PAYROLL MANAGEMENT SYSTEM

Submitted in partial fulfillment of the requirements

of the degree

BACHELOR OF ENGINEERING IN

COMPUTER ENGINEERING

By

Krutika Bhagane_BE_A_03

Nikhil Bhosale_BE_A_07

Abhishek Gambre_BE_A_14

Supervisor

Prof. Snehal Chitale

Department of

Computer Engineering

Pillai HOC College of Engineering and Technology, Rasayani

University of Mumbai

(AY 2022-23)

CERTIFICATE

This is to certify that the Mini Project entitled "**PAYROLL MANAGEMENT SYSTEM"** is a bonafide work of **Krutika Bhagane, Nikhil Bhosale**, **Abhishek Gambre** submitted to the University of Mumbai in partial fulfillment of the requirement for the award of the degree of **"Bachelor of Engineering"** in **"Computer Engineering".**

_____

(Prof. Snehal Chitale)

Supervisor

_____ _____

(Prof. Rohini Bhosale) **(Dr. J.W. Bakal)**

Head of Department Principal

Mini Project Approval

This Mini Project entitled "**PAYROLL MANAGEMENT SYSTEM"** is a bonafide work of **Krutika Bhagane, Nikhil Bhosale**, **Abhishek Gambre** is approved for the degree of **Bachelor of Engineering** in **Computer Engineering.**

Examiners

1……
(Internal Examiner Name & Sign)

2……
(External Examiner name & Sign)

Date:

Place:

Contents

Abstract

"Payroll Management System" is designed to make the existing manual system automatic with the help of computerized equipment and full-edged computer software, fulfilling their requirements, so that their valuable data and information can be stored for a longer period with easy access and manipulation of the same. The required software is easily available and easy to work with. It can maintain and view computerized records without getting redundant entries. The project describes how to manage user data for good performance and provide better services for the user.

Keywords: Information system,Data analysis, table fields, relationship, sqlit queries, python.

Acknowledgement

We express our gratitude to our project guide Prof. Snehal Chitale, who provided us with all the guidance and encouragement. We are also thankful to her for providing us the needed assistance, detailed suggestions to do the project. We also would like to deeply express our sincere gratitude to Project coordinators. We are eager and glad to express our gratitude to the Head of Computer Department. Prof. Rohini Bhosale for her approval of this project. We would like to deeply express our sincere gratitude to our respected principal Dr. J.W. Bakal and the management of Pillai HOC College of Engineering and Technology for providing such an ideal atmosphere to build up this project.

1. Introduction

1.1. Introduction

The proposed project “Payroll Management System” has been developed to overcome the problems faced in the practicing of manual system. This program is built to eliminate and in some cases reduce the hardships faced by the existing system. Moreover this system is designed for particular need of the company to carry out its operations in a smooth and effective manner. This program reduces as much as possible to avoid errors while entering data. It also provides error message while entering invalid data. It is user-friendly as no formal knowledge is required to use the system. Human resource challenges are faced by every organization which has to be overcome by the organization. Every organization has different employee and payroll management needs. Therefore I have design exclusive payroll Management System that are adapted to the organization’s Managerial Requirements.

1.2 Motivation

The purpose of this project is to describe the functionality and specifications of the design of a program for Managing Employees payroll. The expected audiences of this document are the developers and the admin of the user. Now with the help of this system the admin has the information on his finger tips and can easily prepare a good record based on their requirements. Finally, we can say that this system will not only automate the process but save the valuable time of the manager or the admin, which can be well utilized buy his institute. This will be an additional advantage and management of power based on their free time from his normal duty.

1.3 Objective

The goal of this project is to accurately predict the Length of Stay for each patient so that the hospitals can optimize resources and function better.

2. Methodology

Data Exploration

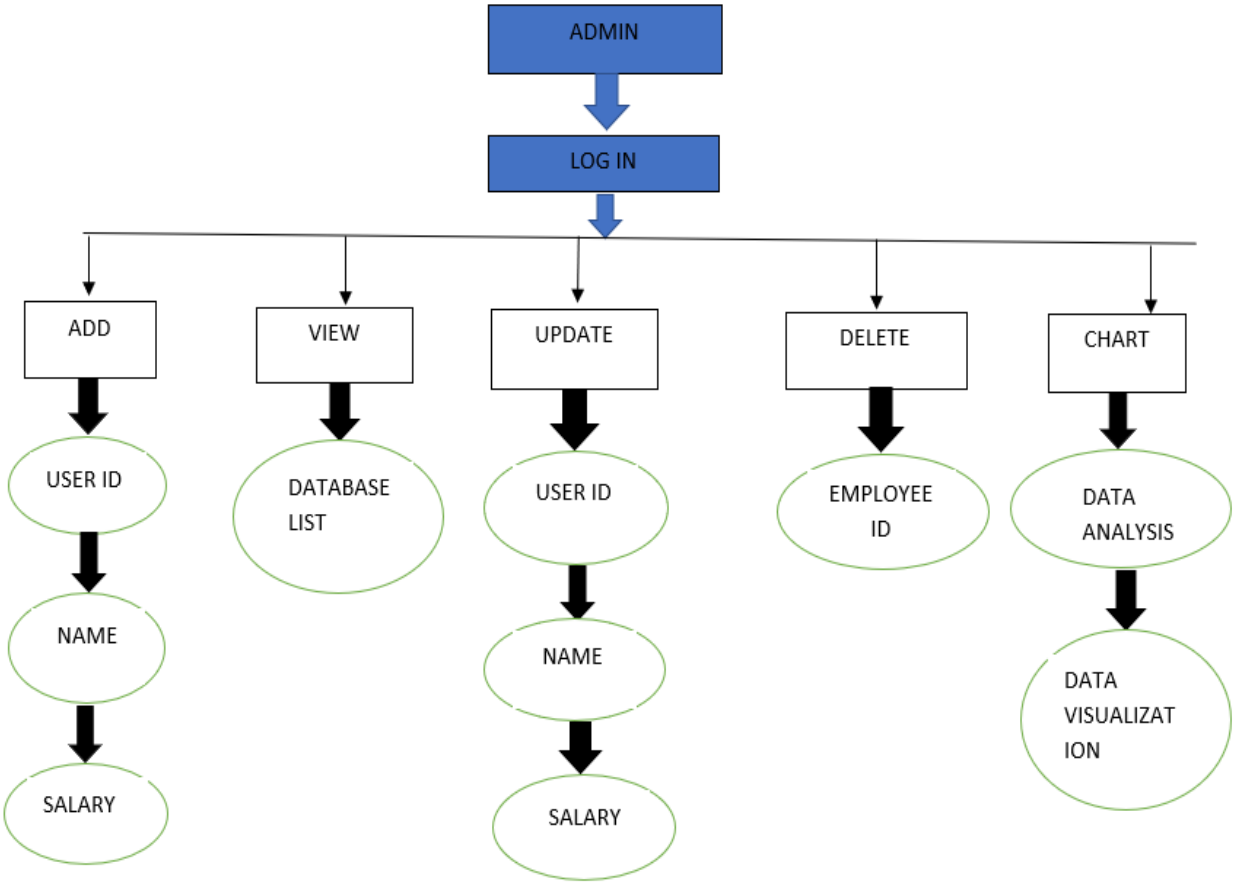
At the very commencement, we proceeded to a decision to carry out the development of my task into the following steps:

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 with in 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 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 in sqlit format.
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.
7. Tests
   To ensure that everything works properly and as it has been expected, test performance has to be done upon the system's functionality.

Case Diagram:-



3. Database design and Implementation

Program:-

```
from tkinter import *
from tkinter.messagebox import *
from tkinter.scrolledtext import *
import requests
import bs4
from sqlite3 import *
import matplotlib.pyplot as plt
import maskpass
def f1():
    add_window.deiconify()
    main_window.withdraw()
def f2():
    main_window.deiconify()
    add_window.withdraw()
def f3():
    update_window.deiconify()
    main_window.withdraw()
def f4():
    main_window.deiconify()
    update_window.withdraw()
def f5():
    view_window.deiconify()
    main_window.withdraw()
    vw_emp_data.delete(1.0,END)
    info=""
    con = None
    try:
        con = connect("D:\Games\internship.db")
        cursor = con.cursor()
        sql="select * from employee"
        cursor.execute(sql)
        data = cursor.fetchall()
        for d in data:
                info = info + " id : " + str(d[0]) + " Name : " + str(d[1]) + "\t"+ " Salary : " + str(d[2])+"\n"
        vw_emp_data.insert(INSERT,info)
    except Exception as e:
        showerror("Issue",e)
    finally:
        if con is not None:
                con.close()
def f6():
    main_window.deiconify()
    view_window.withdraw()
def f7():
    delete_window.deiconify()
    main_window.withdraw()
def f8():
    main_window.deiconify()
    delete_window.withdraw()
def f9():
```

```
    login_window.deiconify()
    main_window.withdraw()
def f10():
    main_window.deiconify()
    login_window.withdraw()
def save():
    id = int(aw_ent_id.get())
    name = aw_ent_name.get()
    salary = int(aw_ent_salary.get())
    if id <0:
      showerror("Issue","Enter Positive Value")
    elif len(name)==0:
      showerror("Issue","Name Should Not be Empty")
    elif len(name)<2:
      showerror("Issue","Name Must contain 3 Alphabet")
    elif salary < 8000:
      showerror("Issue","Salary must be greater than 8K")
    else:
      con = None
      try:
            con=connect("D:\Games\internship.db")
            cursor = con.cursor()
            sql = "insert into employee values('%d','%s','%d')"
            cursor.execute(sql % (id,name,salary))
            con.commit()
            showinfo("Success","Record Added")
            aw_ent_id.delete(0,END)
            aw_ent_name.delete(0,END)
            aw_ent_salary.delete(0,END)
      except Exception as e:
            con.rollback()
            showerror("Issue",e)
            aw_ent_id.delete(0,END)
            aw_ent_name.delete(0,END)
            aw_ent_salary.delete(0,END)
      finally:
            if con is not None:
                  con.close()
def update():
    con = None
    try:
      id = int(uw_ent_id.get())
      name = uw_ent_name.get()
      salary = int(uw_ent_salary.get())
      con = connect("D:\Games\internship.db")
      cursor = con.cursor()
      sql = "update employee set name = '%s',salary = '%d' where id = '%d'"
      cursor.execute(sql % (name,salary,id))
      if cursor.rowcount == 1:
            con.commit()
            showinfo("Success","Record Updated")
            uw_ent_id.delete(0,END)
            uw_ent_name.delete(0,END)
            uw_ent_salary.delete(0,END)
      else:
```

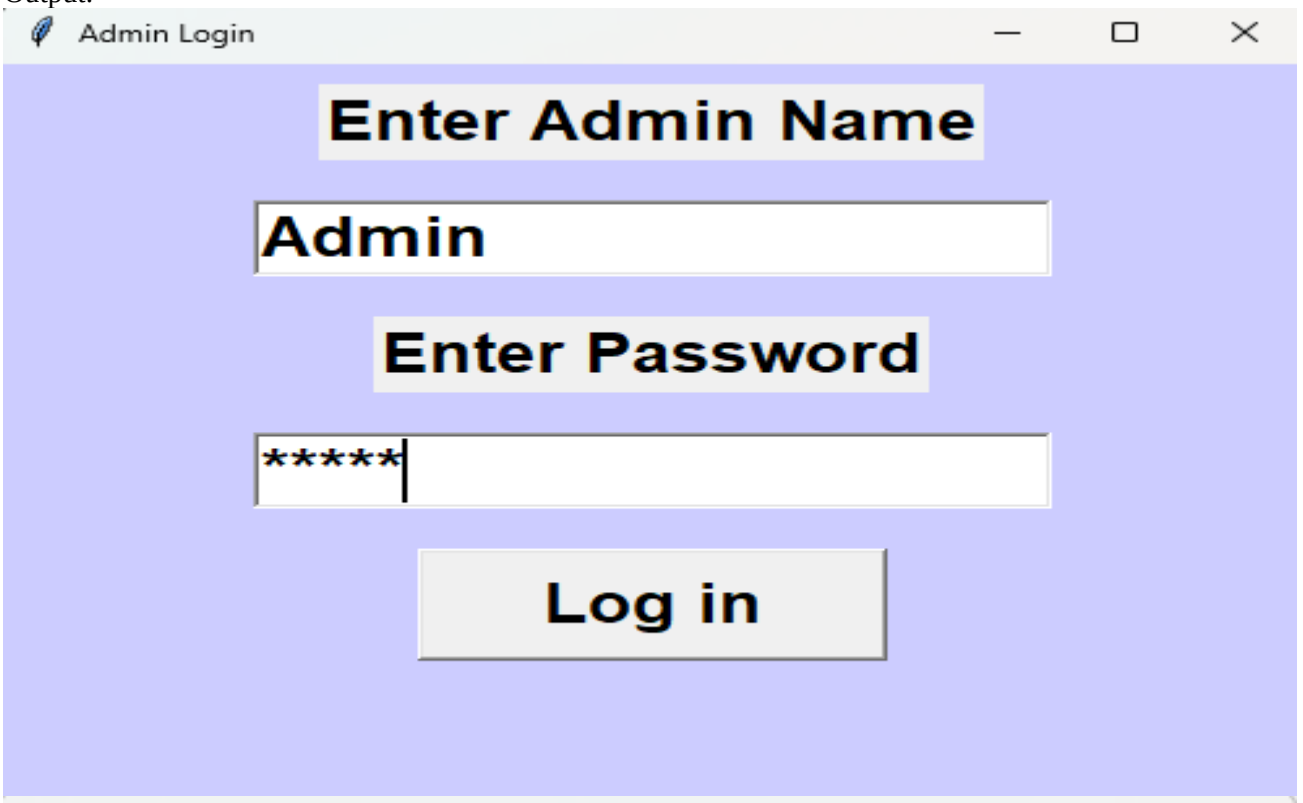
```
                showerror("Issue",id," does not exists")
        except Exception as e:
            print("Issue = ",e)
        finally:
            if con is not None:
                con.close()
                print("closed")
    def delete():
        con = None
        try:
            con = connect("D:\Games\internship.db")
            cursor = con.cursor()
            sql = "delete from employee where id = '%d'"
            id = int(dw_ent_id.get())
            cursor.execute(sql %(id))
            if cursor.rowcount == 1:
                con.commit()
                showinfo("Success","Record deleted")
                dw_ent_id.delete(0,END)
            else:
                print(id," does not exists")
        except Exception as e:
            showerror("Issue",e)
        finally:
            if con is not None:
                con.close()
    def barchart():
        con = None
        try:
            info=[]
            subjects = []
            name=['id1','id2','id3','id4','id5']
            con = connect("D:\Games\internship.db")
            cursor = con.cursor()
            sql = "select salary,name from employee order by salary desc limit 5"
            cursor.execute(sql)
            data = cursor.fetchall()

            #info.append(data)
            #showinfo("success",info)
            for i in data:
                for d in i:
                    info.append(i[0])
            for i in data:
                for d in i:
                    subjects.append(i[1])
            plt.bar(subjects,info,color=["red","green","pink","yellow","blue"])
            plt.xlabel("Employee")
            plt.ylabel("Salary")
            plt.title("Employee Salary")
            plt.show()
        except Exception as e:
            showerror("Issue",e)
        finally:
            if con is not None:
```

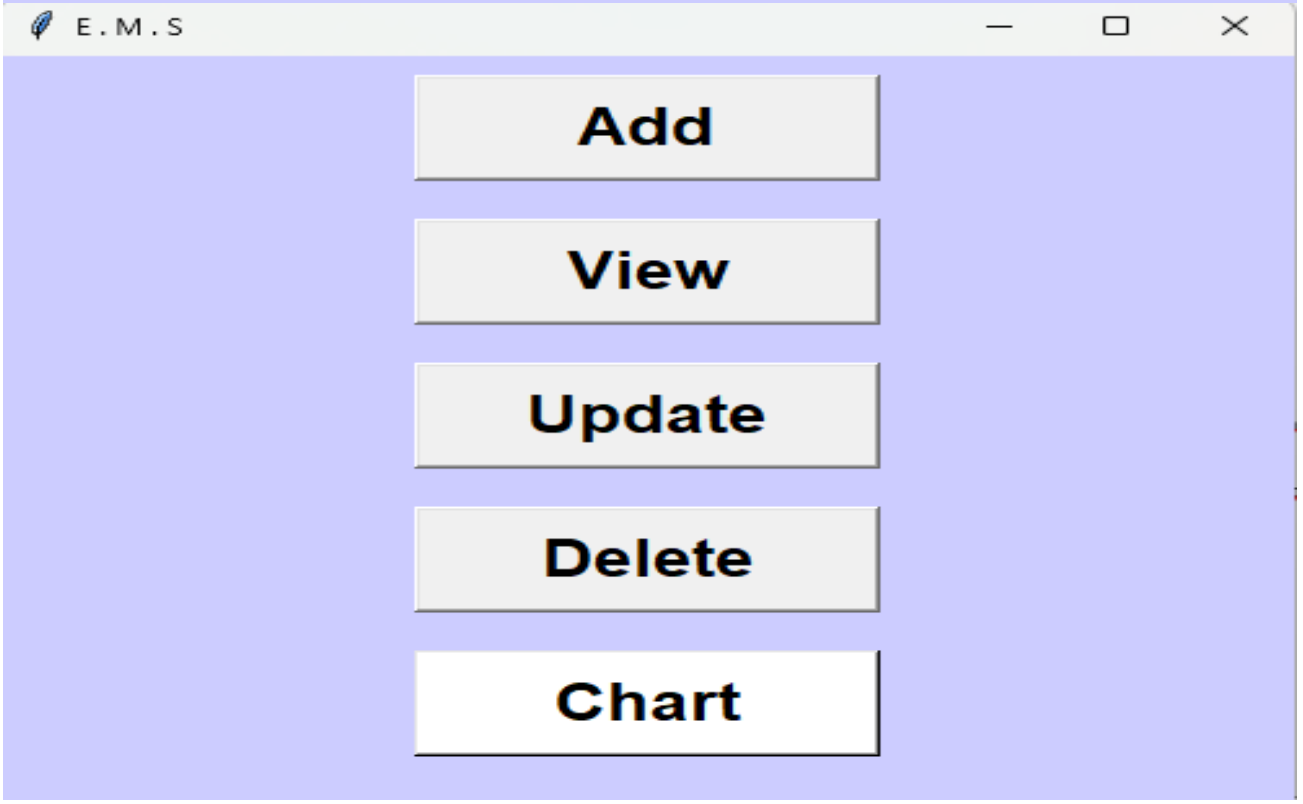
```
                con.close()
def login():
    user = lw_ent_user.get()
    pas = lw_ent_pas.get()
    if user == "student" and pas == "student3714":
        main_window.deiconify()
        lw_ent_pas.delete(0,END)
        login_window.withdraw()
    else:
        showerror("Issue","Incorrect Password")
        lw_ent_pas.delete(0,END)
#-----------------------------------------------------------------------------------------------
f = ("Arial",20,"bold")
fq = ("Arial",15,"bold")
fv = ("Arial",16,"bold")

login_window = Tk()
login_window.title("Admin Login")
login_window.geometry("500x500+400+100")
login_window.configure(bg="#CCCCFF")

lw_lab_user=Label(login_window,text="Enter Admin Name",font=f)
lw_ent_user=Entry(login_window,font=f,bd=2)
lw_lab_user.pack(pady=10)
lw_ent_user.pack(pady=10)



lw_lab_pas=Label(login_window,text="Enter Password",font=f)
lw_ent_pas=Entry(login_window,font=f,bd=2,show= "*")
lw_lab_pas.pack(pady=10)
lw_ent_pas.pack(pady=10)
lw_btn_login = Button(login_window,text = "Log in",font=f,width=10,command=login)
lw_btn_login.pack(pady=10)

#-----------------------------------------------------------------------------------------------
main_window = Toplevel(login_window)
main_window.title("E . M . S")
main_window.geometry("500x500+400+100")
main_window.configure(bg="#CCCCFF")


mw_btn_add = Button(main_window,text = "Add",font=f,width=10,command=f1)
mw_btn_add.pack(pady=10)

mw_btn_view = Button(main_window,text = "View",font=f,width=10,command=f5)
mw_btn_view.pack(pady=10)

mw_btn_update = Button(main_window,text = "Update",font=f,width=10,command=f3)
mw_btn_update.pack(pady=10)

mw_btn_delete = Button(main_window,text = "Delete",font=f,width=10,command=f7)
mw_btn_delete.pack(pady=10)

mw_btn_chart = Button(main_window,text = "Chart",font=f,width=10,bg="white",command=barchart)
```

```
mw_btn_chart.pack(pady=10)

wa = "https://www.brainyquote.com/quote_of_the_day"
res = requests.get(wa)
#print(res)

data = bs4.BeautifulSoup(res.text, "html.parser")
#print(data)

info = data.find("img", {"class":"p-qotd"})
#print(info)

#quote = info["alt"]
#print("Quote = ",quote)
quote_label                                                                                    =
    Label(main_window,text="quote",font=fq,width=45,height=10,fg="white",bg="#7393B3")
quote_label.pack(pady=13)
main_window.withdraw()

#-----------------------------------------------------------------

add_window = Toplevel(main_window)
add_window.title("Add Emp")
add_window.geometry("500x500+400+100")
add_window.configure(bg="#CCCCFF")

aw_lab_id=Label(add_window,text="Enter id",font=f)
aw_ent_id=Entry(add_window,font=f,bd=2)
aw_lab_id.pack(pady=10)
aw_ent_id.pack(pady=10)


aw_lab_name=Label(add_window,text="Enter Name",font=f)
aw_ent_name=Entry(add_window,font=f,bd=2)
aw_lab_name.pack(pady=10)
aw_ent_name.pack(pady=10)

aw_lab_salary=Label(add_window,text="Enter salary",font=f)
aw_ent_salary=Entry(add_window,font=f,bd=2)
aw_lab_salary.pack(pady=10)
aw_ent_salary.pack(pady=10)

aw_btn_save = Button(add_window,text = "Save",font=f,width=10,command=save)
aw_btn_save.pack(pady=10)

aw_btn_back = Button(add_window,text = "Back",font=f,width=10,command=f2)
aw_btn_back.pack(pady=10)
add_window.withdraw()

#-------------------------------------------------------------

view_window = Toplevel(main_window)
view_window.title("View Emp")
view_window.geometry("500x500+400+100")
view_window.configure(bg="#CCCCFF")
```

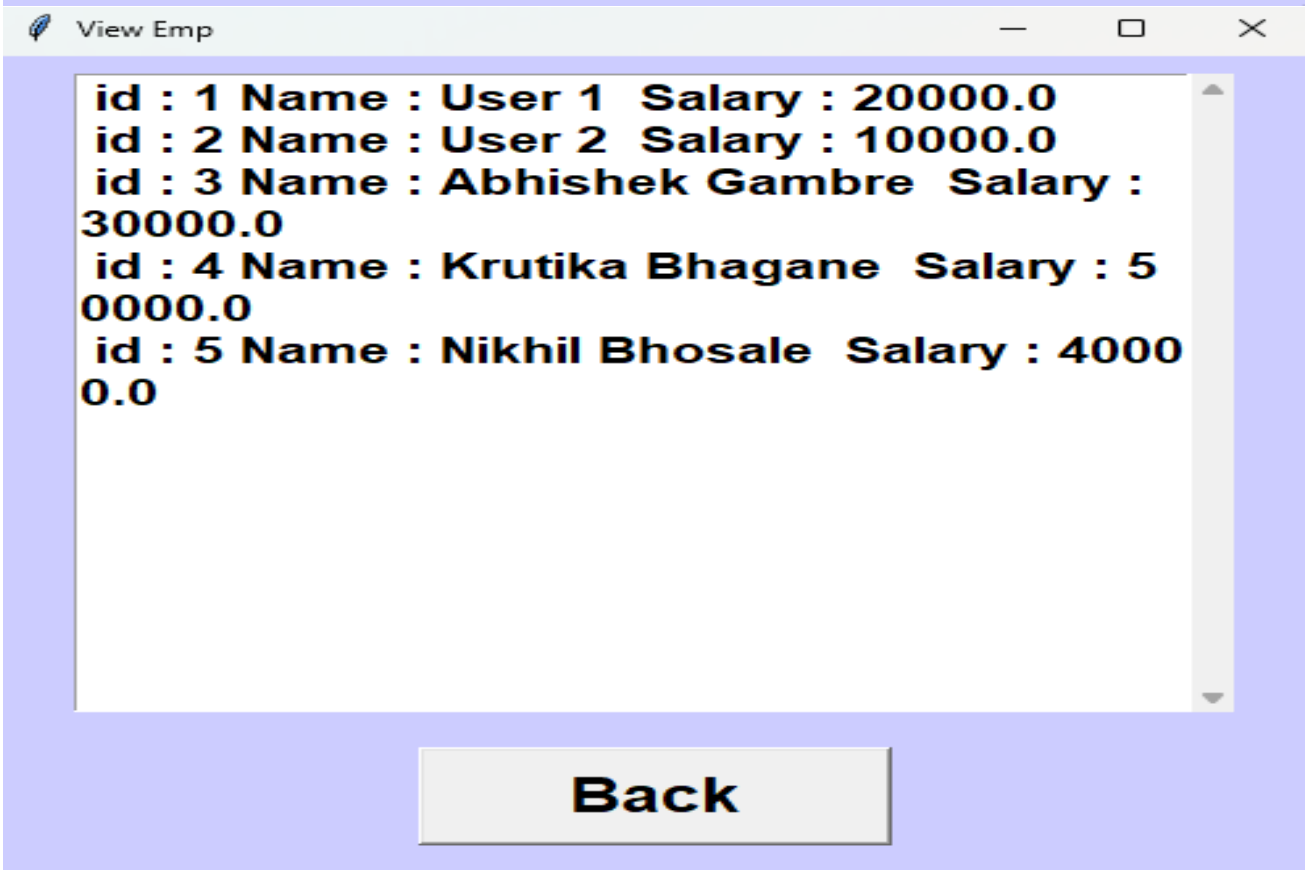
```
vw_emp_data = ScrolledText(view_window,width=35,height=15,font=fv)
vw_emp_data.pack(pady=10)

vw_btn_back = Button(view_window,text = "Back",font=f,width=10,command=f6)
vw_btn_back.pack(pady=10)
view_window.withdraw()
#------------------------------------------------------------
update_window = Toplevel(main_window)
update_window.title("Update Emp")
update_window.geometry("500x500+400+100")
update_window.configure(bg="#CCCCFF")

uw_lab_id=Label(update_window,text="Enter id",font=f)
uw_ent_id=Entry(update_window,font=f,bd=2)
uw_lab_id.pack(pady=10)
uw_ent_id.pack(pady=10)

uw_lab_name=Label(update_window,text="Enter Name",font=f)
uw_ent_name=Entry(update_window,font=f,bd=2)
uw_lab_name.pack(pady=10)
uw_ent_name.pack(pady=10)

uw_lab_salary=Label(update_window,text="Enter salary",font=f)
uw_ent_salary=Entry(update_window,font=f,bd=2)
uw_lab_salary.pack(pady=10)
uw_ent_salary.pack(pady=10)

uw_btn_chart = Button(update_window,text = "Save",font=f,width=10,command=update)
uw_btn_chart.pack(pady=10)

uw_btn_chart = Button(update_window,text = "Back",font=f,width=10,command=f4)
uw_btn_chart.pack(pady=10)
update_window.withdraw()
#------------------------------------------------------------
delete_window = Toplevel(main_window)
delete_window.title("delete Emp")
delete_window.geometry("500x500+400+100")
delete_window.configure(bg="#CCCCFF")

dw_lab_id=Label(delete_window,text="Enter id",font=f)
dw_ent_id=Entry(delete_window,font=f,bd=2)
dw_lab_id.pack(pady=10)
dw_ent_id.pack(pady=10)

dw_btn_chart = Button(delete_window,text = "Delete",font=f,width=10,command=delete)
dw_btn_chart.pack(pady=10)

dw_btn_chart = Button(delete_window,text = "Back",font=f,width=10,command=f8)
dw_btn_chart.pack(pady=10)
delete_window.withdraw()

main_window.mainloop()
```

Output:-


Admin Login
Enter Admin Name
Admin
Enter Password
*****
Log in


E.M.S
Add
View
Update
Delete
Chart


Add Emp
Enter id
1
Enter Name
User Name
Enter salary
10000
Save
Back


View Emp
id : 1 Name : User 1 Salary : 20000.0
id : 2 Name : User 2 Salary : 10000.0
id : 3 Name : Abhishek Gambre Salary : 30000.0
id : 4 Name : Krutika Bhagane Salary : 50000.0
id : 5 Name : Nikhil Bhosale Salary : 40000.0
Back

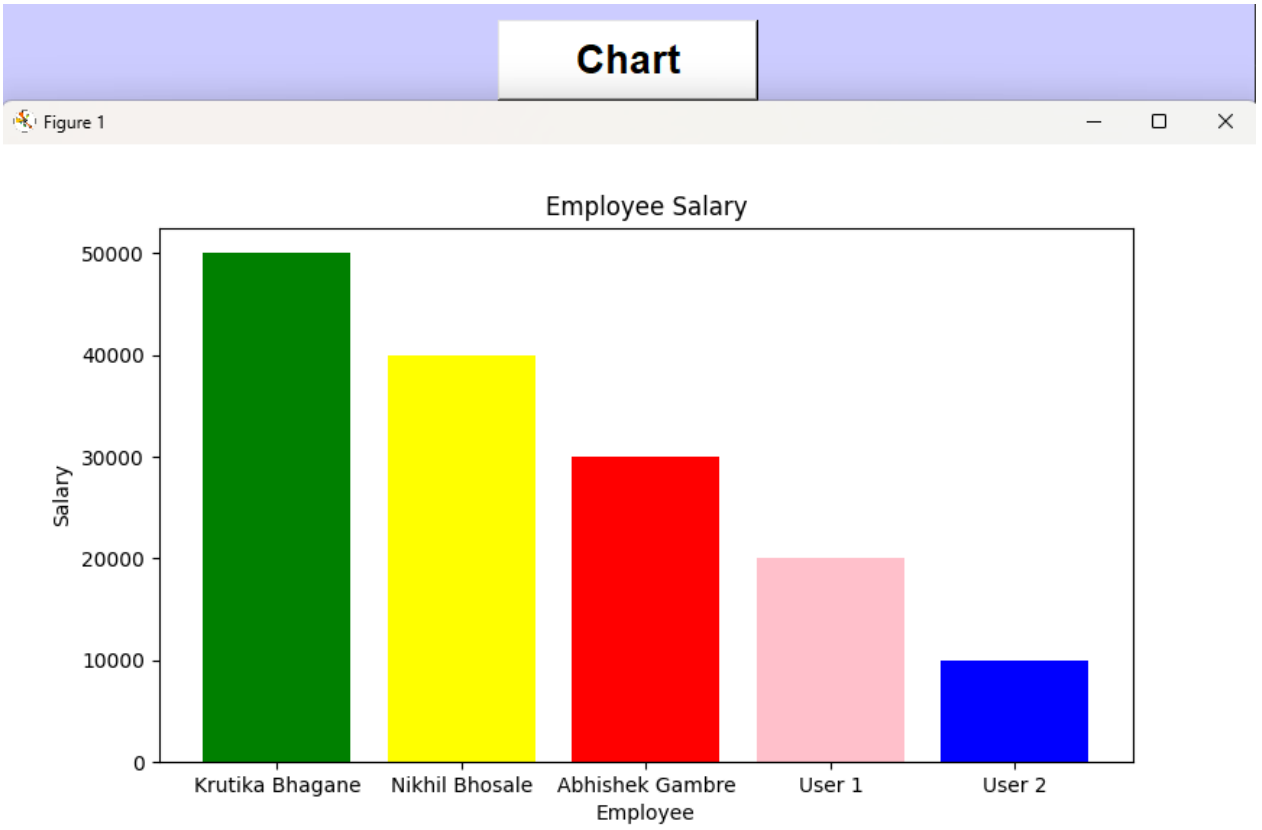


Chart
Figure 1
Employee Salary
50000
40000
30000
20000
10000
0
Salary
Krutika Bhagane
Nikhil Bhosale
Abhishek Gambre
User 1
User 2
Employee

4. Conclusion

In this report, an information system's development has been presented. It was emphasized on the basic steps, consequently taken during the project's development course as a particular attention was turned to the basic operative functions performed upon the data into the database. The report's content comprises the whole task solution, starting from the programming environments have been selected, going through the data analysis, the program analyze and construction, and finishing with the code-implementation and test-samples. As a future work, some additional stuff could be implemented and integrated into the application code making it much more reliable and flexible; especially what concerns a pay-roll module, for instance.

Apparently, the role of such systems is basic and essential within each company that wants to keep a really good control and record concerning its personnel data, functionality and performance on all levels in its structure. Every organization, in nowadays, has the necessity of managing its staff on a really good level as the staff has definitely the greatest merit of building up a company as such as it is. The well managed staff means giving the appropriate financial award-ness and all kind of benefits as such as they have been deserved. That's why the development of such systems is not just a programming business – a lot of people are ordinarily involved in such projects and one of the basic requirements is the reliability of the system, especially what concerns the storage of data and all of the operations that will be performed upon it.

5. References

- Anscombe, F. (1973), Graphs in Statistical Analysis, *The American Statistician*, pp. 195-199.
- What Is Naive Bayes Algorithm in Machine Learning? - Rohit Dwivedi
- Anscombe, F. and Tukey, J. W. (1963), The Examination and Analysis of Residuals, *Technometrics*, pp. 141-160.
- Filliben, J. J. (February 1975), The Probability Plot Correlation Coefficient Test for Normality, *Technometrics*, pp. 111-117.