



5B PROGRAMACIÓN

ACCIONES CRUD EN MONGODB

BASE DE DATOS PARA UNA RED
SOCIALES COLAR



5B PROGRAMACIÓN



OBJETIVO DEL PROYECTO



El objetivo de este proyecto es diseñar, configurar y manipular una base de datos NoSQL utilizando MongoDB Atlas, demostrando la correcta implementación de las cuatro operaciones CRUD: Crear, Leer, Actualizar y Eliminar.

A través de este trabajo se busca fortalecer las habilidades técnicas en el manejo de bases de datos en la nube y la integración con una aplicación web desarrollada con el stack MERN (MongoDB, Express, React, Node.js).



5B PROGRAMACIÓN



TEMA DE LA BASE DE DATOS

TEMA ELEGIDO: RED SOCIAL
ESCOLAR

El tema de la base de datos se centra en una red social diseñada para estudiantes, docentes y personal administrativo de una institución educativa.

La plataforma permite compartir publicaciones, comentar, y gestionar perfiles de usuario, fomentando la comunicación interna y el aprendizaje colaborativo.



5B PROGRAMACIÓN

DISEÑO DEL MODELO DE DATOS

La base de datos cuenta con las siguientes colecciones principales:

- **Usuarios**

Campos: _id, nombre, correo, contraseña, rol, fechaRegistro

- **Publicaciones**

Campos: _id, titulo, contenido, autor, fechaPublicacion, likes

- **Comentarios**

Campos: _id, texto, autor, postId

```
← → social-network-cbtis-copia 2 - copia 4
auth.routes.js post.routes.js index.js post.controller.js post.model.js X
| > features > posts > post.model.js > ...
import mongoose from 'mongoose';

const postSchema = new mongoose.Schema({
  title: {
    type: String,
    default: '',
    trim: true,
  },
  content: {
    type: String,
    required: true,
  },
  author: {
    type: mongoose.Schema.Types.ObjectId,
    ref: 'User', // Regresa a 'User'
    required: true,
  },
  createdAt: {
    type: Date,
    default: Date.now,
  }
});

PROBLEMAS CONSOLA DE DEPURACIÓN TERMINAL PUERTOS

Users\aldoa\Documents\social-network-cbtis-copia 2 - copia 4> npm start
vite

ioredis 3.1.10
ioredis to restart at any time, enter `rs`
ioredis watching path(s): *.*
ioredis watching extensions: js,mjs,cjs,json
ioredis starting `node server.js`

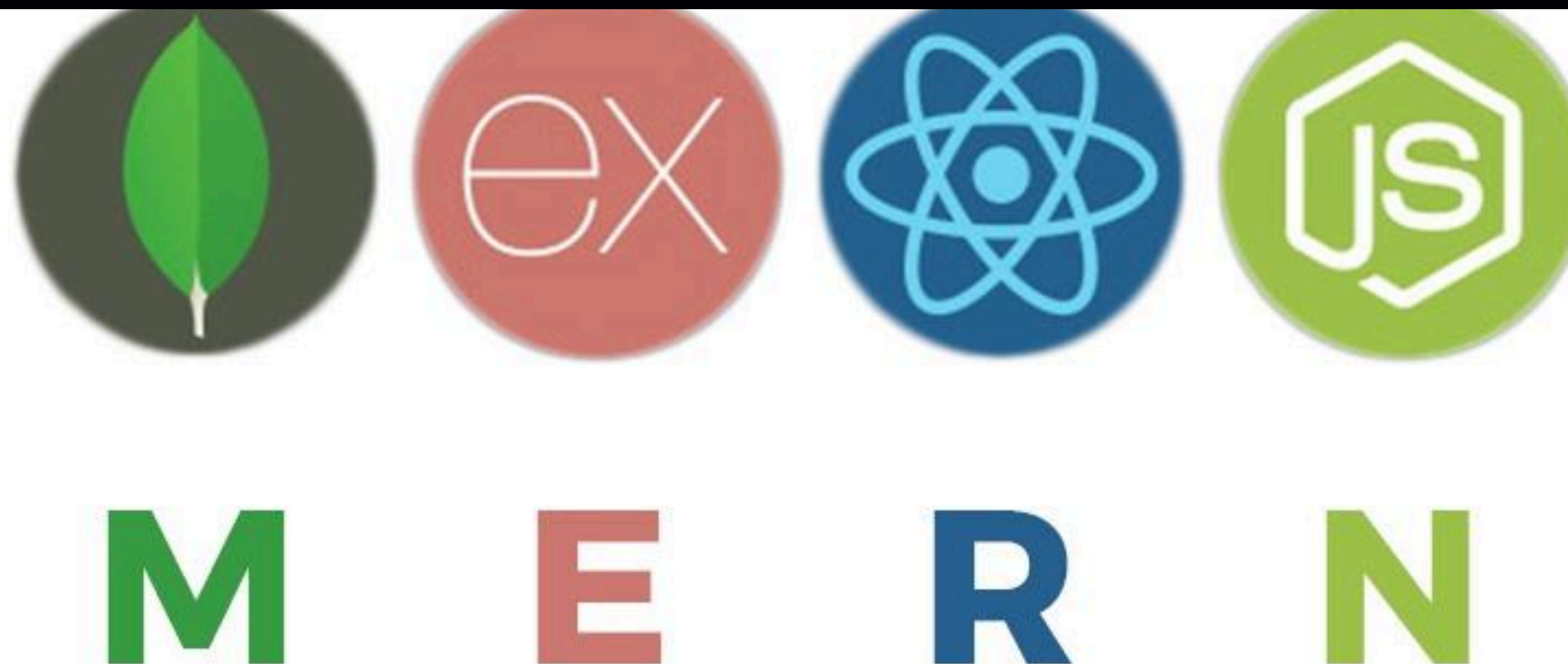
VITE v5.4.19 ready in 1734 ms

→ Local: http://localhost:5173/
→ Network: use --host to expose
rvidor corriendo en el puerto 3000
ngoDB conectado: localhost
```



5B PROGRAMACIÓN

STACK TECNOLÓGICO



CONFIGURACIÓN EN MONGODB ATLAS

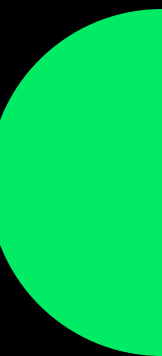


Configuración en MongoDB Atlas

```
mongodb+srv://usuario123:password@cluster0.mongodb.net/redsocial
```

Se creó un cluster en MongoDB Atlas para almacenar la base de datos de la red social.

- Se configuró un usuario con permisos de lectura y escritura
- Se añadió la IP local a la whitelist
- Se obtuvo la cadena de conexión (URI) para enlazar con la aplicación





CONEXIÓN DESDE NODE.JS

```
import mongoose from 'mongoose';

mongoose.connect('mongodb+srv://usuario123:password@cluster0.mongodb.net/redsocial')
  .then(() => console.log('Conexión exitosa a MongoDB Atlas'))
  .catch(err => console.error('Error de conexión:', err));
```



SECCIÓN CRUD

```
const nuevoPost = new Post({  
  titulo: "Nueva publicación",  
  contenido: "Este es mi primer post.",  
  autor: "Aldo Ayala"  
});  
await nuevoPost.save();
```

```
// Consultar todos los posts  
const posts = await Post.find();  
  
// Consultar posts de un autor específico  
const postsAutor = await Post.find({ autor: "Aldo Ayala" });
```

```
await Post.updateOne(  
  { titulo: "Nueva publicación" },  
  { $set: { likes: 20 } }  
);
```

```
await Post.deleteOne({ titulo: "Nueva publicación" });
```

