

Proyecto

Curso: Python para programadores

El proyecto del curso consistirá en una pequeña aplicación desarrollada en Python. El desarrollo de esta aplicación se realizará de forma incremental a lo largo del curso, con entregas parciales cada semana y una entrega final.

1 Requisitos generales del proyecto

1.1 Funcionales

Se desarrollará una aplicación de gestión con interfaz en modo texto de la temática elegida por el/la estudiante.

El programa tendrá las siguientes opciones:

- Añadir elemento: pedirá al usuario los valores de los campos del elemento y lo guardará.
- Borrar elemento: pedirá al usuario el identificador de un elemento y lo borrará.
- Mostrar lista de elementos: mostrará al usuario la lista de elementos, mostrando solo el identificados y el nombre.
- Mostrar detalle de un elemento: pedirá al usuario el identificador de un elemento y mostrará toda su información.
- Opción adicional: opción definida por el/la estudiante que tenga sentido para la temática elegida.
- Terminar: se terminará la ejecución de la aplicación.

1.2 No funcionales

- Se utilizará programación orientada a objetos.
- La aplicación se dividirá en varios módulos.
- Se documentarán con docstrings los módulos, clases y funciones.
- Se realizará persistencia de los datos a un fichero. No hay restricción respecto al tipo y formato del fichero utilizado para cargar/guardar los datos se puede elegir cualquiera que permita realizar la funcionalidad requerida.
- **Opcional:** generar documentación con Sphinx en formato HTML y/o LaTeX.

2 Entregas del proyecto

Se han establecido 5 entregas incrementales para las que se ha del proyecto se extienden a lo largo del curso. Para cada entrega existe una tarea en Platega con las siguientes fechas de entrega:

Entregas	Fechas de entrega
Entrega 1	07/10
Entrega 2	14/10
Entrega 3	21/10
Entrega 4	28/10
Entrega 5 (final)	12/11

En los apartados siguientes, se describe cada una de las entregas.

2.1 Entrega 1

Consiste en describir el tema del proyecto el tema del proyecto, así como el modelo de datos y la opción adicional del menú que se indica en los requisitos.

La entrega se realizará en un documento pdf.

2.1.1 Ejemplo

Un ejemplo de tema de proyecto, podría ser la gestión de empleados de una empresa, donde la entidad básica sería “Empleado”, que puede ser de dos tipos: fijo o temporal.

Para gestionar los empleados de una empresa, necesitamos definir la información requerida para cada empleado.

Los datos básicos que queremos almacenar de cada empleado son:

- NIF
- Nombre
- Fecha de nacimiento
- Sueldo mensual

Tendremos dos tipos de empleados con la siguiente información adicional:

- **Empleado fijo:** año de alta en la empresa, complemento anual.
- **Empleado temporal:** fecha de alta y de baja del contrato.

La opción adicional del menú en este caso sería calcular el sueldo mensual actual considerando que:

- En los empleados temporales el sueldo mensual es fijo.
- En los empleados fijos el sueldo es el resultado de sumarle a la base el complemento anual multiplicado por el número de años en la empresa.

El menú de la aplicación para este ejemplo sería:

- (1) Añadir empleado
- (2) Borrar empleado
- (3) Mostrar lista empleados
- (4) Mostrar detalle de un empleado

- (5) Mostrar sueldo mensual de un empleado
- (6) Terminar

2.2 Entrega 2

En esta entrega se comenzará el desarrollo del programa, tareas a realizar:

- Pensar y empezar a crear la estructura del proyecto.
- Desarrollo del menú principal con una interfaz de texto que permita al usuario ver las opciones del menú y seleccionar un apartado. Se realizará la comprobación de la entrada de datos.
- Se comenzará el desarrollo del apartado “Añadir un elemento”, realizando la petición de datos al usuario y la comprobación de los datos que introduce el usuario sin guardar todavía esta información.

2.3 Entrega 3

Se continuará con el desarrollo en Python. Tareas a realizar:

- Desarrollar las clases necesarias. Ejemplo: tomando el caso de ejemplo de la gestión de empleados descrito en la Entrega 1, tendríamos las clases Empleado, EmpleadoFijo y EmpleadoTemporal para el modelo de datos. Adicionalmente, se pueden crear clases adicionales para la mejor estructuración del programa.
- Continuar el desarrollo del apartado “Añadir” almacenando los datos por ahora en objetos en memoria.
- Desarrollar el apartado “Mostrar la lista de elementos”.

2.4 Entrega 4

Se continuará con el desarrollo en Python. Tareas a realizar:

- Añadir persistencia de datos, seleccionando el formato de archivo que se considere más adecuado.
- Finalizar el desarrollo del apartado “Añadir”, añadiendo la persistencia de datos.
- Desarrollar el desarrollo del apartado “Modificar” (con persistencia de datos).

2.5 Entrega 5 (entrega final)

Se finalizará el proyecto para lo que se realizará el desarrollo de los apartados restantes.

Como tarea opcional se pide la generación de documentación del proyecto con Sphinx.

3 Evaluación

Se evaluará que el proyecto cumpla los requisitos funcionales y no funcionales definidos en este documento.

Tras cada entrega se proporcionará feedback, en el cual se podrán sugerir mejoras.

En cada entrega se evaluarán las tareas específicas de esa entrega, así como las mejoras realizadas de las entregas anteriores.