Siddaganga Institute of Technology, Tumakuru

(An Autonomous Institution affiliated to Visvesvaraya Technological University, Belagavi, Approved by AICTE, New Delhi, Accredited by NAAC and ISO 9001:2015 certified)

Open Ended Project

On

EMPLOYEE RECORD MANAGEMENT

submitted

in the partial fulfilment of the requirements IV semester

Bachelor of Engineering

In

Computer Science and Engineering

By

RAMYA G M -1SI20CS090

SAGARI K M – 1SI20CS096

YOGANANDINI.J -1SI20CS138

Department of Computer Science & Engineering

(Program Accredited by NBA)

Siddaganga Institute of Technology

B.H Road, Tumakuru-572 103, Karnataka, India.

Siddaganga Institute of Technology, Tumakuru

(An Autonomous Institution affiliated to Visvesvaraya Technological University, Belagavi, Approved by AICTE, New Delhi, Accredited by NAAC and ISO 9001:2015 certified)

Department of Computer Science and Engineering

(Program Accredited by NBA)

CERTIFICATE

This is to certify that open ended problem titled "SORTIFY- Employee Record Management system" is a bonafide work carried out by **RAMYA G M(1SI20CS090) SAGARI K M(1SI20CS096), Yoganandini J (1SI20CS138)** of IV semester **Bachelor of Engineering in Computer Science and Engineering** of the **SIDDAGANGA INSTITUTE OF TECHNOLOGY** (An Autonomous Institution, affiliated to VTU, Belagavi, approved by AICTE, New Delhi, Accredited by NAAC and ISO 9001:2015 certified) during the academic year 2021-2022.

Name of the Panel Members: **Signature with Date** :

- **Mr. GURURAJ SIR**

ABSTRACT

At the heart of every corporate success lies the primary role of strong management. Employers must display strong leadership and mentorship qualities to bring the best out of the team. In this scenario, one of the main requirements is elevating employee management.

It will enable one to achieve operational efficiency with minimum overheads in the process. In this regard, it becomes imperative to master this management aspect immediately. Through this article, let's understand the concept of employee management in depth:

What is Employee Management?

In short, it blankets all the essential duties of an HR department to help the workers perform smoothly in a company. It includes processes of recruiting, employee engagement and performance review, and everything in between.

An Employee's Management System (EMS) is a software built to handle the primary housekeeping functions of a company. EMS help companies keep track of all the employees and their records. It is used to manage the company using computerized system.

CONCEPTS USED:

- CLASSES AND OBJECTS
- HIERARCHICAL INHERITANCE
- ARRAYS

TABLE OF CONTENTS:

INTRODUCTION

Employee Management system is an application that enables users to create and store Employee Records. The application also provides facilities of a payroll system which enables user to generate Pay slips too. This application is helpful to department of the organization which maintains data of employees related to an organization.

CLASS:

A class in C++ is the building block that leads to Object-Oriented programming. It is a user-defined data type, which holds its own data members and member functions, which can be accessed and used by creating an instance of that class. A C++ class is like a blueprint for an object.

OBJECT:

An **Object** is an instance of a Class. When a class is defined, no memory is allocated but when it is instantiated (i.e. an object is created) memory is allocated.

INHERITANCE:

The capability of a class to derive properties and characteristics from another class is called Inheritance. Inheritance is one of the most important features of Object-Oriented Programming.

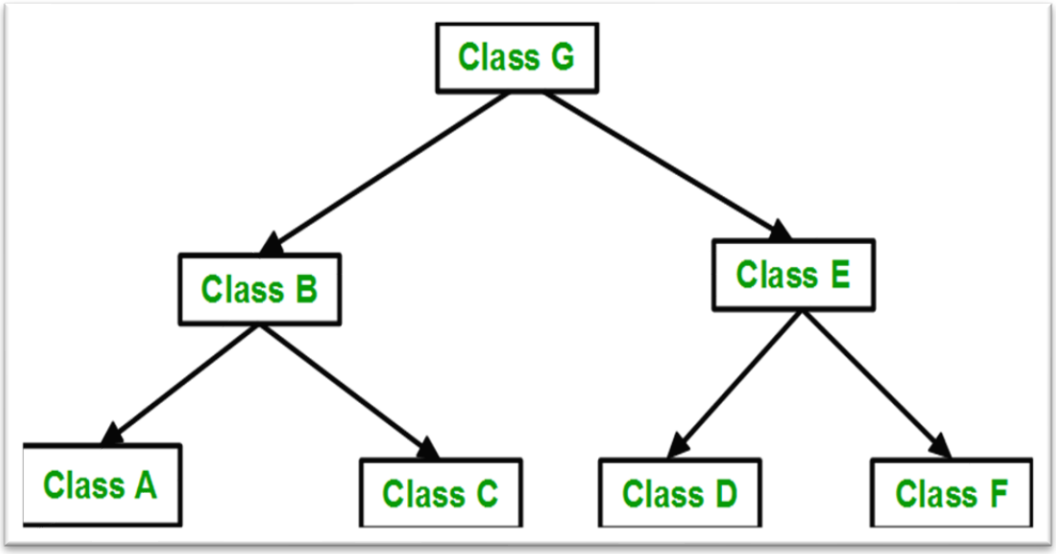
Inheritance is a feature or a process in which, new classes are created from the existing classes. The new class created is called “derived class” or “child class” and the existing class is known as the “base class” or “parent class”. The derived class now is said to be inherited from the base class.

TYPES OF INHERITANCE:

Single Inheritance, Multiple Inheritance, Multilevel Inheritance, Hierarchical Inheritance, Hybrid Inheritance.

HIERARCHICAL INHERITANCE:

In this type of inheritance, more than one subclass is inherited from a single base class. i.e. more than one derived class is created from a single base class.



OBJECTIVE

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 in an organization.

This project simplifies the task of maintain records because of its userfriendly nature.

PROBLEM STATEMENT:

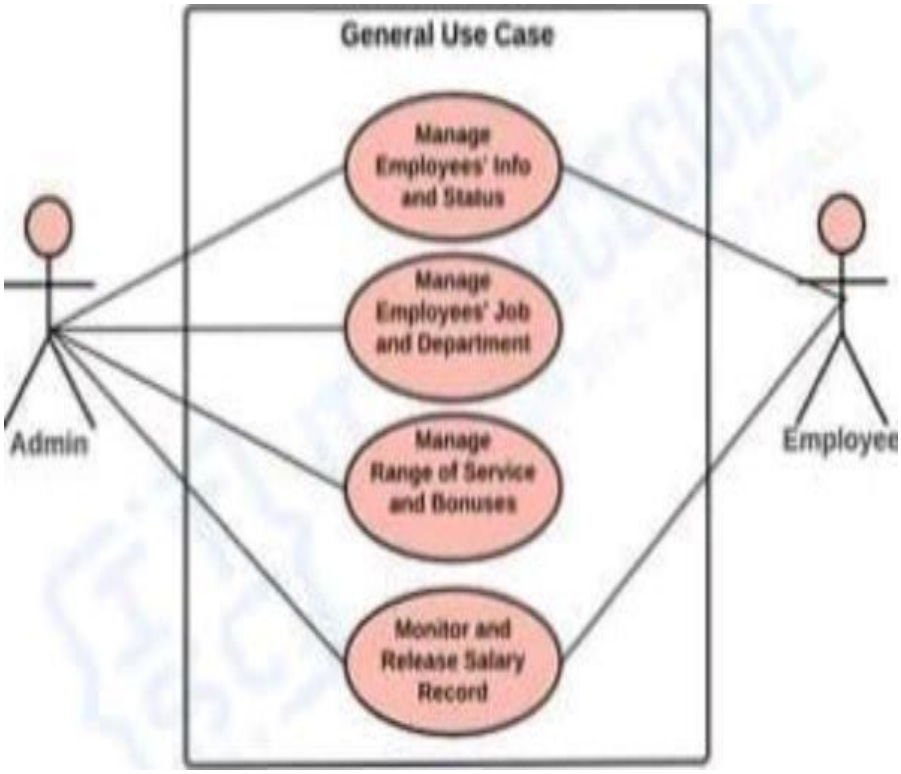
The need for an organization or a small company to sort its employee's records such as sorting based on their method of work that is full time work or part time work. etc.

PROBLEM DESCRIPTION:

What is Employee Management System?

An Employee management system is software useful for the human resource department that allows them to view details of any employee working in the organization or those who worked in the past. It is

designed to simplify the process of record maintenance of employees in an organization.



There are six common problems that firms have with their records management systems:

1. Difficulty inventorying and tracking files.
2. Inability to produce actionable reports from record software
3. Lack of statistics on files
4. Inefficient records disposition
5. No system for managing electronic records
6. A cumbersome interface

IMPLEMENTATION OF CODE

```
#include<iostream>
using namespace std;
const int m=50;

class emp {
public:
        int empno;
char empname [30];
public:         void get ()
        {
            cout<<"\n Enter Employee No.:  ";
cin>>empno;             cout<<"\n Enter
Employee Name:  ";
cin>>empname;
        }
}; class fulltime: public
emp
{
public:
        float daily_rate;
int days;           int
salary;     public:
        void getdata ()
        {
            cout<<"\n Enter Daily Rate:  ";
cin>>daily_rate;             cout<<"\n
Enter No. of Days:  ";
cin>>days;
```

```
        }
void cal ()
        {
salary=daily_rate*days;
cout<<"\n Salary:  "<<salary;
        }
void show ()
        {                   cout<<"\n ------------------------
---------\n";                  cout<<"\n Employee Number:
"<<empno;                     cout<<"\n Employee Name:
"<<empname;                  cout<<"\n Salary:  "<<salary;
cout<<"\n Status:  Fulltime";                  cout<<"\n ----
-----------------------------\n";
        }
}; class parttime:
publicemp
{
public:
        int hourly_rate;
int working_hours;
int salary1;
    public:
void get1()
        {
              cout<<"\n Enter Hourly Rate:  ";
cin>>hourly_rate;                  cout<<"\n Enter
Working_hours:  ";
cin>>working_hours;
```

```
} void cal1() {
salary1=hourly_rate*working_hours;
cout<<"\n Salary:  "<<salary1<<endl;
        }
void show1()
        { cout<<"\n ------------------------
---------\n"; cout<<"\n Employee No:
"<<empno; cout<<"\n Employee Name:
"<<empname; cout<<"\n Salary:  "<<salary1;
cout<<"\n Status:  Part time"; cout<<"\n ---
------------------------------\n";
        }
}; int
main ()
{

    int const cnt=5;
    int var=0;
int var1=0;
fulltime f1[cnt];
parttime p1[cnt];
     int x, i;    do    {
cout<<"\n";          cout<<"\n 1. Enter
Record";          cout<<"\n 2. Display
Record";          cout<<"\n 3. Search
Record";          cout<<"\n 4. Quit";
cout<<"\n\n Enter Your Choice: ";
```

```
cin>>x;              switch(x)              {
case 1:                          int y;
                        cout<<"\n 1. Fulltime Employee";
cout<<"\n 2. Parttime Employee \n";
cout<<"\n Enter: ";                          cin>>y;
switch(y)                        {
case 1:
                                 f1[var]. get ();
                                 f1[var]. getdata ();
f1[var]. cal ();
var++;                                   break;
case 2:
                                 p1[var1]. get ();
p1[var1]. get1();
p1[var1]. cal1();
var1++;                                  break;
                        }
break;                    case 2:
for (i=0; i<var; i++)
                        {
f1[i]. show ();
                        }
for (i=0; i<var1; i++)
                        {
p1[i]. show1();
                        }
break;
```

```
case 3:
int a;
                    cout<<"\n Enter Employee No.: ";
cin>>a;                         for (int i=0; i<var; i++)
                    {
if (f1[i]. empno==a)
                        {
f1[i]. show ();
                        }
if(p1[i]. empno==a)
                        {
p1[i]. show1();
                        }
                    }
            }

      } while (x! =4); return
0;
}
```

OUTPUT:

```
1.Enter Record
2.Display Record
3.Search Record
4.Quit

Enter Your Choice : 1

1. Fulltime Employee
2. Parttime Employee

Enter : 1

Enter Employee No.   :  1

Enter Employee Name  :  SITA

Enter Daily Rate     :  500

Enter No. of Days    :  25

Salary               :  12500

1.Enter Record
2.Display Record
3.Search Record
4.Quit

Enter Your Choice : 1

1. Fulltime Employee
2. Parttime Employee
```

```
Enter : 2

Enter Employee No.    :  2

Enter Employee Name   :  RAM

Enter Hourly Rate     :  700

Enter Working Hours   :  7

Salary                :  4900


1.Enter Record
2.Display Record
3.Search Record
4.Quit

Enter Your Choice : 1

1. Fulltime Employee
2. Parttime Employee

Enter : 1

Enter Employee No.    :  3

Enter Employee Name   :  LAXMAN

Enter Daily Rate      :  900

Enter No. of Days     :  25
```

```
Salary              :  22500

1.Enter Record
2.Display Record
3.Search Record
4.Quit

Enter Your Choice : 2

---------------------------------

Employee Number    :  1
Employee Name      :  SITA
Salary             :  12500
Status             :  Fulltime
---------------------------------

---------------------------------

Employee Number    :  3
Employee Name      :  LAXMAN
Salary             :  22500
Status             :  Fulltime
---------------------------------

---------------------------------

Employee No     :  2
Employee Name   :  RAM
Salary          :  4900
Status          :  Part time
---------------------------------
```

```
Salary               :  22500

1.Enter Record
2.Display Record
3.Search Record
4.Quit

Enter Your Choice : 2

---------------------------------

Employee Number    :  1
Employee Name      :  SITA
Salary             :  12500
Status             :  Fulltime
---------------------------------

---------------------------------

Employee Number    :  3
Employee Name      :  LAXMAN
Salary             :  22500
Status             :  Fulltime
---------------------------------

---------------------------------

Employee No     :  2
Employee Name   :  RAM
Salary          :  4900
Status          :  Part time
---------------------------------
```

```
1.Enter Record
2.Display Record
3.Search Record
4.Quit

Enter Your Choice : 3

Enter Employee No. : 2

-----------------------------------

Employee No      :  2
Employee Name    :  RAM
Salary           :  4900
Status           :  Part time
-----------------------------------


1.Enter Record
2.Display Record
3.Search Record
4.Quit

Enter Your Choice : 3

Enter Employee No. : 3

-----------------------------------

Employee Number    :  3
Employee Name      :  LAXMAN
```

```
Employee Number    :  3
Employee Name      :  LAXMAN
Salary             :  22500
Status             :  Fulltime
--------------------------------


1.Enter Record
2.Display Record
3.Search Record
4.Quit

Enter Your Choice : 4

Process returned 0 (0x0)   execution time : 292.863 s
Press any key to continue.
```

CONCLUSION

In this open ended project we have designed c++ code to maintain employee record through which we can search as well as display employee details like full time employee or part time employee and also their salary.
We have used inheritance, classes and object concept here.

REFERENCES

- Programming in C++, 8th Edition- E. Balagurusamy.
- Internet
- Geeksforgeeks