# Guia de instalacion

Las siguientes instalaciones y pruebas se realizaron en el sistema operativo Red Hat Enterprise Linux 8.2.

## NodeJs

El siguiente punto demuestra el paso a paso para instalar el NodeJs.

**- Ejecutar el siguiente comando en una terminal de linux.**

*sudo yum -y install -y curl*

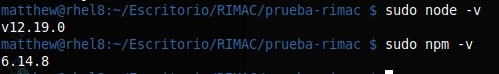
*sudo curl -sL https://rpm.nodesource.com/setup\_13.x | bash -*

*sudo yum install -y nodejs*

**- Verificamos la version**

*node -v*

*npm -*

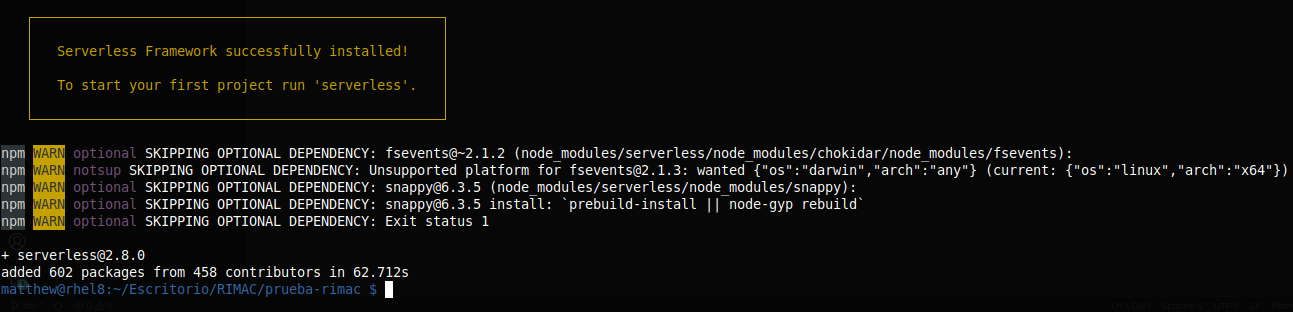


## Serverless

El siguiente punto demuestra el paso a paso para instalar el framework de Serverless.

**- Ejecutar el siguiente comando en una terminal de linux.**

*npm install serverless -g*



## AWS Cli

El siguiente punto demuestra el paso a paso para instalar el cliente de AWS.

**- Ejecutar el siguiente comando en una terminal de linux.**

sudo curl "https://awscli.amazonaws.com/awscli-exe-linux-x86\_64.zip" -o "awscliv2.zip"

sudo unzip awscliv2.zip

sudo ./aws/install



## Serverless - AWS

El siguiente punto demuestra el paso a paso para loguearse el cliente de AWS.

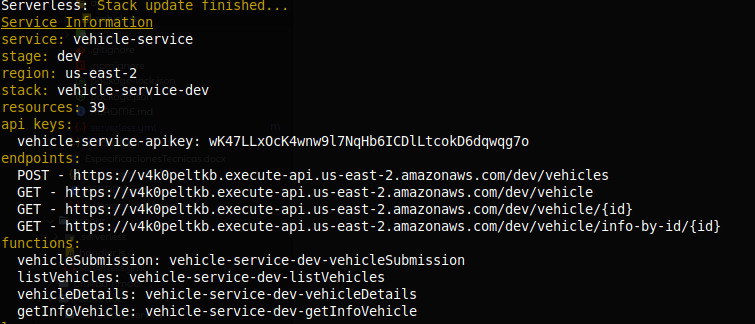
**- Ejecutar el siguiente comando en una terminal de linux.**

serverless config credentials --provider aws --key “{KEY}” --secret “{SECRET}”

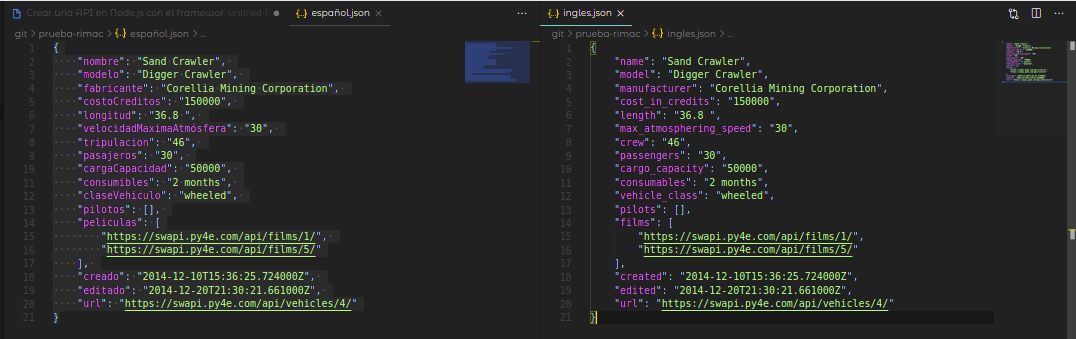


# Reto Técnico

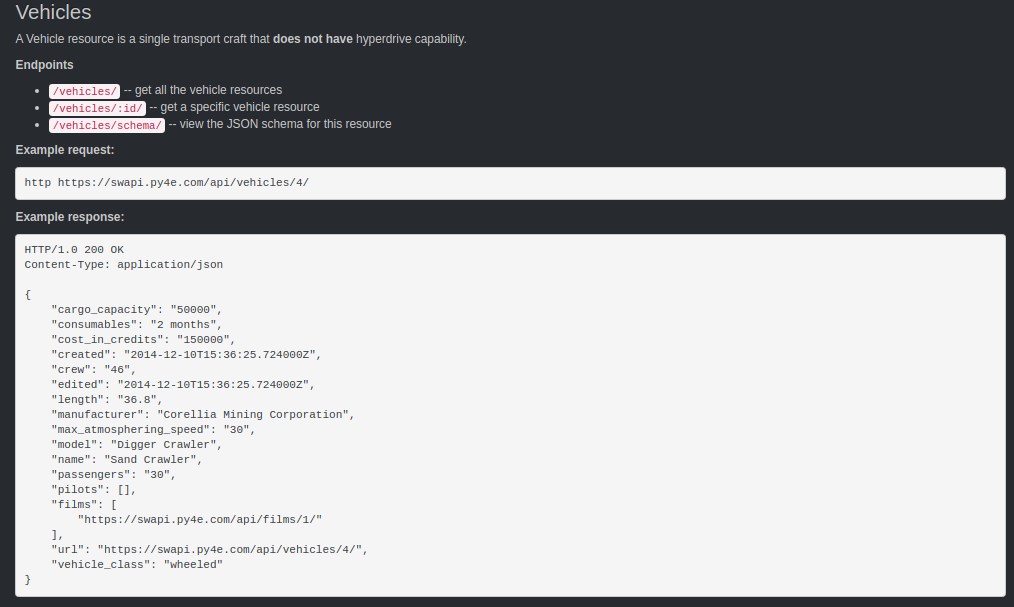
- Crear una API en Node.js con el framework Serverless para un despliegue en AWS



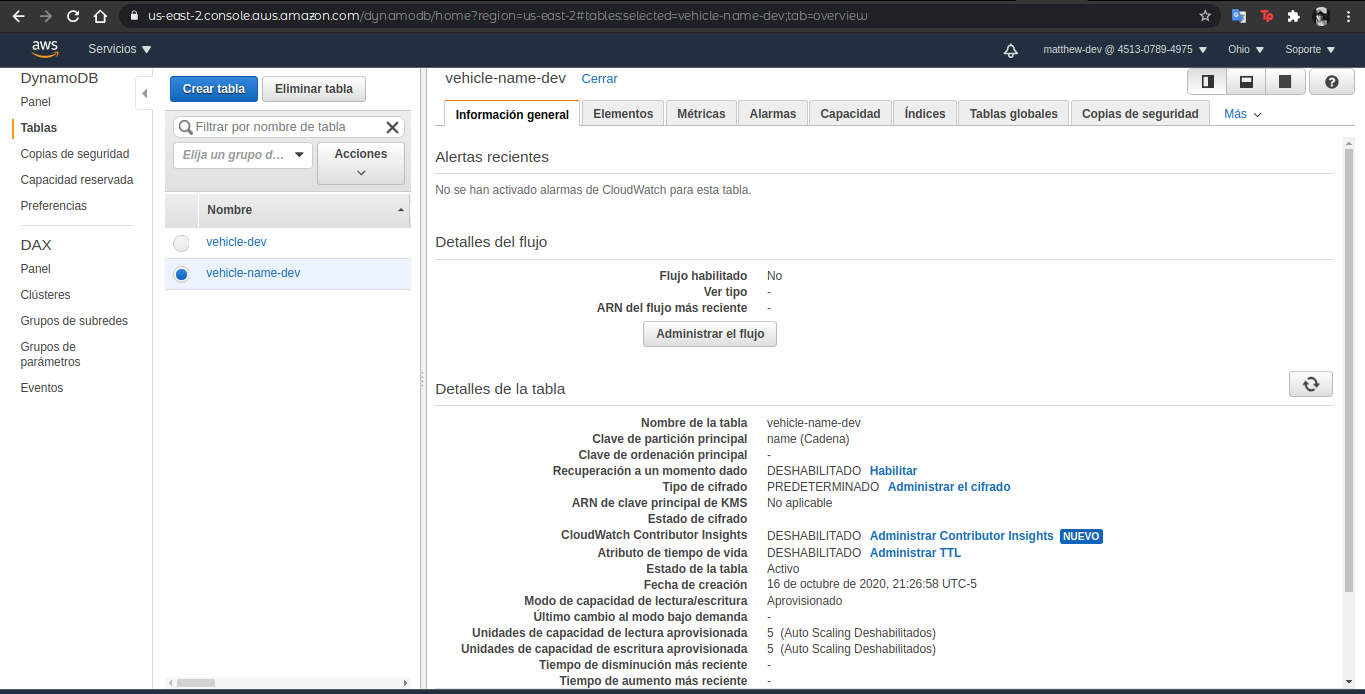
- Adaptar y transformar los modelos de la API de prueba. Se tienen que mapear todos los nombres de atributos modelos del inglés al español (Ej: name -> nombre).



- Integrar la API de prueba StarWars API (lineas abajo está el link) se deben integrar uno o más endpoints.



- Crear un modelo de su elección mediante el uso de un endpoint POST, la data se tendrá que almacenar dentro de una base de datos.



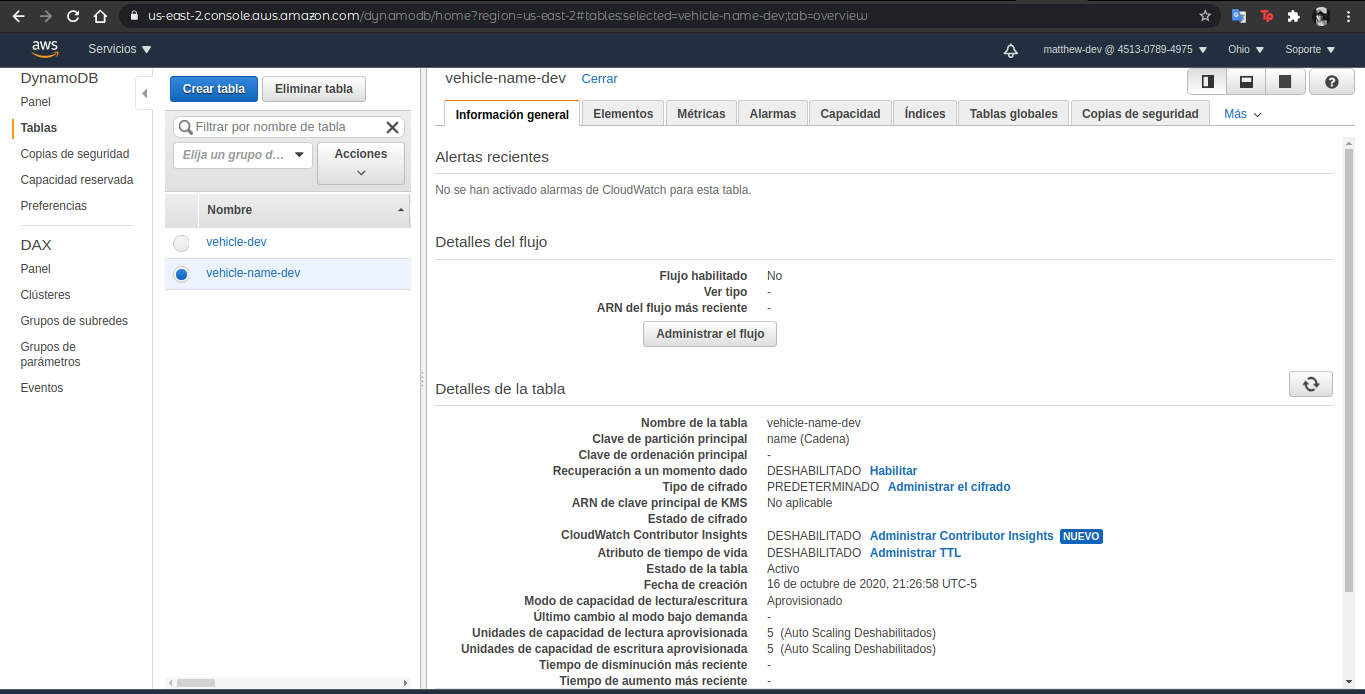
- Crear un endpoint GET que muestre la data almacenada.

# PRUEBAS

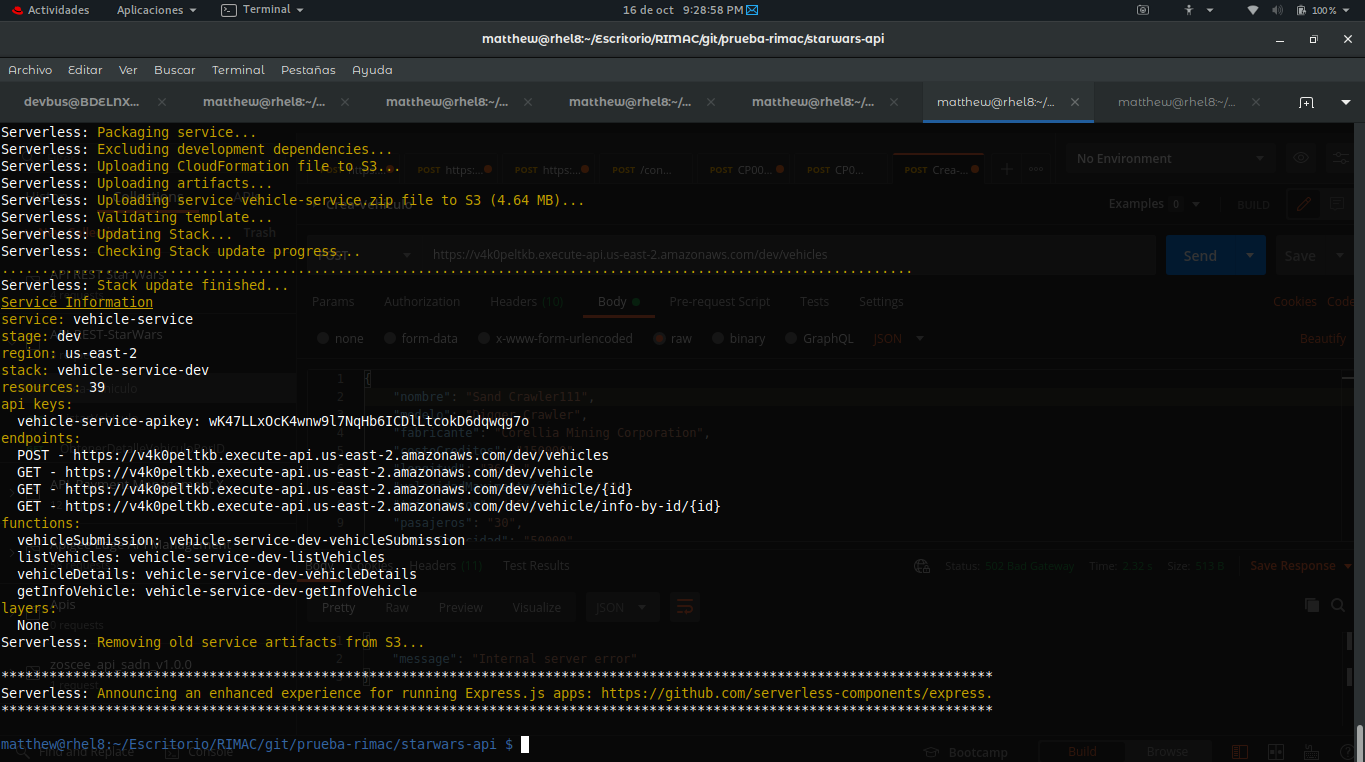
-Pruebas unitarias

-Minimo 2 endpoints, GET para recuperar la información y POST para crear un elemento

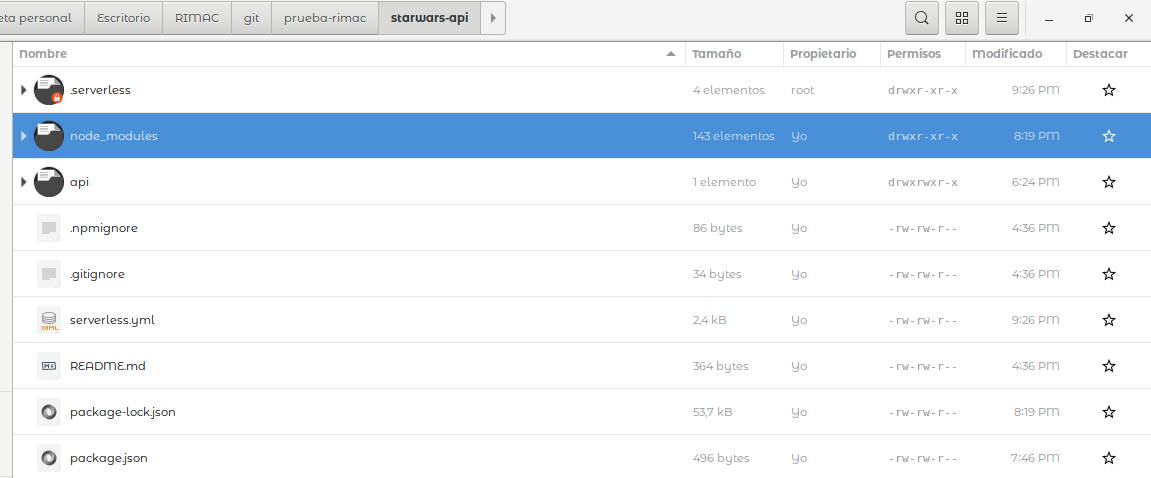
-Integración con una base de datos (DynamoDB o MySQL)



-Uso de Serverless Framework



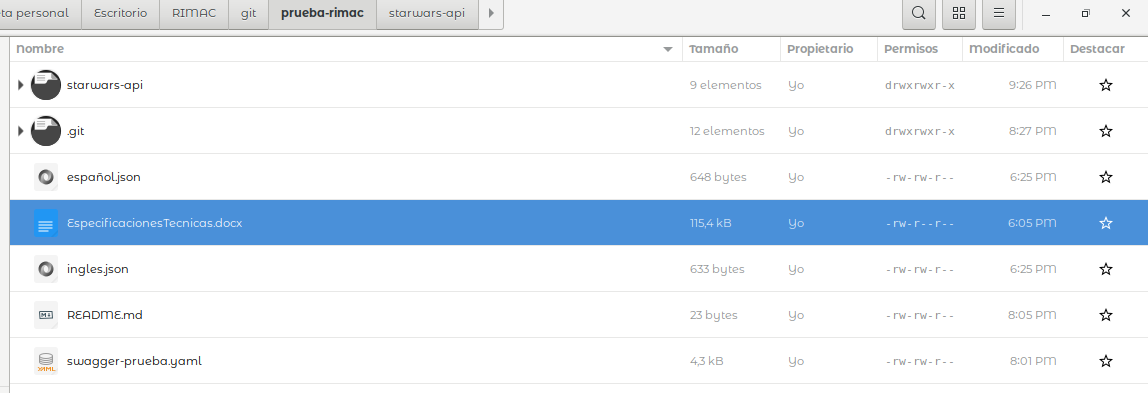
-Uso de Node.js



- Respeto de las buenas prácticas de desarrollo

# Puntos bonus

- Documentación de uso



- Documentación en Open API/Swagger



- Desplegar sin errores en AWS con el comando deploy del framework serverless

## 

- Mayor complejidad de Integración

