A

Project Based Learning Report

on

"GENERATE SALARY SLIP OF EMPLOYEES"

By

MR.GAWANDE SAHIL S	**349**	**F190100187**
MR.GHUGE HRUSHIKESH D	**358**	**F190100196**
MR.GHUGE RUSHIKESH U	**360**	**F190100198**
MR.GHUGE TEJAS B	**361**	**F190100199**
MR.GHULE SAIRAJ S	**363**	**F190100201**

Under the guidance
of**prof. s.b.Chaubari**

Department of First Year
Engineering,Amrutvahini College of
Engineering, Sangamner-422608

2021-2022

CERTIFICATE

Amrutvahini Sheti & ShikshanVikas Sanstha's

Amrutvahini College of Engineering, Sangamner

This is to certify that

MR.GAWANDE SAHIL S	**349**	**F190100187**
MR.GHUGE HRUSHIKESH D	**358**	**F190100196**
MR.GHUGE RUSHIKESH U	**360**	**F190100198**
MR.GHUGE TEJAS B	**361**	**F190100199**
MR.GHULE SAIRAJ S	**363**	**F190100201**

have completed a project on

"Generate Salary Slip Of Employees"

under the guidance of'

Prof. S. B. CHAUHARI

as a part of

Project Based Learning during academic year 2021-2022

Prof. S.B CHAUDHARI	**Prof.**	**Prof.**
Mentor	**Co- Mentor (If any)**	**Examiner**

ACKNOWLEDGEMENT

It gives us immense pleasure in bringing out the Project entitled **‘GENERATE SALARY SLIP OF EMPLOYEE’**. We express our deep sense of gratitude and sincere regards to our project mentor **Prof.** Chaudhri S.B giving his valuable supervision, cooperation and devotion of time that has given to our Project work. We are grateful to F.E. coordinator **Prof. Dr. V. Y. Patil** for his facilities extended during projectwork and for his personal interest and inspiration.

We wish to express profound thanks to **Prof M. A. Venkatesh**, Principal Amrutvahini College of Engineering, for providing necessary facilities to make this projectsuccessful. Finally, we like to thanks all those who directly or indirectly helped us during the work. We also owe our sincere thanks to all faculty members of Mechanical Departmentwho have always extended a helping hand.

MR.GAWANDE SAHIL S	[F190100187]
MR.GHUGE HRUSHIKESH D	[F190100196]
MR.GHUGE RUSHIKESH U	[F190100198]
MR.GHUGE TEJAS B	[F190100199]
MR.GHULE SAIRAJ S	[F190100201]

Content

ABSTRACT

"Salary slip of employees in an organization" is designed to make the existing manual system automatic with the help of computerised 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. This web application can maintain and view computerised records without getting redundant entries. The project describes how to manage user data for good performance and provide better services for the client.

1. INTRODUCTION

The proposed project “Employee Payslip Generator” has been Developed to overcome the problems faced in the practicing of manual system. This software 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 web application is reduced 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 Employee and payroll Management System that are adapted To the organization’s Managerial Requirements.

1.1 Project Outline

The organization we are working for, is dealing with payroll management of companies. The task is to build a salary management system for this organization. Current salary system is manual therefore the organization wants to switch To an automated computerized salary management system. After building. This system we have to integrate it with the existing computerized system. Employee salary management system is a web application, enabling the Organization to handle salaries of employees of company. This system should be capable enough to calculate the salaries of employee. Moreover, the system should be able to calculate tax deductions of every Employee.

1.2 Motivation

All calculations such as employee salary, employee tax, Organization tax calculations etc, are being done manually at the moment Which is a time consuming task. Hence, a system is required that can perform All above said operations automatically. Moreover, the system should be user Friendly, flexible, fast and highly secure.

1.3 Problem Identification

The organization is using a manual system to calculate salaries and tax of Employees. They also have to calculate and keep record of of whole organization manually.this work requires a lot of Paper work, is extremely time consuming job, and accordingly costly as well, As they have to hire more man power. Since there is always a risk of human errors present in a manual system so the chances of errors are very high and to figure out such errors is also a very lengthy procedure. Therefore, the organization decided to switch from a manual system to an automated computerized salary management system. The requirement of the organization is to develop a web based application That is able to deal with salary calculation of employees within the Organization and maintain its data base. Furthermore this system should be Able to generate automatic files such as, salary slips. In response of queries from the data stored in the database.

2. COMPARISON

In this chapter we will compare a system of our own choice with different Salary calculation softwares available in the market. On the basis of this comparison we will describe the motivation of developing customized solution.

2.1 Why Customized Solution

There are a variety of software tools available in the market that are able to calculate the salaries of employees. Now the question arises, why is it required to build a customized software?

2.2 Softwares Available in the Market

There are many softwares available in the market that are being used by the organizations to calculate the salaries of its employees. We are discussing software here for example.

2.3 Analysis Salary Calculator

- Salary Calculation Software

This software is able to do calculations of employee salaries on the basis of data provided by the work tracking system.

The software has the following features :

- “Planning Project”

- “Keeping track of working results and helping managers to define work results”

- “Helps to Calculate salary from work results”

2.4 Employee Salary Management System

The System we are going to develop is according to the user requirements which will perform salary calculation. Moreover, the system will be user friendly and flexible enough to be enhanced according to the needs of the users in future.

3. LITERATURE REVIEW

➢ Performance Evaluation Of The Virtualization Environment Of A Microservices-Based Payroll System

Castro et, Klayton, Polyane Wercelens focused on A shift for a modern approach requires a secure and modular pattern. The micro services architecture running on top of containers\re-executed by VMs has many layers involved. Perhaps, it could also bring some performance issues.

Lack of co-scheduling between guest OS and the hypervisor is a significant research\problem that has been studied by many researchers. The probability of interference between workloads in unbalanced systems increases with the use of containers, increasing the need for v-CPUs to be triggered synchronously and programmatically. Their goal is to evaluate the impacts of architectural choices on a private cloud infrastructuring. They simulated an intense workload due to the payslip generation step. The system under test is the VMware ESXi1, with K8s3 worker nodes running on it.

“We examined the implications to do a significant innovation involving a Public Agency involving its Payroll System and using. private cloud infrastructure. Considering that containerization and MSA made that much more comfortable, we believe that is the deployment format for the future”.

➢ An Analysis of China’s Physician Salary Payment System

Ran ,Focused on An Analysis of China’s Physician Salary Payment System. Physician payment system(PPS) Is a principal incentive system to motivate doctors to provide excellent care for patients. During the past decade, physician remuneration in China has not been in proportional to physician’s average work load and massive responsibilities. This paper reviewed the constitution of the PPS in China, and further discussed the problems and issues to be addressed with respect to pay for performance.

Author's study indicated that the lower basic salary and bonus distribution tied to “profits” was the major contributor to the physician’s profit-driven incentive and the potential cause for the speedy growth of health expenditures. Author recommend that government funding to hospitals should be increased to fully cover physicians’ basic salary, a flexible human resource and talent management mechanism needs to be established that severs personal interest between physicians and hospitals, and modern performance assessment and multiplexed payment systems should be piloted to encourage physicians to get the more legitimate compensation.

➢ A study of Taiwan’s travel agent salary system: an agency theory perspective

Klayton Castro and Lucas M. C. e Martins [2020ڨ] focused on the Performance Evaluation of the Vertualization Environment of a Microservices-Based PayrollSystem. For better usage of idle resources in a symmetric multiprocessing environment, cloud computing providers often exploit the boundaries of parallelism by imposing high CPU subscription rates over their virtualization systems. Moreover, unsuitable resource allocation can significantly impair the per-formance during intensive work loads and increase infrastructure expenditures unnecessarily. It becomes an increasing challenge when dealing with microservices architecture and container encapsulated applications, for these are approaches that add more intricacy layers to the work load scheduling over the hardware.

This paper presents a case study on a public agency private cloud, which aims to evaluate the performance of hypervisors and virtual machines, that are serving a Kubernetes container or chestration cluster and running a microservices-based Payroll System during CPU bound behaviour tasks. Author also propose a model for sizing the virtual environment according to the best choices for the hardware characteristics and the workload needs. Index Terms-payroll system, microservices, kubernetes virtualization, performance evaluation.

➢ Performance, seniority, and wages: formal salary systems and individual earnings profiles

It demonstrates that formal salary systems cause serial correlation in wage growth and “ Green Card’’ effects.

Model (1)- demonstrates that higher education levels, firm-specific experience, and

pre- company experience raise wages for both types of workers

Model (2)-Regression model

Model (3)-mimics model

Such a salary system with a deferred compensation component is likely to encourage workers to make productivity enhancing investments that are difficult to verify as it creates confidence among workers that they will be rewarded for improving their productivity, and it is likely to reduce turnover. This helped us to design proper Or formal salary system

➢ NFA Based Formal Verification of Automatic Payroll Processing System

Aniqa Rehman [2016] focused on the basic objective of 'Automatic payroll processing system (APPS) by formal method. With the passage of time the needs of an organization grow, therefore, the operational work and responsibilities are also grown. Hence there is a need of hiring more employees in an organization. Calculating salaries of employees requires a lot of work and chances of occurring errors increase. Automated payroll processing system (APPS) processes large amounts of salaries. APPS system based on an NFA model that captures system behaviour in a very efficient manner. Formal methods that are mathematical statements provides a mechanism for validation and verification of systems. Formal verification of APPS provided by VDM-SL that ensure the correctness of a system and detail analysis of a system. Types, invariants, operations, functions and all pre and post conditions are evaluated through VDM-SL.

In this paper, we examined that they present Automatic Payroll Processing System (APPS) that processes large amounts of salaries and reduces the numbers of errors that occurs in manual calculation. It saves the time and increase productivity by providing fast access to all records.

4. REQUIREMENT ANALYSIS

In this chapter, we are going to discuss the requirements, design of the system and analyze them using some appropriate software models.

4.1 Methodology

The most suitable and appropriate software development methodology we found for our software development is waterfall software development model, which gave us a clear view about our software and helped us to achieve our goal.

4.2 The SDLC Waterfall

The small and medium size software are usually divided into six stages that are interrelated with each other in a top-down approach named as waterfall. The input of one specific stage is the output of previous stage which initializes the next stage. At every step or stage of the model some additional information is also added up into the input of that stage and generating the results. But that additional information is restricted in scope and previous stages are directly traceable from that stage.

The first stage of the waterfall model is planning stage in which the most critical task is to clearly mention the high level requirements or goals of the software. It helps to make feasibility and risks associated with the project and also provides the basic project structure.

The next stage Is the requirement definition stage that takes goals as input from the planning stage and then these goals are defined into a set of one or more requirements. Major functionalities, initial data entities and the operational data areas are defined under this stage. The Input of design stage is the out put of the requirement stage which is an approved requirements documentation. In this stage the design elements are defined with the help of interviews, prototypes and workshops conducted. These design elements consist of functional hierarchy , business process diagrams, pseudo code and entity-relationship diagram. These design elements provide detail description about the software and each element is related to a specific requirement.

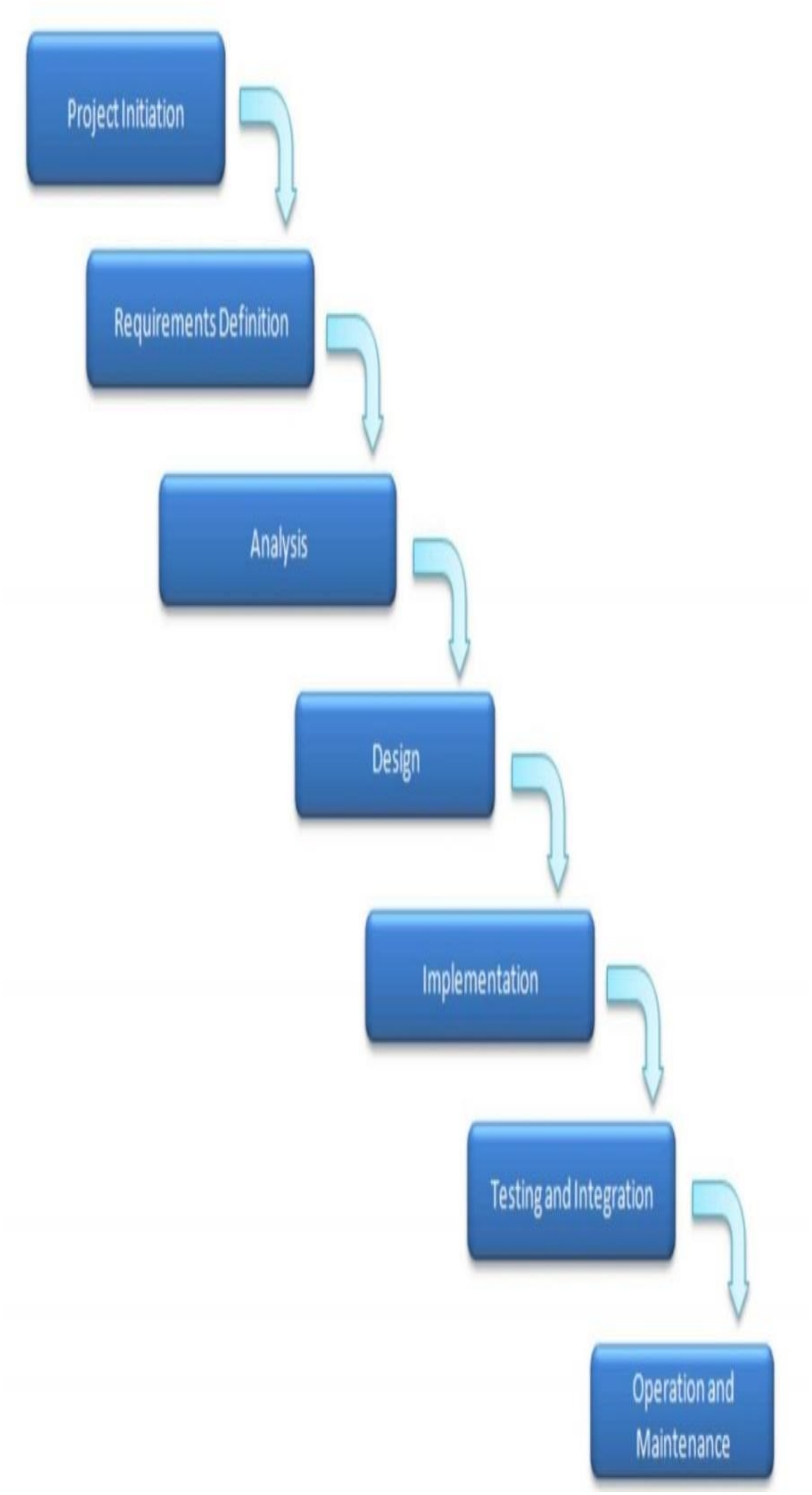
The development stage is initialized by the previous design stage. At this stage the code for the design elements of software is written and it provides the functional software components.

Integration and test stage is provided the output of the development stage. At this stage the different software functioning components developed in the previous stage are integrated with each other to provide full fledge software project which is providing all the high-level requirements. This software is also passed through different test cases to check the validity, correctness, completeness and hence enables us to achieve our goals regarding our project.

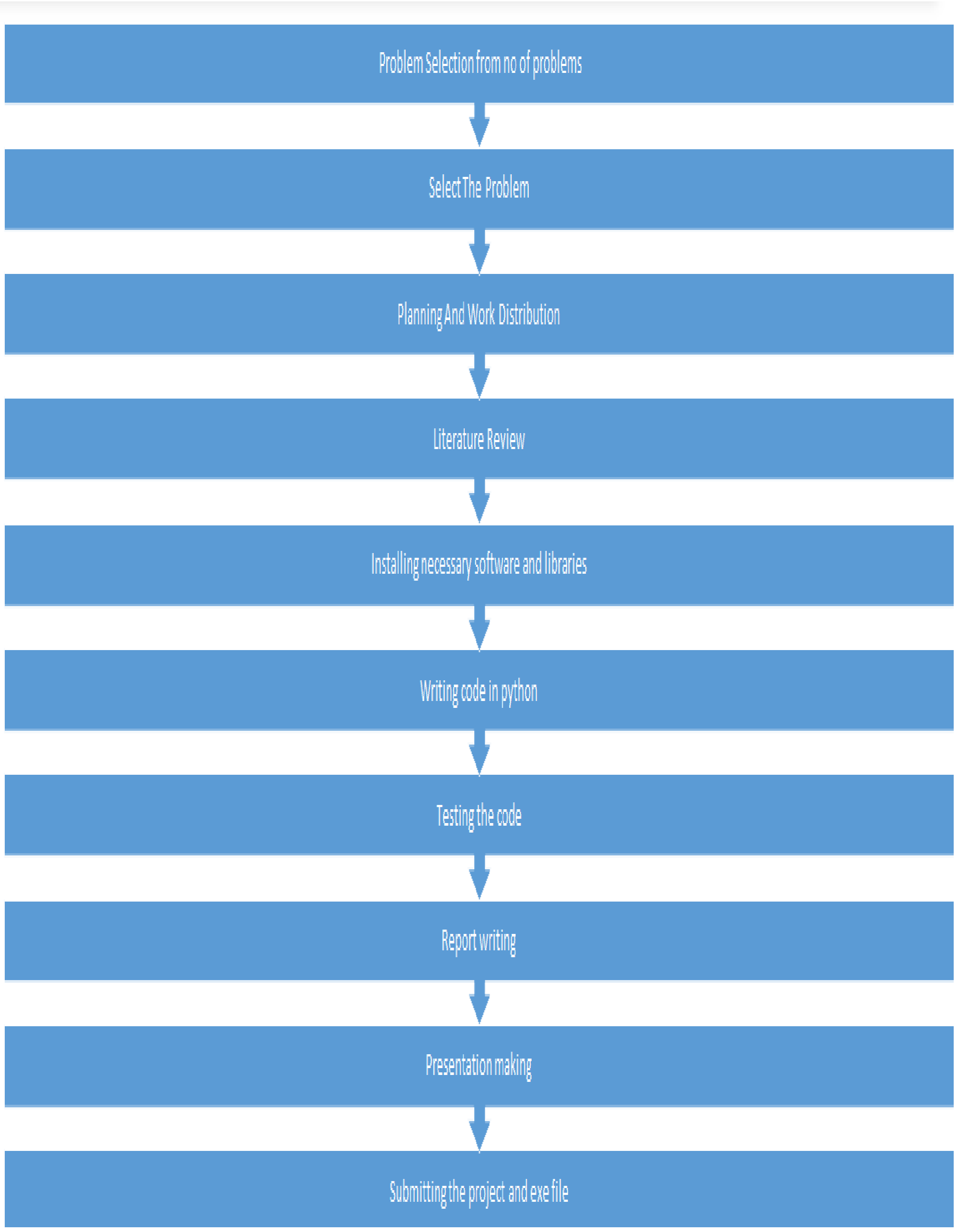
Finally, the installation and acceptance stage arrives. The software is loaded to the server at site of customers and tested with different test cases once again to check the correct working. If all this shows adequate results satisfying the customer then the software is handed over to the customer Formally

4.3 Reasoning to Choose Waterfall Model

As in this model all the phases are in a sequence and are dependent with one another , therefore a phase can not be started until the previous phase is completed and fully documented. This approach is most appropriate for our project, because all the requirements and goals of our project are very clear. Secondly, it is easy to do work in components and waterfall model is providing this approach. After the completion of all the phases individually, they are integrated together. This model is very economical and risk free due to its sequential approach


Project Initiation
Requirements Definition
Analysis
Design
Implementation
Testing and Integration
Operation and Maintenance

4.4 CHART OF METHODOLOGY


Problem Selection from no of problems
Select The Problem
Planning And Work Distribution
Literature Review
Installing necessary software and libraries
Writing code in python
Testing the code
Report writing
Presentation making
Submitting the project and exe file

4.5 SYSTEM REQUIREMENTS :

4.5.1 Available Technologies:

Languages: python

RDBMS: Online MySQL

Web Server: WAMP

4.5.2 Tools Used:

Editor Used: Spyder

Operating System: Windows 10

4.5.3 Hardware Used:

Processor: Intel core i5

RAM: 8GB

Hard Disk: 1TB

5. ACTUAL CODE

5.1 CODE 1:-

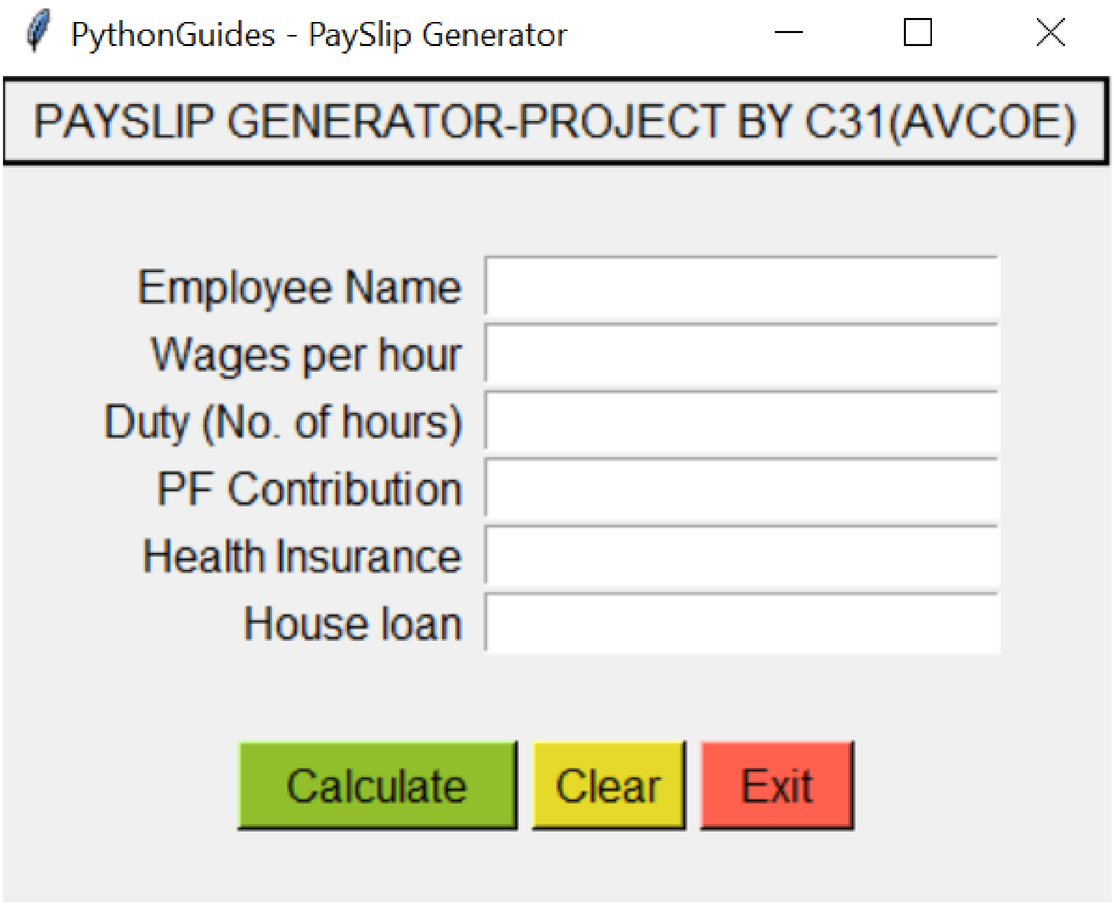
```
# modules
from tkinter import *
import calsal2

# tkinter object
ws = Tk()
ws.title('PythonGuides - PaySlip Generator')
ws.geometry('400x300')

f = 'sans-serif, 12'

# functions
def clearbtn():
    emp_name.delete(0, 'end')
    rph.delete(0, 'end')
    duty.delete(0, 'end')
    s3_contrib.delete(0, 'end')
    phealth.delete(0, 'end')
    hloan.delete(0, 'end')



def main():
    ename = emp_name.get()
    r = rph.get()
    d = duty.get()
    contri = s3_contrib.get()
    ph = phealth.get()
    hl = hloan.get()

    r = float(r)
    d = float(d)
    contri = float(contri)
    ph = float(ph)
    hl = float(hl)

    gross_sal = r * d
    tax = gross_sal * 0.1
    total_deduction = contri + ph + hl + tax
    net_sal = gross_sal - total_deduction
    # calling function
    calsal2.payslip(ename, d, r, gross_sal, contri, ph, hl, tax, total_deduction, net_sal,
ws, f)

Label(
    ws,
    text="PAYSLIP GENERATOR-PROJECT BY C31(AVCOE)",
    font=('sans-serif, 12'),
    relief=SOLID,
    padx=10,
    pady=5
).pack()

# frame widget
mainframe = Frame(ws, padx=5, pady=5)
mainframe.pack(expand=True)

# label widget
Label(
    mainframe,
    text='Employee Name',
    font=f
    ).grid(row=2, column=0, sticky='e')

Label(
    mainframe,
    text='Wages per hour',
    font=f
    ).grid(row=3, column=0, sticky='e')

Label(
    mainframe,
    text='Duty (No. of hours)',
    font=f
    ).grid(row=4, column=0, sticky='e')

Label(
    mainframe,
    text='PF Contribution',
    font=f
    ).grid(row=5, column=0, sticky='e')

Label(
    mainframe,
    text='Health Insurance',
    font=f
    ).grid(row=6, column=0, sticky='e')

Label(
    mainframe,
    text='House loan',
    font=f
    ).grid(row=7, column=0, sticky='e')


# Entry widgets
emp_name = Entry(mainframe, font=f)
rph = Entry(mainframe, font=f)
duty = Entry(mainframe, font=f)
s3_contrib = Entry(mainframe, font=f)
phealth = Entry(mainframe, font=f)
hloan = Entry(mainframe, font=f)

# geometry method - Grid
emp_name.grid(row=2, column=1, padx=5)
rph.grid(row=3, column=1, padx=5, sticky='w')
duty.grid(row=4, column=1, padx=5, sticky='w')
s3_contrib.grid(row=5, column=1, padx=5, sticky='w')
phealth.grid(row=6, column=1, padx=5, sticky='w')
hloan.grid(row=7, column=1, padx=5, sticky='w')

# default values in the entry widget
emp_name.insert('0',"")
rph.insert('0','')
duty.insert('0','')
s3_contrib.insert('0','')
phealth.insert('0','' )
hloan.insert('0','' )

# frame for buttons
frame = Frame(mainframe)
frame.grid(row=8, columnspan=3, pady=(30, 0))

# button widget
Button(
    frame,
    text='Calculate',
    width=10,
    command=main,
    font=f,
    bg='#91BF2C'
).pack(side=LEFT, expand=True, padx=(0, 5))

Button(
    frame,
    text='Clear',
    width=5,
    font=f,
    bg='#E6D92A',
    command=clearbtn
).pack(side=LEFT, expand=True, padx=(0, 5))

Button(
    frame,
    text='Exit',
    width=5,
    font=f,
    bg='#FF614F',
    command=lambda:ws.destroy()
).pack(side=LEFT, expand=True, padx=(0, 5))

# infinite loop
ws.mainloop()
```

5.2 CODE 2 :-

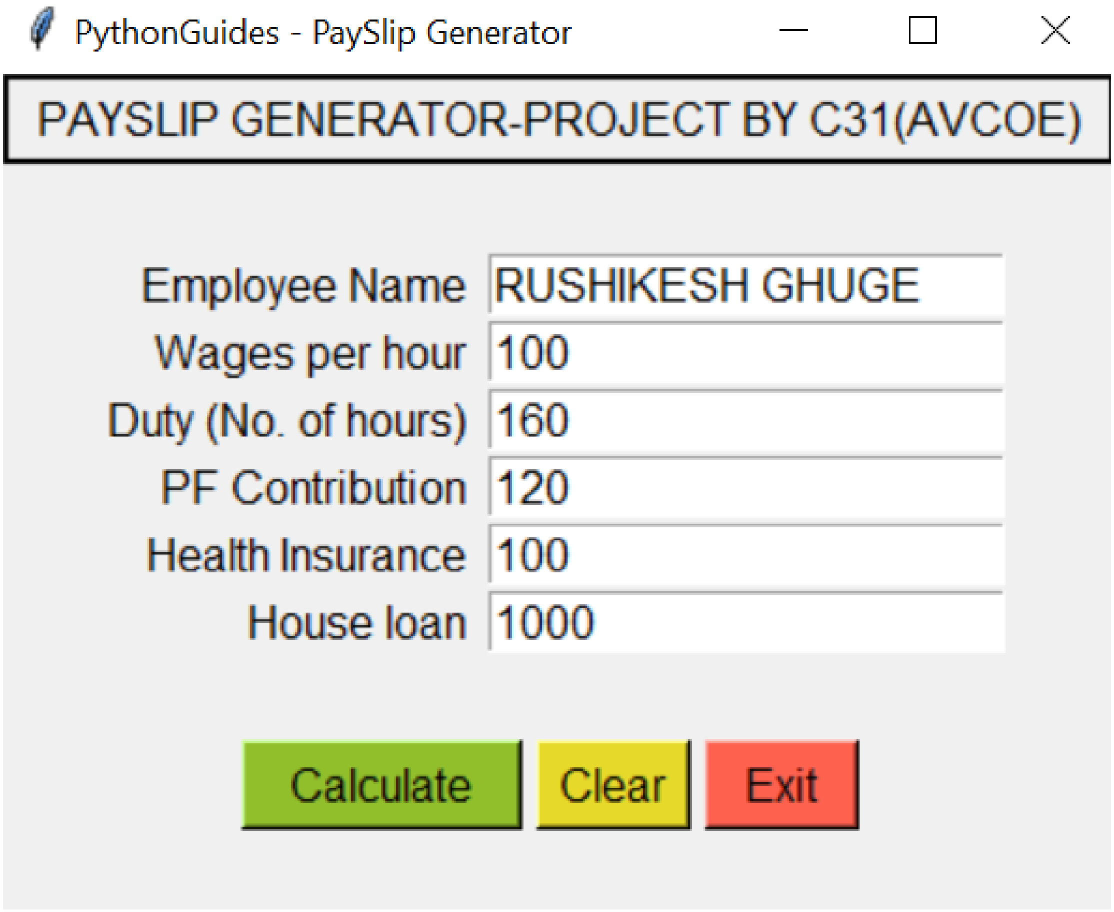
```
# module
from tkinter import *

# function
def payslip(ename, d, r, gross_sal, contri, ph, hl, tax, total_deduction, net_sal, ws, f):
    cf = ('sans-serif 12 bold')
    win = Toplevel(ws)
    win.geometry('500x450+500+200')

    Label(
        win,
        text='EMPLOYEES INFORMATION',
        font='sans-serif 14',
        relief=SOLID, padx=5,
        pady=10
        ).place(x=110 , y=10)

    Label(
        win,
        text='Name: ',
        font=cf
        ).place(x=10, y=70)

    Label(
        win,
        text=f'{ename}',
        font=f
        ).place(x=65, y=70)

    Label(
        win,
        text='Working Hours: ',
        font=cf
        ).place(x=250, y=70)

    Label(
        win,
        text=f'{d}',
        font=f
        ).place(x=390, y=70)

    Label(
        win,
        text='Rate per hour: ',
        font=cf
        ).place(x=10, y=110)

    Label(
        win,
        text=f'{r}',
        font=f
        ).place(x=125, y=110)

    Label(
        win,
        text='Gross Salary: ',
        font=cf
        ).place(x=250, y=110)

    Label(
        win,
        text=f'{gross_sal}',
        font=f
        ).place(x=390, y=110)


    Label(
        win,
        text='DEDUCTIONS',
        font='sans-serif 14', relief=SOLID,
        pady=5, padx=10
        ).place(x=170, y=180)


    Label(
        win,
        text='PF: ',
        font=cf
        ).place(x=10, y=240)

    Label(
        win,
        text=f'{contri}',
        font=f
        ).place(x=65, y=240)

    Label(
        win,
        text='Health Insurance: ',
        font=cf
        ).place(x=250, y=240)

    Label(
        win,
        text=f'{ph}',
        font=f
        ).place(x=390, y=240)

    Label(
        win,
        text='Loans:',
        font=cf
        ).place(x=10, y=280)

    Label(
        win,
        text=f'{hl}',
        font=f
        ).place(x=105, y=280)

    Label(
        win,
        text='Tax: ',
        font=cf
        ).place(x=250, y=280)

    Label(
        win,
        text=f'{tax}',
        font=f
        ).place(x=390, y=280)

    Label(
        win,
        text='Total Deductions: ',
        font=cf
        ).place(x=10, y=320)

    Label(
        win,
        text=f'{total_deduction}',
        font=f
        ).place(x=150, y=320)

    Label(
        win,
        text='Net Salary:',
        font=cf
        ).place(x=250, y=320)

    Label(
        win,
        text=f'{net_sal}',
        font=f
        ).place(x=390, y=320)

    Button(
        win,
        text='Close',
        padx=10,
        pady=5,
        font=f,
        bg='#FF614F',
        command=lambda:win.destroy()
        ).place(x=220, y=390)
```

6. IMPLEMENTATION & RESULT

6.1 OUTPUT 1-



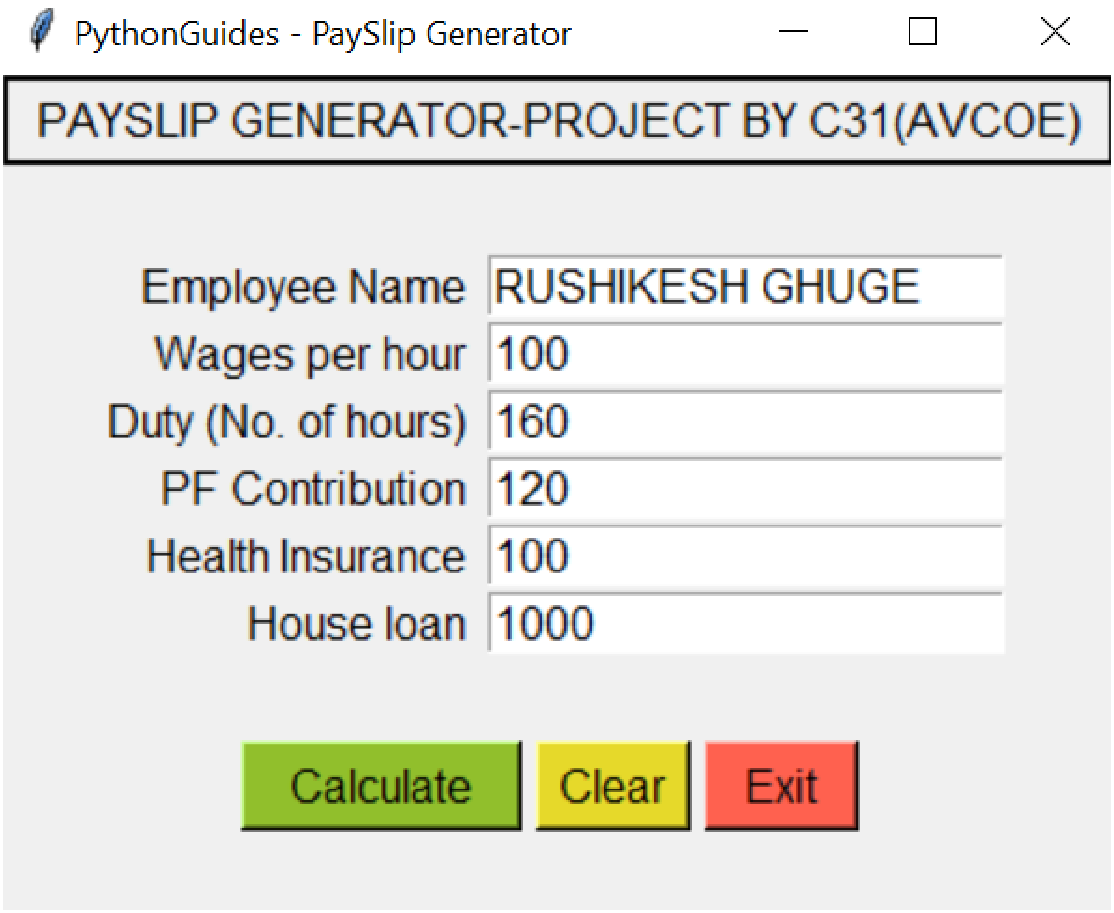
6.2 FILLING DATA IN OUTPUT 1:-



PythonGuides - PaySlip Generator
PAYSLIP GENERATOR-PROJECT BY C31(AVCOE)
Calculate
Clear
Exit

Employee Name RUSHIKESH GHUGE
Wages per hour 100
Duty (No. of hours) 160
PF Contribution 120
Health Insurance 100
House loan 1000

6.3 OUTPUT 2-



PythonGuides - PaySlip Generator

PAYSLIP GENERATOR-PROJECT BY C31(AVCOE)

Employee Name RUSHIKESH GHUGE

Wages per hour 100

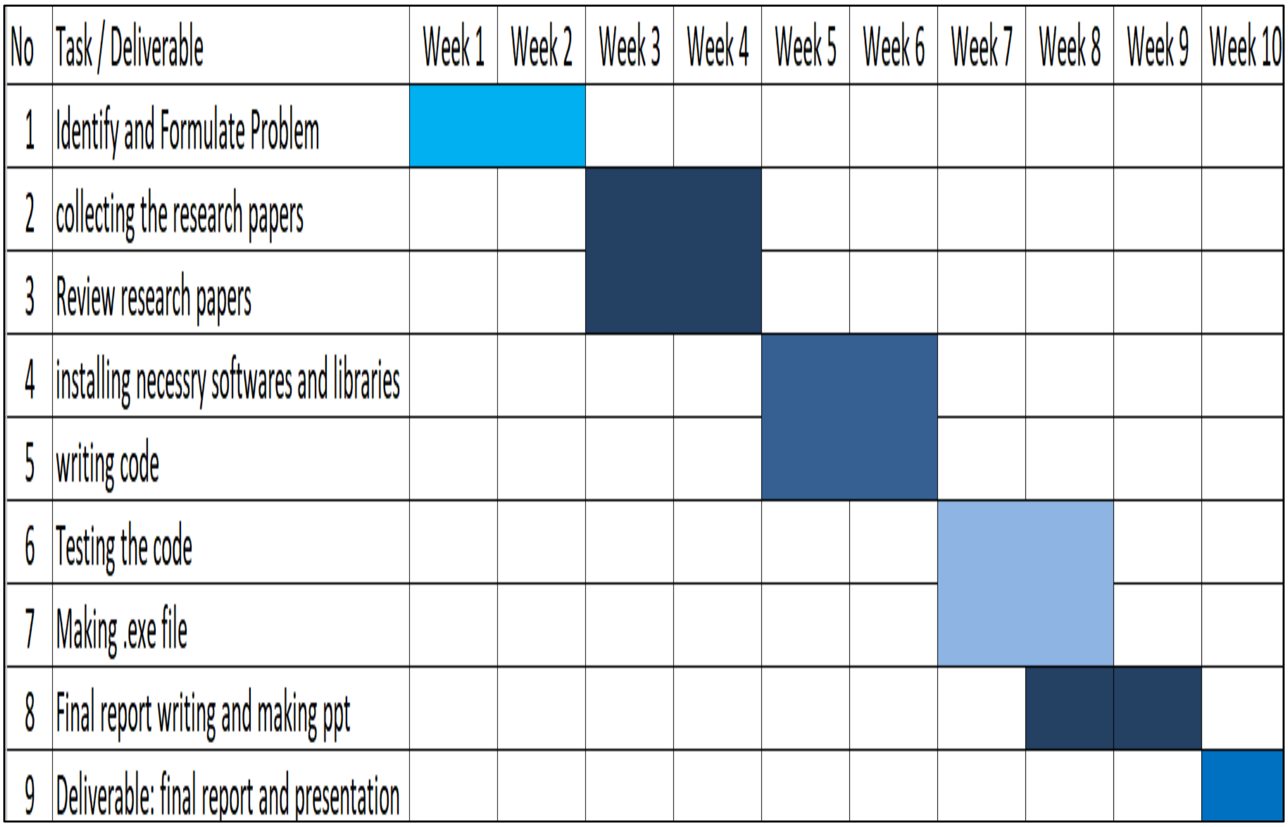
Duty (No. of hours) 160

PF Contribution 120

Health Insurance 100

House loan 1000

7. Gantt Chart



No	Task / Deliverable	Week 1	Week 2	Week 3	Week 4	Week 5	Week 6	Week 7	Week 8	Week 9	Week 10
1	Identify and Formulate Problem	■	■								
2	collecting the research papers			■	■						
3	Review research papers			■	■						
4	installing necessry softwares and libraries					■	■				
5	writing code					■	■				
6	Testing the code							■	■		
7	Making .exe file							■	■		
8	Final report writing and making ppt								■	■	
9	Deliverable: final report and presentation										■

8. ADVANTAGES

- Easy to use.
- It is completely secure.
- This system is easily compatible with most of the web browsers.
- It is very interactive and saves time.
- Reduces paper works.
- Calculations are automated so it is highly accurate.

9. CONCLUSION

We are trying to make this project more accurate in future ,

- Our first objective Generate a salary slip by using python programming is completed.
- The project is useful to reduce the efforts of the person.
- It is very interactive and time saving.
- It helps in growth of company/industry.

FUTURE SCOPE

I. The option to print the records In future.

II. print the record.

III. I would like to implement a regular backup mechanism to back up the employee database to avoid disasters.

IV. The system can be developed in such a way that its existing features can be modified to better versions.

10.REFERENCES

1. Castro et, Klayton, Polyane Wercelens, Rafael Padilha, Italo Gervasio, Flávio EG de Deus, William F. Giozza, and Rafael T. de Sousa. "Performance evaluation of the virtualization environment of a microservices-based payroll system." In *2020 15th Iberian Conference on Information Systems and Technologies (CISTI)*, pp. 1-6. IEEE, 2020.
2. Ran et, Li-mei, Kai-jian Luo, Yun-cheng Wu, Lan Yao, and You-mei Feng. "An analysis of China’s physician salary payment system." *Journal of Huazhong University of Science and Technology [Medical Sciences]* 33, no. 2 (2013): 309-314.
3. Huang et, Chung, Pei-Wen Huang, and Yi-Jung Chen. "A study of Taiwan's travel agent salary system: an agency theory perspective." *Tourism Management* 25, no. 4 (2004): 483-490.
4. Dohmen et, Thomas J. "Performance, seniority, and wages: formal salary systems and individual earnings profiles." *Labour Economics* 11, no. 6 (2004): 741-763.
5. Rehman et, Aniqa, and Nazir Ahmad Zafar. "NFA based formal verification of automatic payroll processing system." In *2016 International Conference on Emerging Technologies (ICET)*, pp. 1-6. IEEE, 2016.