

Ejercicio de entrega de NodeJs

Enunciado

Se requiere una API REST que permita manejar librerías y los libros asociados a cada una de las librerías. Utilizar Node.Js Express, Sequelize y Passport como la infraestructura para crear el servicio. Usar SQLite como motor de base de datos.

Entidades

Libreria

- **Descripción:** Una librería puede tener desde 0 a muchos libros.
- **Ruta:** /library
- **Acciones**
 - **Crear librería (AUTH)**
 - **Obtener una librería**
Debe traer también todos los libros
 - **Obtener todas las librerías**
Debe traer también todos los libros
 - **Modificar una librería (AUTH)**
 - **Eliminar una librería (**) (AUTH)**
 - **Agregar un libro nuevo (*) (AUTH)**

- **Entidad**

id	Int	El identificador de la librería
name	String	Nombre de la librería. Eg: El Librote
location	String	Dirección física de la librería. Eg: Av. Libertador 1460
telefono	String	Número de teléfono. Eg: 3514563344

Libro

- **Descripción:** Un libro tiene todos los datos del mismo, puede pertenecer a una librería o no y representan la instancia (copia) de un libro. Puede haber más de un libro con los mismos datos, excepto el id que es único para esa instancia.
- **Ruta: /book**
- **Acciones**
 - Crear libro (*) (AUTH)
 - Obtener un libro en particular
 - Obtener todos los libros
 - Modificar un libro (AUTH)
 - Eliminar un libro (**) (AUTH)
- **Entidad**

id	Int	El identificador de este libro en particular
isbn	Int	Este identificador es único en todo el mundo y representa el libro, la versión del autor y el año de edición
titulo	String	Título del libro
autor	String	Autor del libro
year	String	Año de edición del libro
library	Int	El identificador de la librería en donde este libro se encuentra

(*): Para crear un libro, pueden hacerlo de las dos formas:

- Haciendo que la librería tenga un método para agregar un libro nuevo
- Crear un libro directamente con /book y enviar el id de la librería

(**) El borrado, siempre de forma lógica. Esto quiere decir que no borramos de la base de datos si no que marcamos que fué borrado

Usuario

- **Descripción:** Un usuario del sistema. Estos usuarios van a ser creados en la base de datos cuando se inicia el sistema
- **Ruta: /user**
- **Acciones**
 - Login
- **Entidad (A definir)**

Debe existir un usuario con name: **admin** y password: **admin**

Puntos a evaluar:

- Que la API permite hacer un CRUD de librerías
- Que la API permite hacer un CRUD de libros
- Que las acciones marcadas con (AUTH) sólo se puedan ejecutar si el usuario está autenticado
- Definir y crear la entidad de Usuario
- Describir el proceso de desarrollo. (Cómo fueron creados los archivos y por qué)
- Bonus:
 - Que haya validación de las entidades al momento de crearse/actualizarse

HAVE FUN!!