

Crud NodeJS

(Part two - Crud NodeJS with Mongo DB)

Versión : 0.1.1(20190303)

Author :

Ing. Duván Albeiro Mejia Cortes (software.bugs.io@gmail.com)

Objective:

In this short course (tutorial) we are going to build a Crud NodeJS web application with one NoSQL (MongoDB). To access the database we will use JavaScript programming language. Also we will build a server uses NodeJS.

Note: This is the Second part of the tutorial.

Software Requirements:

Bellow there is the list of the required Software to build a Crud Nodejs With Mongo DB.

Programming languages:

- JavaScript
- NodeJS v11.10.1
- Visual Studio code ([Version 1.31](#))
- Express
- MongoDB(NoSQL)

Download the folder

Download the [crud-nodejs-mongodb](#) folder:

Clone this repository

```
git clone https://github.com/DuvanSGF/NodeJS.git
```

Or download the zip file and unpack it:

```
https://github.com/DuvanSGF/NodeJS/archive/master.zip/
```

unzip master.zip

1. Database

1.1 NoSQL database

Run the MongoDB server

To create a database in MongoDB, use the mongod.exe command in a terminal to start (run) the server

```
CA: Símbolo del sistema - mongod.exe
C:\Program Files\MongoDB\Server\4.0\bin>mongod.exe
2019-03-03T16:44:06.354-0500 I CONTROL [main] Automatically disabling TLS 1.0, to force-enable TLS 1.0 specify --sslDis
abledProtocols 'none'
2019-03-03T16:44:07.479-0500 I CONTROL [initandlisten] MongoDB starting : pid=128100 port=27017 dbpath=C:\data\db\ 64-b
it host=LAPTOP-5TJA26HR
2019-03-03T16:44:07.479-0500 I CONTROL [initandlisten] targetMinOS: Windows 7/Windows Server 2008 R2
2019-03-03T16:44:07.480-0500 I CONTROL [initandlisten] db version v4.0.2
2019-03-03T16:44:07.480-0500 I CONTROL [initandlisten] git version: fc1573ba18aee42f97a3bb13b67af7d837826b47
2019-03-03T16:44:07.480-0500 I CONTROL [initandlisten] allocator: tcmalloc
2019-03-03T16:44:07.480-0500 I CONTROL [initandlisten] modules: none
2019-03-03T16:44:07.480-0500 I CONTROL [initandlisten] build environment:
2019-03-03T16:44:07.480-0500 I CONTROL [initandlisten] distmod: 2008plus-ssl
2019-03-03T16:44:07.480-0500 I CONTROL [initandlisten] distarch: x86_64
2019-03-03T16:44:07.480-0500 I CONTROL [initandlisten] target_arch: x86_64
2019-03-03T16:44:07.515-0500 I STORAGE [initandlisten] options: {}
2019-03-03T16:44:07.515-0500 I STORAGE [initandlisten] Detected data files in C:\data\db\ created by the 'wiredTiger' s
torage engine, so setting the active storage engine to 'wiredTiger'.
2019-03-03T16:44:07.516-0500 I STORAGE [initandlisten] wiredtiger_open config: create,cache_size=5566M,session_max=2000
0,eviction=(threads_min=4,threads_max=4),config_base=false,statistics=(fast),log=(enabled=true,archive=true,path=journal
,compressor=snappy),file_manager=(close_idle_time=100000),statistics_log=(wait=0),verbose=(recovery_progress),
2019-03-03T16:44:08.169-0500 I STORAGE [initandlisten] WiredTiger message [1551649448:168792][128100:140711558597088],
txn-recover: Main recovery loop: starting at 13/135424
2019-03-03T16:44:08.327-0500 I STORAGE [initandlisten] WiredTiger message [1551649448:327058][128100:140711558597088],
txn-recover: Recovering log 13 through 14
2019-03-03T16:44:08.559-0500 I STORAGE [initandlisten] WiredTiger message [1551649448:558314][128100:140711558597088],
txn-recover: Recovering log 14 through 14
2019-03-03T16:44:08.668-0500 I STORAGE [initandlisten] WiredTiger message [1551649448:668255][128100:140711558597088],
txn-recover: Set global recovery timestamp: 0
2019-03-03T16:44:09.034-0500 I RECOVERY [initandlisten] WiredTiger recoveryTimestamp. Ts: Timestamp(0, 0)
2019-03-03T16:44:10.007-0500 I CONTROL [initandlisten]
```

npm init --yes

Nos inicializa un package json-- Nos inicializa un archivo de configuracion

Luego

Mr Mejia@LAPTOP-5TJA26HR MINGW64 /c/Ingenieria_Software/Web_Development/Nodejs/crud-nodejs-mongodb (master) \$ npm install express

se encarga de escribir el codigo del servidor de nodejs de una manera mucho mas rapida, optimizada y profesional

\$ npm install express mongoose

Para conectar con mongo db , modelar los datos dentro de la base datos , y el crud

mostrar vistas

\$ npm install express mongoose ejs

Nos permite ejecutar pequeños js para construir las vistas

\$ npm install express mongoose ejs morgan

Para el modulo de desarrollo , multiples usuarios y morgan nos ayuda

```
Mr Mejia@LAPTOP-5TJA26HR MINGW64 /c/Ingenieria_Software/Web_Development/Nodejs/crud-nodejs-mongodb (master) $ npm install express mongoose ejs morgan
```

luego

```
npm i nodemon -D
```

Para evitar reiniciar el servidor cada vez que hagamos un cambio

Para iniciar

```
npm start
```

```
"dev": "nodemon src/app.js"
```

```
Mr Mejia@LAPTOP-5TJA26HR MINGW64 /c/Ingenieria_Software/Web_Development/Nodejs/crud-nodejs-mongodb (master) $ npm run dev
```

Esto para ejecutar el servido para guardalos sin necesidad de refrescar

```
const path = require('path')
```

Con esto evitamos los problemas de rutas en las carpetas

Motor de vista ejs

Middleware

Procesa los datos antes que llehen por la ruta lleguen a la consola por morgan