



## Desafío - Autenticación con Devise

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

El objetivo de esta actividad es la implementación de un Sistema de autenticación manual para permitir el logeo de un usuario en una aplicación.

### Setup:

En esta actividad se utilizó la gema carrierwave para el manejo de archivos por lo que antes de empezar tendrás que instalar imagemagick en tu computador. Si ya lo has instalado antes, omite este paso.

Para saber si ya está instalado vamos a escribir:

```
convert -version
```

Si obtenemos un mensaje de command not found entonces debemos instalar la herramienta.

### Instalando ImageMagick

En **OSX**:

```
brew update  
brew install imagemagick
```

EN **Ubuntu**:

```
sudo apt-get update  
sudo apt-get install imagemagick
```

Más información en [Documentación carrierwave](#)

### Comienza la actividad

Pictory es una aplicación para que diversos usuarios guarden sus historias y puedan compartirlas, pero esta aplicación no está terminada, el cliente necesita:

- Crear un modelo **user** con los campos *name* (string), *email* (string) y *password* (string).
- Generar la ruta para la creación y la vista de new de un usuario.

```
get 'users/sign_up', to: "users#new"
post 'users', to: "users#create"
```

- En la terminal ejecutar **rails routes** para corroborar las ruta creada. La ruta generada debe apuntar a los métodos **users#new** y **users#create**.
- Crear un controlador de *users* vacío.
- En el controller **users** crear el método **new**.
- En el controller **users** crear el método **user\_params**. Este método debe permitir y retornar los campos necesarios para la creación de un nuevo usuario, es decir, name, email y password.
- En el controller **users** crear el método **create**. Este método debe generar una nueva instancia de *User* recibiendo como argumento **user\_params** y almacenarlo en la BD. Luego, si el usuario es creado exitosamente, agregar **@user.id** a una variable de session (**session[:user\_id]**) y redireccionar al *root\_path*, en caso de error, que haga render del método *new*.
- En la vista *New* de users se debe agregar un formulario que permita ingresar un nuevo usuario.
  - El formulario debe ser generado utilizando el helper *form\_with* añadiendo el modelo y debe implementar las clases de bootstrap (revisar docs).

```
<%= form_with(model: @user) do |form| %>
```

- Donde **@user** debe ser declarado en el método correspondiente (new) como una nueva instancia de **User**.
- El formulario debe tener el campo para *name*, *email* y *password*.
- Crear los métodos *current\_user* y *logged?* en **UsersHelper**.
- Añadir ruta de **sessions** para crear y destruir sesion de usuario.

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

- Crear controller de **sessions** vacío.
- En controlador de **sessions** crear método *destroy* reseteando las variables de sesion y redireccionando a la página root.
- Agregar un link para que el usuario pueda cerrar su sesión (solo en caso que haya iniciado sesión).
- Agregar un link para que el usuario pueda registrarse o iniciar sesión (solo en caso que NO haya iniciado sesión).
- Agregar las rutas para logear un usuario.
- Crear el método new de **sessions**, agregar su ruta respectiva, y añadir la vista con un formulario de logueo.
- Añadir usuario a history.

Utilizar *current\_user* para guardar el dato en la base de datos en el método create.

- Crear vista con las historias que le pertenecen al usuario.