
 <p>IES ISIDRA DE GUZMÁN ALCALÁ DE HENARES</p>  <p>Dirección General de Educación Secundaria, Formación Profesional y Régimen Especial VICEPRESIDENCIA, CONSEJERÍA DE EDUCACIÓN Y UNIVERSIDADES</p>	<p>FECHA Y HORA: 15 NOVIEMBRE 2024 – 10.05H LABORATORIO PRÁCTICO REST API DURACIÓN PRUEBA: 2H MÓDULO: DWES CURSO: 2 º DESARROLLO DE APLICACIONES WEB CURSO 2024 – 2025 AULA D22</p> <p>CALIFICACIÓN: 0 A 10 PORCENTAJE NOTA FINAL TRIMESTRE : 10 %</p>	<p>NOMBRE Y APELLIDOS:</p> <hr/> <p>CALIFICACIÓN:</p> <hr/>
---	--	---

Contexto

Una empresa tecnológica necesita gestionar dispositivos IoT conectados en distintos hogares. Para ello, desea un sistema que registre los dispositivos, permita actualizarlos, eliminarlos y consultar su estado actual. La empresa también desea que el sistema gestione de forma robusta los errores para asegurar un flujo de datos limpio y confiable.

Objetivo

Implementar una **Minimal API en .NET C#** que ofrezca las operaciones CRUD sobre una base de datos SQLite, con gestión completa de errores en cada una de las rutas.

Requisitos Técnicos

1. Conexión a Base de Datos SQLite Externa

- Utiliza una base de datos SQLite llamada `devices.db` ya creada, con las siguientes tablas:
 - `Devices` (`Id`, `Name`, `Type`, `UserId`, `LocationId`)
 - `Users` (`Id`, `Username`, `Email`)
 - `Locations` (`Id`, `LocationName`, `Description`)
 - `DeviceStatusHistory` (`Id`, `DeviceId`, `Status`, `Timestamp`)



Resumen de Almacenamiento y Relaciones

- **Usuarios** (`Users`) son los propietarios de los dispositivos.
- **Ubicaciones** (`Locations`) representan las ubicaciones físicas donde se colocan los dispositivos.
- **Dispositivos** (`Devices`) están vinculados a un usuario y a una ubicación específica, y almacenan detalles como el nombre y tipo de dispositivo.
- **Historial de Estado de Dispositivos** (`DeviceStatusHistory`) guarda un historial de cambios de estado de los dispositivos, permitiendo un seguimiento cronológico de su funcionamiento.

El fichero tanto de creación de las tablas como de inserción de valores ejemplo están en `devices.sql` en la ruta raíz del repositorio.

2. Rutas CRUD con Especificaciones de Respuesta JSON:

- **GET /devices:** Obtiene todos los dispositivos.
 - **Respuesta:** Código 200 con lista de dispositivos.
- **GET /devices/{id}:** Obtiene un dispositivo por `ID`.

 <p>IES ISIDRA DE GUZMÁN ALCALÁ DE HENARES</p>  <p>Dirección General de Educación Secundaria, Formación Profesional y Régimen Especial VICEPRESIDENCIA, CONSEJERÍA DE EDUCACIÓN Y UNIVERSIDADES</p>	<p>FECHA Y HORA: 15 NOVIEMBRE 2024 – 10.05H LABORATORIO PRÁCTICO REST API DURACIÓN PRUEBA: 2H MÓDULO: DWES CURSO: 2 º DESARROLLO DE APLICACIONES WEB CURSO 2024 – 2025 AULA D22</p> <p>CALIFICACIÓN: 0 A 10 PORCENTAJE NOTA FINAL TRIMESTRE : 10 %</p>	<p>NOMBRE Y APELLIDOS:</p> <hr/> <p>CALIFICACIÓN:</p> <hr/>
---	--	---

- **Respuesta:**
 - **200 OK:** Retorna el dispositivo encontrado.
 - **404 NOT FOUND:** Si el dispositivo no existe.
- **POST /devices:** Crea un nuevo dispositivo.
 - **Parámetros:** Name, Type, UserId, LocationId
 - **Respuesta:**
 - **201 CREATED:** Si el dispositivo se crea correctamente, incluye el dispositivo en la respuesta.
 - **400 BAD REQUEST:** Si `UserId` o `LocationId` no existen o si los datos son inválidos.
- **PUT /devices/{id}:** Actualiza un dispositivo existente.
 - **Parámetros:** Name, Type, UserId, LocationId
 - **Respuesta:**
 - **204 NO CONTENT:** Si el dispositivo se actualiza correctamente.
 - **404 NOT FOUND:** Si el dispositivo no existe.
 - **400 BAD REQUEST:** Si `UserId` o `LocationId` no existen o si los datos son inválidos.
- **DELETE /devices/{id}:** Elimina un dispositivo.
 - **Respuesta:**
 - **204 NO CONTENT:** Si el dispositivo se elimina correctamente.
 - **404 NOT FOUND:** Si el dispositivo no existe.

Descripción de las Pruebas en Postman (1 punto correcto)

- **Get All Devices:** Prueba que la ruta de obtener todos los dispositivos funcione y devuelva un código 200.
- **Get Device by ID - Valid:** Obtiene un dispositivo válido, debe devolver 200.
- **Get Device by ID - Not Found:** Consulta un ID inexistente y espera un 404.
- **Create Device - Valid:** Prueba de creación de un dispositivo válido, espera un 201.
- **Create Device - Invalid User/Location:** Intenta crear un dispositivo con `UserId` y `LocationId` inválidos, debe devolver 400.
- **Update Device - Valid:** Actualiza un dispositivo existente, debe devolver 204.
- **Update Device - Not Found:** Intenta actualizar un dispositivo inexistente, espera un 404.
- **Delete Device - Valid:** Elimina un dispositivo existente, debe devolver 204.
- **Delete Device - Not Found:** Prueba de eliminación de un dispositivo inexistente, debe devolver 404.
- **Formateo de Clases Modelo y Código Limpio.**