



Guía Unidad “2”

Programación Orientada a Objeto Seguro

TI3021

## Unidad 2: Herencia, encapsulamiento, polimorfismo y métodos en Python

|  |  |  |  |
| --- | --- | --- | --- |
| **Área Académica** | Tecnologías de la Información y Ciberseguridad | **Carrera** | Analista Programador |
| **Asignatura** | Programación orientada a objeto seguro | **Código** | TI3021 |
| **Unidad de aprendizaje** | Herencia, encapsulamiento, polimorfismo y métodos en Python. | **Aprendizaje esperado** | 2.1 Construye software en Python para resolver una problemática, aplicando paradigma de programación orientada a objetos y su conexión segura a una base de datos. |
| **Duración** | 24 horas | **Fecha** |  |

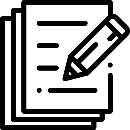
AAAA

|  |
| --- |
| Criterios de evaluación  2.1.1 Verifica diagrama de clases con solución propuesta, considerando el estándar UML.  2.1.2 Desarrollo el código Python con solución propuesta, considerando la herencia, encapsulamiento, polimorfismo.  2.1.3 Utiliza librerías Python de conexión a base datos, considerando repositorios oficiales, la pertinencia con el sistema gestor de base de datos y la solución propuesta.  2.1.4 Ejecuta métodos para creación, actualización, eliminación y listado de registros, de acuerdo con diagrama de clases y conexión a base de datos.  2.1.5 Resuelve errores de ejecución de aplicación en Python, considerando las excepciones originadas, su estructura. |

AAAA

## Forma Descripción generada automáticamente con confianza bajaEtapa II del Proyecto

1. Presentación



La siguiente guía contiene la información necesaria para realizar la segunda entrega del producto/proyecto ***“Sistema de Gestión de Empleados para una Empresa”*** que inicio en la unidad anterior de esta asignatura. Los lineamientos y orientaciones presentes en este recurso de aprendizaje permitirán identificar claramente los pasos a realizar para **desarrollar la solución diseñada en la unidad anterior para responder a las necesidades planteadas**.

Para la realización de esta guía es importante abarcar las siguientes etapas de desarrollo:

1. Análisis de Requerimientos.
2. Diseño del Modelo de Datos.
3. Configuración de la Base de Datos.
4. Desarrollo de Clases y Métodos.
5. Conexión a la Base de Datos desde Python.
6. Implementación de las Operaciones CRUD.
7. Desarrollo de la Interfaz de Usuario.
8. Pruebas.

Esta guía implica la realización de un **Programa en Python aplicando el paradigma orientado a objetos e integrado con base de datos**, correspondiente a la Evaluación Sumativa 2 (ES2) que consiste en el **desarrollo de un programa en Python, que permite realizar un CRUD conectado a base de datos según requerimientos.**

Para la realización de esta entrega, contarás con el instrumento de evaluación “Pauta de cotejo N°1”, previamente confeccionado, que constituye el estándar mínimo a cumplir en la realización de esta actividad.

1. **Instrucciones generales:**
2. El proyecto que será desarrollado durante este semestre es de carácter grupal (entre 2-3 estudiantes), conformados en equipos de trabajo colaborativo, sin embargo, durante la primera unidad, correspondiente al diseño de la solución, el trabajo fue individual**.**

Una vez conformados los grupos de trabajo, deben establecer el rol de Líder de Proyecto, quien se relacionará con el docente, y será responsable del cumplimento de las actividades a desarrollar. Asimismo, establecerá el control de los avances del proyecto, asignaciones de trabajo, reuniones de seguimiento; y, además, será el encargado de reportar al docente de asignatura el nivel de cumplimiento de los integrantes del grupo, dado que la elaboración de este proyecto es de carácter grupal con evaluación individual. En resumen, el líder, es el responsable de llevar a buen término la ejecución del proyecto. Este rol debe ir rotando entre los miembros del equipo durante el semestre, con la finalidad de que todos los integrantes tengan la oportunidad de desarrollar esta habilidad.

1. Con el proyecto se realiza la Evaluación Sumativa 2 con una ponderación de 30%.

1. Se sugiere revisar la bibliografía de la asignatura para sustentar la problemática a abordar. Además, revisar el instrumento de evaluación. Esto les permitirá tener claridad de los aspectos que serán evaluados, optimizar los tiempos y priorizar las actividades que permitan alcanzar en los plazos dados, la meta fijada para esta unidad de aprendizaje.
2. **Actividades**

***Actividad 1: Análisis de Requerimientos***

* Revisar los requerimientos del sistema y asegurarse de comprender completamente las operaciones CRUD necesarias.
* Utilizar el diagrama de clases para identificar las entidades clave y las relaciones entre ellas.

***Actividad 2: Diseño del Modelo de Datos***

* Asegurar que el modelo de datos refleje adecuadamente las entidades y relaciones definidas en el diagrama de clases.
* Definir las tablas y relaciones en la base de datos según el modelo.

***Actividad 3: Configuración de la Base de Datos***

* Crear la base de datos y las tablas de acuerdo con el diseño del modelo de datos.
* Configurar las conexiones y permisos necesarios para acceder a la base de datos desde Python.

***Actividad 4: Desarrollo de Clases y Métodos***

* Utilizar el diagrama de clases para guiar la creación de clases en Python que representen las entidades del sistema.
* Implementar métodos para realizar las operaciones CRUD en cada clase.

***Actividad 5: Conexión a la Base de Datos desde Python***

* Utilizar una biblioteca de acceso a base de datos compatible con Python (por ejemplo, pymysql o mysqlclient para MySQL) para establecer la conexión con la base de datos desde tu programa.

***Actividad 6: Implementación de las Operaciones CRUD***

* Para cada entidad en tu sistema, implementa funciones o métodos que permitan crear, leer, actualizar y eliminar registros en la base de datos.

***Actividad 7: Desarrollo de la Interfaz de Usuario***

* Crear una interfaz de usuario, considerando desarrollar las pantallas necesarias para interactuar con las operaciones CRUD.

***Actividad 8: Pruebas***

* Realizar pruebas exhaustivas para asegurarse que todas las operaciones CRUD funcionan correctamente.
* Considerar casos límite y situaciones de error.

***Actividad (ES2) Programa en Python aplicando el paradigma orientado a objetos e integrado con base de datos***

La Evaluación Sumativa 2 consiste en que los estudiantes de manera grupal conformados por equipos de trabajo entre 2 a 3 integrantes, desarrollan un programa en Python, que permita realizar un CRUD conectado a base de datos según requerimientos.

Para la entrega debe considerar los siguientes elementos:

* Utilización de Herramientas de programación.
* Programa en lenguaje Python propuesto para la solución considerando:
* Los siguientes requisitos del sistema:
  + Registro de Empleados.
  + Gestión de Departamentos.
  + Asignación de Empleados a Departamentos.
  + Registro de Tiempo.
  + Gestión de Proyectos.
  + Asignación de Empleados a Proyectos.
* Para la codificación de cada clase considere:
  + Constructores.
  + Atributos.
  + Métodos.
  + Visibilidad.
* Relaciones entre clases.
* Herencia de clases.
* Polimorfismo.
* Librerías de conexión a base de datos.
* Clase de conexión a base de datos.
* Métodos para creación, actualización, eliminación y listado de registros.
* Procedimientos de seguridad en conexiones a base de datos.
* Estructuras para el manejo de excepciones.

# **Título del caso: Sistema de Gestión de Empleados para una Empresa**.

## Contexto:

Considerando que eres un desarrollador de software independiente, la empresa "EcoTech Solutions" ha solicitado sus servicios para el desarrollo de una solución informática segura a sus requerimientos. EcoTech Solutions es una empresa en crecimiento que se especializa en tecnologías sostenibles. Han experimentado un rápido aumento en su fuerza laboral y necesitan una solución segura y efectiva para gestionar a sus empleados y proyectos. Es por esta razón que se han acercado a usted para desarrollar un nuevo Sistema de Gestión de Empleados con un enfoque en programación orientada a objetos (POO), sin dejar de lado los temas asociados a la seguridad informática.

## Requisitos del Sistema:

### § *Registro de Empleados:*

El sistema debe permitir a los administradores de recursos humanos registrar nuevos empleados. Cada empleado tendrá información personal como nombre, dirección, número de teléfono, dirección de correo electrónico, fecha de inicio de contrato y salario. Además, se debe asignar automáticamente un ID único a cada empleado.

### § *Gestión de Departamentos:*

La empresa tiene varios departamentos, como Desarrollo Sostenible, Investigación y Desarrollo, Ventas, Recursos Humanos, etc. El sistema debe permitir la creación, edición, búsquedas y eliminación de departamentos. Cada departamento tendrá un nombre y un gerente asociado.

### § *Asignación de Empleados a Departamentos:*

Los empleados pueden ser asignados a un departamento en particular. Cada empleado solo puede pertenecer a un departamento a la vez. El sistema debe permitir la asignación y reasignación de empleados a departamentos.

### § *Registro de Tiempo:*

El sistema debe permitir a los empleados registrar las horas trabajadas. Los empleados deben poder ingresar la fecha, la cantidad de horas trabajadas y una breve descripción de las tareas realizadas. Estos registros de tiempo deben estar asociados a un empleado y a un proyecto específico.

### 

### § *Gestión de Proyectos:*

La empresa trabaja en varios proyectos a la vez. El sistema debe permitir la creación, edición y eliminación de proyectos. Cada proyecto tendrá un nombre, una descripción y una fecha de inicio.

### § *Asignación de Empleados a Proyectos:*

Los empleados pueden ser asignados a uno o varios proyectos. El sistema debe permitir la asignación y des asignación de empleados a proyectos.

### 

### § *Generación de Informes:*

Los administradores deben poder generar informes que muestren la información de los empleados, proyectos, departamentos y registros de tiempo.

## Consideraciones Técnicas:

* Utiliza el paradigma de programación orientada a objetos para diseñar y desarrollar el sistema.
* Crea programa en Python para desarrollar métodos CRUD de empleados, departamentos, proyectos y registros de tiempo, además de la asignación de empleados a departamentos y la asignación de empleados a proyectos.
* Utiliza librerías de Python para la conexión a base de datos.
* Utiliza herencia y polimorfismo de manera efectiva para evitar duplicación de código.
* Utiliza una base de datos para almacenar la información del sistema.
* Desarrolla una interfaz de usuario para que los usuarios puedan interactuar con el sistema de manera intuitiva.
* Utiliza estructuras para el manejo de excepciones para resolver errores de ejecución.

|  |
| --- |
| **Revisión de documentos en el AAI**.  AAAA  En el AAI revisa los siguientes recursos:   * Instrumento de evaluación |

**Indicaciones de entrega**

|  |
| --- |
| * El programa se codificará de forma grupal según las instrucciones dadas por el docente. * La revisión será presencial, por tanto, para ello en la fecha correspondiente debe desplegar el programa en el equipo de laboratorio considerando la base de datos y el programa en lenguaje Python. * En la plataforma AAI se habilitará la tarea para cargar la evidencia de su trabajo, para ello debe subir en una carpeta comprimida los archivos del programa (códigos) junto al script de la base de datos, hasta el término de la clase del día agendado. * Una vez finalizado el plazo, se bloqueará el acceso a subir los archivos. * **Se sugiere que la carga No sea realizada a último minuto, ya que el sistema pudiese fallar y se presentarían problemas de conexión.** * NO SE RECIBIRÁN ENTREGAS POR CORREO. * Siendo este trabajo en equipo, el docente se reserva el derecho a interrogar a algún integrante del equipo en caso de presentar dudas en la autoría de los códigos. * Finalmente, y no menos importante, quien no asista a las fechas de evaluación, se le evaluará con la nota mínima, salvo que justifique según los protocolos establecidos en el Reglamento Académico su inasistencia y que informe oportunamente al docente vía correo electrónico. |



AAAA

