



LionMane Software, Inc.

World-class Software Development, Cloud, and QA Services

www.LionMane.com

PRUEBA ANGULAR

DESARROLLAR APLICACIÓN PARA VER RAZAS DE PERROS

INSTRUCCIONES

Desarrollar una aplicación web utilizando Angular y Node con Typescript. El propósito de esta aplicación será para explorar las diferentes razas de perros que nos proveen un API.

Deberás utilizar el API que te indicaremos para obtener la información de las diferentes razas de perros. Finalmente, tendrás que desarrollar la aplicación utilizando Angular y Node con Typescript.

Para obtener la información de las razas y sub-razas de perros, utilizarás el siguiente API: <https://dog.ceo/dog-api/documentation/>.

Este API categoriza los perros en razas (por ejemplo, *Hound*) y luego por sub-razas (por ejemplo, *Afghan Hound*, *English Foxhound*, etc). Es importante preservar esta jerarquía y mostrar las razas y sub-razas de perros estructuradamente.

Lineamientos de evaluación

1. No necesitas diseño gráfico, sin embargo la aplicación debe ser visualmente atractiva y organizada
2. La experiencia de usuario (UX) es importante
3. La calidad de tu código es muy importante
4. Deberás mostrar buenas prácticas de programación, y buenas prácticas utilizando Git

Instrucciones para entregar la aplicación

1. Crear un repositorio público en Github
 - a. Favor hacer múltiples commits durante el desarrollo de la aplicación. Mientras mas detallados, mejor.
2. Al finalizar la aplicación, enviar un email a daniela.salan@lionmane.io con el link de tu repositorio Github.
 - a. Incluir instrucciones de como instalar dependencias y correr la aplicación.

API

Deberás utilizar el siguiente API: <https://dog.ceo/dog-api/documentation/>

Estas son algunas de las rutas que provee el API:

- Listar todas las razas:
 - <https://dog.ceo/api/breeds/list/all>
- Listar todas las sub-razas de una raza:
 - [https://dog.ceo/api/breed/\\$RAZA/list](https://dog.ceo/api/breed/$RAZA/list)
 - Ejemplo:
 - \$RAZA = "hound"
 - <https://dog.ceo/api/breed/hound/list>
- Obtener todas las imágenes de una sub-raza:
 - [https://dog.ceo/api/breed/\\$RAZA/\\$SUB-RAZA/images](https://dog.ceo/api/breed/$RAZA/$SUB-RAZA/images)
 - Ejemplo:
 - \$RAZA = "hound"
 - \$SUB-RAZA = "afghan"
 - <https://dog.ceo/api/breed/hound/afghan/images>

REQUERIMIENTOS

Tu aplicación deberá constar con todos los siguientes requerimientos.

Requerimientos Visuales

Tu aplicación deberá constar con las siguientes dos pantallas como mínimo (puedes agregar más pantallas si crees que mejorarán la calidad de la aplicación):

1. Pantalla de Registración y Login

- a. Mostrar una opción para hacer login utilizando Email y Password
 - i. Mostrar un error si los credenciales utilizados no existen en el sistema o no están correctos
- b. Mostrar una opción para registrar un nuevo usuario usando Email y Password
 - i. Mostrar un error si el Email a registrar ya existe en el sistema
 - ii. Mostar un mensaje de éxito si la registración concluye con éxito

2. Pantalla de Inicio

- a. Mostrar el nombre de la aplicación
- b. Mostrar el nombre y una imagen de la sub-raza favorita, si es que una fue elegida
 - i. Esto únicamente si ya se cuenta con una sub-raza favorita. De lo contrario, mostrar un mensaje que aún no se a elegido una sub-raza favorita.
- c. Mostrar un listado vertical de todas las Razas de Perros
 - i. Al seleccionar una raza, desplegar la pantalla #2 para ver las sub-razas que conforman la raza de perros seleccionada.

3. Pantalla de Sub-razas de Perros

- a. Mostrar el nombre de la raza que se está visualizando
- b. Mostrar un listado de todas las sub-razas de perros que conforman esta raza. Por ejemplo, si la raza es "Hound", entonces las sub-razas incluirán: "Blood Hound", "Afghan Hound", "American Foxhound", etc.
 - i. Cada sub-raza en la lista debe tener los siguientes datos:

1. Un thumbnail (imagen) de la sub-raza
 2. Nombre de la sub-raza
- ii. Al pulsar una sub-raza, mostrar un popup que muestra lo siguiente:
 1. 3 imágenes de la sub-raza
 - a. Si hay menos imagenes disponibles, crear lógica para acomodar menos imagenes
 2. Nombre de la sub-raza
 3. Opción para seleccionar dicha sub-raza como la raza favorita

Requerimientos Back-end

Desarrollar un back-end muy sencillo utilizando Node que controle el registro de usuarios y el login.

Requerimientos Técnicos

- La aplicación debe ser escrita usando Angular, Node y desarrollado en Javascript o Typescript (de preferencia Typescript)
- Puedes utilizar cualquier librería que te facilite la implementación