

UNIVERSIDAD PERUANA DE CIENCIAS APLICADAS

ESTUDIOS PROFESIONALES PARA EJECUTIVOS



Trabajo Final
Fundamentos de Programación
2021-01

“Sistema para cálculo de planilla”

Presentada por:

Pumantico Bazo, Jose Antonio
Quispe Mayta, Christian Alexander
Salazar Reque, Juan Miguel
Vidal Chumbiriza, Giber Mario

LIMA – PERÚ

Año 2021





Contenido

A. Situación Actual	4
B. Propuesta de innovación.....	4
1. Detalle del nuevo proceso	4
2. Algoritmo Propuesto	5
3. Herramienta y Tecnología por usar	11
4. Gestión del proyecto. Product Backlog en Trello	11
C. Programación	13
1. Control de versiones en GIT	13
D. Conclusiones y recomendaciones	16
E. Anexos	16



A. Situación Actual

La empresa White Lion Nuts S.A.C. actualmente lleva el cálculo de su planilla de forma manual, la cual consiste en que los trabajadores marcan su ingreso, salida a refrigerio, entrada a refrigerio y salida en un marcador manual o como también se le conoce “marcador de tarjeta”; este marcador genera cartillas impresas las cuales son guardadas por el auxiliar de marcaciones y posteriormente son revisadas manualmente en un cuaderno de cálculo para así elaborar los respectivos cálculos de :

- Inasistencias
- Tardanzas
- Descuentos
- Sueldo final

B. Propuesta de innovación

1. Detalle del nuevo proceso

Se necesita automatizar el proceso del cálculo de las planillas, para lo cual se requiere lo siguiente:

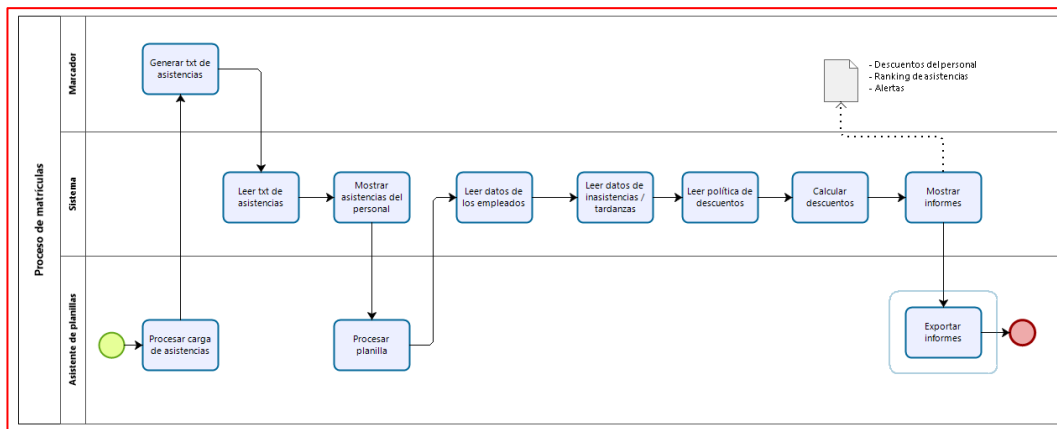
- Leer mediante el software el archivo de texto en el cual se registran las planillas.
- Obtener los datos del personal , política de asistencias , inasistencias y tardanzas
- Realizar el cálculo de los sueldos y los descuentos
- Mostrar informes :
 - Listar a los empleados indicando su sueldo , monto descontado , total de días con inasistencias , total de horas de tardanza , monto a pagar y una alerta indicando si el personal debe ser cesado
 - Ranking de tardanzas / faltas por área

El sistema deberá simular las asistencias de un mes x.



Figura 1

Flujo de procesos para el sistema propuesto



2. Algoritmo Propuesto

i. Leer Datos de los empleados

El sistema leerá un archivo de texto externo, para luego devolver una matriz con los datos del personal.

- Datos de entrada : Archivo de texto plano, cantidad de empleados
- Datos de salida : Matriz bidimensional con los datos del personal (dni, nombres y área)

Figura 2

Diagrama de flujo para el algoritmo: "Leer datos de los empleados"

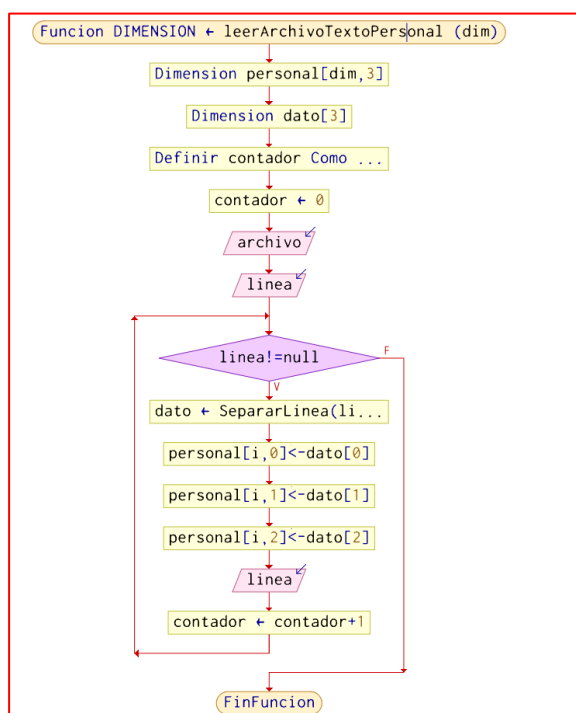




Figura 3

Pseudocódigo para el algoritmo: “Leer datos de los empleados”

```
Funcion Dimension <- leerArchivoTextoPersonal (dim)
  Dimension personal[dim,3]
  Dimension dato[3]
  Definir contador Como Entero

  contador ← 0
  Leer archivo
  Leer linea;

  Mientras linea != null Hacer
    dato ← SepararLinea(linea)
    personal[i,0] ← dato[0]
    personal[i,1] ← dato[1]
    personal[i,2] ← dato[2]

    Leer linea;
    contador←contador+1
  FinMientras
FinFuncion
```



ii. Simular ingresos del personal

El sistema simulará los ingresos y salidas del personal.

- Datos de entrada : Matriz bidimensional con los datos del personal (dni , nombres y área)
- Datos de salida : Matriz bidimensional con los datos de las asistencias (dni, fecha, flag_asistió,hora_ingreso,hora_salida)

Figura 4

Diagrama de flujo para el algoritmo: “Simular ingresos del personal”

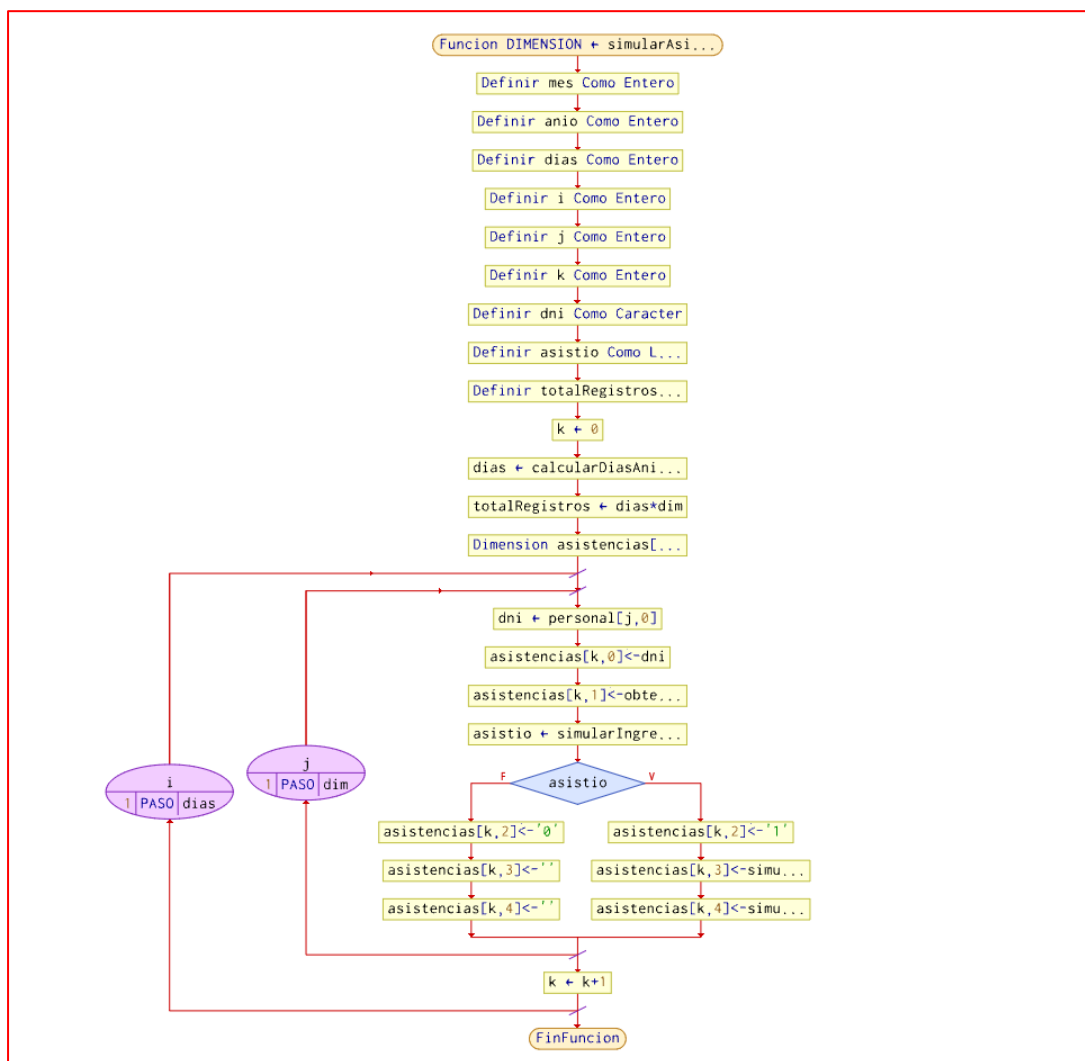




Figura 5

Pseudódigo para el algoritmo: "Simular ingresos del personal"

```
Funcion Dimension <- simularAsistenciasPersonal(personal,dim)
  Definir mes como Entero
  Definir anio como Entero
  Definir dias como Entero
  Definir i como Entero
  Definir j como Entero
  Definir k Como Entero
  Definir dni como cadena
  Definir asistio Como Logico
  Definir totalRegistros Como Entero

  k<-0
  dias <- calcularDiasAnio(anio,mes)
  totalRegistros <- dias * dim

  Dimension asistencias[totalRegistros,5]

  Para i<=1 Hasta dias Con Paso 1 Hacer
    Para j<=1 Hasta dim Con Paso 1 Hacer
      dni <- personal[j,0]

      asistencias[k,0] <-dni
      asistencias[k,1] <-obtenerFecha(i)

      asistio <- simularIngreso()

      Si asistio Entonces
        asistencias[k,2]<- "1"
        asistencias[k,3]<- simularHoraIngreso()
        asistencias[k,4]<- simularHoraSalida()
      SiNo
        asistencias[k,2]<- "0"
        asistencias[k,3]<- ""
        asistencias[k,4]<- ""
      Fin Si
    Fin Para
    k<-k+1
  Fin Para
FinFuncion
```




iii. Simular ingreso

El sistema genera un número aleatorio del 1 al 10.

- Datos de entrada:
 - Si el número es menor o igual que 2 , el empleado no asistió
 - Si el número es mayor que 2 el usuario asistió
- Datos de salida : Booleano que indica si el personal asistió

Figura 6

Diagrama de flujo para el algoritmo: "Simular ingreso"

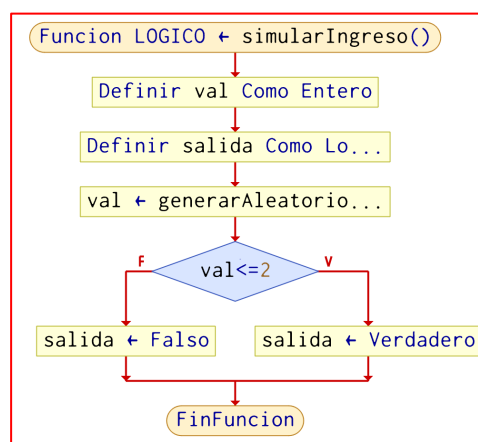


Figura 7

Pseudocódigo para el algoritmo: "Simular ingreso"

```
Funcion Logico <- simularIngreso()
  Definir val como Entero
  Definir salida como Logico
  val ← generarAleatorio(1,10)
  Si val ≤ 2 Entonces
    .....
    salida ← Verdadero
  SiNo
    .....
    salida ← Falso
  Fin Si
FinFuncion
```



iv. Simular hora de ingreso

El sistema parte de la hora base: 8 y genera un número aleatorio del 0 al 59 , los cuales van a ser los minutos que ingreso el personal

- Datos de salida : Cadena con la hora de ingreso del personal

Figura 8

Diagrama de flujo para el algoritmo: "Simular hora de ingreso"

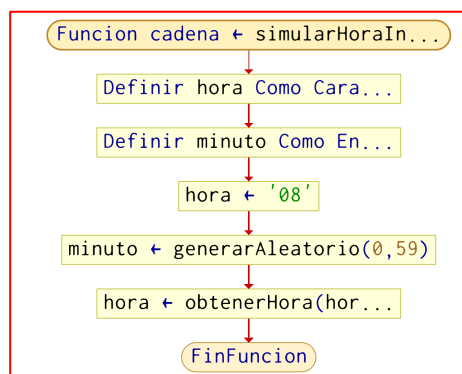


Figura 9

Pseudocódigo para el algoritmo: "Simular hora de ingreso"

```
Funcion cadena <- simularHoraIngreso()
  Definir hora Como Caracter
  Definir minuto Como Entero

  hora ← "08"
  minuto ← generarAleatorio(0,59)

  hora ← obtenerHora(hora,minuto)
FinFuncion
```



3. Herramienta y Tecnología por usar

- Bizagi modeler (Herramienta para modelar procesos)
- IntelliJIDEA (IDE para desarrollar en java)
- Gitlab (Herramienta para el control de versiones)
- Trello (Herramienta para gestionar las tareas)
- PSeint (Herramienta para realizar diagramas de flujo)
- JFreeChart (Librería de java para generar gráficos)

4. Gestión del proyecto. Product Backlog en Trello

- Creación del espacio de trabajo

Figura 10

Creación de un espacio de trabajo en Trello

Vamos a crear un Espacio de trabajo
Impulse su productividad facilitándole a todos el acceso a los tableros en una única ubicación.

Nombre del Espacio de trabajo
Fundamentos de programación
Este es el nombre de su empresa, equipo u organización.

Tipo de Espacio de trabajo
Ingeniería y TI

Descripción del Espacio de trabajo Opcional
Espacio de trabajo del grupo 5 de programación I

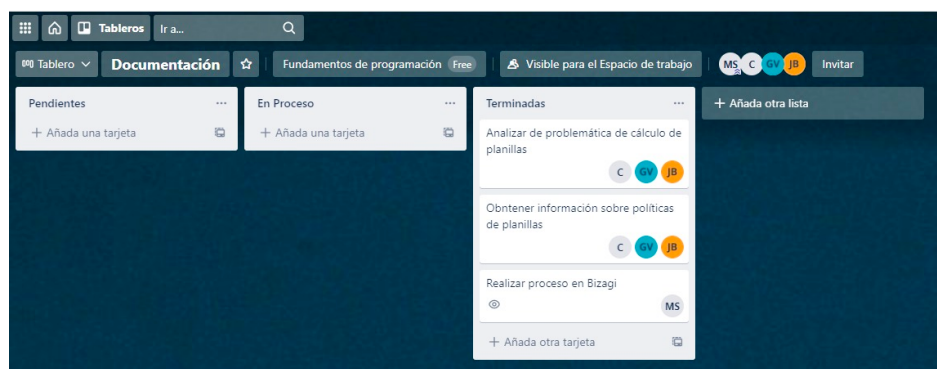
Incorpora a los miembros con unas cuantas palabras sobre tu Espacio de trabajo.

Continuar

- Se trabajaron con 2 tableros , en los cuales se asignó tareas a los miembros del equipo de trabajo

Figura 11

Tablero de proyecto “Documentación” en Trello

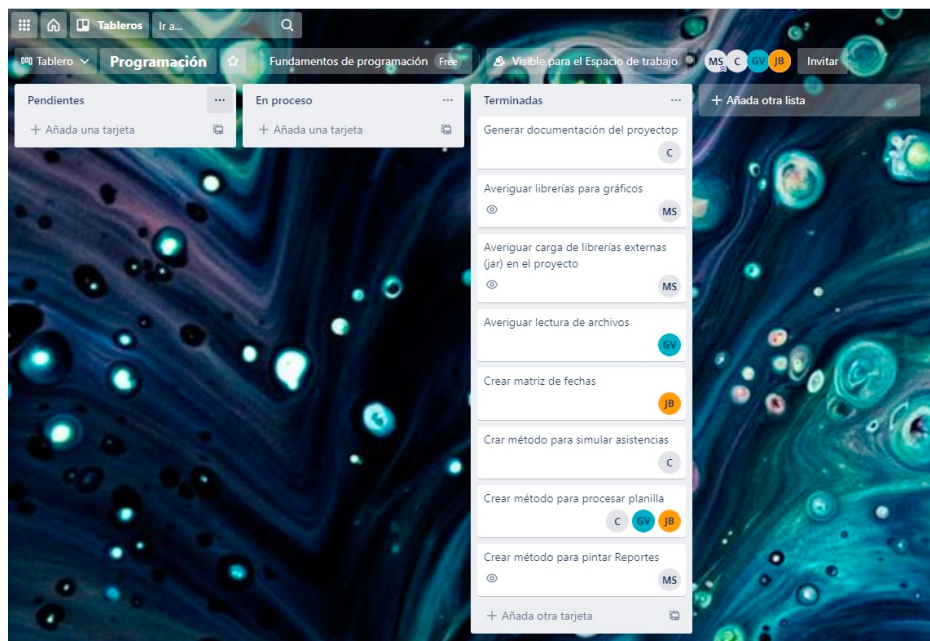




iii. Tareas del tablero de documentación

Figura 12

Tablero de proyecto “Programación” en Trello





C. Programación

1. Control de versiones en GIT

- ✓ Creación del proyecto

Figura 13

Formulario de creación de proyecto en GitHub

New project › Create blank project/repository

Project name
Lion White

Project URL
https://gitlab.com/Kerberos0711/

Project slug
lion-white

Want to house several dependent projects under the same namespace? [Create a group.](#)

Project description (optional)
Control de versiones para el proyecto de fundamentos de programación

Visibility Level [?](#)

☒ Private
Project access must be granted explicitly to each user. If this project is part of a group, access will be granted to members of the group.

☐ Public
The project can be accessed without any authentication.

☐ Initialize repository with a README
Allows you to immediately clone this project's repository. Skip this if you plan to push up an existing repository.

Create project Cancel

- ✓ Comandos del proyecto

Figura 14

Proyecto de GitHub sin commits o "en blanco"

Command line instructions

You can also upload existing files from your computer using the instructions below.

Git global setup

```
git config --global user.name "Miguel Salazar"
git config --global user.email "juan.salazar@outlook.com"
```

Create a new repository

```
git clone https://gitlab.com/Kerberos0711/lion-white.git
cd lion-white
touch README.md
git add README.md
git commit -m "add README"
git push -u origin master
```

Push an existing folder

```
cd existing_folder
git init
git remote add origin https://gitlab.com/Kerberos0711/lion-white.git
git add .
git commit -m "Initial commit"
git push -u origin master
```

Push an existing Git repository

```
cd existing_repo
git remote rename origin old-origin
git remote add origin https://gitlab.com/Kerberos0711/lion-white.git
git push -u origin --all
git push -u origin --tags
```



- ✓ Carga de las actualizaciones

Figura 15

Ubicación de la carpeta y clonación del proyecto GitHub creado

```
MINGW64:/e/01.Cursos/11.UPC/Ciclo 01/Fundamentos de programación/Proyecto
juans@Kerberos MINGW64 ~
$ cd E:
juans@Kerberos MINGW64 /e
$ cd E:/01.Cursos/11.UPC/Ciclo 01/Fundamentos de programación/Proyecto/
juans@Kerberos MINGW64 /e/01.Cursos/11.UPC/Ciclo 01/Fundamentos de programación/Proyecto
$ git clone https://gitlab.com/Kerberos0711/lion-white.git com.javax.lion
```

Figura 16

Inicialización del proyecto git clonado y primer commit

```
Demo.txt 05/05/2021 21:54 Documento de te... 0 KB
MINGW64:/e/01.Cursos/11.UPC/Ciclo 01/Fundamentos de programación/Pr...
juans@Kerberos MINGW64 /e/01.Cursos/11.UPC/Ciclo 01/Fundamentos de programación/Proyecto/com.javax.lion (master)
$ git init
Reinitialized existing Git repository in E:/01.Cursos/11.UPC/Ciclo 01/Fundamentos de programación/Proyecto/com.javax.lion/.git/
juans@Kerberos MINGW64 /e/01.Cursos/11.UPC/Ciclo 01/Fundamentos de programación/Proyecto/com.javax.lion (master)
$ git add .
juans@Kerberos MINGW64 /e/01.Cursos/11.UPC/Ciclo 01/Fundamentos de programación/Proyecto/com.javax.lion (master)
$ git commit -m "Primera carga"
[master (root-commit) 73abed9] Primera carga
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 Demo.txt
juans@Kerberos MINGW64 /e/01.Cursos/11.UPC/Ciclo 01/Fundamentos de programación/Proyecto/com.javax.lion (master)
$ git push
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Writing objects: 100% (3/3), 222 bytes | 222.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To https://gitlab.com/Kerberos0711/lion-white.git
* [new branch] master -> master
juans@Kerberos MINGW64 /e/01.Cursos/11.UPC/Ciclo 01/Fundamentos de programación/Proyecto/com.javax.lion (master)
$ |
```



- ✓ Control de modificaciones del proyecto

Figura 17

Archivo modificado en un commit visto desde GitHub

```
src/ui/frmpincipal.java 0 - 100644
582 + planillas[i][4] = ""+(int)xdescuentos[0]; //Falta justificada
583 + planillas[i][5] = ""+faltas_injustificadas; //Falta injustificada
584 + planillas[i][6] = ""+xdescuentos[2]; //Tardanza
585 +
586 + //-----|-----|-----|
587 + // | Nivel 1 | Nivel 2 |
588 + //-----|-----|-----|
589 + //|Faltas |Justificada |25 x día |10 x día |
590 + //| |Injustificada |150 x día |60 x día |
591 + //-----|-----|-----|
592 +
593 + dsctos_faltaj = (nivel == 1 ? 25 : 10) * xdescuentos[0];
594 + dsctos_faltasj = (nivel == 1 ? 150 : 60) * xdescuentos[1];
595 +
596 + //-----|-----|-----|
597 + // | Nivel 1 | Nivel 2 |
598 + //-----|-----|-----|
599 + //|Tardanzas |1 - 4 | 10 | 5 |
600 + //| |4.1 a mas | 15 | 10 |
601 + //-----|-----|-----|
602 +
603 + if(xdescuentos[2]<=4){
604 + dsctos_tardanza = (nivel == 1 ? 10 : 5) * xdescuentos[2];
605 + }else{
606 + dsctos_tardanza = (nivel == 1 ? 15 : 10) * xdescuentos[2];
607 + }
608 +
609 + planillas[i][7] = ""+dsctos_tardanza;
610 + planillas[i][8] = ""+dsctos_faltaj;
611 + planillas[i][9] = ""+dsctos_faltasj;
612 +
613 + sueldo_net = sueldo - dsctos_faltaj - dsctos_faltasj - dsctos_tardanza;
614 +
615 + planillas[i][10] = ""+sueldo_net;
```

- ✓ Repositorio del proyecto

Figura 18

Vista del proyecto en GitHub con los archivos ya subidos

Lion White

Project ID: 26440534

3 Commits 1 Branch 0 Tags 3.2 MB Files 3.2 MB Storage

Control de versiones para el proyecto de fundamentos de programación.

Auto DevOps

It will automatically build, test, and deploy your application based on a predefined CI/CD configuration.

Learn more in the [Auto DevOps documentation](#)

Enable in settings

master lion-white

History Find file Web IDE Clone

Comentarios modificados

Miguel Salazar authored 9 minutes ago

c65c3621

Upload File

Add README

Add LICENSE

Add CHANGELOG

Add CONTRIBUTING

Add Kubernetes cluster

Set up CI/CD

Configure integrations

Name	Last commit	Last update
idea	Documentación del proyecto	13 minutes ago
data	Carga de proyecto	1 hour ago
images	Carga de proyecto	1 hour ago
out/production/com.java.lion	Documentación del proyecto	13 minutes ago
resources	Carga de proyecto	1 hour ago
src/ui	Comentarios modificados	9 minutes ago
Comandos git.txt	Carga de proyecto	1 hour ago
Trabajo Final Fundamentos de Pr...	Documentación del proyecto	13 minutes ago



D. Conclusiones y recomendaciones

Se puede concluir a partir del proyecto propuesto que, se puede optimizar de manera exponencial el proceso de control de planillas con un programa relativamente no complejo.

Bajo la conclusión ya expuesta recomendamos a la empresa:

- A corto plazo, delegar la tarea de mantener el software a un empleado con conocimientos en el lenguaje Java, de caso contrario contratar a una persona con ese perfil técnico.
- A mediano plazo, evaluar opciones de mejora, agregar nuevos modulos relacionados al control de planilla.
- A largo plazo, al ser un software con impacto positivo para la empresa se recomienda llevarlo a dispositivos móviles, ya que estos cuentan con tecnologías para la identificación biometrica y además que es de fácil transporte.

E. Anexos

- Cargos y sueldo del personal

Cargo	Codigo	Sueldo Neto	Nivel
Ingeniería	IN001	3100.00	1
Administrativos	AD001	2500.00	1
Tecnicos de planta	TP001	2000.00	2
Operadores	OP001	1500.00	2

- Horario (Tolerancia 10 min)

Día	Hora de ingreso	Hora de salida
Lunes - Viernes	08:00	18:00



- Política de descuentos

Tipo	Estado	Nivel 1	Nivel 2	Observación
Falta	Justificada	25.00 x día	10.00 x día	
	No justificada	150.00 x día	60.00 x día	3 faltas injustificadas es causal de despido
Tardanzas		1 – 4 tardanzas : 10.00 por tardanza	1 – 4 tardanzas : 5.00 por tardanza	
		5 a más tardanzas: 15.00 por tardanza	5 a más tardanzas: 10.00 por tardanza	