

Curso Python

Trabajo Final

Proyecto Django

1. Desarrolle una aplicación en Django que tenga las siguientes funciones:
 - a. Debe tener el objeto Alumno con los siguientes atributos:
 - i. Nombre: string
 - ii. Apellido: string
 - iii. DNI: integer (clave primaria)
 - iv. Teléfono: string
 - v. Correo Electrónico: mail
 - vi. Curso: Curso
 - b. Debe tener el objeto Curso con los siguientes atributos:
 - i. Nombre: string
 - ii. Descripción: string
 - iii. BandaHoraria: BandaHoraria
 - iv. Nota: integer
 - c. Debe tener el objeto BandaHoraria con los siguientes atributos:
 - i. Nombre: string
 - ii. Horario_inicio: datetime
 - iii. Horario_fin: datetime
 - d. Debe tener una URL/Endpoint llamada /cargarAlumnos que llame a una función de la vista que lea un archivo csv de alumnos (con columnas equivalentes a sus atributos) y guarde esos alumnos verificando que no se repitan por su clave primaria.
 - e. Debe tener una URL/Endpoint llamada /listarAlumnos que llame a una función de la vista que liste esos alumnos.
 - f. Debe tener una URL/Endpoint llamada /alumno que reciba el dni de un alumno por parámetro (GET) y retorne la información en formato csv del alumno incluidos el curso y la banda horaria.



- g. **Endpoint para Modificar Alumnos:** Crea un nuevo endpoint `/modificarAlumno` que permita la actualización de los datos de un alumno existente utilizando el método HTTP PUT. Esta función debe recibir el DNI del alumno y los datos actualizados en formato JSON para modificar el registro correspondiente en la base de datos. Debe verificar que el alumno exista y si no existe debe devolver el error HTTP correspondiente.
- h. **Endpoint para Eliminar Alumnos:** Implementa un endpoint `/eliminarAlumno` que permita la eliminación de un alumno existente utilizando el método HTTP DELETE. Al recibir el DNI de un alumno por parámetro en la URL del endpoint, la función correspondiente debe eliminar este registro de la base de datos. Debe verificar que el alumno exista y si no existe debe devolver el error HTTP correspondiente.
- i. **Endpoint para Asignar Curso a Alumnos:** Crea una funcionalidad en la API que permita asignar un curso a un alumno en particular. Implementa un endpoint `/asignarCurso` que reciba el DNI del alumno y el ID del curso deseado para asociar el alumno con dicho curso. Debe verificar que el alumno y el curso existan y si no existen debe devolver el error HTTP correspondiente.
- j. **Endpoint para Consultar Alumnos por Curso:** Desarrolla un endpoint `/alumnosPorCurso` que reciba el ID del curso como parámetro (GET) y devuelva una lista de todos los alumnos matriculados en ese curso, proporcionando información básica de cada alumno en formato JSON.
- k. **Usted debe crear los cursos y las bandas horarios como prefiera para probar las funciones.**
- l. **En todos los endpoints se debe verificar que el objeto a consultar exista y si no existe debe devolver el error HTTP correspondiente.**

