

DIPLOMADO DE ACTUALIZACION EN NUEVAS TECNOLOGIAS PARA  
DESARROLLO DE SOFTWARE

TALLER UNIDAD 3 FRONTEND

PREPARADO POR:  
MILER ANDRES ESPAÑA  
COD: 2130341131

PREPARADO PARA:  
VICENTE AUX

UNIVERSIDAD DE NARIÑO  
INGENIERIA DE SISTEMAS  
ENERO 2024

1. Modificamos base de datos y BackEnd (Taller002Backend)

- a. Primero empezamos creando nuevos campos en nuestra base de datos llamada “mascotas”, creamos los campos de raza, tipo de mascota y estado de la mascota.

```
MySQL [mascotas]> describe mascotas;
```

Field	Type	Null	Key	Default	Extra
id	int	NO	PRI	NULL	auto_increment
nombre	varchar(100)	YES		NULL	
edad	int	YES		NULL	
estado_mascota	varchar(10)	NO		NULL	
tipo_mascota	varchar(10)	NO		NULL	
raza	varchar(20)	YES		NULL	

- b. Agregamos los campos agregados en la base de datos al modelo del proyecto BackEnd, en el archivo mascotasModelo.js

```
nombre:{
  type: Sequelize.STRING,
  allowNull: true
},
edad:{
  type: Sequelize.INTEGER,
  allowNull:true
},
estado_mascota:{
  type: Sequelize.STRING,
  allowNull: true
},
tipo_mascota:{
  type: Sequelize.STRING,
  allowNull: true
},
raza:{
  type: Sequelize.STRING,
  allowNull: false
}
```

- c. Ahora, actualizamos el archivo mascotasControler.js con los nuevos campos de la base de datos, dentro de las funciones de crear y actualizar.

```
const crear = (req,res)=>{
  if(!req.body.nombre){
    res.status(400).json({
      mensaje: "El nombre no puede estar vacio."
    });
    return;
  }
  const dataset={
    nombre: req.body.nombre,
    edad: req.body.edad,
    tipo_mascota: req.body.tipo_mascota,
    raza: req.body.raza
  };
};
```

```
const actualizar=(req,res)=>{
  const id= req.params.id;
  if(!req.body.nombre && !req.body.edad && !req.body.tipo_mascota && !req.body.raza ){
    res.status(400).json({
      mensaje: `No se encontraron Datos para Actualizar`
    });
    return;
  }
  else{
    const nombre= req.body.nombre;
    const edad=req.body.edad;
    const tipo_mascota= req.body.tipo_mascota;
    const raza= req.body.raza;
    mascotas.update({nombre,edad,tipo_mascota,raza},{where:{id}})
    .then((resultado)=>{
      res.status(200).json({
        mensaje: `Registro Actualizado`
      });
    });
  }
};
```

- d. Verificamos que los cambios estén funcionando correctamente

The screenshot displays a REST client interface with a 'Send Request' tab on the left and a response view on the right. The request is a POST to `http://localhost:8000/mascotas/crear` with a JSON body: `{ "nombre": "Stuart", "edad": 2, "tipo_mascota": "Gato", "raza": "Russian" }`. The response shows headers like `Content-Type: application/json; charset=utf-8` and a status of 200, with a JSON body: `{ "mensaje": "Registro creado correctamente" }`. A red box highlights the request body, and a red arrow points from it to the response body.

```
Send Request
POST http://localhost:8000/mascotas/crear HTTP/1.1
Content-Type: application/json

{
  "nombre": "Stuart",
  "edad": 2,
  "tipo_mascota": "Gato",
  "raza": "Russian"
}

4 Content-Type: application/json; charset=utf-8
5 Content-Length: 43
6 ETag: W/"2b-q67xZLRgZ8v2LE0Rt4QP7gs1ZYI"
7 Date: Wed, 10 Jan 2024 20:18:06 GMT
8 Connection: close
9
10 {
11   "mensaje": "Registro creado correctamente"
12 }
```

```

1  HTTP/1.1 200 OK
2  X-Powered-By: Express
3  Access-Control-Allow-Origin: *
4  Content-Type: application/json; charset=utf-8
5  Content-Length: 95
6  ETag: W/"5f-5GvjJgkY+/4jmFbaCJtvOnefa7w"
7  Date: Wed, 10 Jan 2024 20:20:45 GMT
8  Connection: close
9
10 {
11   "id": 14,
12   "nombre": "Stuart",
13   "edad": 2,
14   "estado_mascota": "",
15   "tipo_mascota": "Gato",
16   "raza": "Russian"
17 }

```

## 2. Modificamos el FrontEnd (mascotas-fe).

- a. Empezamos insertando una imagen de portada, el cual tendrá el nombre de la fundación de mascotas, debemos importar la imagen desde la carpeta images.

```

import portada from '../images/adopta.png';

<div className="App">
  <img
    src={portada}
    alt="Portada de la página"
    style={{ height: '250px', width: '100%' }}
  />

```




- b. Ahora modificamos el modal del botón “añadir mascota”, para que reciba los nuevos campos creados en la base de datos, ya hacemos las validaciones pertinentes de nuestro modal.

```
<input
  type="text"
  id="tipo_mascota"
  className="form-control"
  placeholder="Raza Mascota"
  value={raza}
  onChange={(e)=>setRaza(e.target.value)}
></input>
</div>
<div className="input-group mb-3">
  <span className="input-group-text">
    <i className="fa-solid fa-gift"></i>
  </span>
  <input
    type="text"
    id="tipo_mascota"
    className="form-control"
    placeholder="Perro o Gato"
    value={tipo_mascota}
    onChange={(e)=>setTipo_mascota(e.target.value)}
  ></input>
```

### Registrar Mascota


	Nombre
	Edad
	Raza Mascota
	Perro o Gato

 Guardar


Cerrar


- c. Hacemos lo mismo para editar el modal del botón “editar mascota” para que modifique los nuevos campos creados en la base de datos, ya hacemos las validaciones pertinentes de nuestro modal.

### Editar Mascota

 Stuart

 2

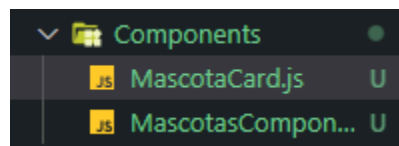
 Raza Mascota

 Perro o Gato

Guardar

Cerrar

- d. Ahora, creamos las tarjetas donde se mostrarán las mascotas que tenemos en la base de datos, creamos un nuevo archivo llamado mascotasCard.js dentro de la carpeta Components. Importamos React, Bootstrap y las imágenes que tendrán las tarjetas.



```
// MascotaCard.js
import React, {useState} from "react";
import 'bootstrap/dist/css/bootstrap.min.css';
import laica from '../images/laica.png';
```

- e. Creamos la tarjeta con los parámetros traídos desde la base de datos y le damos formato y estilo con Bootstrap.

```
return (
  <div className="card" style={{ width: '15rem' }}>
    <img src={laica} className="card-img-top" alt={mascota.nombre} />
    <div className="card-body">
      <h5 className="card-title">Nombre: {mascota.nombre}</h5>
      <h5 className="card-title">Raza: {mascota.raza}</h5>
      <h6 className="card-text">{mascota.tipo_mascota}</h6>
      <p className="card-text">Edad: {mascota.edad}</p>
      <button
        onClick={()=>openModal(mascota.id,mascota.nombre,mascota.edad,mascota.raza)}
        className="btn btn-primary"
        data-bs-toggle="modal"
        data-bs-target="#modalDetalle">
        Detalles
      </button>
    </div>
  </div>
)
```

Exportamos el componente de la tarjeta creada.

```
export default MascotaCard;
```

- f. Importamos el componente de la tarjeta creada y llamamos la tarjeta desde el componente principal mascotasComponente.js, las tarjetas ya vienen con la información como:

- Nombre
- Raza
- Tipo de mascota
- Edad

de cada una de las mascotas, así que, no tenemos que volver a incluirla.

```
import MascotaCard from "../MascotaCard.js";
```

```
<div className="container">
  <div className="row mt-5">
    {mascotas.map(mascota => (
      <div key={mascota.id} className="col-md-3 mb-3">
        <MascotaCard mascota={mascota}/>
        <div className="col-md-9 mb-9">
          <button
            onClick={()=>openModal(2,mascota.id,mascota.nombre,mascota.edad,mascota.raza)}
            className="btn btn-warning col-md-4 mb-4"
            data-bs-toggle="modal"
            data-bs-target="#modalMascotas"
            ><i className="fa-solid fa-edit"></i>
          </button>
        </div>
      </div>
    )
  </div>
</div>
```

+Añadir Mascota



Nombre: Laika  
Raza: Golden  
Perro  
Edad: 4

Detalles



Nombre: Pancho  
Raza: Angora  
Gato  
Edad: 2

Detalles



Nombre: Pepe  
Raza: Comun  
Gato  
Edad: 5

Detalles



Nombre: Kaiser  
Raza: Bernes  
Perro  
Edad: 2

Detalles



- g. Finalmente, agregamos el modal del botón “Detalles” de las mascotas, en el archivo mascotasCard.js, el modal tendrá el nombre y la raza de la mascota traídos desde la base de datos.

Creamos la variable OpenModal para después ejecutarla.

```
const [titulo, setTitulo]=useState("");
const openModal =()=>{
    setTitulo("Detalles");
};
```

```
<div id="modalDetalle" className="modal fade" aria-hidden="true">
<div className="modal-dialog">
  <div className="modal-content">
    <div className="modal-header">
      <label className="h5 text-start">Nombre: {mascota.nombre}</label>
      <label className="h5">Raza: {mascota.raza}</label>
    </div>
    <div className="modal-body">
      <div className="input-group mb-3">
        <span>
          <h5>{mascota.nombre} es una de las mascotas mas queridas y juguetonas
            de nuestro centro de adopcion, es un {mascota.raza} de {mascota.edad} años de edad,
            cuenta con todas las vacunas necesarias para su crecimiento y protección</h5>
          <h4>esta en busca de un dulce hogar!</h4>
        </span>
      </div>
      <div className="d-grid col-6 mx-auto">
        <button
          id="btnCerrarModal"
          type="button"
          className="btn btn-success"
          data-bs-dismiss="modal">
```

Nombre: Laika

Raza: Golden

Laika es una de las mascotas mas queridas y juguetonas de nuestro centro de adopcion, es un Golden de 4 años de edad, cuenta con todas las vacunas necesarias para su crecimiento y protección.

esta en busca de un dulce hogar!

Cerrar