

Desafío - Like Me (Parte I)

- Para realizar este desafío debes haber estudiado previamente todo el material disponible correspondiente a la unidad.
- Desarrollo desafío:
 - El desafío se debe desarrollar de manera Individual.
 - Para la realización del desafío necesitarás apoyarte del archivo *Apoyo Desafío - Like Me*.

Descripción

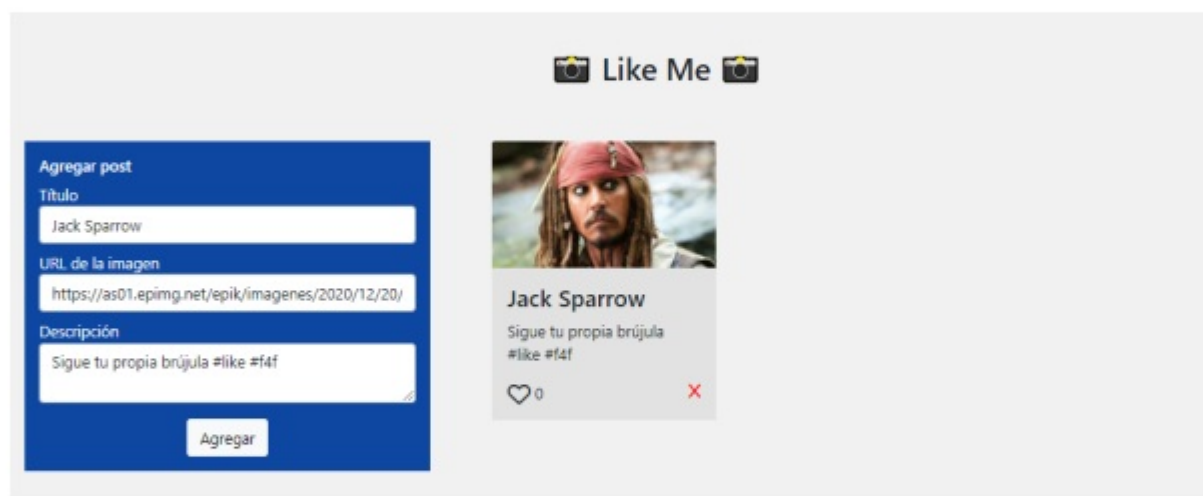
La red social "Like Me" está apenas iniciando el proceso de desarrollo, se encuentra estudiando el mercado y analizando la competencia y ha descubierto que el éxito de esta industria se enfoca en los likes de los posts emitidos por los usuarios, por lo que decidió iniciar con el desarrollo de la lógica correspondiente a la creación de posts y a la interacción de likes que este puede tener.

En este desafío serás un desarrollador full stack developer de la empresa "Like Me" y deberás crear un servidor que disponibilice las rutas **GET** y **POST** para consultas y guardar los posts en una base de datos PostgreSQL con el paquete pg.

Para el desarrollo de este desafío dispondrás de un Apoyo Desafío - Like Me, que contiene una aplicación React con la interfaz que te mostramos en la siguiente imagen.



La siguiente imagen muestra la interacción de la aplicación cliente al registrar un nuevo post.



Esta aplicación React de apoyo incluye las consultas HTTP a un servidor local en el puerto 3000 con las rutas correspondientes al desafío en sus diferentes interacciones.

Para iniciar con el desarrollo de este desafío, ocupa las siguientes instrucciones SQL para crear una base de datos **likeme** y una tabla **posts**.

```
CREATE DATABASE likeme;
```

```
CREATE TABLE posts (id SERIAL, titulo VARCHAR(25), img VARCHAR(1000),  
descripcion VARCHAR(255), likes INT);
```

Requerimientos

1. Habilitar los **cors** en el servidor utilizando el paquete de npm. **(2 puntos)**
2. Usar el paquete **pg** para conectarse e interactuar con la base de datos. **(2 puntos)**
3. Crear una ruta GET con Express para devolver los registros de una tabla alojada en PostgreSQL. **(3 puntos)**
4. Crear una ruta POST con Express que reciba y almacene en PostgreSQL un nuevo registro. **(3 puntos)**



¡Mucho éxito!