"Sistema para el monitoreo, detección y alerta de somnolencia del conductor mediante visión artificial, comunicación inalámbrica y geolocalización"

Primer Reporte Parcial

Lista de actividades

- Definir rutas del frontend
- Diseño de rutas del backend
- Conexión Backend con Mongo DB
- Sistema de acceso con credenciales
- Creación de la base de datos no relacional
- Investigación de modelos de Redes Neuronales Convolucionales
- Diseño de una red neuronal convolucional capaz de detectar ojos cerrados y abiertos

Autores:
Alan Eduardo Gamboa Del Ángel
Maite Paulette Díaz Martínez

Asesores:
M.en C. Niels Henrik Navarrete Manzanilla
Dr. Rodolfo Vera Amaro

30 de Noviembre 2022

Índice

Índice de figuras

Índice de tablas

1. Definir rutas del frontend

1.1. Objetivo

Definir e implementar las rutas que tendrá la aplicación, así como si serán públicas o privadas y la información que se desplegará en cada una de las mismas.

1.2. Descripción

Como primer paso, se necesita crear un proyecto de React utilizando el siguiente comando:

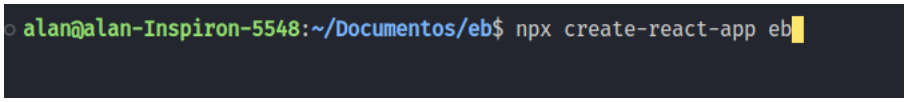

Figura 1: Creación proyecto de react

Posteriomente, se realiza la instalación del moudelo *React Router Dom* utilizando el siguiente comando:

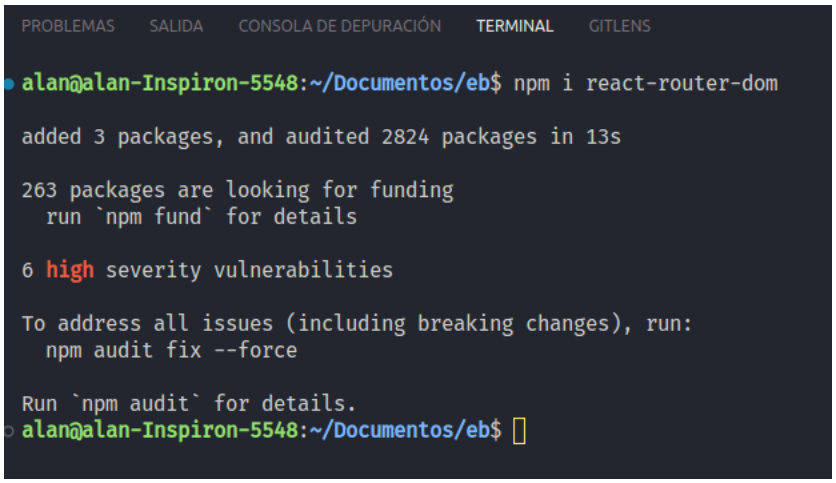

Figura 2: Creación proyecto de react

Dentro de nuestro proyecto creado, ingresamos al archivo *index.js*. Dentro de este archivo, importamos dos librerías ofrecidas por el módulo de *react router dom* utilizando la línea de código siguiente:

Figura 3: Creación proyecto de react

1.3. Resultados

2. Definir rutas del backend

2.1. Objetivo

Crear las rutas mediante las que el cliente realizará las peticiones y tendrá acceso a las operaciones, así como su funcionamiento en cuanto a obtención de datos y comunicación con el resto de la aplicación.

2.2. Descripción

2.3. Resultados

3. Conexión Backend con Mongo DB

3.1. Objetivo

Realizar la conexión de NodeJs con la base de datos MongoDb.

3.2. Descripción

Como primer paso, se debe ingresar a la página web `https:\mongodb.com` y registrarse.

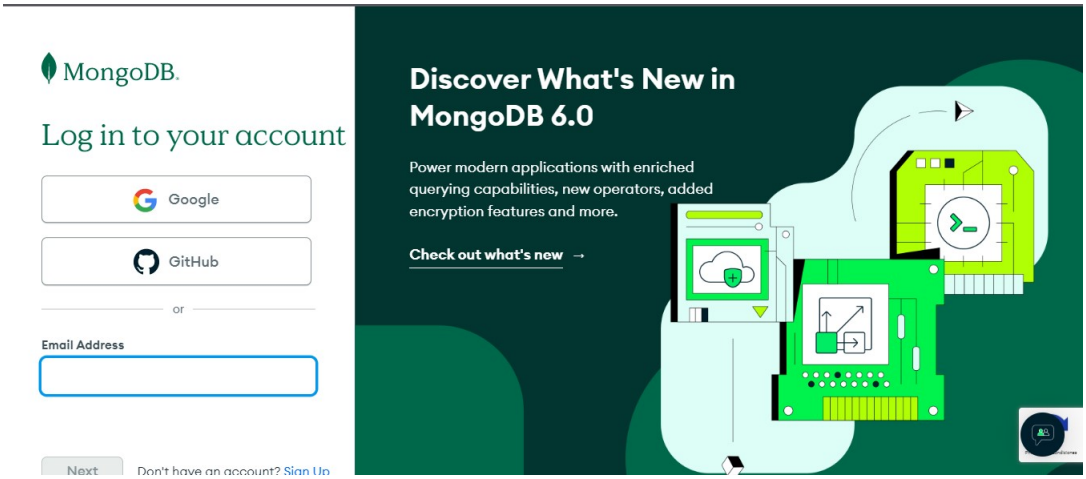

Figura 4: Página web MongoDb Atlas.

Después de haber iniciado sesión, daremos click al botón *Create* para crear un cluster al cuál podremos conectarnos para utilizar los servicios de Mongo Atlas.

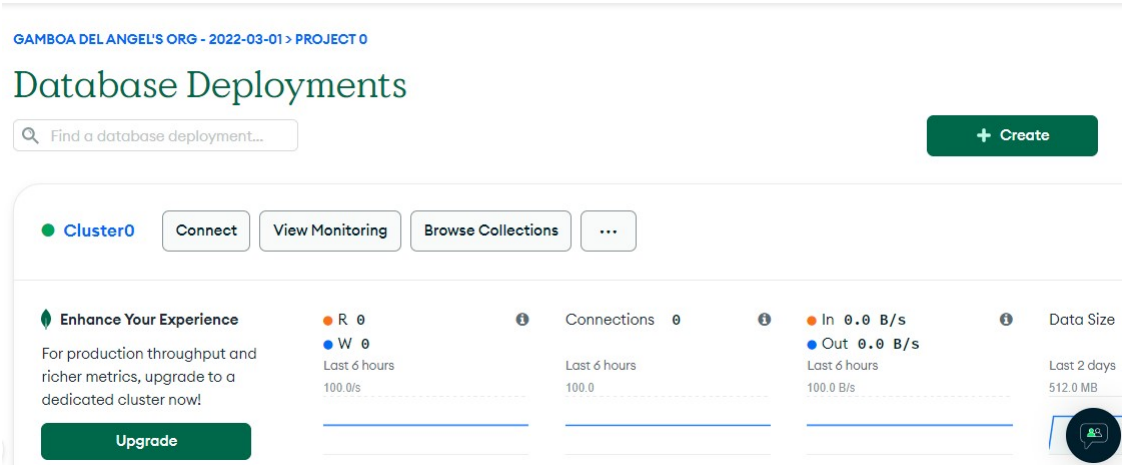

Figura 5: Página Principal MongoDB

Posteriormente, se seleccionarán los ajustes del cluster a crear, será un cluster compartido ya que esta opción es grátis.

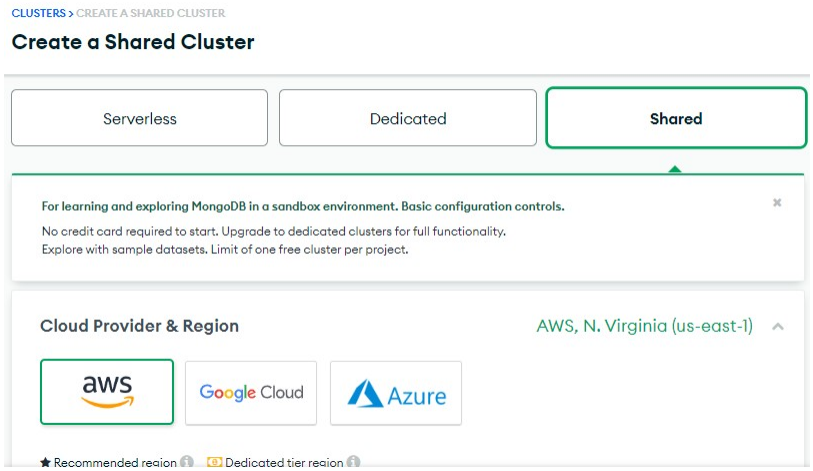

Figura 6: Página Creación de Cluster

De igual manera, se mantendrán los valores por default en cuanto al almacenamiento y la versión de Mongo a utilizar.

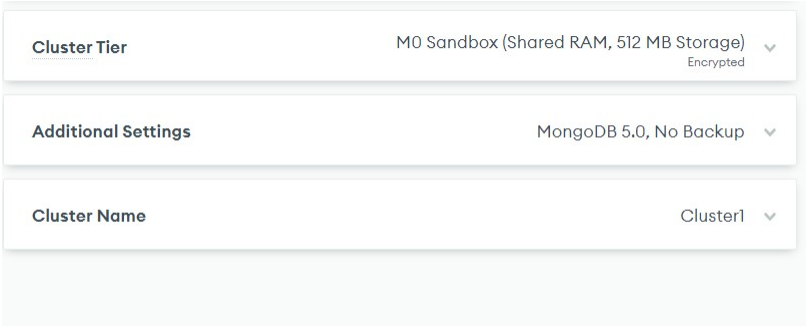

Figura 7: Página Creación de Cluster

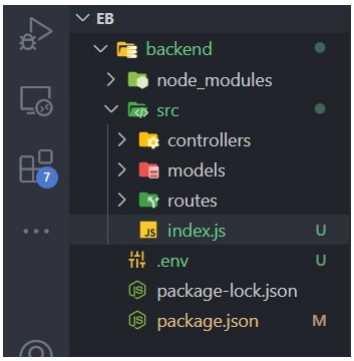

Figura 8: Directorio del Backend

backend > src > index.js > ...

```
const express = require("express");
const mongoose = require("mongoose");
require('dotenv').config({ path: 'env' });

const app = express();
const port = process.env.PORT || 9000;
const URI = 'mongodb+srv://root:250997@cluster0.zrhyx.mongodb.net/?retry

//iniciar servidor
app.get("/", (req,res) =>  {
        res.send("Servidor Listo")
});
```

Figura 9: Directorio del Backend

```
//conexion mongo
mongoose.set("strictQuery", true);
mongoose.connect(URI, {
        useNewUrlParser: true,
        useUnifiedTopology: true
}).then(()=> console.log("Conexión a mongo exitosa"))
.catch((error) => console.log(error));

//levantar servidor
app.listen(port,()=> console.log("servidor escuchando en el puerto", por
```

Figura 10: Directorio del Backend

3.3. Resultados

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER COMMENTS node

```
C:\Users\alang\Documents\TT\eb\backend>npm run start

> backend@1.0.0 start
> nodemon src/index.js

[nodemon] 2.0.20
[nodemon] to restart at any time, enter `rs`
[nodemon] watching path(s): *.*
[nodemon] watching extensions: js,mjs,json
[nodemon] starting `node src/index.js`
servidor escuchando en el puerto 9000
Conexión a mongo exitosa
```

Figura 11: Directorio del Backend

4. Sistema de acceso con credenciales

4.1. Objetivo

Establecer los roles de cada tipo de usuario con sus respectivos permisos de acceso a la aplicación web utilizando Amazon Cognito

4.2. Descripción

4.3. Resultados

5. Creación de la base de datos No Relacional

5.1. Objetivo

Crear la base de datos en MongoDB.

5.2. Descripción

5.3. Resultados

6. Investigación de modelos de Redes Neuronales Convolucionales

6.1. Objetivo

Determinar distintos modelos de redes neuronales convolucionales que ofrezcan mejor eficiancia al clasificar imágenes.

6.2. Descripción

6.3. Resultados

7. Diseño de una red neuronal convolucional capaz de detectar ojos cerrados y abiertos

7.1. Objetivo

Diseñar y realizar pruebas de los modelos de redes neuronales convolucionales previamente investigados para determinar el rendimiento y la precisión de cada uno.

7.2. Descripción

7.3. Resultados

8. Conclusiones

Para el desarrollo de la Estación base, se decidió utilizar la suite de herramientas de Amazon Amplify. Se hará uso de Amplify Hosting, el cuál tiene una integración directa con Github, esto quiere decir, que los cambios que se realicen en el repositorio se reflejarán de manera automática en la aplicación web. Además, AWS Amplify ofrece su servicio de almacenamiento en la nube S3, este será de gran ayuda para almacenar contenido multimedia, en este caso los videos de incidencia de los conductores. Para el manejo de credenciales, se utilizará Amazon Cognito, que se encargará de administrar las credenciales de acceso a la aplicación. Para el manejo de datos, se estará utilizando MongoDB, un manejador NoSQL que trabaja con documentos. Para el desarrollo de la aplicación, se decidió utilizar el lenguaje de programación Javascript, junto con NodeJs que nos ayudará a manejar varias peticiones al mismo tiempo. Finalmente el análisis del sistema de comunicaciones,en un principio sólo se había contemplado el análisis de telemetría pero al ir realizando las actividades del tercer reporte e ir profundizando en algunos temas de comunicaciones, se decidió que también que se necesitaba el análisis de la cobertura y de los datos, es decir,de la transmisión de los fotogramas por lo que se incluyeron también en este reporte.

9. Bibliografia

Referencias