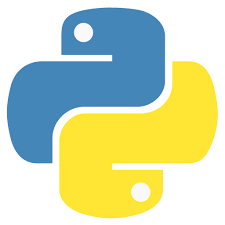
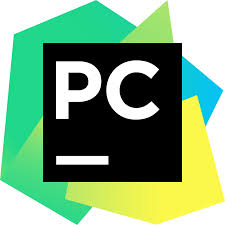
Universidad de Camagüey “Ignacio Agramonte Loynaz”

Facultad de Informática y Ciencias Exactas

**Informe Proyecto Final**

**Proyecto #1: Sistema de Nóminas**

Asignatura: **Programación Orientada a Objeto**.



Integrantes: Reynier Miguelez Remedios

Idel Reyes Pacheco

Alexai Lamas

1er año Ingeniería Informática

**Introducción:**

**Sistema de nóminas:**

Se desea realizar un sistema para gestionar las nóminas de una empresa de proyectos. Se conoce que en la entidad hay trabajadores vinculados que se les paga por productividad y hay trabajadores que se les paga por la estancia en su puesto de trabajo en el horario establecido. De los trabajadores en general se registra el nombre, el sexo, la fecha de nacimiento, y el nivel profesional (que puede ser profesional, técnico medio u obrero calificado); por otro lado, de los trabajadores vinculados se registran los datos del proyecto en que trabaja, su rol dentro del proyecto, el plan mensual de cumplimiento de su tarea, que es un valor numérico entre 1 y 100, y el valor real de cumplimiento de su tarea. De los trabajadores no vinculados se registra, además, su responsabilidad (que puede ser obrero, jefe de departamento o director de la empresa), el departamento a que pertenece, la cantidad de llegadas tarde que lleva en el mes y la cantidad de horas trabajadas en el mes. De los proyectos se registra el nombre del proyecto, el cliente, el costo del proyecto, la fecha de inicio y de culminación y el porciento de terminación. Por último, se conoce que el sistema actualiza su información mensualmente y que tiene una lista con todos los datos de las nóminas de todos los trabajadores de la empresa, el mes y año en que se está trabajando, así como una lista de los proyectos en los que está trabajando la empresa.

El sistema debe permitir las siguientes funcionalidades:

a) Implemente las funcionalidades necesarias para gestionar (insertar, actualizar, eliminar y listar) los datos de los trabajadores de forma independiente (vinculados y no vinculados), los datos de los proyectos, así como actualizar los datos de cumplimiento del trabajo mensual de cada trabajador.

b) Implemente la funcionalidad necesaria para determinar el salario de un trabajador dado su nombre y teniendo en cuenta que este se determina a partir de la bonificación por estudios terminados que es de $80 para los profesionales, $60 para los técnicos medio y $40 para los obreros calificados además del cumplimiento de sus labores que se determina de la siguiente forma:

•Para los trabajadores no vinculados se determina entre la cantidad de horas trabajadas por $12.5 más la bonificación del cargo que es $120 para los jefes de departamento y $200 para el director de la empresa restándole $5 por cada llegada tarde.

•Para los trabajadores vinculados se determina por el porciento de cumplimiento de su tarea del mes por el coeficiente de eficiencia que es 21.5 si el por ciento de cumplimiento de su tarea del mes está entre 95 y 100, es 18.5 si el porciento está entre 80 y 94, 15 para un porciento menor de 80, y 25 para un porciento mayor de 100.

c) Implemente la funcionalidad necesaria para determinar el listado de proyectos que tiene la empresa ordenado en orden descendente por el valor añadido que es el costo por el porciento de terminación.

d) Implemente la funcionalidad necesaria para despedir un trabajador del sistema devolviendo todos sus datos y determinando su salario acumulado en el mes actual.

e) Implemente la funcionalidad necesaria para determinar qué porciento de los trabajadores vinculados cumplen con más de un porciento dado, su labor mensual.

f) Implemente la funcionalidad necesaria para listar a final de mes la nómina de la empresa con los datos de cada trabajador y el salario a cobrar, además de la sumatoria total de todos los salarios.

g) Pruebe que las operaciones implementadas en el modelo funcionan correctamente según los datos de prueba que usted le entró al programa.

**Herramientas y Tecnologías:**

* Python 3.11.1: Es un lenguaje de programación de alto nivel, interpretado y de propósito general. Es conocido por su sintaxis clara y legible, lo que lo hace ideal para principiantes en programación. Python es versátil y se utiliza en una amplia gama de aplicaciones, desde desarrollo web hasta inteligencia artificial y análisis de datos. Además, cuenta con una comunidad activa que ofrece una amplia gama de bibliotecas y herramientas que facilitan el desarrollo de software en Python.
* PyCharm Community: Es un entorno de desarrollo integrado (IDE) utilizado para programar en Python. Es desarrollado por la compañía JetBrains y aparece en dos versiones: una versión gratuita conocida como PyCharm Community y una versión de pago llamada PyCharm Professional.
* Qt Designer 5.11.1: Es una herramienta de diseño gráfico que forma parte del framework QT. Se utiliza para crear interfaces de usuario de manera visual para aplicaciones que utilizan QT, un framework de desarrollo de aplicaciones multiplataforma. QT Designer permite a los desarrolladores diseñar interfaces de usuario de forma rápida y sencilla, utilizando una variedad de widgets y elementos gráficos, y luego integrar estos diseños en sus aplicaciones escritas en C++ o Python, entre otros lenguajes compatibles.
* PyQt5: Es un conjunto de enlaces Python para la biblioteca de desarrollo de software QT. Proporciona una forma poderosa y flexible de crear interfaces gráficas de usuario utilizando el lenguaje de programación Python junto con las capacidades de la biblioteca QT. Con PyQt5, los desarrolladores pueden crear aplicaciones de escritorio multiplataforma que incluyan una amplia variedad de widgets y herramientas gráficas, lo que lo convierte en una opción popular para el desarrollo de interfaces de usuario.
* Visual Paradigm Community Edition 1.17.1: Es una herramienta de modelado de software que permite a los desarrolladores y diseñadores visualizar, diseñar y modificar el diseño de sus aplicaciones de manera rápida y efectiva.
* MySql 8.0: Es un sistema de bases de datos de Oracle que se utiliza en todo el mundo para gestionar bases de datos. Se base en álgebra relacional y se utiliza principalmente para el almacenamiento de datos de diverso servicios web. MySQL almacena, gestiona y muestra datos en tablas. Funciona como un sistema cliente-servidor. Mientras que la base de datos actúa como un servidor en el que se almacena toda la información relevante, el software puede verse como un cliente. Con la ayuda del software, los usuarios de la base de datos relacional pueden formular diversas consultas, denominadas “queries”, en el lenguaje de consultas SQL y enviarlas al sistema de bases de datos. Estos son procesados por MySQL por lo que el acceso a los datos es una parte importante de MySQL. MySQL se caracteriza por un alto grado de independencia de la plataforma. Los usuarios pueden usar MySQL en más de veinte plataformas diferentes, incluidos los sistemas operativos comunes Windows, MacOS y Linux.

**Desarrollo:**

**Análisis del problema:**

Para resolver este problema se creó una aplicación que permite gestionar la información de cada trabajador de la Empresa, así como la información de los proyectos que esta maneja. Estos datos se mostrarán en una interfaz intuitiva y el usuario será capaz de insertar, actualizar, eliminar y listar en tablas toda la información.

**Diagrama de clases:**

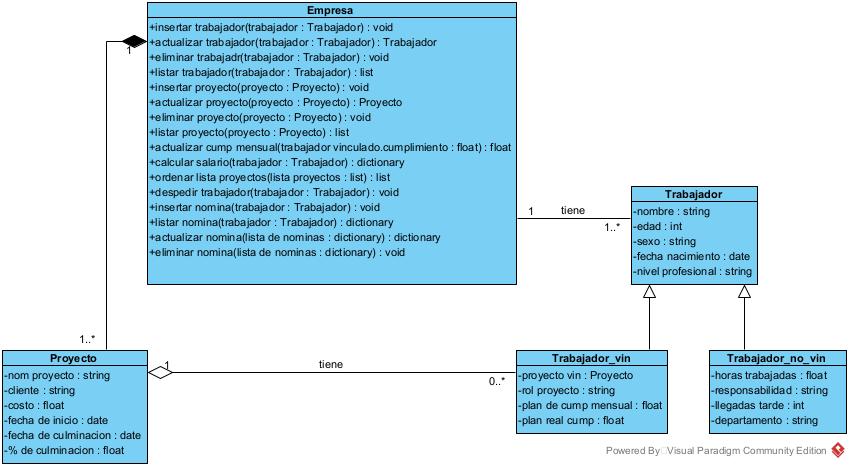


Diagrama de clases en que se muestra el flujo de datos que maneja la aplicación, así como los tipos de datos utilizados.

**Tipos de datos empleados:**

**Numéricos:** Enteros (int) y decimales (float). Utilizados principalmente como identificadores de objetos dentro de la base de datos y para expresar datos específicos como salarios, edades, porcientos, etcétera.

**Cadenas de caracteres:** También conocidos como “strings”. Representan los textos, nombres, encabezados de tablas, etcétera.

**Fechas:** Tipos de datos que manejan fechas (date), expresadas en formato “año-mes-día”.

**Objetos:** Representan entidades del mundo real como trabajadores y proyectos.

**Descripción de métodos:**

**Gestión de datos:**

**Insertar:**

Para insertar trabajadores o proyectos al sistema de deberán rellenar los campos solicitados, luego se presionará en el botón “ADD” representado por un símbolo de (+), el programa insertará en la base de datos los datos del formulario y los mostrará en pantalla a través de una tabla. De no rellenar los datos se inhabilitará el botón de dicha operación, si el dato que se desea insertar ya se encuentra en la base de datos se emitirá una señal especificando que el elemento ya existe y no se realizará la operación.

**Eliminar:**

Para eliminar trabajadores o proyectos del sistema bastará con seleccionar en la tabla correspondiente el elemento que se desea eliminar, luego se presiona en el botón “DELL” representado por el símbolo de un “contenedor de basura”, el programa eliminará el elemento seleccionado de la base de datos y lo removerá de la tabla correspondiente. De no tener ningún elemento seleccionado se emitirá una señal de error señalando que se debe seleccionar un objeto antes de eliminarlo.

**Actualizar:**

Para actualizar un trabajador o un proyecto se deberá seleccionar el elemento que se desea actualizar, el sistema procederá a rellenar los campos del formulario con los datos del elemento seleccionado, el usuario realizará los cambios que desee en los campos y luego deberá oprimir el botón “UPDATE” representado por “una pluma de escribir”, el sistema sobrescribirá los nuevos datos sobre los anteriores en la base de datos y lo mostrará así en la tabla. Si algún campo aún no ha sido rellenado se inhabilitará el botón de dicha operación.

**Ventana principal**

**Nóminas:**

En la ventana principal se mostrarán las nóminas de los trabajadores con todos sus datos, así como el salario de cada trabajador y el total de salarios que deberá pagar la Empresa. Para acceder a dicho total bastará con presionar en el menú superior la opción “Funcionalidades” la cual abrirá un menú deslizante, dentro se presionará en la opción “Total de salarios a pagar”, el sistema devolverá un mensaje con el total de salarios a pagar por la Empresa.

**Porciento de cumplimiento:**

Para determinar el porciento de trabajadores vinculados a proyectos que cumplen con un porciento de cumplimiento de su labor dado bastará con acceder en el menú superior a “Funcionalidades”, luego presionar sobre la opción “porciento de cumplimiento actual”. El sistema abrirá una ventana auxiliar donde se insertará el porciento con el cual se desea comprobar, luego presionar el botón “Determinar”, el sistema devolverá un mensaje con el porciento total.

**Despedir trabajador:**

Si se desea despedir de la Empresa a algún trabajador bastará con seleccionar en la nómina correspondiente de dicho trabajador y luego presionar el botón representado por el símbolo de “trabajador eliminado” el sistema devolverá un mensaje con los datos de dicho trabajador y su salario acumulado en el mes, luego será eliminado de la base de datos y se removerá su nómina de la tabla.

**Ordenar proyectos:**

El sistema cuenta con la funcionalidad de ordenar los proyectos de la Empresa, se ordenará por el costo del proyecto y el porciento de culminación del mismo de manera descendente. Para ordenar los proyectos bastará con pulsar en el botón representado por “líneas horizontales” en la pestaña “proyectos”. El sistema reordenará la tabla de acuerdo a los parámetros antes mencionados.

**Ayuda:**

En el menú superior se encontrará la opción “Ayuda” la cual, al pulsar sobre ella, desplegará dos opciones:

**1-Descripción:** La primera opción se llama “Descripción”. Al pulsar sobre ella el programa devolverá un mensaje con una breve reseña del programa y su objetivo a cumplir.

**2-Acerca de:** La segunda opción llamada “Acerca de:” devolverá información sobre la creación del programa y sus autores.

**Conclusiones:**

Durante el desarrollo de este proyecto, se ha logrado diseñar e implementar un Sistema de Nóminas utilizando Programación Orientada a Objetos (POO) en Python. Se han aplicado conceptos clave de POO, como la definición de clases, encapsulamiento, herencia y polimorfismo para representar entidades del mundo real como trabajadores y proyectos.

Además, se ha empleado un enfoque metodológico sólido que ha permitido descomponer el problema, desarrollar un diagrama de clases UML y definir un conjunto claro de funcionalidades a través de las cuales se ha puesto a prueba la robustez del sistema.

Durante este proceso, se ha adquirido experiencia en el diseño e implementación de sistemas orientados a objetos, así como en el uso de herramientas como MySQL, Qt Designer y PyQt5 para el desarrollo de interfaces gráficas de usuario. En resumen, el proyecto ha representado un desafío significativo que ha permitido adquirir valiosas habilidades en el desarrollo de sistemas orientados a objetos y la implementación de soluciones en Python.

**Recomendaciones:**

Para una mejora futura de la aplicación se recomienda la implementación de una conexión Cliente-Servidor más amplia que permita el manejo de la base de datos de una forma remota. Se recomienda implementar un sistema de seguridad de acceso a la base de datos y un sistema de autentificación de usuario para restringir el personal que puede acceder a los datos.

**Bibliografía**

-Documentación de PyQt5

<https://www.riverbankcomputing.com/static/Docs/PyQt5/>

-Documentación de Python

<https://docs.python.org/3/>

-Qt Assistant versión 5.15.3

-Documentación de MySQL

-Foros de Internet.