Observar el archivo database.js:

```
export const connect = async () => {
    await mongoose.connect('mongodb://localhost/geometria', {
        useNewUrlParser: true,
        useUnifiedTopology: true,
        useCreateIndex: true, // Para que cree el índice único asociado al campo unique
        useFindAndModify: false // para usar findOneAndDelete y findAndModify
    })
    .catch( (error) => {throw 'Error: '+ error}
    )
}

export const disconnect = async () => {
    try {
        await mongoose.disconnect();
    }
    catch (error){
        throw 'Error: ' + error;
    }
}
```

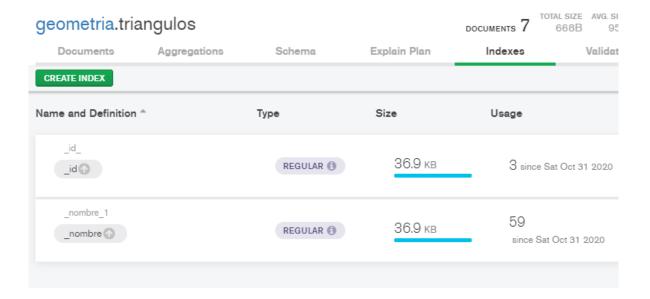
Hemos quitado la escritura en consola con control.log, levantando un error en su lugar y hemos añadido algunos modificadores, entre ellos el relacionado con la creación de índices.

Hemos modificado el Schema para incluir validaciones

Dos tipos de validaciones:

Una llamada unique. Esta crea un índice único en la colección, llamado _nombre_1 para el campo name:

El índice lo podemos ver también con Compass:



Las otras validaciones max y min.

Ver la documentación de las validaciones de mongoose:

https://mongoosejs.com/docs/validation.html

Para que también salten las validaciones en update además de en las inserciones de nuevos documentos hay que añadir la opción "runvalidators: true" en los métodos de update. Lo vemos en la opción 7 del menú:

```
case 7:
    await connect()

// Controlamos el error de validación

// Recordar que hay que poner la opción useFindAndModify: false

await Triangulos.findOneAndUpdate(
    { _nombre: triangulo.nombre },
    {
        _nombre: triangulo.nombre,
        _base: triangulo.lado2,
        _lado2: triangulo.lado2,
        _lado3: triangulo.lado3,
        _altura: triangulo.altura
    },
    {
        runValidators: true // para que se ejecuten las validaciones del Schema
    }
)
    .then(() => console.log('Modificado Correctamente') )
    .catch( (err) => console.log('Error: '+err)) // concatenando con cadena muestra mensaje
    await disconnect()
    break
```

Observar cómo capturamos el error de la validación. Lo hacemos con .catch()

Algo similar hacemos con el método save, para crear un nuevo documento en la colección, ver la opción 5 del menú.