

Propuesta de solución

Caso Práctico 1 – Apartado C

Asignatura	Datos de los alumnos	Fecha
Experto Universitario en DevOps & Cloud	Apellidos: Leon Granda	
	Nombre: Giovanna Victoria	

URL de repositorio solución de GitHub: <https://github.com/GiovannaLeon/todo-list-aws.git>

Reto 1 – Creación pipeline CI

En este reto se solicita un único entregable:

- Relación de comandos, junto con sus capturas de pantalla, empleados para llevar a cabo el clonado del repositorio.

Mi repositorio: <https://github.com/GiovannaLeon/todo-list-aws.git>

1. Clono el repositorio en mi carpeta de trabajo

`git clone https://github.com/anieto-unir/todo-list-aws`

```
C:\Users\amaro\Documents\Giovanna>git clone https://github.com/anieto-unir/todo-list-aws
Cloning into 'todo-list-aws'...
remote: Enumerating objects: 57, done.
remote: Counting objects: 100% (18/18), done.
remote: Compressing objects: 100% (15/15), done.
remote: Total 57 (delta 8), reused 3 (delta 3), pack-reused 39 (from 1)
Receiving objects: 100% (57/57), 17.62 KiB | 360.00 KiB/s, done.
Resolving deltas: 100% (11/11), done.
```

2. Verifico que esta creado el repositorio en mi capeta

`dir`

```
C:\Users\amaro\Documents\Giovanna>dir
El volumen de la unidad C es Windows
El n mero de serie del volumen es: 6475-04AA

Directorio de C:\Users\amaro\Documents\Giovanna

29/01/2025  00:15    <DIR>        .
23/01/2025  19:49    <DIR>        ..
11/12/2024  14:21    <DIR>        casopractico_unir
28/01/2025  23:05    <DIR>        helloworld
29/01/2025  00:15    <DIR>        todo-list-aws
13/12/2024  16:15    <DIR>        wiremock
04/12/2024  23:47      19.347.010 wiremock-standalone-3.10.0.jar
                      1 archivos     19.347.010 bytes
                      6 dirs      565.450.919.936 bytes libres
```

3. Ingreso a la carpeta todo-list-aws

```
C:\Users\amaro\Documents\Giovanna\todo-list-aws>git init
Initialized empty Git repository in C:/Users/amaro/Documents/Giovanna/todo-list-aws/.git/
C:\Users\amaro\Documents\Giovanna\todo-list-aws>
```

4. Creo el repositorio en mi GitHub.

<https://github.com/GiovannaLeon/todo-list-aws.git>

5. Asocio mi carpeta local a mi repositorio creado

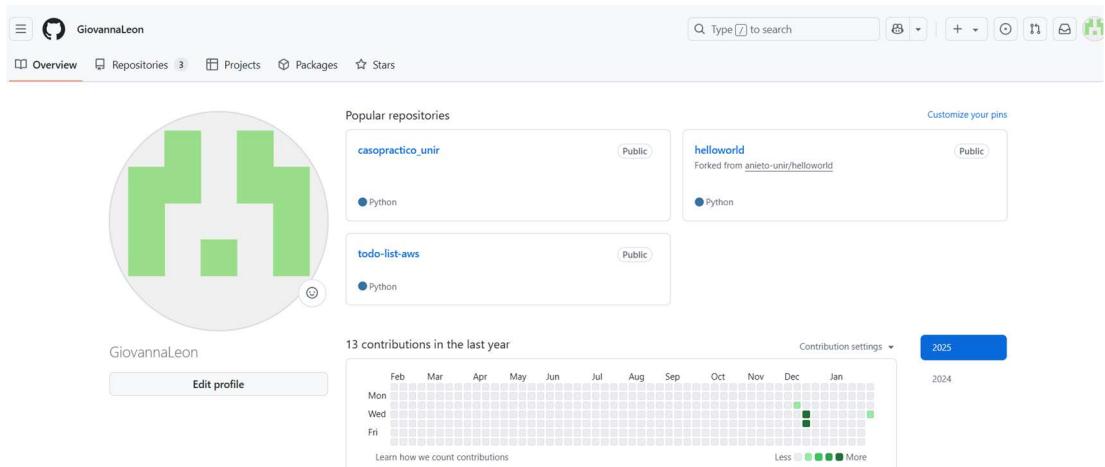
```
git remote set-url origin https://github.com/GiovannaLeon/todo-list-aws.git
```

```
C:\Users\amaro\Documents\Giovanna\todo-list-aws>git remote set-url origin https://github.com/GiovannaLeon/todo-list-aws.git
```

6. Subo la información

```
C:\Users\amaro\Documents\Giovanna\todo-list-aws>git remote set-url origin https://github.com/GiovannaLeon/todo-list-aws.git
C:\Users\amaro\Documents\Giovanna\todo-list-aws>git push -u origin master
Enumerating objects: 57, done.
Counting objects: 100% (57/57), done.
Delta compression using up to 20 threads
Compressing objects: 100% (39/39), done.
Writing objects: 100% (57/57), 17.62 KiB | 5.87 MiB/s, done.
Total 57 (delta 11), reused 57 (delta 11), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (11/11), done.
To https://github.com/GiovannaLeon/todo-list-aws.git
 * [new branch]      master -> master
branch 'master' set up to track 'origin/master'.
```

6. Verifico que se ha subido correctamente

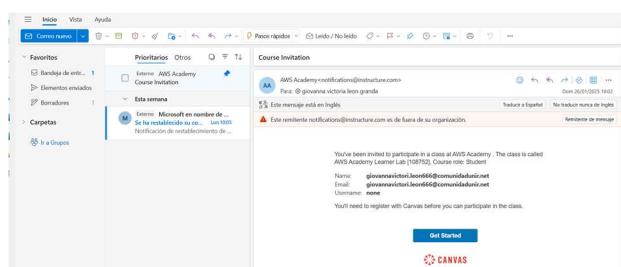


Reto 2 – Infraestructura AWS

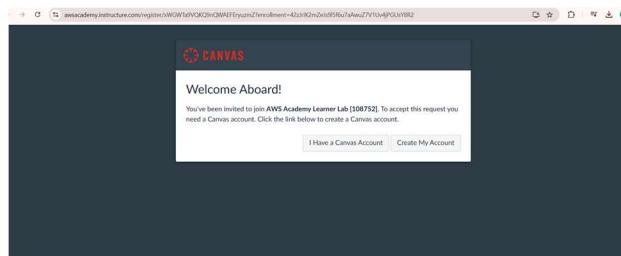
En este reto se solicitan 6 entregables:

- ▶ Pasos realizados hasta acceder a la consola de AWS.

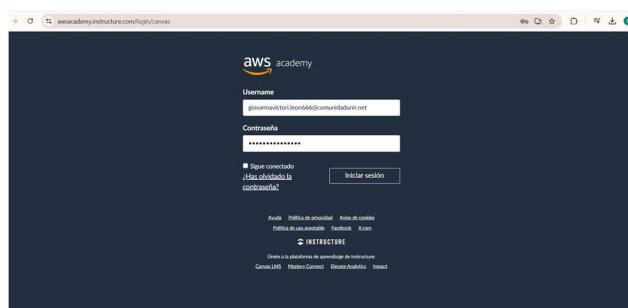
- 1) Envío de credenciales a correo de la comunidad unir. Presionamos “Get Started”



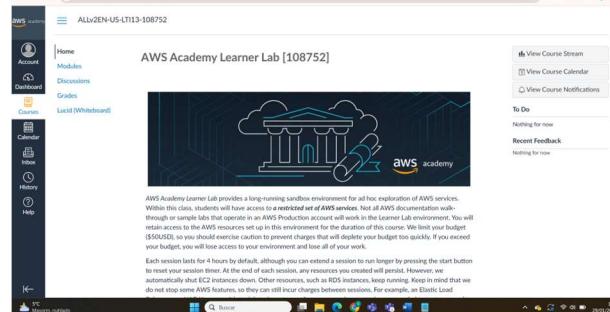
- 2) Seleccionamos “Create My Account” para crear usuario.



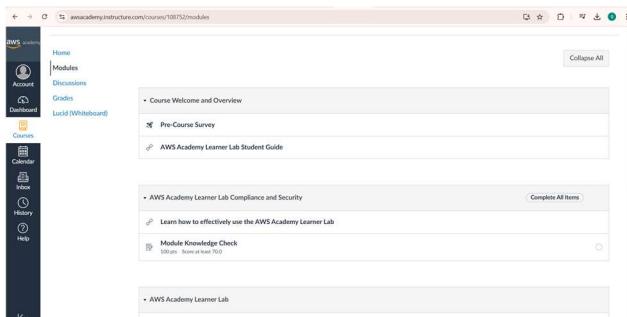
- 3) Creamos nuestras credenciales, e ingreso con el usuario y password registrados.



- 4) Hemos ingresados a la plataforma “aws” como estudiantes.



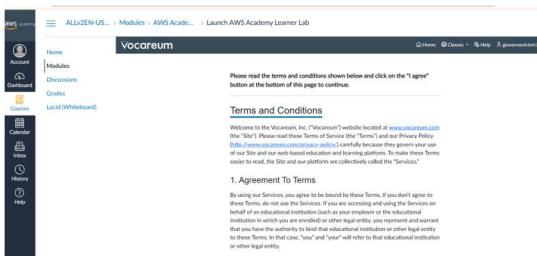
- 5) Seleccionamos “módulos”. Nos muestra la pantalla a continuación y como primer paso debemos de leer el documento “**AWS Academy Lab Student Guide**” que es de lectura obligatoria.



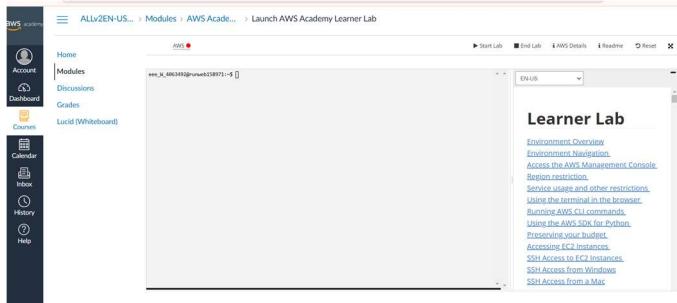
Uso de lectura obligatoria



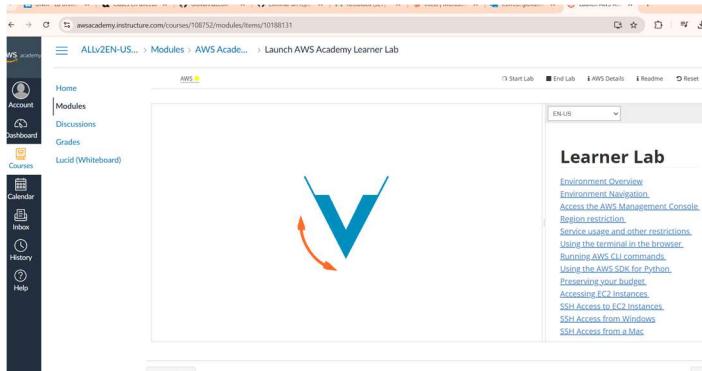
- 6) Una vez leído entramos a “**Launch AWS Academy Learner Lab**”, nos muestra los términos y condiciones, y le damos aceptar.



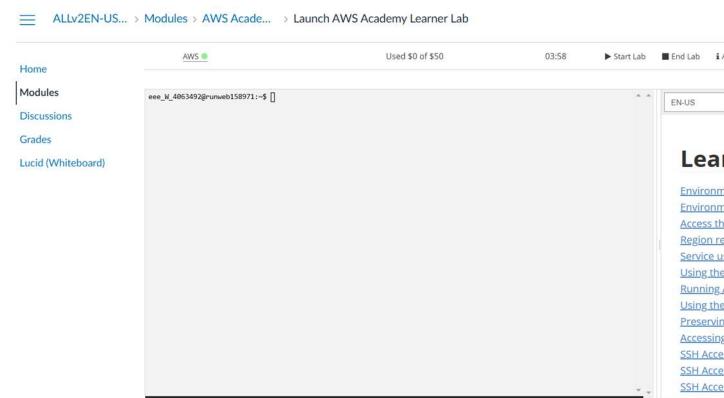
7) Al aceptar los términos Entramos al laboratorio



8) Presionamos “StarLab” se demora unos segundos al arrancar el laboratorio.



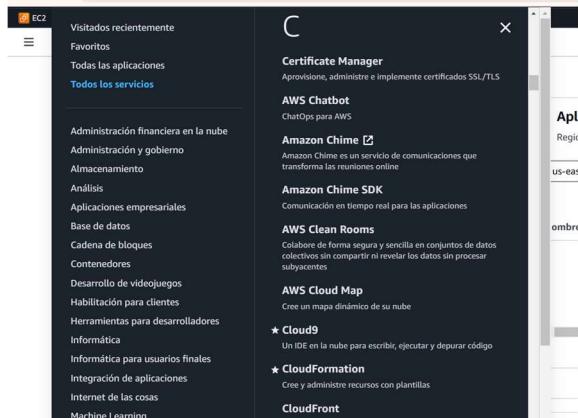
9) Aparece un pequeño terminal con comandos Linux.



10) Presionamos “AWS”, y entramos a configurar la plataforma, en la parte superior encontramos , presionamos 



11) Nos muestra todos los servicios que utiliza “AWS”, donde marcaremos los que vamos a utilizar EC2, Lambda, Cloud9 y CloudFormation como favoritos.

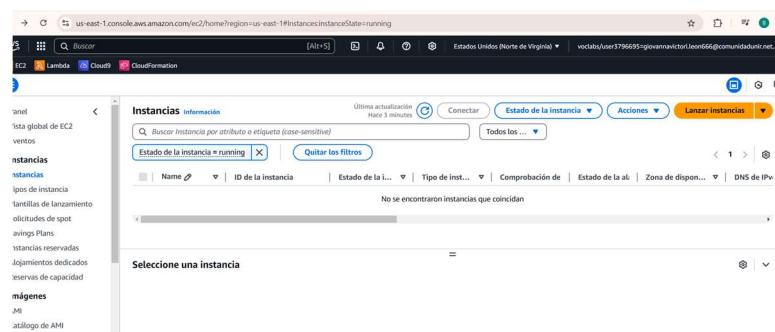


12) Ya nos muestra la barra de “AWS” con los servicios seleccionados.

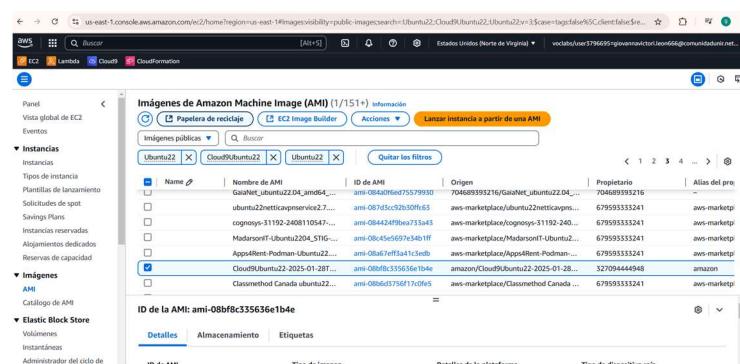
The screenshot shows the AWS Home Page. At the top, there's a navigation bar with the AWS logo, a search bar, and a dropdown menu for the region set to 'Estados Unidos (Norte de Virginia)'. Below the navigation bar, there are four service icons: EC2, Lambda, Cloud9, and CloudFormation. The Cloud9 icon is highlighted with a blue border, indicating it is selected. The main content area is titled 'Página de inicio de la Consola' and contains sections for 'Visitados recientemente' and 'Aplicaciones (0)'. The 'Aplicaciones' section shows a message: 'No hay aplicaciones' and 'Comience creando una aplicación'. There's also a 'Crear aplicación' button.

► Comandos y pasos realizados para la creación de la instancia EC2

- 1) Procedemos a crear la instancia EC2". Presionamos en favoritos EC2, procederemos a crear una instancia, señalamos en la sección izquierda la opción de menú "AMIS".



- 2) Nos muestra una pantalla donde tenemos que indicar el tipo de imagen que es “**ímgenes públicas**”, luego ponemos el texto “Cloud9Ubuntu22” en el recuadro buscar. Nos muestra todas las AMIS que coinciden con la búsqueda, y señalamos “Cloud9Ubuntu22”, presionamos en el botón naranja “**Lanzar instancia a partir de una AMI**”.



- 3) Una vez lanzada la instancia nos muestra una pantalla que nos permite configurarla. Indicamos su configuración con las siguientes características:

a. Tipo de Instancia: t3.micro

Tipo de instancia Información | Obtener asesoramiento

Tipo de instancia

t3.micro	Familia: t3	2 vCPU	1 GiB Memoria	Generación actual: true
Bajo demanda Ubuntu Pro base precio: 0.0139 USD per Hour				
Bajo demanda SUSE base precio: 0.0104 USD per Hour				
Bajo demanda Linux base precio: 0.0104 USD per Hour				
Bajo demanda RHEL base precio: 0.0392 USD per Hour				
Bajo demanda Windows base precio: 0.0196 USD per Hour				

Se aplican costos adicionales a las AMI con software preinstalado

Todas las generaciones

[Comparar tipos de instancias](#)

b. Par de Claves: vockey

▼ Par de claves (inicio de sesión) [Información](#)

Puede utilizar un par de claves para conectarse de forma segura a la instancia. Asegúrese de que tiene acceso al par de claves seleccionado antes de lanzar la instancia.

Nombre del par de claves - obligatorio

vockey

[Crear un nuevo par de claves](#)

c. Configurar almacenamiento: 20GB, de tipo gp2

▼ Configurar almacenamiento [Información](#)

Avanzado

1x GiB Volumen raíz (Sin cifrar)

Los clientes que cumplan los requisitos de la capa gratuita pueden obtener hasta 30 GB de almacenamiento magnético o de uso general (SSD) de EBS

[Agregar un nuevo volumen](#)

d. Detalles avanzados: El Rol con el que se ejecutará esta máquina será el LabInstanceProfile.

▼ Detalles avanzados [Información](#)

Directorio de unión al dominio [Información](#)

Seleccionar [Crear nuevo directorio](#)

Perfil de instancia de IAM [Información](#)

LabInstanceProfile [Crear un nuevo perfil de IAM](#)

arn:aws:iam::582789485636:instance-profile/LabInstanceProfile

e. Creamos un nuevo Grupo de seguridad y presionamos editar

▼ Configuraciones de red [Información](#)

Red [Información](#)
vpc-02Saa4767460c4040

Subred [Información](#)
Sin preferencias (subred predeterminada en cualquier zona de disponibilidad)

Asignar automáticamente la IP pública [Información](#)

Habilitar
Se aplican cargos adicionales cuando no se cumplen los límites del nivel gratuito

Firewall (grupos de seguridad) [Información](#)
Un grupo de seguridad es un conjunto de reglas de firewall que controlan el tráfico de la instancia. Agregue reglas para permitir que un tráfico específico llegue a la instancia.

Crear grupo de seguridad Seleccionar un grupo de seguridad existente

Crearemos un nuevo grupo de seguridad denominado "launch-wizard-1" con las siguientes reglas:

Permitir el tráfico de SSH desde
Ayuda a establecer conexión con la instancia

Permitir el tráfico de HTTPS desde Internet
Para configurar un punto de enlace, por ejemplo, al crear un servidor web

Permitir el tráfico de HTTP desde Internet
Para configurar un punto de enlace, por ejemplo, al crear un servidor web

f. Configuramos el SSH, el HTTP y TCP

- i. **SSH (Secure Shell):** Es un protocolo que te permite acceder de manera remota y segura a la terminal o línea de comandos de tu servidor EC2. Abre el puerto 22 (por defecto para SSH) en el grupo de seguridad de la instancia EC2

- ii. **HTTP (HyperText Transfer Protocol):** Es el protocolo utilizado para la comunicación en la web, es decir, para acceder a páginas web a través de un navegador. En el grupo de seguridad de EC2, permite el tráfico entrante en el puerto 80.
- iii. **TCP (Transmission Control Protocol):** Es un protocolo de comunicación en redes que se usa para asegurar que los datos lleguen correctamente desde un punto A a un punto B. TCP puede usarse para diferentes servicios, no solo para HTTP o SSH.

The screenshot shows the AWS Security Groups configuration interface. It displays three security group rules:

- Regla del grupo de seguridad 2 (TCP, 80, 0.0.0.0/0)**:
Type: ssh, Protocol: TCP, Port: 22, Origin: Anywhere.
- Regla del grupo de seguridad 3 (TCP, 8080, 0.0.0.0/0)**:
Type: HTTP, Protocol: TCP, Port: 80, Origin: Anywhere.
- Regla del grupo de seguridad 1 (TCP personalizado, 8080, 0.0.0.0/0)**:
Type: TCP personalizado, Protocol: TCP, Port: 8080, Origin: Anywhere.

- g. Vemos el resumen de nuestra configuración para estar seguro de que es lo que queremos que tenga nuestra EC2.

The screenshot shows the AWS Launch Wizard 'Resumen' (Summary) step. The summary information is as follows:

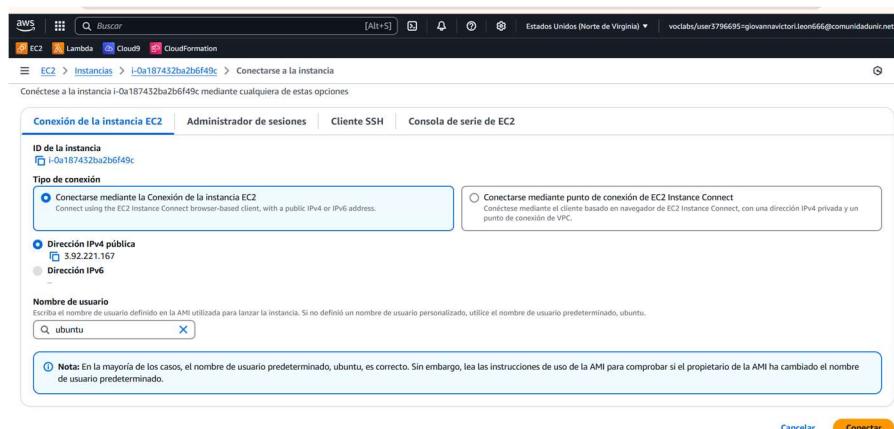
- Número de instancias:** 1
- Imagen de software (AMI):** Cloud9 Cloud9Ubuntu22 AMI
ami-08bf8c355636e1b4e
- Tipo de servidor virtual (tipo de instancia):** t3.micro
- Firewall (grupo de seguridad):** Nuevo grupo de seguridad
- Almacenamiento (volúmenes):** Volúmenes: 1 (20 GiB)

At the bottom, there are buttons for **Cancelar** (Cancel) and **Lanzar instancia** (Launch instance). There is also a link to **Código de versión preliminar**.

- Comandos y salida obtenida para la comprobación de que podemos acceder a la instancia EC2 con sistema Ubuntu.

1. Desde la página de EC2, vemos las instancias y entramos a la creada. Podemos ver la ip publica 3.92.221.167.

La **IP pública de EC2** es una dirección IP asignada a tu instancia de EC2 que es accesible desde cualquier parte del mundo a través de internet. Esta IP se usa para que puedas conectar tu servidor (instancia EC2) desde fuera de la red de AWS. Es decir, cuando quieras acceder a tu instancia EC2 de manera remota, como a través de SSH o navegar por un sitio web alojado en tu EC2, utilizarás la IP pública.



2. Al precionar el botón naranja de la imagen anterior “Conectar” se abre un terminal. Pero las comprobaciones lo haremos en el terminal inicial.
3. Copiamos nuestra dirección IP publica en el portapapeles.

ID de la instancia	Estado de la i...	Tipo de inst...	Comprobación de	Estado de la al:	Zona de dispon...	DNS de IPv4 pública
i-0a187432ba2b6f49c	En ejecución	t3.micro	3/3 comprobador	Ver alarmas +	us-east-1d	ec2-3-92-221-167.com...

ec2-3-92-221-167.compute-1.amazonaws.com 3.92.221.167

4. En el terminal del laboratorio, ingresamos los comandos de las pantallas a continuación para comprobar si todo es correcto.

- a. **Ls -la** : listamos lo que han en nuestra terminal.

```
eee_W_4063492@runweb159147:~$ ls -la
total 28
drwxrwx--- 5 eee_W_4063492 apache 6144 Jan 28 17:27 .
drwxrwx--- 5 eee_W_4063492 apache 6144 Jan 28 17:27 ..
dr-xr-xr-x 2 eee_W_4063492 apache 6144 Jan 28 17:27 .aws
-rw-rw-r-- 1 eee_W_4063492 apache 72 Jan 28 17:27 .gitconfig
drwxr-xr-x 2 eee_W_4063492 apache 6144 Jan 28 17:27 .ssh
-rw-r--r-- 1 root      root   4084 Jan 29 15:25 .termrc
dr-xr-xr-x 2 root      root   6144 Jan 28 17:22 .voc
eee_W_4063492@runweb159147:~$ []
```

- b. **Ls -la .ssh: labsuser.pem**: Este es un archivo de clave privada utilizado para la conexión SSH.

```
eee_W_4063492@runweb159147:~$ ls -la .ssh
total 12
drwxr-x--- 2 eee_W_4063492 apache 6144 Jan 28 17:27 .
drwxrwx--- 5 eee_W_4063492 apache 6144 Jan 28 17:27 ..
-r----- 1 eee_W_4063492 apache 1678 Jan 29 14:07 labsuser.pem
```

C. Conexión SSH:

~\$ ssh -i ~/ssh/labuser.pem ubuntu@3.92.221.167

```
eee_W_4063492@runweb159147:~$ ssh -i ~/ssh/labuser.pem ubuntu@3.92.221.167
Welcome to Ubuntu 22.04.5 LTS (GNU/Linux 6.8.0-1021-aws x86_64)
```

```
* Documentation: https://help.ubuntu.com
* Management: https://landscape.canonical.com
* Support: https://ubuntu.com/pro
```

```
System information as of Wed Jan 29 23:43:58 UTC 2025
```

```
System load: 0.0          Processes:           114
Usage of /: 30.1% of 19.20GB  Users logged in:     0
Memory usage: 33%          IPv4 address for ens5: 172.31.84.29
Swap usage: 67%
```

```
Expanded Security Maintenance for Applications is not enabled.
```

```
0 updates can be applied immediately.
```

```
Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status
```

```
New release '24.04.1 LTS' available.
Run 'do-release-upgrade' to upgrade to it.
```

```
Last login: Wed Jan 29 22:57:42 2025 from 18.206.107.29
:~ $ []
```

```
Last login: Wed Jan 29 22:57:42 2025 from 18.206.107.29
:~ $ ls
environment
:~ $ ls -la
total 44
drwxr-x--- 6 ubuntu ubuntu 4096 Jan 29 23:43 .
drwxr-xr-x 3 root   root   4096 Jan 28 05:21 ..
-rw----- 1 ubuntu ubuntu  28 Jan 29 23:46 .bash_history
-rw-r--r-- 1 ubuntu ubuntu 220 Jan  6 2022 .bash_logout
-rw-r--r-- 1 ubuntu ubuntu 972 Jan 28 04:22 .bashrc
lrwxrwxrwx 1 root   root   7 Jan 28 05:37 .c9 -> /opt/c9
drwxr-xr-x 2 ubuntu ubuntu 4096 Jan 29 22:57 .cache
-rw-rw-r-- 1 ubuntu ubuntu 102 Jan 29 23:43 .gitconfig
drwxrwxr-x 3 ubuntu ubuntu 4096 Jan 28 05:40 .npm
-rw-r--r-- 1 ubuntu ubuntu 807 Jan  6 2022 .profile
drwxr-xr-x 2 ubuntu ubuntu 4096 Jan 28 05:21 .ssh
-rw-r--r-- 1 root   root   0 Jan 28 05:28 .sudo_as_admin_successful
drwxr-xr-x 2 ubuntu ubuntu 4096 Jan 28 05:29 environment
:~ $ []
```

Una vez que estés conectado a la instancia EC2, puedes usar el siguiente comando para verificar el sistema operativo:

d. Uname -a

```
:~ $ uname -a
Linux ip-172-31-84-29 6.8.0-1021-aws #23~22.04.1-Ubuntu SMP Tue Dec 10 16:50:46 UTC 2024 x86_64 x86_64 x86_64 GNU/Linu
x
```

e. Df -h: Para verificar el espacio de almacenamiento en la instancia EC2, utiliza el siguiente df -h

```
:~ $ df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/root       20G   5.8G  14G  31% /
tmpfs          458M    0  458M  0% /dev/shm
tmpfs          183M  888K 182M  1% /run
tmpfs          5.0M    0  5.0M  0% /run/lock
efivarfs        128K  3.8K 120K  4% /sys/firmware/efi/efivars
/dev/nvme0n1p15 105M  6.1M  99M  6% /boot/efi
tmpfs          92M   4.0K  92M  1% /run/user/1000
```

f. Cat /etc/os-release: También puedes usar el siguiente comando para obtener más detalles específicos del sistema operativo.

```
:~ $ cat /etc/os-release
PRETTY_NAME="Ubuntu 22.04 LTS"
NAME="Ubuntu"
VERSION_ID="22.04"
VERSION="22.04.5 LTS (Jammy Jellyfish)"
VERSION_CODENAME=jammy
ID=ubuntu
ID_LIKE=debian
HOME_URL="https://www.ubuntu.com/"
SUPPORT_URL="https://help.ubuntu.com/"
BUG_REPORT_URL="https://bugs.launchpad.net/ubuntu/"
PRIVACY_POLICY_URL="https://www.ubuntu.com/legal/terms-and-policies/privacy-policy"
UBUNTU_CODENAME=jammy
:~ $ []
```

g. Free -h: Verificar la memoria.

```
UBUNTU_CODENAME=jammy
:~ $ free -h
              total        used        free      shared  buff/cache   available
Mem:      914Mi       263Mi       90Mi      4.0Mi      560Mi      493Mi
Swap:     488Mi       322Mi      166Mi
:~ $ []
```

h. Ip a: Verificar las interfaces de red.

```
:~ $ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
2: ens5: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 9001 qdisc mq state UP group default qlen 1000
    link/ether 12:86:70:6a:2a:8f brd ff:ff:ff:ff:ff:ff
    altname enp0s5
    inet 172.31.84.29/20 metric 100 brd 172.31.95.255 scope global dynamic ens5
        valid_lft 2949sec preferred_lft 2949sec
    inet6 fe80::1086:70ff:fe6a:2a8f/64 scope link
        valid_lft forever preferred_lft forever
3: docker0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc noqueue state DOWN group default
    link/ether 02:42:dd:6b:59:75 brd ff:ff:ff:ff:ff:ff
    inet 172.17.0.1/16 brd 172.17.255.255 scope global docker0
        valid_lft forever preferred_lft forever
```

i. Verificar los procesos en ejecución:

```
top - 00:03:39 up 1:12, 1 user, load average: 0.00, 0.00, 0.00
Tasks: 112 total, 1 running, 111 sleeping, 0 stopped, 0 zombie
%Cpu(s): 0.2 us, 0.2 sy, 0.0 ni, 99.5 id, 0.0 wa, 0.0 hi, 0.0 si, 0.2 st
MiB Mem : 914.1 total, 88.4 free, 264.1 used, 561.6 buff/cache
MiB Swap: 488.3 total, 166.3 free, 322.0 used. 492.6 avail Mem

PID USER PR NI VIRT RES SHR S %CPU %MEM TIME+ COMMAND
713 mysql 20 0 1785884 88384 21120 S 0.7 9.4 0:29.16 mysqld
2259 ubuntu 20 0 10908 3848 3200 R 0.3 0.4 0:00.01 top
  1 root 20 0 101904 12792 8312 S 0.0 1.4 0:01.80 systemd
  2 root 20 0 0 0 0 S 0.0 0.0 0:00.00 kthreadd
  3 root 20 0 0 0 0 S 0.0 0.0 0:00.00 pool_workqueue_release
  4 root 0 -20 0 0 0 I 0.0 0.0 0:00.00 kworker/R-rCU_p
  5 root 0 -20 0 0 0 I 0.0 0.0 0:00.00 kworker/R-rCU_p
  6 root 0 -20 0 0 0 I 0.0 0.0 0:00.00 kworker/R-slab_
  7 root 0 -20 0 0 0 I 0.0 0.0 0:00.00 kworker/R-netns
  9 root 0 -20 0 0 0 I 0.0 0.0 0:00.00 kworker/0:H-events_highpri
 10 root 20 0 0 0 0 I 0.0 0.0 0:00.17 kworker/0:1:cgroup_destroy
 12 root 0 -20 0 0 0 I 0.0 0.0 0:00.00 kworker/R-mm_pe
 13 root 20 0 0 0 0 I 0.0 0.0 0:00.00 rCU_tasks_rude_kthread
 14 root 20 0 0 0 0 I 0.0 0.0 0:00.00 rCU_tasks_trace_kthread
 15 root 20 0 0 0 0 S 0.0 0.0 0:00.05 ksftirqd/0
 16 root 20 0 0 0 0 I 0.0 0.0 0:00.20 rCU_sched
 17 root rt 0 0 0 0 S 0.0 0.0 0:00.02 migration/0
 18 root -51 0 0 0 0 S 0.0 0.0 0:00.00 idle_inject/0
 19 root 20 0 0 0 0 S 0.0 0.0 0:00.00 cpuhp/0
 20 root 20 0 0 0 0 S 0.0 0.0 0:00.00 cpuhp/1
 21 root -51 0 0 0 0 S 0.0 0.0 0:00.00 idle_inject/1
 22 root rt 0 0 0 0 S 0.0 0.0 0:00.07 migration/1
 23 root 20 0 0 0 0 S 0.0 0.0 0:00.05 ksftirqd/1
 25 root 0 -20 0 0 0 I 0.0 0.0 0:00.00 kworker/1:H-events_highpri
```

► Integración con Cloud9

Pasos y comandos empleados para la creación de la instancia Cloud9 corriendo dentro de la máquina EC2.

1. Presionamos cloud9.

The screenshot shows the AWS Cloud9 'Create environment' interface. At the top, there's a navigation bar with AWS logo, search bar, and account information ('Estados Unidos (Norte de Virginia)'). Below it, a breadcrumb trail shows 'AWS Cloud9 > Entornos > Crear entorno'. A message box says: 'Para obtener capacidades similares a las de AWS Cloud9, explore los kits de herramientas de AWS en su propio IDE y AWS CloudShell en la consola de administración de AWS. Más información'.

Crear entorno Info

Detalles

Nombre Límite de 60 caracteres alfanuméricos y únicos por usuario.

Descripción: opcional Límite de 200 caracteres.

Tipo de entorno Info Nueva instancia de EC2. Determina en qué se ejecutará el IDE de Cloud9.

Computación existente Ya tiene una instancia o un servidor que desea usar.

2. En computación existente copiamos la clave en el portapapeles y vamos al terminal inicial.

Computación existente

AWS Cloud9 usa la clave pública SSH para conectarse de forma segura a su servidor.
Para crear una conexión segura, agregue nuestra clave pública a su archivo `~/.ssh/authorized_keys` e ingrese sus credenciales de inicio de sesión a continuación.

Usuario
El nombre de inicio de sesión que usa para conectarse a la instancia o al servidor. Se recomienda que el nombre de inicio de sesión esté asociado a los permisos administrativos o a un usuario administrador de la instancia o el servidor.
`ubuntu`

Host
La dirección IP pública (preferida) o el nombre de host de la instancia o el servidor.
`3.92.221.167`

Número de puerto
El puerto que desea que AWS Cloud9 utilice para conectarse a la instancia o al servidor, o utilice el puerto predeterminado 22.
`22`

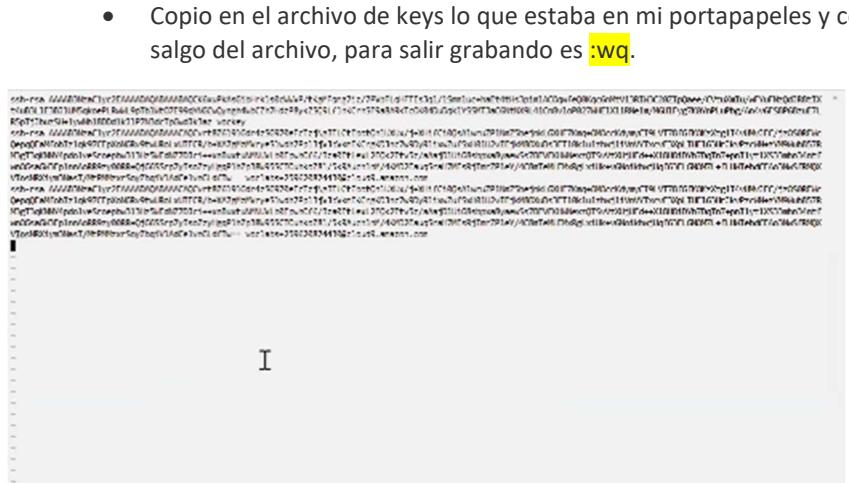
► **Detalles adicionales: opcional**
Ruta del entorno, ruta al binario node.js y host de salto SSH.

3. En el terminal ponemos

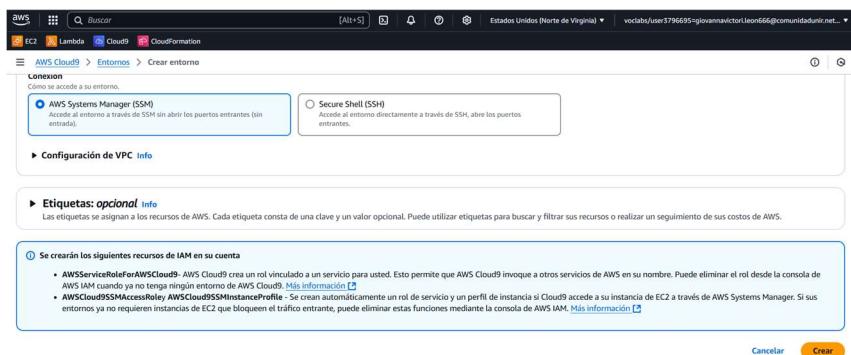
- `~$ pwd`
`/home/Ubuntu`
- Editamos el archivo de claves
`~$ sudo vi .ssh/authorized_keys`

```
27 root      0 -20      0      0      0 I   0.0  0.0  0:00.00 kworker/R-inet_
:~ $ pwd
/home/ubuntu
:~ $ sudo vi .ssh/authorized_keys
```

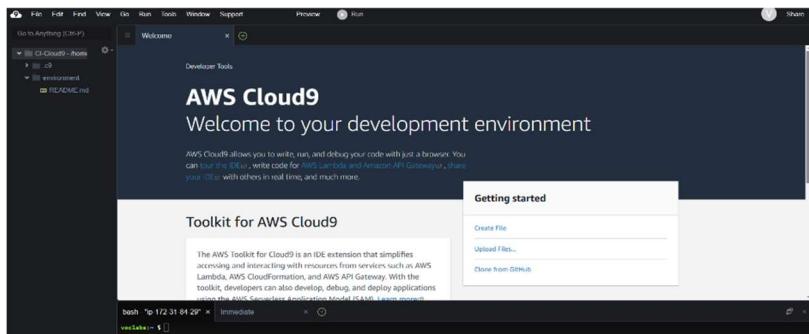
- Copio en el archivo de keys lo que estaba en mi portapapeles y con el comando wq salgo del archivo, para salir grabando es `:wq`.



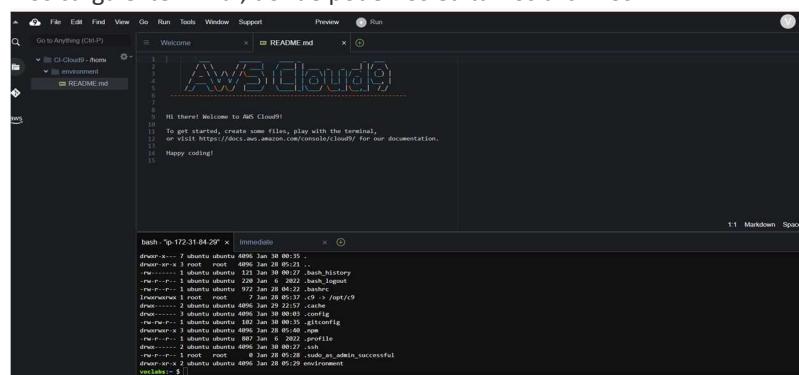
5. Volvemos a Cloud9, en la parte inferior de la página de creación de entornos presiono el botón inferior naranja “Crear”.



6. Si todo esta correcto se crea el entorno para cloud9, y presionamos en el hipervínculo abrir para entrar a CLOUD9.



7. Nos carga el terminal, donde podemos editar los archivos.



Cloud9 Corriendo en mi navegador

The screenshot shows a Cloud9 IDE interface with several tabs open. The terminal tab displays a series of curl commands being executed against an AWS Lambda function endpoint. The browser tab shows the results of one of these requests, which returns an 'Internal server error' message.

```

$ curl -X POST https://axjbrp1w8j.execute-api.us-east-1.amazonaws.com/Prod/todos
{
    "error": "Internal server error"
}
  
```

► Instalación de Jenkins.

1. Estos son los comandos para la instalación de Jenkins.

The screenshot shows the Jenkins website's documentation page for Debian/Ubuntu. It includes a 'Long Term Support release' section and a terminal command for keyring installation:

```

sudo wget -O /usr/share/keyrings/jenkins-keyring.asc \
  https://pkg.jenkins.io/debian-stable/jenkins.io-2023.key
  
```

2. En el terminal de cloud9 vemos el certificado del repositorio.

```
sudo wget -O /usr/share/keyrings/jenkins-keyring.asc \
  https://pkg.jenkins.io/debian-stable/jenkins.io-2023.key
```

The screenshot shows the Cloud9 terminal executing the command to download the Jenkins repository keyring. The output shows the file being saved to /usr/share/keyrings/jenkins-keyring.asc.

```

$ sudo wget -O /usr/share/keyrings/jenkins-keyring.asc \
  https://pkg.jenkins.io/debian-stable/jenkins.io-2023.key
  
```

3. Añadimos el repositorio.

```
echo "deb [signed-by=/usr/share/keyrings/jenkins-keyring.asc]" \
https://pkg.jenkins.io/debian-stable binary/ | sudo tee \
/etc/apt/sources.list.d/jenkins.list > /dev/null

voclab:- $ echo "deb [signed-by=/usr/share/keyrings/jenkins-keyring.asc]" \
> https://pkg.jenkins.io/debian-stable binary/ | sudo tee \
> /etc/apt/sources.list.d/jenkins.list > /dev/null
voclab:- $
```

4. Actualizo

```
sudo apt-get update
```

```
sudo -p-172-31-84-29* x Immediate
Get:36 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/main amd64 c-n-f Metadata [388 B]
Get:37 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/restricted amd64 c-n-f Metadata [116 B]
Get:38 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/universe amd64 Packages [30.0 kB]
Get:39 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/universe Translation-en [16.6 kB]
Get:40 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/universe amd64 c-n-f Metadata [672 B]
Get:41 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/multiverse amd64 c-n-f Metadata [116 B]
Get:42 https://pkg.jenkins.io/debian-stable binary/ Packages [28.4 kB]
Get:43 http://security.ubuntu.com/ubuntu jammy-security/main amd64 Packages [2075 kB]
Get:44 http://security.ubuntu.com/ubuntu jammy-security/main Translation-en [324 kB]
Get:45 http://security.ubuntu.com/ubuntu jammy-security/restricted amd64 Packages [2836 kB]
Get:46 http://security.ubuntu.com/ubuntu jammy-security/restricted Translation-en [498 kB]
Get:47 http://security.ubuntu.com/ubuntu jammy-security/universe amd64 Packages [961 kB]
Fetched 37.7 MB in 8s (4959 kB/s)
Reading package lists... Done
voclab:- $
```

5. Y por último instalamos

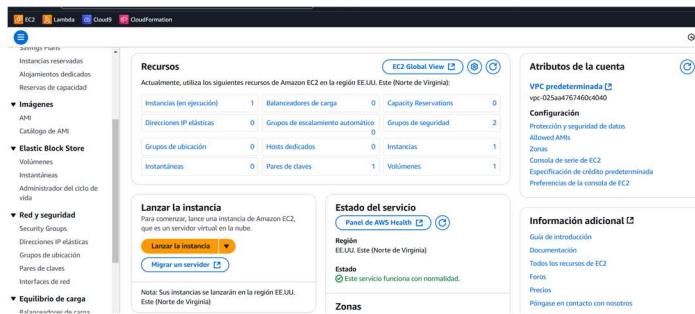
```
sudo apt-get install Jenkins
```

```
sudo -p-172-31-84-29* x Immediate
Get:47 http://security.ubuntu.com/ubuntu jammy-security/universe amd64 Packages [961 kB]
Fetched 37.7 MB in 8s (4959 kB/s)
Reading package lists... Done
voclab:- $ sudo apt-get install jenkins
Reading package lists... Done
Building dependency tree...
Reading state information...
The following NEW packages will be installed:
  jenkins net-tools
The following NEW packages will be installed:
  jenkins net-tools
0 upgraded, 2 newly installed, 0 to remove and 5 not upgraded.
Need to get 95.1 MB of archives.
After this operation, 97.7 MB of additional disk space will be used.
Do you want to continue? [Y/n] :
```

Ya está instalado Jenkins

► Creación de IP elástica asociada a la máquina EC2

1. En la ventana EC2, en la sección izquierda seleccionamos “Direcciones IP Elásticas”.



2. Presionamos “Asignar dirección IP Elástica”.



3. Y luego presionamos asignar, del botón naranja que está en la parte inferior derecha.



4. Marcamos “Asociar esta dirección IP elástica” del botón verde superior a la derecha



5. Nos muestra la pantalla donde nos indica que esta asociada. IP 3.210.124.61



6. Ahora ya tenemos nuestra ip elástica asociada a nuestra instancia

The screenshot shows the AWS Cloud9 interface with the 'Entornos' tab selected. A modal window titled 'Direcciones IP elásticas (1/1)' is open, displaying a single entry for the IP address 3.210.124.61. The entry includes columns for Name, Dirección IPv4 asignada, Tipo, ID de asignación, and Registro DNS inverso. The 'Name' column shows '3.210.124.61', 'Tipo' shows 'IP pública', and 'ID de asignación' shows 'ipalloc-0d7e026e4dc788cc0'. Below the table, a note says 'Vea el uso de las direcciones IP y las recomendaciones para liberar las IP no utilizadas con Public IP insights.' At the bottom of the modal, there are tabs for 'Resumen' and 'Etiquetas'.

7. Para que mi Cloud9 funcione, tengo que ir a mi Cloud9 y poner la ip elástica.

The screenshot shows the AWS Cloud9 interface with the 'Entornos' tab selected. A table lists one environment named 'CI-Cloud9'. The columns include Nombre, Cloud9 IDE, Tipo de entorno, Conexión, Permiso, and ARN del propietario. The 'Nombre' column shows 'CI-Cloud9', 'Tipo de entorno' shows 'Computación existente', 'Conexión' shows 'Secure Shell (SSH)', and 'Permiso' shows 'Propietario'. The 'ARN del propietario' column shows 'arn:aws:ssts::582789485636:assumed-role/voclabs/user3796695:giovannavictori.leon666@comunidadunir.net'.

8. Presionamos el botón “Editar”.

The screenshot shows the AWS Cloud9 interface with the 'Entornos' tab selected. It displays the 'CI-Cloud9' environment. Below the main table, there are two sections: 'Computación existente' and 'Etiquetas'. The 'Computación existente' section contains fields for 'Usuario' (ubuntu), 'Número de puerto' (22), 'Host' (5.92.221.167), and 'Trayectoria del entorno' (/home/ubuntu). The 'Etiquetas' section is currently empty.

9. Y modificamos la ip pública por la elástica

The screenshot shows the AWS Cloud9 interface with the 'Entornos' tab selected. A modal dialog titled 'AWS Cloud9 > Entornos > CI-Cloud9 > Editar' is open. Inside the dialog, there is a warning message: 'AWS Cloud9 usa la clave pública SSH para conectarse de forma segura a su servidor. Para crear una conexión segura, agregue nuestra clave pública a su archivo ~/.ssh/authorized_keys e ingrese sus credenciales de inicio de sesión a continuación.' Below this, there are fields for 'Usuario' (ubuntu), 'Host' (3.210.124.61), 'Número de puerto' (22), and 'Detalles adicionales: opcional' (Ruta del entorno, ruta al binario node.js y host de salto SSH). At the bottom right of the dialog are 'Cancelar' and 'Guardar cambios' buttons.

10. Guardamos los cambios.

Reto 3 – Stack Serverless

<https://github.com/GiovannaLeon/todo-list-aws.git>

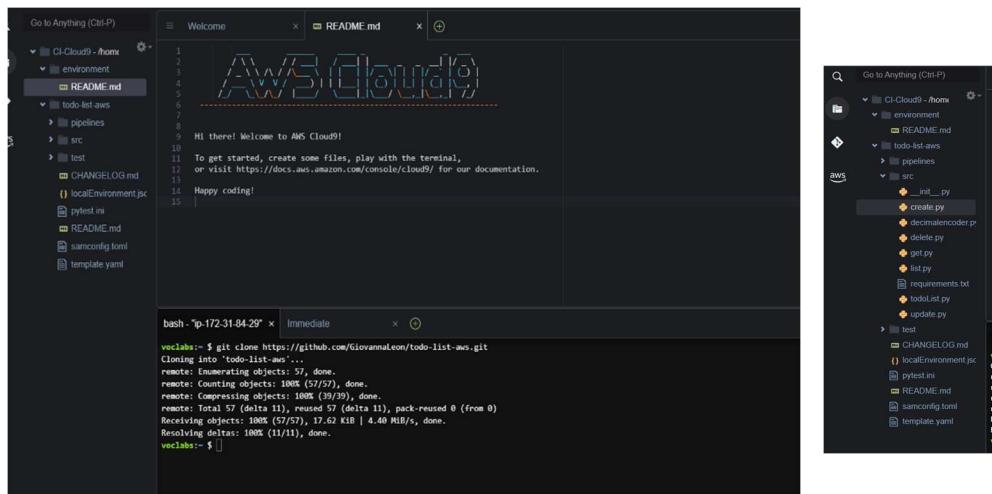
En este reto se solicitan 4 entregables:

- ▶ Descarga del repositorio del proyecto todo-list-aws, en la máquina EC2.

1. En awsCloud9 indicar el comando

2. En la sección izquierda se muestra el contenido de nuestro repositorio de github

Git clone https://github.com/GiovannaLeon/todo-list-aws.git



- ▶ Construcción de la plantilla SAM

1. Ingreso a la carpeta todo-list-aws

vclabs:~ \$ cd todo-list-aws/

2. Listado el contenido, vemos el template, los archivos de código etc

vclabs:~/todo-list-aws (master) \$ ls -la

```

voclabs:~ $ git clone https://github.com/GiovannaLeon/todo-list-aws.git
voclabs:~ $ git clone https://github.com/GiovannaLeon/todo-list-aws.git
Cloning into 'todo-list-aws'...
remote: Enumerating objects: 57, done.
remote: Counting objects: 100% (57/57), done.
remote: Compressing objects: 100% (39/39), done.
remote: Total 57 (delta 11), reused 57 (delta 11), pack-reused 0 (from 0)
Receiving objects: 100% (57/57), 17.62 KiB | 4.40 MiB/s, done.
Resolving deltas: 100% (11/11), done.
voclabs:~ $ ls
environment todo-list-aws
voclabs:~ $ cd todo-list-aws/
voclabs:~/todo-list-aws (master) $ ls -la
total 60
drwxrwxr-x 6 ubuntu ubuntu 4096 Jan 30 09:27 .
drwxr-x--- 8 ubuntu ubuntu 4096 Jan 30 09:27 ..
drwxrwxr-x 8 ubuntu ubuntu 4096 Jan 30 09:27 .gitignore
-rw-rw-r-- 1 ubuntu ubuntu 72 Jan 30 09:27 CHANGELOG.md
-rw-rw-r-- 1 ubuntu ubuntu 318 Jan 30 09:27 README.md
-rw-rw-r-- 1 ubuntu ubuntu 5819 Jan 30 09:27 localEnvironment.json
-rw-rw-r-- 1 ubuntu ubuntu 775 Jan 30 09:27 pipelines
drwxrwxr-x 6 ubuntu ubuntu 4096 Jan 30 09:27 pytest.ini
-rw-rw-r-- 1 ubuntu ubuntu 965 Jan 30 09:27 samconfig.toml
drwxrwxr-x 2 ubuntu ubuntu 4096 Jan 30 09:29 src
-rw-rw-r-- 1 ubuntu ubuntu 4135 Jan 30 09:27 template.yaml
drwxrwxr-x 4 ubuntu ubuntu 4096 Jan 30 09:27 test
voclabs:~/todo-list-aws (master) $ 

```

3. Ahora compilamos la plantilla

```
voclabs:~/todo-list-aws (master) $ sam build
```

```

Building codeuri: /home/ubuntu/todo-list-aws/src runtime: python3.10 metadata: {} architecture: x86_64 functions: CreateTodoFunction, ListTodosFunction, GetTodoFunction
UpdateTodoFunction, DeleteTodoFunction
Running PythonPipBuilder:ResolveDependencies
Running PythonPipBuilder:CopySource
Build Succeeded
Built Artifacts : .aws-sam/build
Built Template : .aws-sam/build/template.yaml

Commands you can use next
=====
[*] Validate SAM template: sam validate
[*] Invoke Function: sam local invoke
[*] Test Function in the Cloud: sam sync --stack-name {{(stack-name)}} --watch
[*] Deploy: sam deploy --guided

SAM CLI update available (1.112.0); (1.112.0 installed)
To download: https://docs.aws.amazon.com/serverless-application-model/latest/developerguide/serverless-sam-cli-install.html
voclabs:~/todo-list-aws (master) $ 

Building codeuri: /home/ubuntu/todo-list-aws/src runtime: python3.10 metadata: {} architecture: x86_64 functions: CreateTodoFunction, ListTodosFunction, GetTodoFunction
UpdateTodoFunction, DeleteTodoFunction
Running PythonPipBuilder:ResolveDependencies
Running PythonPipBuilder:CopySource
Build Succeeded
Built Artifacts : .aws-sam/build
Built Template : .aws-sam/build/template.yaml

Commands you can use next
=====
[*] Validate SAM template: sam validate
[*] Invoke Function: sam local invoke
[*] Test Function in the Cloud: sam sync --stack-name {{(stack-name)}} --watch
[*] Deploy: sam deploy --guided

SAM CLI update available (1.112.0); (1.112.0 installed)
To download: https://docs.aws.amazon.com/serverless-application-model/latest/developerguide/serverless-sam-cli-install.html
voclabs:~/todo-list-aws (master) $ sam deploy --guided

Configuring SAM deploy
=====
Looking for config file [samconfig.toml]: Found
Reading default arguments : Success
Setting default arguments for 'sam deploy'
=====
Stack Name [todo-list-aws]: 

```

► Despliegue en AWS del stack y revisión de los Outputs

Comandos y parámetros empleados, junto con sus salidas.

Explicación de qué nos indican las salidas “Outputs”.

Stack Name [todo-list-aws]: El nombre de la pila que será desplegada (aceptar el valor predeterminado o personalizar).

AWS Region [us-east-1]: La región en la que se desplegará la pila (puedes cambiarla si es necesario).

Parameter Stage [default]: El stage de la aplicación (puedes personalizarlo si tienes diferentes entornos).

Allow SAM CLI IAM role creation [Y/n]: Permitir que SAM CLI cree un rol IAM (responder Y para permitirlo).

Disable rollback [y/N]: Si deshabilitar el rollback (responder n para mantenerlo habilitado y revertir cambios en caso de error)

Stack Name [todo-list-aws]:

AWS Region [us-east-1]:

Parameter Stage [default]:

#Shows you resources changes to be deployed and require a 'Y' to initiate deploy

Si queremos confirmar los cambio antes de desplegarlo:

Allow SAM CLI IAM role creation [Y/n]: y

Deseo que cree errores desimos que si

Allow SAM CLI IAM role creation [Y/n]: y

Desabilitamos el rollback decimos que no

Disable rollback [y/N]:

4. Si queremos guardar el fichero decimos que si

```
Looking for config file [samconfig.toml] : Found
Reading default arguments : Success

Setting default arguments for 'sam deploy'
-----
Stack Name [todo-list-aws]:
AWS Region [us-east-1]:
Parameter Stage [default]:
#Shows you resources changes to be deployed and require a 'Y' to initiate deploy
Confirm changes before deploy [y/N]:
#Shows you resources changes to be deployed and require a 'Y' to initiate deploy      #SAM needs permission

Allow SAM CLI IAM role creation [Y/n]: y
Error: Invalid input
Allow SAM CLI IAM role creation [Y/n]: y
#Preserves the state of previous provisioned resources when an operation fails
Disable rollback [y/N]:
CreateTodoFunction has no authentication. Is this okay? [y/N]: y
ListTodosFunction has no authentication. Is this okay? [y/N]: y
GetTodoFunction has no authentication. Is this okay? [y/N]: y
UpdateTodoFunction has no authentication. Is this okay? [y/N]: y
DeleteTodoFunction has no authentication. Is this okay? [y/N]: y
Save arguments to configuration file [Y/n]:
```

En la sesión default

```
CreateTodoFunction has no authentication. Is this okay? [y/N]: y
ListTodosFunction has no authentication. Is this okay? [y/N]: y
GetTodoFunction has no authentication. Is this okay? [y/N]: y
UpdateTodoFunction has no authentication. Is this okay? [y/N]: y
DeleteTodoFunction has no authentication. Is this okay? [y/N]: y
Save arguments to configuration file [Y/n]:
SAM configuration file [samconfig.toml]:
SAM configuration environment [default]:
```

5. Ahora empieza a calcular todo lo que se necesita para este stage

Resultado

```

CloudFormation outputs from deployed stack
-----
Outputs
-----
Key     BaseUrlApi
DescriptionBase URL of API
Value    https://axjbrpiw8j.execute-api.us-east-1.amazonaws.com/Prod

Key      DeleteTodoApi
Description API Gateway endpoint URL for ${opt:stage} stage for Delete TODO
Value    https://axjbrpiw8j.execute-api.us-east-1.amazonaws.com/Prod/todos/{id}

Key      ListTodosApi
Description API Gateway endpoint URL for ${opt:stage} stage for List TODO
Value    https://axjbrpiw8j.execute-api.us-east-1.amazonaws.com/Prod/todos

Key      UpdateTodoApi
Description API Gateway endpoint URL for ${opt:stage} stage for Update TODO
Value    https://axjbrpiw8j.execute-api.us-east-1.amazonaws.com/Prod/todos/{id}

Key      GetTodoApi
Description API Gateway endpoint URL for ${opt:stage} stage for Get TODO
Value    https://axjbrpiw8j.execute-api.us-east-1.amazonaws.com/Prod/todos/{id}

Key      CreateTodoApi
Description API Gateway endpoint URL for ${opt:stage} stage for Create TODO
Value    https://axjbrpiw8j.execute-api.us-east-1.amazonaws.com/Prod/todos/
-----

Successfully created/updated stack - todo-list-aws in us-east-1
voclabs:~/todo-list-aws (master) $ 

```

Si pincho la tercera url, para probar se abre en el navegador y veo que es un error.



- Comprobación, en CloudFormation, de que el Stack se ha creado correctamente.

Se ha generado correctamente

1. Entramos a cloudformation , y vemos todo-list-aws.

A screenshot of the AWS CloudFormation console. The left sidebar shows navigation options like EC2, Lambda, Cloud9, and CloudFormation. Under CloudFormation, it says "Pilas" (Stacks) and lists three stacks: "todo-list-aws", "aws-sam-cli-managed-default", and "c149604a3851739f9064558t1w582789_485636". The main area shows a summary message: "Presentamos la nueva experiencia y las capacidades de los Enlaces. El nuevo flujo de trabajo de la consola de Enlaces simplifica la creación de enlaces mediante utilizar funciones de Lambda o Lenguaje específico del dominio (DSL) de Guard. También puede más información acerca de los destinos de evaluación ampliados." Below this, there's a table titled "Pilas (3)" with columns: Nombre de la pila, Estado, Hora de creación, and Descripción. The table data is as follows:

Nombre de la pila	Estado	Hora de creación	Descripción
todo-list-aws	CREATE_COMPLETE	2025-01-30 11:09:42 UTC+0100	todo-list-aws Application TODO-LIST with SAM format
aws-sam-cli-managed-default	CREATE_COMPLETE	2025-01-30 11:08:54 UTC+0100	Managed Stack for AWS SAM CLI
c149604a3851739f9064558t1w582789_485636	CREATE_COMPLETE	2025-01-29 02:27:10 UTC+0100	associate Learner Lab template (academy)

2. Se muestra todas las funciones cargadas en todo-list-aws

Información

todo-list-aws

Información de la pila

todo-list-aws

Información general

ID de pila: arn:aws:cloudformation:us-east-1:582789485636:stack/todo-list-aws/52fc40d0-def2-11ef-aa6a-128907f2cd89

Estado: CREATE_COMPLETE

Motivo del estado: -

Pila principal: Pila raíz

Descripción: todo-list-aws Application TODO-LIST with SAM format

Estado detallado: -

Hora de creación: 2025-01-30 11:09:42 UTC+0100

Hora de actualización: 2025-01-30 11:09:48 UTC+0100

Estado de desviación: -

Hora de eliminación: -

Outputs

Nombre	Valor
BaseUrlApi	Base URL of API https://2if2jvbjo5.execute-api.us-east-1.amazonaws.com/Prod
DeleteTodoApi	API Gateway endpoint URL for \${opt:stage} stage for Delete TODO https://2if2jvbjo5.execute-api.us-east-1.amazonaws.com/Prod/todos/{id}
ListTodosApi	API Gateway endpoint URL for \${opt:stage} stage for List TODO https://2if2jvbjo5.execute-api.us-east-1.amazonaws.com/Prod/todos
UpdateTodoApi	API Gateway endpoint URL for \${opt:stage} stage for Update TODO https://2if2jvbjo5.execute-api.us-east-1.amazonaws.com/Prod/todos/{id}
CreateTodoApi	API Gateway endpoint URL for \${opt:stage} stage for Create TODO https://2if2jvbjo5.execute-api.us-east-1.amazonaws.com/Prod/todos/

Reto 3 – Verificación API

En este reto se solicitan 4 entregables:

► Ejecución de los comandos *curl*, con la API recién desplegada.

Estos comandos deben mostrar un error.

```
git:(master) % sam build
git:(master) % sam deploy --guided
? Enter a stage name (optional): Prod
? Enter a profile name (optional): 
? Region: us-east-1
? Stack name: todo-list-aws
? Function name: todo-list-aws
? Handler: lambda_handler.lambda_handler
? Role ARN: arn:aws:lambda:us-east-1:123456789012:role/lambdaBasicExecutionRole
? IAM Policy: LambdaBasicExecutionPolicy
? API Endpoint: https://2if2jvbjo5.execute-api.us-east-1.amazonaws.com/Prod/todos
? API Endpoint: https://2if2jvbjo5.execute-api.us-east-1.amazonaws.com/Prod/todos/{id}
? API Endpoint: https://2if2jvbjo5.execute-api.us-east-1.amazonaws.com/Prod/todos
? API Endpoint: https://2if2jvbjo5.execute-api.us-east-1.amazonaws.com/Prod/todos/{id}

CloudFormation outputs from deployed stack
Outputs
-----
Key          Description          Value
BaseUrlApi  Base URL of API    https://2if2jvbjo5.execute-api.us-east-1.amazonaws.com/Prod
DeleteTodoApi API Gateway endpoint URL for ${opt:stage} stage for Delete TODO https://2if2jvbjo5.execute-api.us-east-1.amazonaws.com/Prod/todos/{id}
ListTodosApi API Gateway endpoint URL for ${opt:stage} stage for List TODO   https://2if2jvbjo5.execute-api.us-east-1.amazonaws.com/Prod/todos
UpdateTodoApi API Gateway endpoint URL for ${opt:stage} stage for Update TODO https://2if2jvbjo5.execute-api.us-east-1.amazonaws.com/Prod/todos/{id}
CreateTodoApi API Gateway endpoint URL for ${opt:stage} stage for Create TODO  https://2if2jvbjo5.execute-api.us-east-1.amazonaws.com/Prod/todos/
```

Successfully created/updated stack - todo-list-aws in us-east-1



```
{
  "errorMessage": "The 'dynamodb-dbddb' resource does not exist.\n\nThe available resources are:\n  - cloudformation\n  - cloudwatch\n  - dynamodb\n  - ec2\n  - glacier\n  - iam\n  - opsworks\n  - sns\n  - sqs",
  "errorType": "ResourceNotFoundException",
  "requestId": "4c687743-8e09-4f14-a535-9d5680cc51e4",
  "stackTrace": [
    "File \"./var/task/list.py\", line 8, in list\n        result = todolist.get_items()\n      ^\n    File \"./var/task/todolist.py\", line 42, in get_items\n        table = get_table(dynamodb)\n      ^\n    File \"./var/task/todolist.py\", line 18, in get_table\n        dynamodb = boto3.resource('dynamodb-dbddb')\n      ^\n    File \"./var/runtime/boto3/_init_.py\", line 101, in resource\n        return _get_default_session().resource(*args, **kwargs)\n      ^"
  ]
}
```

- ▶ Ejecución manual de la función lambda del listado de To-Do's, para obtener el detalle del error producido en la invocación a esta función.

1. Pero hay un error, para localizarlo entramos a todo.list.aws de cloud9

ID de pila	Descripción
arn:aws:cloudformation:us-east-1:582789485636:stack/todo-list-aws/52fc40d0-def2-11ef-aa6a-128907f2cd89	todo-list-aws Application TODO-LIST with SAM format

Estado	Estado detallado
CREATE_COMPLETE	-

Motivo del estado	Pila raíz
-	-

Pila principal	Hora de creación
-	2025-01-30 11:09:42 UTC+0100

Pila secundaria	Hora de actualización
-	2025-01-30 11:09:48 UTC+0100

Hora de eliminación	Estado de desviación
-	-

2. En recursos, buscamos la función ListTodosFunction,

ID lógico	ID físico	Tipo	Estado
permissionprod	arn:aws:iam::prod-nP8tBmfsfb086	AWS::Lambda::Permission	CREATE_COMPLETE
GetTodoFunction	todo-list-aws-GetTodoFunction-R8ayvGjMKn6a	AWS::Lambda::Function	CREATE_COMPLETE
GetTodoFunctionCreatePermissionProd	todo-list-aws-GetTodoFunctionCreatePermissionProd-TLUGhV4maKsr	AWS::Lambda::Permission	CREATE_COMPLETE
ListTodosFunction	todo-list-aws-ListTodosFunction-msr35sd8kExL	AWS::Lambda::Function	CREATE_COMPLETE
ListTodosFunctionCreatePermissionProd	todo-list-aws-ListTodosFunctionCreatePermissionProd-GHVB5sWQpuWm	AWS::Lambda::Permission	CREATE_COMPLETE
ServerlessRestApi	axjbrpw8j	AWS::ApiGateway::RestAPI	CREATE_COMPLETE

3. Entramos para que nos lleve a la función lamda, donde nos muestra la información de esta función.

The screenshot shows the AWS Lambda console. In the top navigation bar, it says 'Lambda > Funciones > todo-list-aws-ListTodosFunction-msF3SdsbKEIx'. Below the title 'todo-list-aws-ListTodosFunction-msF3SdsbKEIx' are buttons for 'Limitación', 'Copiar ARN', and 'Accion'. A note says 'Esta función pertenece a una aplicación. Haga clic aquí para administrarla.' On the left, there's a sidebar with 'Información general de la función' and 'Diagrama'. The main area shows a diagram where the function is connected to an 'API Gateway' endpoint. To the right, there's a 'Descripción' section with details like 'Última modificación hace 14 minutos', 'ARN de la función arn:aws:lambda:us-east-1:582789485636:function:todo-list-ListTodosFunction-msF3SdsbKEIx', and an 'Aplicación todo-list-aws'.

5. Entramos a api Gateway

The screenshot shows the AWS API Gateway 'Desencadenadores' (Triggers) list. It lists one trigger named 'todo-list-aws' which points to the 'todo-list-aws' Lambda function. The URL for the trigger is https://zsjbrpwj.execute-api.us-east-1.amazonaws.com/Prod/todos.

6. Así es como funciona nuestra función lamda

The screenshot shows the AWS API Gateway 'Recursos' (Resources) page for the '/todos' endpoint. It shows the execution flow: Client → Solicitud de método → Solicitud de integración → Integración de proxy → Lambda. The ARN is arn:aws:execute-api:us-east-1:582789485636:axbrpwj/GET/todos and the ID de recurso is 4fh5ty.

7. Pero hay un error, entonces vamos a probar, donde nos detallarán el error.

The screenshot shows the AWS Lambda function details page with the 'Probar' (Test) tab selected. It shows the configuration for the test and the results of the last run, which failed with the message 'No se pudo cargar el código fuente del archivo zip. El archivo zip no contiene el archivo index.js o el archivo index.js no tiene la función de ejecución correcta.'

The screenshot shows the AWS Lambda console interface. In the top navigation bar, it says "Lambda > Funciones > todo-list-aws-ListTodosFunction-msf3SdabKEIx". Below this, a red box highlights the "Ejecutando la función: error (registros)" section. The error message is displayed in red text:

```

{
  "errorMessage": "The 'dynamodb-dbdbdb' resource does not exist.\nThe available resources are:\n - cloudformation\n - cloudwatch\n - dynamodb\n - ec2\n - glacier\n - iam\n - opsworks\n - s3\n - sns\n - sqs\n",
  "errorType": "ResourceNotFoundException"
}

```

Below the error message, there's a "Resumen" section with the following details:

- Código SHA-256: 1+u+jnijc5M2QzP61b0deI0ixJw8t2XU1Yp9QZ1cQCaA+
- ID de solicitud: 4c687743-8b49-4774-a535-9d568b651e4
- Duración de inicialización:
- Tiempo de ejecución: hace 2 segundos
- Versión de la función: \$LATEST
- Duración:

En la sección detalles podemos ver el error exactamente

```

{
  "errorMessage": "The 'dynamodb-dbdbdb' resource does not exist.\nThe available resources are:\n - cloudformation\n - cloudwatch\n - dynamodb\n - ec2\n - glacier\n - iam\n - opsworks\n - s3\n - sns\n - sqs\n",
  "errorType": "ResourceNotFoundException"
}

```

Se está buscando el recurso 'dynamodb-dbdbdb' , pero solo tenemos con este nombre dynamodb\n

► Corrección del código fuente y explicación del motivo del fallo

1. El error es que se está buscando el recurso 'dynamodb-dbdbdb' , pero solo tenemos con este nombre dynamodb\n en todolist.py

```

{
  "errorMessage": "The 'dynamodb-dbdbdb' resource does not exist.\nThe available resources are:\n - cloudformation\n - cloudwatch\n - dynamodb\n - ec2\n - glacier\n - iam\n - opsworks\n - s3\n - sns\n - sqs\n",
  "errorType": "ResourceNotFoundException"
}

```

2. Para solucionarlo vamos al código fuente, vemos que el error está en el fichero todoList, vamos a cloud9, al fichero todolist.py, y en la función gettable corrijo la línea de código que está mal.

The screenshot shows the Cloud9 IDE interface. On the left, there's a file tree with files like .gitignore, .htaccess, environment, README.md, template.yaml, create.py, todolist.py, update.py, requirements.txt, CHangelog.md, localenvironment.json, pytest.ini, and README.md. The main window shows the code for todolist.py:

```

def get_table(dynamodb=None):
    if not dynamodb:
        URL = os.environ['ENDPOINT_OVERRIDE']
        if URL:
            dynamodb = boto3.resource('dynamodb')
            dynamodb.meta.client.meta.endpoint_url = URL
        else:
            dynamodb = boto3.resource('dynamodb')
            dynamodb.meta.client.meta.endpoint_url = os.environ['ENDPOINT_OVERRIDE']

    table = dynamodb.Table(os.environ[DYNAMODB_TABLE])
    return table

```

A red box highlights the line "dynamodb = boto3.resource('dynamodb')". A tooltip for this line shows the error message: "API Gateway endpoint URL for \${opt:stage} stage for List TODO https://axxerplid4j.execute-api.us-east-1.amazonaws.com/Prod/todos".

3. Vemos que el error está en la Línea 18, la muestro en color rojo.

```

def get_table(dynamodb=None):
    if not dynamodb:
        URL = os.environ['ENDPOINT_OVERRIDE']
        if URL:
            print('URL dynamoDB:' + URL)
            boto3.client = functools.partial(boto3.client, endpoint_url=URL)

```

```

boto3.resource = functools.partial(boto3.resource,
                                    endpoint_url=URL)
dynamodb = boto3.resource("dynamodb")
# fetch todo from the database
table = dynamodb.Table(os.environ['DYNAMODB_TABLE'])
return table

```

4. Esta modificación se realiza en local, para pasarla a nuestro servidor es nueva sam build, donde revisa que este bien el código fuente

```
voclabs:~/todo-list-aws (master) $ sam build
voclabs:~/todo-list-aws (master) $ sam deploy --guided
```

```

$ sam build
$ sam deploy --guided

  sam -p 172.31.84.20 - x Immediate
  Key      Description          Value
  ListTodos          API Gateway endpoint URL for $opt(stage) stage for List TODO
  https://axjbrpiw8j.execute-api.us-east-1.amazonaws.com/Prod/todos
  GetTodos           API Gateway endpoint URL for $opt(stage) stage for Update TODO
  https://axjbrpiw8j.execute-api.us-east-1.amazonaws.com/Prod/todos/{id}
  GetTodosDl         API Gateway endpoint URL for $opt(stage) stage for Get TODO
  https://axjbrpiw8j.execute-api.us-east-1.amazonaws.com/Prod/todos/{id}
  CreateTodos        API Gateway endpoint URL for $opt(stage) stage for Create TODO
  https://axjbrpiw8j.execute-api.us-east-1.amazonaws.com/Prod/todos/
  UpdateTodos        API Gateway endpoint URL for $opt(stage) stage for Update TODO
  https://axjbrpiw8j.execute-api.us-east-1.amazonaws.com/Prod/todos/{id}

  Successfully created/updated stack - todo-list-aws in us-east-1
  voclabs:~/todo-list-aws (master) $ []

```

5. Actualizamos la url y listo, no vemos error.



- ▶ Ejecución de todos los comandos *curl*, que ahora sí deben mostrar datos de la lista de tareas.

El orden para seguir para la ejecución de los comandos será:

```

Successfully created/updated stack - todo-list-aws in us-east-1
voclabs:~/todo-list-aws (master) $ curl https://axjbrpiw8j.execute-api.us-east-1.amazonaws.com/Prod/todos
[]voclabs:~/todo-list-aws (master) $ []

```

- Alta de un elemento To-Do

```
curl -X POST https://axjbrpiw8j.execute-api.us-east-1.amazonaws.com/Prod/todos --data '{
  "text": "Realizar el CP1.1"
}'
```

```

Successfully created/updated stack - todo-list-aws in us-east-1
voclabs:~/todo-list-aws (master) $ curl https://axjbrpiw8j.execute-api.us-east-1.amazonaws.com/Prod/todos
[]voclabs:~/todo-list-aws (master) $ curl -X POST https://axjbrpiw8j.execute-api.us-east-1.amazonaws.com/Prod/todos --data '{
  "text": "Realizar el CP1.1"
}' --status-code
{"statusCode": 200, "body": "{\"id\": \"f1b1bfff-deef-48bc-194aab275ac1\", \"text\": \"Realizar el CP1.1\", \"checked\": false, \"createdAt\": \"2023-08-17T18:06:54.718866Z\", \"updatedAt\": \"2023-08-17T18:06:54.718866Z\""}[]voclabs:~/todo-list-aws (master) $ []

```

- Alta de otro elemento To-Do

```
curl -X POST https://axjbrpiw8j.execute-api.us-east-1.amazonaws.com/Prod/todos --data '{
  "text": "Aprender AWS"
}'
```

- Listamos todas la tareas

```
voclabs:~/todo-list-aws (master) $ curl https://axjbrpiw8j.execute-api.us-east-1.amazonaws.com/Prod/todos
```

```
voclabs:~/todo-list-aws (master) $ curl https://axjbrpiw8j.execute-api.us-east-1.amazonaws.com/Prod/todos
[{"checked": false, "createdAt": "1738235054.718066", "text": "Realizar el CP1.1", "id": "f13b1ffc-def9-11ef-85bc-194aab275ac1", "updatedAt": "1738235054.718066"}, {"checked": false, "createdAt": "1738235054.718066", "text": "Aprender AWS", "id": "6401baae-defa-11ef-aeff-194aab275ac1", "updatedAt": "1738235054.718066"}]voclabs:~/todo-list-aws (master) $
```

- Obtención un To-Do a partir de su identificador

```
curl https://axjbrpiw8j.execute-api.us-east-1.amazonaws.com/Prod/todos/f13b1ffc-def9-11ef-85bc-194aab275ac1
```

```
voclabs:~ $ curl https://axjbrpiw8j.execute-api.us-east-1.amazonaws.com/Prod/todos
[{"checked": false, "createdAt": "1738235054.718066", "text": "Realizar el CP1.1", "id": "f13b1ffc-def9-11ef-85bc-194aab275ac1", "updatedAt": "1738235054.718066"}]voclabs:~ $
voclabs:~ $ curl https://axjbrpiw8j.execute-api.us-east-1.amazonaws.com/Prod/todos/f13b1ffc-def9-11ef-85bc-194aab275ac1
{"checked": false, "createdAt": "1738235054.718066", "text": "Realizar el CP1.1", "id": "f13b1ffc-def9-11ef-85bc-194aab275ac1", "updatedAt": "1738235054.718066"}voclabs:~ $
```

- Modificacion de un To-Do a partir de su identificador

```
curl -X PUT https://axjbrpiw8j.execute-api.us-east-1.amazonaws.com/Prod/todos/f13b1ffc-def9-11ef-85bc-194aab275ac1 \ -H "Content-Type: application/json" \ -d '{"text": "Realizar el CP1.3 este ejemplo update", "checked": true, "createdAt": "1738235054.718066", "updatedAt": "1738235054.718066"}'
```

```
voclabs:~ $ curl -X PUT https://axjbrpiw8j.execute-api.us-east-1.amazonaws.com/Prod/todos/f13b1ffc-def9-11ef-85bc-194aab275ac1 \ -H "Content-Type: application/json" \ -d '{"text": "Realizar el CP1.3 este ejemplo update", "checked": true, "createdAt": "1738235054.718066", "updatedAt": "1738235054.718066"}'
{"checked": true, "createdAt": "1738235054.718066", "text": "Realizar el CP1.3 este ejemplo update", "id": "f13b1ffc-def9-11ef-85bc-194aab275ac1", "updatedAt": "1738235054.718066"}voclabs:~ $
```

- Borrado del segundo To-Do

```
voclabs:~/todo-list-aws (master) $ curl -X DELETE https://axjbrpiw8j.execute-api.us-east-1.amazonaws.com/Prod/todos/6401baae-defa-11ef-aeff-194aab275ac1
```

```
voclabs:~/todo-list-aws (master) $ curl https://axjbrpiw8j.execute-api.us-east-1.amazonaws.com/Prod/todos/6401baae-defa-11ef-aeff-194aab275ac1
{"checked": false, "createdAt": "1738235054.718066", "text": "Realizar el CP1.1", "id": "f13b1ffc-def9-11ef-85bc-194aab275ac1", "updatedAt": "1738235054.718066"}voclabs:~/todo-list-aws (master) $ curl https://axjbrpiw8j.execute-api.us-east-1.amazonaws.com/Prod/todos/6401baae-defa-11ef-aeff-194aab275ac1
{"checked": false, "createdAt": "1738235054.718066", "text": "Realizar el CP1.1", "id": "f13b1ffc-def9-11ef-85bc-194aab275ac1", "updatedAt": "1738235054.718066"}voclabs:~/todo-list-aws (master) $
```