

# Introducción a Serverless Framework

Presenta



**Martin Balmaceda**  
Dev



## Resumiendo ... La Computación sin servidor es ..

Un modelo de ejecución en el que el proveedor en la nube es responsable de ejecutar un fragmento de código mediante la asignación dinámica de los recursos.

Y cobrando solo por la cantidad de recursos utilizados para ejecutar el código. El código, generalmente, se ejecuta dentro de contenedores sin estado que pueden ser activados por una variedad de eventos que incluyen solicitudes HTTP, eventos de base de datos, servicios de colas, alertas de monitoreo, carga de archivos, eventos programados (trabajos cron), etc.

El código que se envía a al proveedor en la nube para la ejecución es generalmente en forma de una función. Por lo tanto, serverless a veces se denomina “*Funciones como servicio*” o “*FaaS*”. Las siguientes son las ofertas de FaaS de los principales proveedores en la nube:



# Lenguajes

Los lenguajes soportados por AWS Lambda son:

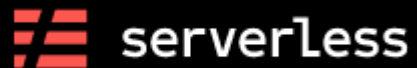


AWS Lambda



# Serverless Framework

Es un marco web gratuito y de código abierto escrito con Node.js. Serverless Framework es el primer marco desarrollado para crear aplicaciones en AWS Lambda

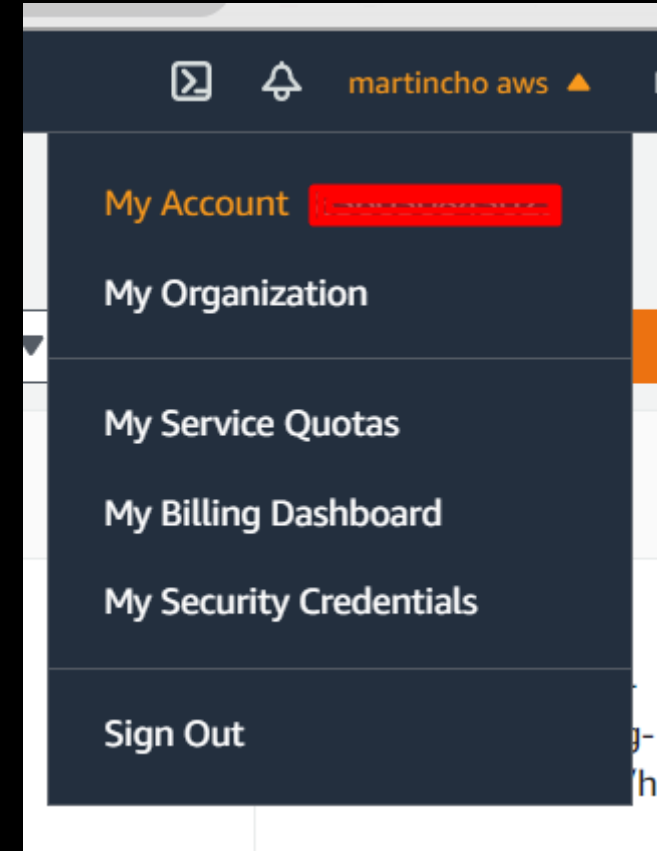


AWS Lambda

# Serverless Framework – Credenciales

Antes de empezar a crear proyectos

Se debe indicar en el la carpeta aws de nuestro SO  
Las credenciales de nuestras cuentas de aws



# Serverless Framework - Instalación

A través del gestor de paquetes npm  
Escribir el comando para esta version específica

```
npm install -g serverless@1.78.1
```

```
serverless --v
```

```
sls --v
```

```
Framework Core: 1.78.1  
Plugin: 3.8.4  
SDK: 2.3.2  
Components: 2.34.9
```



# Serverless Framework – Primer Proyecto

A través del gestor de paquetes npm  
Escribir el comando para esta versión específica

```
serverless create --template aws-nodejs --path firstApp
```



# Serverless Framework – El archivo YML

Sección Provider:

Indica entre sus atributos mas importantes:

- Name: Nombre del proveedor (Aws, Google, Azure)
- Runtime: La versión del lenguaje compatible (node, go, etc)
- Region: region en la cual vamos a deployar
- iamRolesStatatements: Configuración de los Roles iam de AWS
- Enviroments: variables de entorno para ser usados en el código

```
provider:
  name: aws
  runtime: nodejs12.x
  region: us-east-1
  iamRoleStatements:
```





# Serverless Framework – El archivo YML

Sección Functions:

Indica entre sus atributos mas importantes:

- Nombre de la función Lambda
- Handler: Entrada de la función.
- Evento: Disparador de la lambda

```
functions:
  sender:
    handler: sender.handler
    events:
      - http:
          path: sls-sqs/sender
          method: post
```



# Serverless Framework – El archivo YML

Sección Recursos:

Indica entre sus atributos mas importantes:

- Nombre del recurso
- Type: tipo del recurso (SNS, SQS, DynamoDB, etc)
- Propiedades: Nombre, arn, etc del recurso

```
resources:  
  Resources:  
    MyQueue:  
      Type: "AWS::SQS::Queue"  
      Properties:  
        QueueName: "MyQueue"
```



## Recursos

El repositorio con la información, esta diapositiva y ejemplos en:

<https://gitlab.com/atalivar77/taller-serverless-framework>

```
resources:
  Resources:
    MyQueue:
      Type: "AWS::SQS::Queue"
      Properties:
        QueueName: "MyQueue"
```





**Muchas Gracias!!!!**

