << "\n";

cout << "\t\t\tSalary: " << salary << "\n";

cout << "\t\t\tLeaves: " << leaves << "\n";

found = true;

break;

}

}

if (!found)

{

cout << "\n\t\t\t Employee ID Not Found...";

}

}

file.close();

}

void employee::deleteRecord()

{

system("cls");

fstream file, temp;

bool found = false;

string delete\_emp\_id;

cout << "\n----------------------------------------------------" << endl;

cout << "------------ Delete Employee Record ------------" << endl;

file.open("employeeRecord.txt", ios::in);

temp.open("temp.txt", ios::out);

if (!file)

{

cout << "\n\t\t\tNo Data is Present..";

}

else

{

cout << "\nEnter Employee ID to Delete: ";

cin >> delete\_emp\_id;

while (file >> name >> emp\_id >> position >> email\_id >> address >> salary >> leaves)

{

if (delete\_emp\_id != emp\_id)

{

temp << name << " " << emp\_id << " " << position << " " << email\_id << " " << address << " " << salary << " " << leaves << "\n";

}

else

{

found = true;

cout << "\n\t\t\tSuccessfully Deleted Data";

}

}

if (!found)

{

cout << "\n\t\t\t Employee ID Not Found...";

}

}

file.close();

temp.close();

remove("employeeRecord.txt");

rename("temp.txt", "employeeRecord.txt");

}

void employee::manageSalary()

{

system("cls");

fstream file, temp;

bool found = false;

string manage\_emp\_id;

double new\_salary;

cout << "\n----------------------------------------------------" << endl;

cout << "------------ Manage Employee Salary ------------" << endl;

file.open("employeeRecord.txt", ios::in);

temp.open("temp.txt", ios::out);

if (!file)

{

cout << "\n\t\t\tNo Data is Present..";

}

else

{

cout << "\nEnter Employee ID to Manage Salary: ";

cin >> manage\_emp\_id;

while (file >> name >> emp\_id >> position >> email\_id >> address >> salary >> leaves)

{

if (manage\_emp\_id != emp\_id)

{

temp << name << " " << emp\_id << " " << position << " " << email\_id << " " << address << " " << salary << " " << leaves << "\n";

}

else

{

found = true;

cout << "\n\t\t\tCurrent Salary: " << salary << "\n";

cout << "\nEnter New Salary: ";

cin >> new\_salary;

temp << name << " " << emp\_id << " " << position << " " << email\_id << " " << address << " " << new\_salary << " " << leaves << "\n";

cout << "\n\t\t\tSalary Updated Successfully";

}

}

if (!found)

{

cout << "\n\t\t\t Employee ID Not Found...";

}

}

file.close();

temp.close();

remove("employeeRecord.txt");

rename("temp.txt", "employeeRecord.txt");

}

void employee::manageLeaves()

{

system("cls");

fstream file, temp;

bool found = false;

string manage\_emp\_id;

int new\_leaves;

cout << "\n----------------------------------------------------" << endl;

cout << "------------ Manage Employee Leaves ------------" << endl;

file.open("employeeRecord.txt", ios::in);

temp.open("temp.txt", ios::out);

if (!file)

{

cout << "\n\t\t\tNo Data is Present..";

}

else

{

cout << "\nEnter Employee ID to Manage Leaves: ";

cin >> manage\_emp\_id;

while (file >> name >> emp\_id >> position >> email\_id >> address >> salary >> leaves)

{

if (manage\_emp\_id != emp\_id)

{

temp << name << " " << emp\_id << " " << position << " " << email\_id << " " << address << " " << salary << " " << leaves << "\n";

}

else

{

found = true;

cout << "\n\t\t\tCurrent Leaves: " << leaves << "\n";

cout << "\nEnter New Leaves: ";

cin >> new\_leaves;

temp << name << " " << emp\_id << " " << position << " " << email\_id << " " << address << " " << salary << " " << new\_leaves << "\n";

cout << "\n\t\t\tLeaves Updated Successfully";

}

}

if (!found)

{

cout << "\n\t\t\t Employee ID Not Found...";

}

}

file.close();

temp.close();

remove("employeeRecord.txt");

rename("temp.txt", "employeeRecord.txt");

}

int main()

{

employee emp;

emp.menu();

return 0;

}

**Conclusion:** In conclusion, the Employee Management System represents a pivotal and transformative solution that revolutionizes the way organizations handle their workforce. By leveraging advanced technologies and automation, this comprehensive software streamlines HR processes, enhances operational efficiency, and improves decision-making, all while fostering a positive employee experience.

Through its robust features, such as employee database management, recruitment and on boarding, attendance tracking, performance management, leave management, training and development, payroll processing, and employee self-service, the system provides a holistic approach to managing the entire employee lifecycle.

One of the system's key advantages is its ability to increase data accuracy, mitigating errors and ensuring that HR personnel have reliable, real-time information at their disposal. Additionally, the system ensures data security and compliance with data protection regulations, safeguarding sensitive employee information.

Moreover, the Employee Management System empowers employees through self-service capabilities, enabling them to take control of their personal information, leaves, and other HR-related activities. This self-service functionality not only boosts employee satisfaction but also reduces the administrative burden on HR staff.

By centralizing and automating various HR processes, the system significantly improves overall efficiency, freeing up valuable time for HR professionals to focus on strategic initiatives and talent development.

Ultimately, the Employee Management System plays a crucial role in enhancing the overall organizational performance, driving productivity, and promoting a collaborative and harmonious work environment. It empowers businesses to make data-driven decisions, optimize workforce planning, and achieve greater success.

As technology continues to evolve, the Employee Management System will remain a vital tool for organizations seeking to stay competitive, attract top talent, and nurture a highly engaged and satisfied workforce. Its implementation not only sets the foundation for streamlined HR operations but also positions companies for long-term growth and success in a dynamic and ever-changing business landscape.