



Título del Proyecto

Gestor de Servicio de Urgencias Hospitalarias en Python

Nombre del estudiante

Victor Guerra Rubio

Asignatura

Python Programming

Año

2025

Solución al proyecto de aplicación

El proyecto consiste en una aplicación desarrollada en Python que permite gestionar de forma sencilla el servicio de urgencias de un hospital. Se trabajó con dos estructuras de datos principales:

- Diccionarios para almacenar información de pacientes y médicos. El problema de esta estructura es que es efímera, es decir, una vez se cierra el programa se restaura a los valores preseleccionados en un principio.
- Ficheros CSV para registrar las historias clínicas de los pacientes de forma persistente.

Implementamos distintas funciones que cumplen tareas específicas:

- Gestión de acceso: La función `gestion_acceso(medicos)` valida que los médicos registrados puedan entrar al sistema con un identificador y contraseña, permitiendo tres intentos antes de lanzar un `ValueError`.
- Gestor de pacientes:
 - `registra_paciente(pacientes)` permite registrar nuevos pacientes asegurando que la edad esté entre 0 y 120.
 - `alta_paciente(pacientes)` actualiza la fecha de alta de un paciente.
 - `consulta_info_paciente(pacientes, id_paciente)` muestra información detallada de un paciente.

Ejemplo de registro de paciente:

```
def registra_paciente(pacientes):
```

```
    dni = input("Introduce el DNI del paciente: ")
    nombre = input("Introduce el nombre: ")
    edad = int(input("Introduce la edad: "))
    fecha_ingreso = datetime.today().strftime('%d-%m-%Y')
    pacientes[dni] = [nombre, edad, fecha_ingreso, None]
```

- Historias clínicas:

- `consulta_historia_clinica(id_paciente)` lee las visitas desde `historias_clinicas.csv`.
- `registra_historia_clinica(id_medico)` añade una nueva visita y actualiza el CSV.

Ejemplo:

```
df = pd.read_csv('historias_clinicas.csv')
df = pd.concat([df, nueva_visita], ignore_index=True)

df.to_csv('historias_clinicas.csv', index=False)
```



Instituto
Europeo
de Posgrado

Aplicación práctica del conocimiento

La realización de este proyecto me permitió integrar y aplicar conocimientos adquiridos durante la asignatura de Python Programming, como:

- Creación de funciones estructuradas.
- Uso de estructuras de datos como listas y diccionarios.
- Manejo de ficheros CSV con pandas.
- Validación de datos de entrada del usuario.
- Control de errores y excepciones.

Este tipo de aplicación es trasladable al ámbito profesional en hospitales, clínicas o empresas que necesiten herramientas de gestión esto es un ejemplo de un sencillo CRM para un hospital, también se podría aplicar ciertos cambios para que pueda ser usado como un CRM de gestión de usuarios o incluso para la gestión de una tienda o inventario.

Además, reforcé habilidades clave como el diseño modular, flujo de control, y manipulación eficiente de datos. Esto tiene aplicación directa en desarrollo de software profesional.

Referencias

Python Software Foundation. (2024). Python Documentation. <https://docs.python.org/3/>