

Desafío - CRUD anidado

Para realizar este desafío debes haber revisado todo el material disponibilizado en la plataforma.

El objetivo de esta actividad es la implementación de un CRUD anidado de Categorías y Productos, donde una categoría puede tener N cantidad de productos, y cada producto pertenece a una categoría (relación 1 a N).

1)

- Iniciar un nuevo proyecto en Rails 5.1
- Añadir el **CDN de Bootstrap** al layout
- Añadir el **CDN de jQuery** al layout
- Crear un scaffold **Category** con el campo *name* (string)
- Revisar y correr la migración
- Convertir el *index* de *categories* en la página de inicio.
- Crear un modelo **Product** con los campos *name* (string), *price* (integer) y *category:references*.
- Revisar y correr la migración.
- En el modelo *Category* agregar la relación con producto (**has_many**) y verificar que el modelo *Product* se haya creado con la relación correcta (**belongs_to**)

Recordar la relación entre categoría y producto en el enunciado.

- Crear el controller **products**.
- Generar la ruta para la creación de un producto asociado a una categoría.
 - Para ello modificar la ruta *resources* de *Categories* que ahora recibirá un bloque y dentro crearemos la ruta que apuntará al método *create* de *products*.

```
resources :products, only: [:create]
```

- En la terminal ejecutar **rails routes** para corroborar la ruta creada. La ruta generada debe contener el **:category_id** y apuntar al método **products#create**.
- Para poder almacenar un registro en el método *create* primero debemos generar el filtro de parámetros conocido como **strong params**, para ello:
 - En el controller **products** crear el método **product_params**. Este método debe permitir y retornar los campos necesarios para la creación de un nuevo producto, es decir, *name* y *price*.

- En el controller **products** crear el método **create**. Este método debe generar una nueva instancia de *Product* recibiendo como argumento **product_params** y almacenarlo en la BD. Luego redireccionar a la vista *show* de la categoría involucrada.
- En la vista *Show* de *categories* además del detalle de la categoría se debe agregar un formulario que permita ingresar un nuevo producto a la categoría seleccionada.
 - El formulario debe ser generado utilizando el helper *form_with* anidando los dos modelos y debe implementar las clases de bootstrap (revisar docs).

```
<%= form_with(model: [ @category, @product ]) do |form| %>
```

- Donde **@product** debe ser declarado en el método correspondiente como una nueva instancia de **Product**.
- El formulario debe tener campos para *name* y *price*.
- En la vista *show*, bajo el formulario, se deben listar todos los productos correspondientes a esa categoría en una tabla con los campos *name* y *price*.
- Junto a cada registro de producto en la tabla se debe añadir un botón para eliminar el producto. Para ello:
 - Agregar el método *destroy* a la ruta anidada.

```
resources :products, only: [:create, :destroy]
```

- Crear el método correspondiente en el controller.
- Agregar el botón con el *method: :delete* a cada registro de productos en la tabla.
- En la vista *Index* de *categories*:
 - Los registros deben estar listados en una tabla (bootstrap) que contenga el nombre de la categoría y la cantidad de productos que existe por categoría (**count**).
 - Al hacer click en el nombre de la categoría debe redireccionar a la vista *show* de esa categoría.
 - Cada categoría además debe ir acompañada de botones para *Edit* y *Destroy* utilizando la clases de bootstrap.
- En *application.html.erb*:
 - En una vista parcial, agregar un navbar (**fixed**) con bootstrap con los link para acceder al home (index de categorías) y para agregar nueva categoría.