



# CheckMed - Instalación

Fecha: 30 de Junio de 2021  
Autor: Santillán Marcelo  
Diseño de Bases de Datos 2020

## INDICE

[Iniciar el proyecto con node.js](#)

[Dependencias:](#)

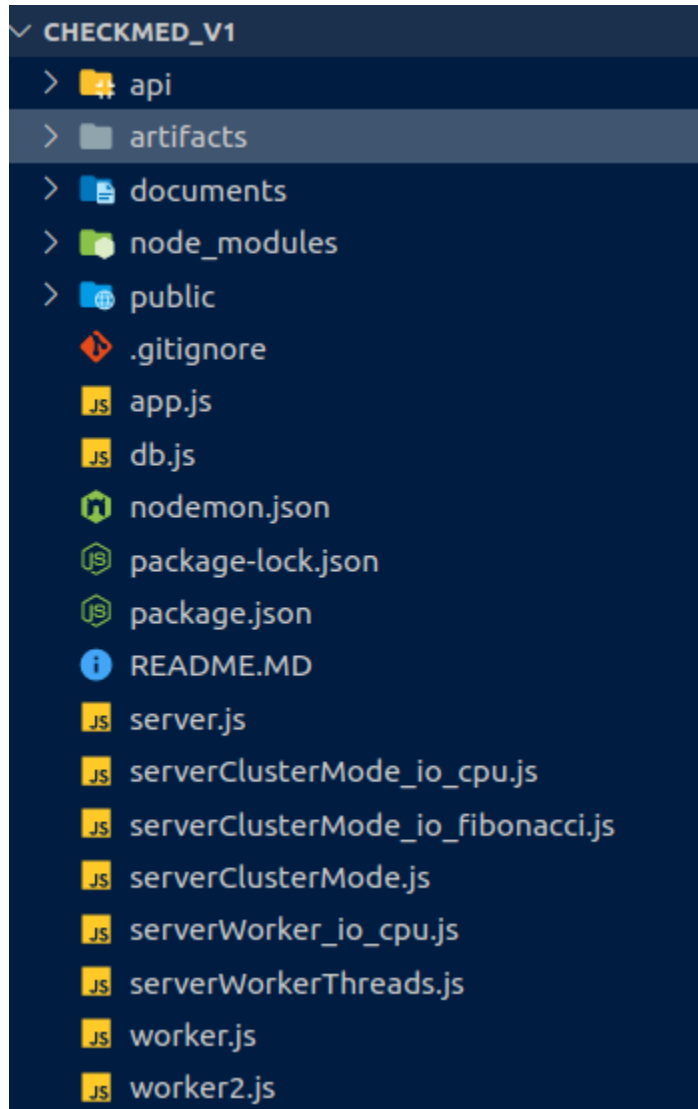
[Base de Datos](#)

[Datos de Prueba](#)

[Realización de pruebas](#)

Formato del proyecto:





## Iniciar el proyecto con node js

Las versiones que utilice fueron la últimas verificadas por linux

A terminal window with a dark background. The title bar shows 'marcelo880@marcelo880-N56VB: ~/Desktop/checkmed\_v1'. The terminal text shows the user running 'node -v' which returns 'v10.19.0', then 'npm -v' which returns '7.12.1', and finally a blank prompt.

```
marcelo880@marcelo880-N56VB:~/Desktop/checkmed_v1$ node -v
v10.19.0
marcelo880@marcelo880-N56VB:~/Desktop/checkmed_v1$ npm -v
7.12.1
marcelo880@marcelo880-N56VB:~/Desktop/checkmed_v1$
```

Si npm no está instalado, el siguiente comando lo instala de forma global

```
$> npm install -g npm
```

- Si npm no está actualizado, el siguiente comando lo actualiza de forma global

```
$> npm update -g npm
```

```
$> npm install
```

npm install se pueden colocar las dependencias que se mencionan a continuación

## Dependencias:

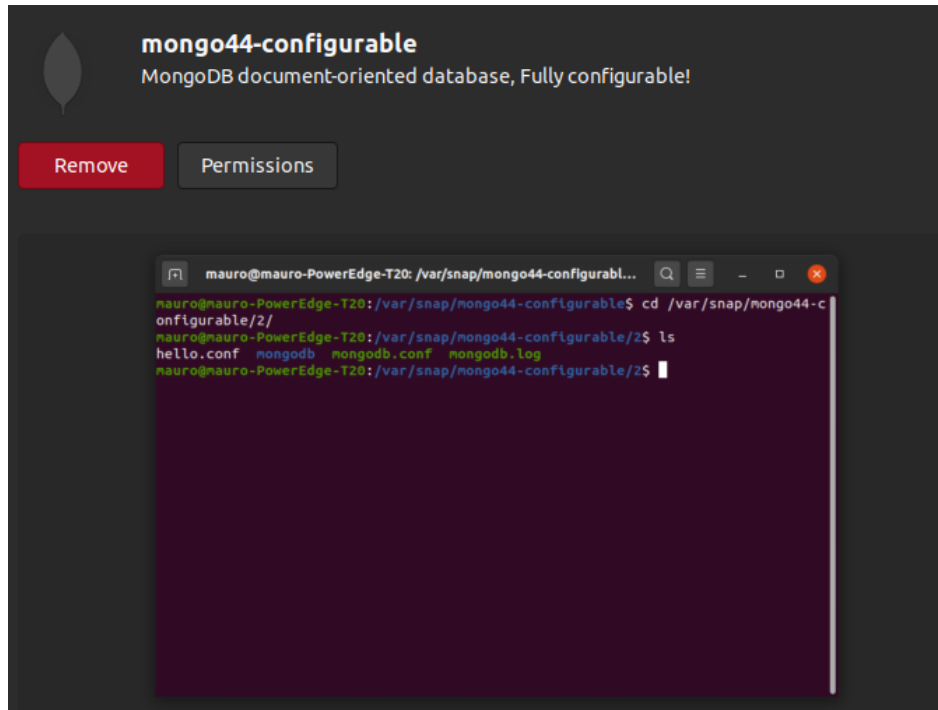
http, express, node-cron, os, faker, worker\_threads, morgan, fs, uuid, body-parser, mongoose

- <https://www.npmjs.com/package/node-cron>
- <https://www.npmjs.com/package/worker-thread>
- express <https://www.npmjs.com/package/express>
- body-parser (para peticiones post a jsons)  
<https://www.npmjs.com/package/body-parser>
- npm i mongoose <https://www.npmjs.com/package/mongoose> orm para MongoDB.

## Base de Datos

Instalación en Ubuntu 20.04

<https://www.digitalocean.com/community/tutorials/how-to-install-mongodb-on-ubuntu-20-04-es>



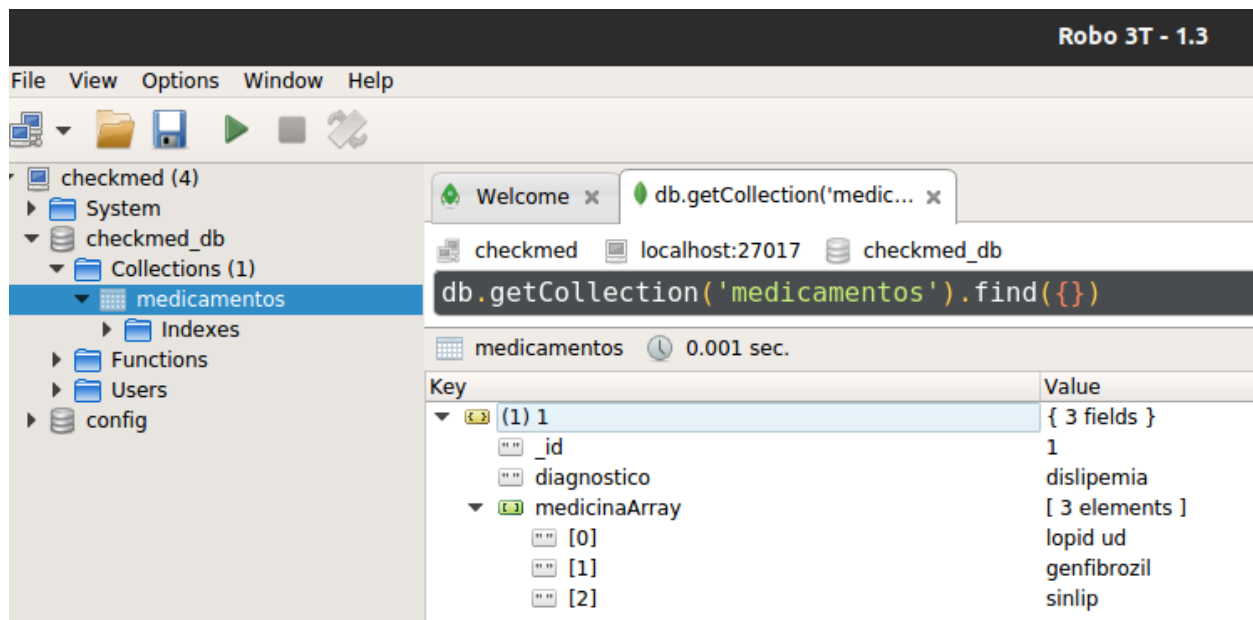
```
marcelo880@marcelo880-N56VB:~$ mongo
MongoDB shell version v4.4.5
connecting to: mongodb://127.0.0.1:27017/?compressors=disabled&gssapiServiceName=mongodb
Implicit session: session { "id" : UUID("edec928f-afda-4784-9703-fe85b241e14b") }
MongoDB server version: 4.4.5
---
```

Inserción de registro de medicinas:

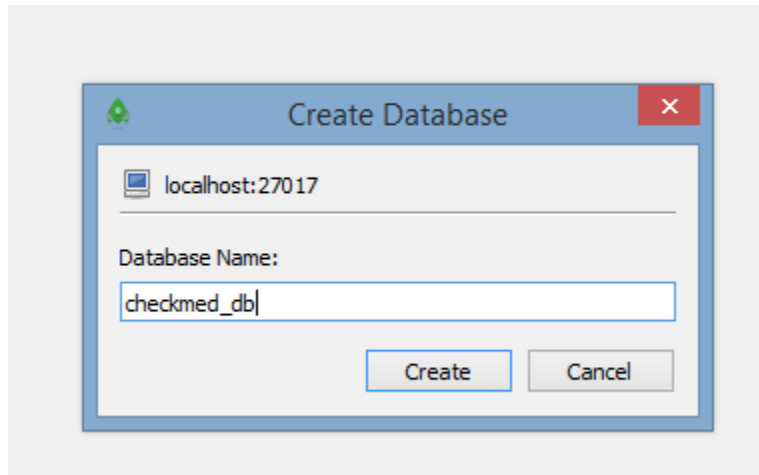
<https://kb.objectrocket.com/mongo-db/how-to-add-elements-into-an-array-in-mongodb-1195>

```
use checkmed_db
switched to db checkmed_db
```

```
> db.medicamentos.insert({
...   "_id": "1",
...   "diagnostico": "dislipemia",
...   "medicinaArray": ['lopid ud', 'genfibrozil', 'sinlip']
... });
```



Voy a utilizar MongoDB con estos datos:



MONGODB\_URI = mongodb://localhost/checkmed\_db

#### Colecciones de la base de datos:

+ Medicamentos

+ Tickets

---> comments

<https://stackoverflow.com/questions/17297632/automatic-persistence-of-node-js-objects-in-database>

## Datos de Prueba

Para generar los datos de prueba, utilizo la librería faker js lo cual me permite generar datos aleatorios

<https://www.npmjs.com/package/faker>

Ejemplo en libfaker.js

## Realización de pruebas

Dado que tengo instalado Ubuntu la versión más actualizada es la 10, y para ejecutar los scripts que contienen worker threads se utiliza este comando:

**node --experimental-worker serverWorker**

Para las mediciones utilicé Apache Benchmark

Apache Benchmark:

```
marcelo880@marcelo880-N56VB:~$ ab -V
This is ApacheBench, Version 2.3 <$Revision: 1843412 $>
Copyright 1996 Adam Twiss, Zeus Technology Ltd, http://www.zeustech.net/
Licensed to The Apache Software Foundation, http://www.apache.org/

marcelo880@marcelo880-N56VB:~$
```

<https://bobcares.com/blog/apache-benchmark-install-ubuntu/>