



**UNIVERSIDAD  
AUTONOMA DE  
SINALOA**



## **FACULTAD DE INFORMATICA CULIACAN**

**Tema: CRUD COMPLETO**

**Materia: DESARROLLO WEB**

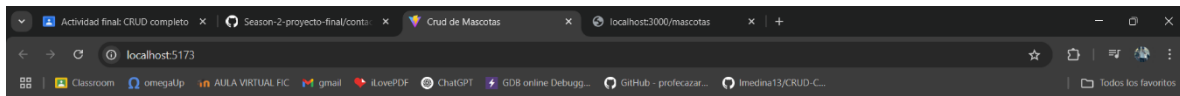
**Profesor: JOSE MANUEL CAZAREZ  
ALDERETE**

**Alumno: Medina Cebreros Luis Angel**

**Grupo: 2-1**

**Fecha: 30/MAYO/2024**

## EVIDENCIA DE LA EJECUCION



### CRUD de Mascotas

Nombre  Edad

#### Listado de Mascotas

• koki -- 12 años



Se inicia el express, cors, y son configurados para recibir las petición y se inicia el servidor de igual manera

```
JS app.js > ...
1  const express = require('express');
2  const bodyParser = require('body-parser');
3  const cors = require('cors');
4  const mascotas = require('./mascotas');
5  const puerto = 3000;
6
7  const app = express();
8  app.use(bodyParser.json());
9  app.use(cors());
10
11 app.listen(puerto, () => {
12   console.log('Servicio de mascotas iniciado en el puerto', puerto);
13 });
14
```

Esta parte sirve para insertar una nueva mascota a la base de datos y el otro para regresar todas las mascotas que ya han sido ingresadas

```

14
15 // Crear mascota
16 app.post('/mascotas', async (req, res) => {
17   const { nombre, tipo, edad } = req.body;
18   const data = await mascotas.create({
19     nombre, tipo, edad
20   });
21   res.send(data);
22 });
23
24 // Obtener todas las mascotas
25 app.get('/mascotas', async (req, res) => {
26   const data = await mascotas.findAll();
27   res.send(data);
28 });
29

```

El primero servirías para modificar a la mascota seleccionada y el otro para poder eliminarla

```

30 // Actualizar mascota
31 app.put('/mascotas/:id', async (req, res) => {
32   const { nombre, tipo, edad } = req.body;
33   const { id } = req.params;
34   const data = await mascotas.update({
35     nombre, tipo, edad
36   }, {
37     where: { id }
38   });
39   res.send(data);
40 });
41
42 // Eliminar mascota
43 app.delete('/mascotas/:id', async (req, res) => {
44   const { id } = req.params;
45   const data = await mascotas.destroy({
46     where: { id }
47   });
48   res.send(data);
49 });
50

```

En esta imagen se ve como esta definida la estrucutra de la tabla de la base de datos de las mascotas

```

JS mascotas.js > Mascota > id
1  const { DataTypes } = require('sequelize');
2  const sequelize = require('./conexion');
3
4  const Mascota = sequelize.define('Mascota', {
5    id: {
6      type: DataTypes.INTEGER,
7      primaryKey: true,
8      autoIncrement: true
9    },
10   nombre: {
11     type: DataTypes.STRING,
12     allowNull: false
13   },
14   tipo: {
15     type: DataTypes.STRING,
16     allowNull: false
17   },
18   edad: {
19     type: DataTypes.INTEGER,
20     allowNull: false
21   }
22 }, {
23   tableName: 'mascotas',
24   timestamps: false
25 });
26
27 module.exports = Mascota;
28

```

En esta sería la conexión con la base de datos sequelize

```

1  const Sequelize = require('sequelize');
2
3  const sequelize = new Sequelize({
4    dialect: 'sqlite',
5    storage: 'mascotas.sqlite'
6  });
7
8  module.exports = sequelize;

```

permite agregar o actualizar datos de una mascota, manejando su estado y cambios según si está en modo edición o creación.

```

function MascotaForm({ onAgregar, onActualizar, editando }) {
  const [form, setForm] = useState({ nombre: '', especie: '', edad: '' });

  useEffect(() => {
    if (editando) setForm(editando);
  }, [editando]);

  const handleChange = (e) =>
    setForm({ ...form, [e.target.name]: e.target.value });

  const handleSubmit = (e) => {
    e.preventDefault();
    if (editando) {
      onActualizar(editando.id, form);
    } else {
      onAgregar(form);
    }
    setForm({ nombre: '', especie: '', edad: '' });
  };
}

```

Esta muestra una lista de mascotas con botones para editar o eliminar cada una, y registra en consola el array de mascotas recibido.

```

function MascotaList({ mascotas, onEditar, onEliminar }) {
  console.log("Mascotas:", mascotas);
  return (
    <div>
      <h2>Listado de Mascotas</h2>
      <ul>
        {mascotas.map((mascota) => (
          <li key={mascota.id}>
            {mascota.nombre} - {mascota.especie} - {mascota.edad} años
            <button onClick={() => onEditar(mascota)}>Editar</button>
            <button onClick={() => onEliminar(mascota.id)}>Eliminar</button>
          </li>
        ))}
      </ul>
    </div>
  );
}

export default MascotaList;

```