

## TP 7 – ORM – CRUD - Sequelize

### Tarea 1 – Cree la siguiente aplicación: Generador de CRUDS

Genere una aplicación que permita generar dinámicamente una solución CRUD para un recurso.

Para llevarlo a cabo el usuario accedera a una página donde podrá especificar el nombre del recurso y cada uno de sus atributos. Cada atributo deberá registrarse con su tipo de dato.

Los tipos de datos disponibles serán (entero, varchar (50), bool, enum)

Una vez registrada la información el programa deberá generar la tabla correspondiente a ese recurso.

El programa generará dinámicamente una “librería” con la siguiente funcionalidad:

Una clase modelo del recurso.

Un módulo con funciones manejadoras para cada uno de los endpoints del recurso.

Un módulo para enrutar cada uno de los endpoints del recurso:

GET	/recurso/{id}	Retorna datos de un recurso
GET	/recurso/create	Muestra formulario para nuevo recurso
POST	/recurso/	registra nuevo recurso
GET	/recurso/	Retorna listado de los recursos
GET	/recurso/{id}/edit	Muestra formulario para editar recurso
PUT	/recurso/{id}	Actualiza los datos del recurso
DELETE	/recurso/{id}	Elimina un recurso

Tener en cuenta que los formularios tengan elementos adecuados al tipo de datos de los atributos del recurso.

### Tarea 2

a.- Se pide reconvertir el ejercicio ‘To-do’ del práctico de motor de plantillas para que los ítems sean almacenados en la base de datos.

Los ítems tendrán la siguiente información (Titulo, fecha de creación, fecha de resolución, descripción, prioridad y una fecha límite), donde prioridad puede ser baja, media y alta. Cada Item tendrá un estado que puede ser (Sin resolver, resolviendo y resuelta).

La aplicación podrá gestionar listas de tareas. Una lista de tareas tiene un título, fecha de creación, fecha de resolución, las tareas que las componen y un estado (pendiente, en proceso, resuelta).

Todos las tareas deben pertenecer a una y solo una lista de tareas.

\*El usuario podrá gestionar tareas y/o listas de tareas, pudiendo ver el detalle de las tareas y actualizar su estado (a excepción de las tareas resueltas las cuales no podrán cambiar el estado).

\*\* En esta aplicación no habrá gestión de usuarios por lo cual las tareas y listas de tareas son compartidas para todo aquel que use la aplicación.

\*\*\* El usuario podrá consultar los listados de tareas y ordenarlos por fecha de creación, fecha límite si existiera y prioridad.

\*\*\*\* Las tareas pueden ser borradas sin posibilidad de recuperar la información.

\*\*\*\*\* La lista puede siempre y cuando no tenga tareas.

\*\*\*\*\* La lista de tareas puede ser archivada siempre y cuando esté resuelta. Una lista archivada deja de ser parte de las listas activas, pero pueden consultarse a través de la opción “consultar tareas archivadas”

\*\*\*\*\* Una lista pasa a estar resuelta solo cuando todas las tareas de la lista están resueltas.