

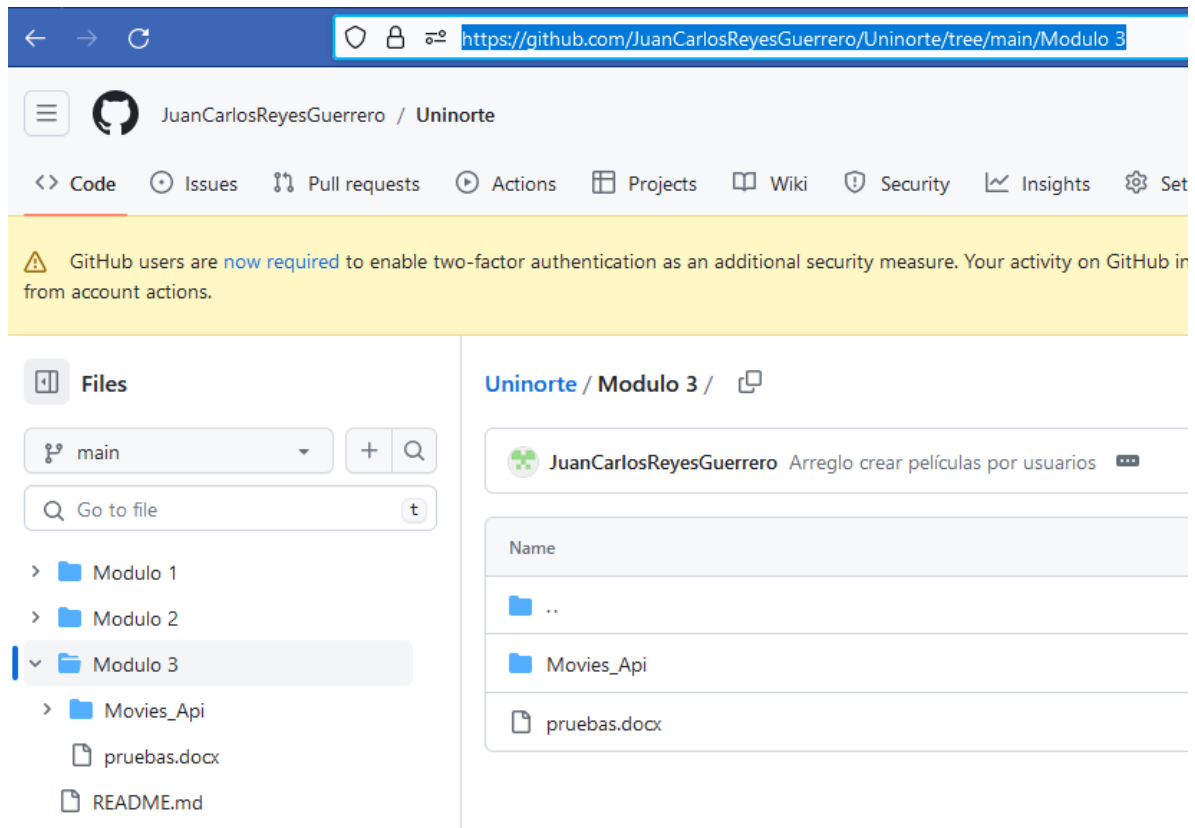
MODULO 3

ServicioWeb: My 100 Movie List

Presentado Por:
Juan Carlos Reyes

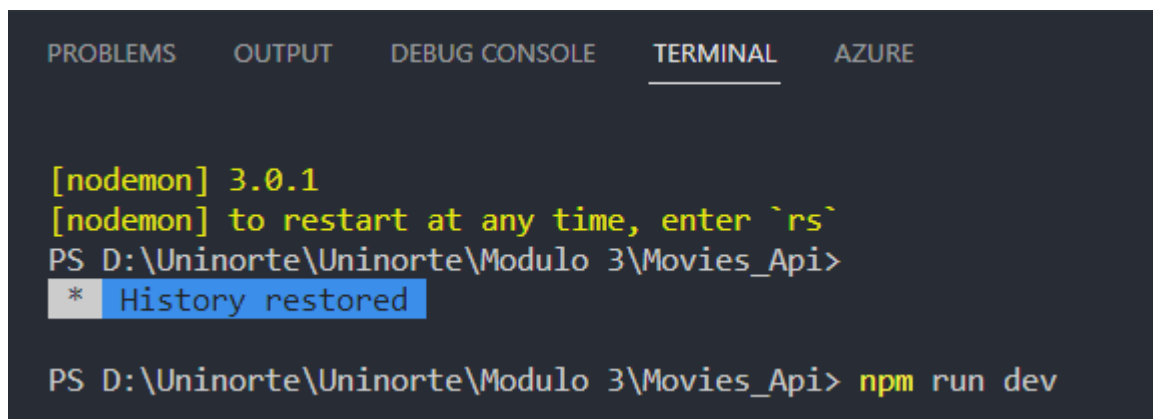
1. Ruta Entrega Módulo 3

<https://github.com/JuanCarlosReyesGuerrero/Uninorte/tree/main/Modulo%203>



2. Correr el proyecto

npm run dev



```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  AZURE

[nodemon] to restart at any time, enter `rs`
[nodemon] watching path(s): *.*
[nodemon] watching extensions: js,mjs,cjs,json
[nodemon] starting `node src/index.js`
[Function: verifyToken]
servidor corriendo en: 9000
Base de datos conectada
```

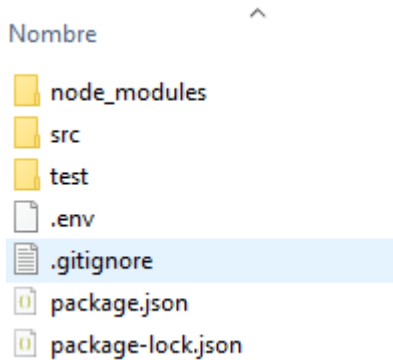
3. Version instalada de NodeJs

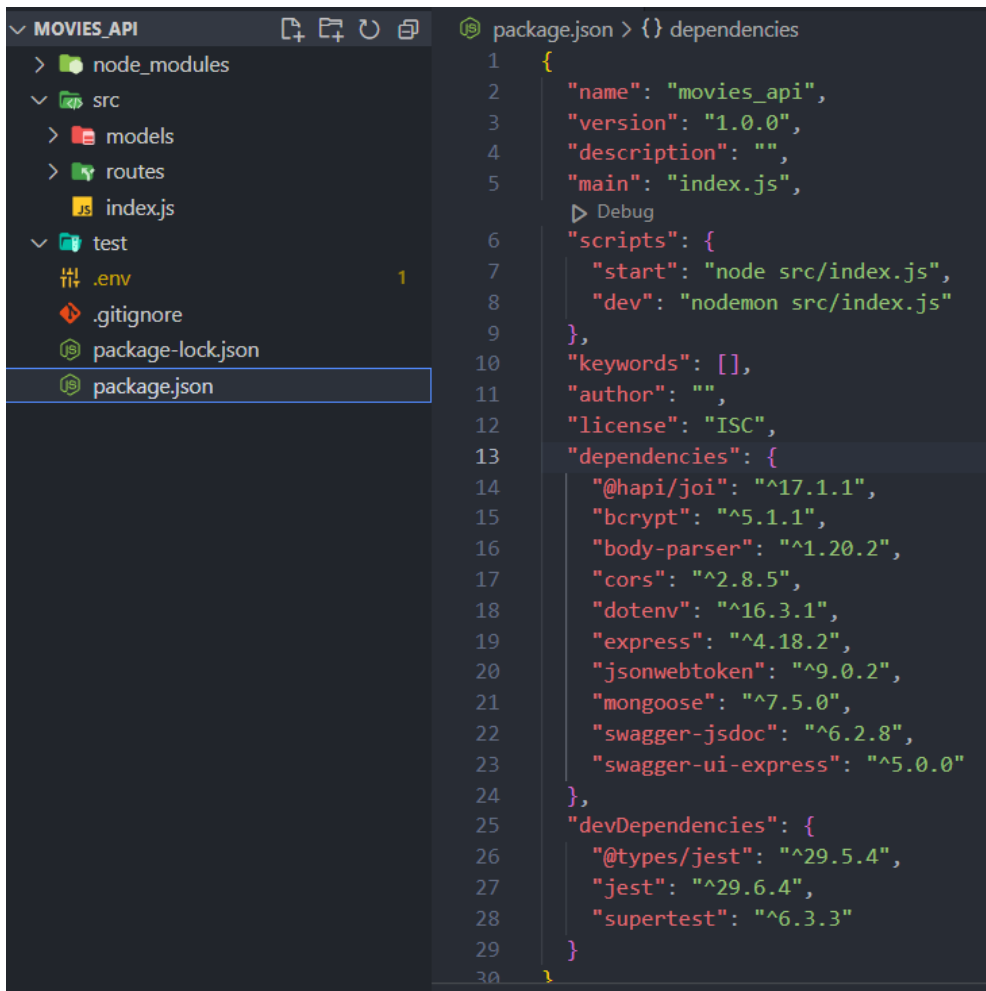
```
C:\> Símbolo del sistema

C:\Users\quind>node --version
v16.18.0

C:\Users\quind>
```

4. Configuración del Proyecto





```
package.json > {} dependencies
1  {
2    "name": "movies_api",
3    "version": "1.0.0",
4    "description": "",
5    "main": "index.js",
6    "scripts": {
7      "start": "node src/index.js",
8      "dev": "nodemon src/index.js"
9    },
10   "keywords": [],
11   "author": "",
12   "license": "ISC",
13   "dependencies": {
14     "@hapi/joi": "^17.1.1",
15     "bcrypt": "^5.1.1",
16     "body-parser": "^1.20.2",
17     "cors": "^2.8.5",
18     "dotenv": "^16.3.1",
19     "express": "^4.18.2",
20     "jsonwebtoken": "^9.0.2",
21     "mongoose": "^7.5.0",
22     "swagger-jsdoc": "^6.2.8",
23     "swagger-ui-express": "^5.0.0"
24   },
25   "devDependencies": {
26     "@types/jest": "^29.5.4",
27     "jest": "^29.6.4",
28     "supertest": "^6.3.3"
29   }
30 }
```

5. Funciones controladoras del Proyecto

```
// import routes
const authRoutes = require('./routes/auth');
const movieRoutes = require('./routes/movie');
const movieByUserRoutes = require('./routes/movie-by-user');
const verifyToken = require('./routes/validate-token');
app.use("/api-doc", swaggerUI.serve, swaggerUI.setup(swaggerJsDoc(swaggerSpec)))
```

6. Middlewares del Proyecto

```
// route middlewares
app.use('/api/user', authRoutes);
app.use('/api/movie', verifyToken, movieRoutes);
app.use('/api/movie-by-user', verifyToken, movieByUserRoutes);
```

7. Autenticación y autorización del Proyecto

```
const jwt = require('jsonwebtoken')

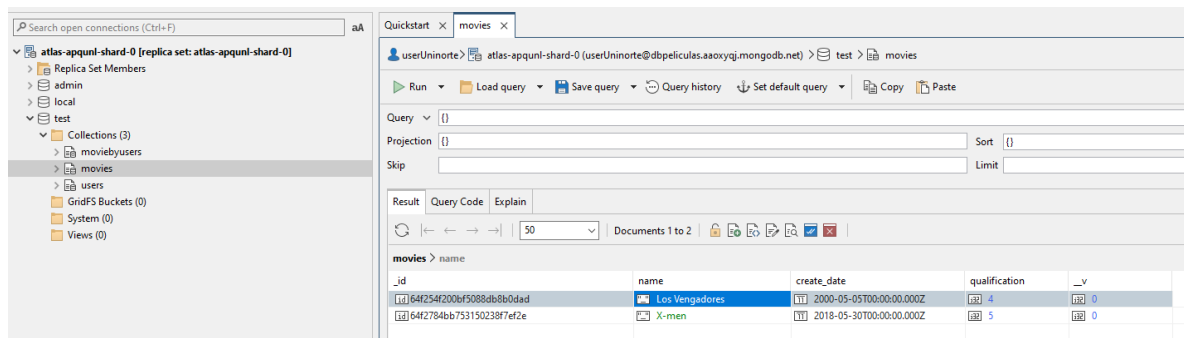
// middleware para validar token (rutas protegidas)
const verifyToken = (req, res, next) => {
  const token = req.header('auth-token')
  if (!token) return res.status(401).json({ error: 'Acceso denegado' })
  try {
    const verified = jwt.verify(token, process.env.TOKEN_SECRET)
    req.user = verified
    next()
  } catch (error) {
    res.status(400).json({error: 'token no es válido'})
  }
}

module.exports = verifyToken;
```

8. Conexión a la base de datos

```
// Conexión a Base de datos
//const uri = `mongodb+srv://${process.env.USER}:${process.env.PASSWORD}@`
const uri = process.env.MONGODB_URI;
mongoose.connect(uri,
  { useNewUrlParser: true, useUnifiedTopology: true }
)
.then(() => console.log('Base de datos conectada'))
.catch(e => console.log('error db:', e))
```

La base de datos se encuentra en un repositorio de MongoDB Atlas en la Nube



9. Herramienta para pruebas de funcionamiento de APIs

Se utilizó el software Postman

Overview

POST https://localhost:443/ GET http://localhost:9000/api/

+ ...

HTTP uninorte / http://localhost:9000/api/movie-by-user/

GET http://localhost:9000/api/movie-by-user/

Params Authorization Headers (9) Body ● Pre-request Script Tests Settings

Headers 8 hidden

	Key	Value
<input checked="" type="checkbox"/>	auth-token	eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXV
	Key	Value

Body Cookies Headers (7) Test Results

Pretty Raw Preview Visualize JSON

```
1  {
2    "error": null,
3    "data": {
4      "title": "Películas por usuario",
5      "movies": [
6        {
7          "_id": "64f7290f418fac0fd338c662",
8          "id_user": "64f135fd79fd42e25ba7a688",
9          "id_movie": "64f254f200bf5088db8b0dad",
10         "name_user": "Maria Madroñero",
11         "name_movie": "Los Vengadores",
12         "qualification": 3,
13         "date": "2023-09-05T13:11:43.420Z",
14         "__v": 0
15       }
16     ]
17   }
18 }
```

```
Windows PowerShell
Terminal Windows se puede establecer como la aplicación de terminal predeterminada en la configuración. Abrir Configuración

PS D:\Cursos\Uninorte\Uninorte\Modulo 3\Peliculas_API> npm test

> peliculas@1.0.0 test
> jest

PASS test/api/users.route.spec.js
Pruebas sobre la API de users
  GET /api/users
    ✓ La ruta funciona (53 ms)
    ✓ La petición nos devuelve un array de users (13 ms)
  POST /api/users
    ✓ La ruta funcione (35 ms)
    ✓ Se inserta correctamente (10 ms)
    ✓ Error en la inserción (8 ms)
  PUT /api/users
    ✓ La ruta funciona (24 ms)
    ✓ Se actualiza correctamente (15 ms)
  DELETE /api/users
    ✓ La ruta funciona (10 ms)
    ✓ Borra correctamente (15 ms)

Test Suites: 1 passed, 1 total
Tests: 9 passed, 9 total
Snapshots: 0 total
Time: 1.771 s, estimated 3 s
Ran all test suites.
PS D:\Cursos\Uninorte\Uninorte\Modulo 3\Peliculas_API>
```

10. La documentación de las apis se realizó con Swagger

