



Universidad  
**Tecmilenio**®





# Especialización DevOps

GitHub Actions CI/CD

Semana 4





# Introducción

GitHub Actions es una herramienta muy popular, ya que facilita la automatización, los flujos de trabajo para despliegues, pruebas y los cambios en casi cualquier plataforma. Este proceso se realiza por medio de un solo archivo YAML (GitHub, s.f.-a).



## GitHub Actions

Crea automatización de eventos, se basa en las preferencias del usuario que realiza de manera autónoma las pruebas, despliegues y ejecuciones continuas (GitHub, s.f.-a).

- Desarrollo en Git Hub.
- Variedad de plantillas CI.
- Pruebas de contenedores múltiples.

### Componentes de Git Hub

Paso (step).

Trabajo (job).

Flujo de trabajo  
(workflow).

Evento (event).

Acciones (actions).

Runner.



## Flujos de trabajo

Se almacenan como archivos YAML por separado en el repositorio de código en `.github/workflows`.

Se crea un archivo llamado `learn-github-actions.yml` y se agrega código.

Las características que considera GitHub Actions cuando se quiere personalizar un flujo de trabajo (GitHub, s.f.-b):

Variables.  
Scripts.

Compartir datos entre trabajos.

```
1 name: pruebas-github-actions
2 on: [push]
3 jobs:
4   check-bats-version:
5     runs-on: ubuntu-latest
6     steps:
7       - uses: actions/checkout@v2
8       - uses: actions/setup-node@v2
9         with:
10          node-version: '14'
11       - run: npm install -g bats
12       - run: bats -v
13
```

## Administrar flujos de trabajo

- Almacenar secretos.
- Crear jobs independientes.
- Usar matriz de compilaciones.
- Almacenar dependencias de caché.
- Usar bases de datos y contenedores.
- Utilizar etiquetas para enrutar flujos de trabajo.
- Reutilizar flujos de trabajo.
- Utilizar ambientes.
- Utilizar flujos de trabajo iniciales.

## Compartir flujos de trabajo

- Reutilizar flujos de trabajo.
- Utilizar flujos de trabajo iniciales.
- Compartir secretos dentro de una organización.
- Compartir ejecutores autohospedados dentro de una organización.

## Creación de un flujo de trabajo inicial manual

- No debe haber flujo de trabajo previamente iniciado.
- Solo se pueden integrar en repositorios públicos.

GitHub tiene plantillas, las cuales ayudan a iniciar un proyecto y posteriormente poder editarlo; esto ayuda a optimizar el tiempo (Heller, 2021).

### Eventos que desencadenan flujos de trabajo

Generan la configuración de prueba, ejecución o despliegue de manera controlada. Esto hace que inicien cuando un evento sea lanzado.

Se puede configurar para programarlo con comandos de ejecución.

Branch\_protection\_rule

Check\_run

Check\_suite

Create

Delete

Deployment

```
1 name: Tecmilenio
2
3 on:
4   push:
5     branches: [ $default-branch ]
6   pull_request:
7     branches: [ $default-branch ]
8
9 jobs:
10  build:
11    runs-on: ubuntu-latest
12
13    steps:
14      - uses: actions/checkout@v2
15
16      - name: Run a one-line script
17        run: echo Hola mundo desde Tecmilenio
```

## Implementación de canalización CI/CD

Es una práctica de desarrollo que acelera el proceso de publicaciones de funciones sin comprometer la calidad.



La implementación ocurre cuando un cambio recién integrado se implementa automáticamente en el entorno.

```
1 on:
2   push:
3     branches:
4       - main
5 jobs:
6   deploy:
7     runs-on: ubuntu-latest
8     steps:
9       - uses: actions/checkout@v2
10      - uses: actions/setup-python@v2
11      - uses: aws-actions/setup-sam@v1
12      - uses: aws-actions/configure-aws-credentials@v1
13      with:
14        aws-access-key-id: ${ secrets.AWS_ACCESS_KEY_ID }
15        aws-secret-access-key: ${ secrets.AWS_SECRET_ACCESS_KEY }
16        aws-region: us-east-2
17      - run: sam build --use-container
18      - run: sam deploy --no-confirm-changeset --no-fail-on-empty-changeset
```



GitHub permite administrar tareas al crear una automatización de los flujos de trabajo, algo que lo convierte en una de las herramientas más utilizadas para las tareas de DevOps.

Esto es gracias a todas las opciones que ofrece GitHub Actions, como contar con el repositorio más grande para el control de versiones GitHub.



Para esta actividad, será necesario contar con una cuenta de GitHub. En caso de no tener una, deberás crearla en el sitio oficial de GitHub. Ya con una cuenta, inicia sesión y realiza las siguientes instrucciones:

1. Crea un nuevo repositorio y llámalo "misgitactions".
2. Crea un nuevo flujo de trabajo (workflow) bajo la ruta: `misgitactions/.github/workflows/pruebagithubactions.yml`.
3. Coloca el siguiente contenido en el archivo recién creado (recuerda respetar la estructura yaml).

```
1 name: GitHub Actions Practica
2 on: [push]
3 jobs:
4   Explorando-GitHub-Actions:
5     runs-on: ubuntu-latest
6     steps:
7       - run: echo "El flujo de trabajo ha sido ejecutado en el evento: ${github.event_name}"
8       - run: echo "El trabajo se esta ejecutando en un servidor ${runner.os} hospedado en GitHub"
9       - run: echo "El nombre de la rama es: ${github.ref} y tu repositorio es: ${github.repository}."
10      - name: Verifica el codigo del repositorio
11        uses: actions/checkout@v2
12      - run: echo "El repositorio: ${github.repository} ha sido colonado al runner."
13      - run: echo " El flujo de trabajo esta listo para para verificar el código en el runner."
14      - name: Listar archivos en el repositorio
15        run: |
16          ls ${github.workspace}
17      - run: echo "El estado del trabajo es: ${job.status}."
```

4. Crea una nueva rama, haz clic en el botón commit e inicia una solicitud pull (para crearla, debes hacer clic en el botón Propose new file).
5. Busca los resultados del ciclo de trabajo. Para ello, visita la página principal del repositorio en la página oficial de GitHub, luego dirígete a la pestaña "Actions", después, a la izquierda, encontrarás el panel de "Workflows" con el conjunto de todos los trabajos. Haz clic en el trabajo llamado "Explorando-GitHub-Actions".
6. Analiza los registros del Log, documenta el proceso y concluye tus resultados.
7. Plasma tus evidencias (reporte del proceso), resultados y conclusiones en un archivo en formato PDF.

# Bibliografía



- GitHub. (s.f.-a). *Entender las GitHub Actions*. Recuperado de <https://docs.github.com/es/actions/learn-github-actions/understanding-github-actions>
- GitHub. (s.f.-b). *Características esenciales de las GitHub Actions*. Recuperado de <https://docs.github.com/es/actions/learn-github-actions/essential-features-of-github-actions>
- Heller, P. (2021). *Automating WorkFlows with GitHub Actions*. Reino Unido: Packt Publishing.



# Especialización DevOps

Esenciales de AWS. Parte 1

Semana 4



En los últimos años, AWS se ha convertido en una de las plataformas de microservicios más grandes e importantes, ya que los servicios web que contiene son rápidos, exponenciales y es el más utilizado a nivel mundial.

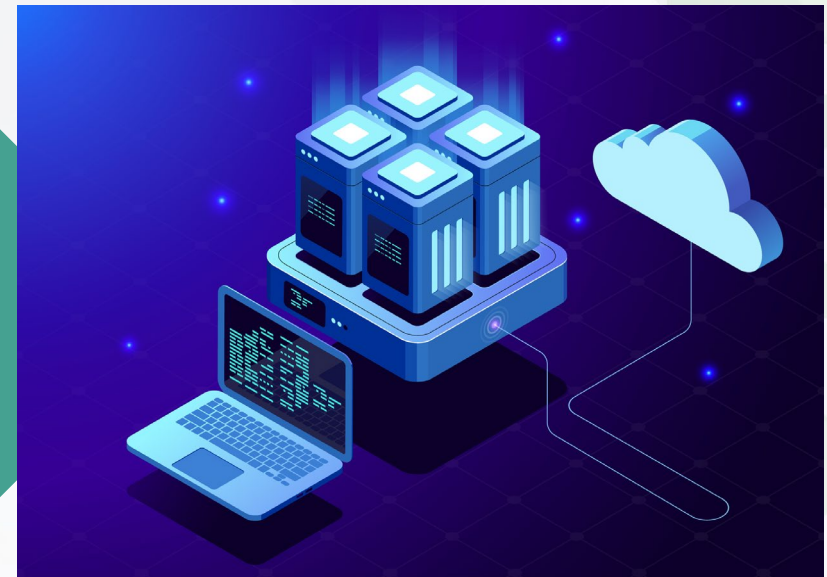


## AWS

Resulta sencillo aprender AWS, ya que se pueden crear aplicaciones con cualquier lenguaje de programación (Amazon Web Services, s.f.-a).

Cinco pilares:

1. Seguridad.
2. Eficacia del rendimiento.
3. Fiabilidad.
4. Excelencia operativa.
5. Optimización de costos.



AWS cuenta con una plataforma para poder navegar a otras consolas de servicios individuales.

Basado en un navegador que puede iniciar directamente desde la consola de administración AWS, Cloudshell puede ejecutar comandos de AWS CLI en servicios utilizando ShellBash y PowerShell o Z.



Los cinco pilares que ofrece AWS son para el diseño y operación de los sistemas de la nube para que sean de bajo costo, seguros y eficientes.

## Administración de accesos en AWS



Para facilitar esta acción, existe la tecnología para la administración de accesos de identidades (IAM), Amazon Web Services, (s.f.-b).

Registra y administra a los usuarios y los permisos de acceso automáticamente, considerando los privilegios que se conceden según los roles.

1. Principios.
2. Acciones.
3. Recursos.

Principio de mínimo privilegio con Zero Trust.



## Políticas en AWS

### Políticas basadas en recursos

Otorgan permisos para realizar acciones específicas en algún recurso y se define bajo qué condiciones se aplican.

Para crear el usuario, se debe seleccionar el medio y después se establecen los permisos mediante grupos creados manualmente o de forma individual.

### Políticas basadas en identidades

Documentos de políticas de permisos JSON que controlan las acciones que puede realizar una identidad (usuarios, grupos de usuarios y roles), en qué recursos y en qué condiciones.

Políticas administradas.

Políticas integradas.

## Ejercicio

1. Investiga en fuentes confiables sobre los aspectos fundamentales de la plataforma AWS. Puedes utilizar la lista de *whitepapers* de AWS en:

Los siguientes enlaces son externos a la Universidad Tecmilenio, al acceder a ellos considera que debes apegarte a sus términos y condiciones.

<https://d0.awsstatic.com/whitepapers/aws-overview.pdf>

2. Define los siguientes conceptos con base en las fuentes que consultaste:

La infraestructura básica global de AWS.

Los principios de arquitectura de AWS.

Los servicios clave y los casos de uso más comunes.

Los aspectos de seguridad básica que integra AWS.

AWS ofrece mucho como plataforma de microservicios que ayudan a la infraestructura para el cumplimiento de actividades dentro de los procesos de programación de una aplicación y estos microservicios optimizan el tiempo, los costos y mejoran la seguridad para generar proyectos robustos en poco tiempo.



Cierre



# Bibliografía



- Amazon Web Services. (s.f.-a). *ASPECTOS FUNDAMENTALES DE AWS*. Recuperado de <https://aws.amazon.com/es/getting-started/fundamentals-core-concepts/?e=gs2020&p=fundoverview>
- Amazon Web Services. (s.f.-b). *AWS Identity and Access Management (IAM)*. Recuperado de <https://aws.amazon.com/es/iam/>



# Curso DevOps autodirigido

Esenciales de AWS. Parte 2

Semana 4



Con los avances tan rápidos que hay en el mundo digital, AWS se encargó de crear un servicio para almacenar servidores virtuales, donde el cliente puede configurar la seguridad, la red y la administración del almacenamiento.

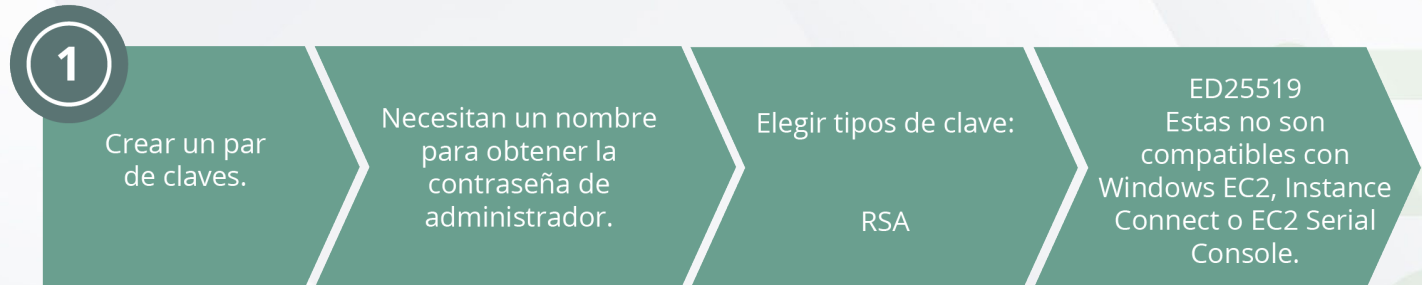
Completamente en la nube, Amazon Elastic Compute Cloud (EC2) elimina la necesidad de invertir en hardware, hace más rápidos los procesos administrativos y da la opción de crear tantos servidores como sean posibles. Lo mejor de todo es que es escalable y disminuye la necesidad de proyectar el tráfico.



## Amazon EC2

- → Entornos informáticos virtuales.
- → Amazon Machine Images (AMI).
- → Configuraciones de CPU, memoria, almacenamiento y capacidad de red.
- → Información de inicio de sesión.
- → Volúmenes de almacenamiento para datos temporales.
- → Volúmenes de almacenamiento persistentes de datos (Amazon EBS).
- → Cortafuegos que especifican los rangos de las direcciones IP de origen.
- → Direcciones IPV4 estáticas para computación en la nube dinámica.
- → Metadatos que crean y asignan etiquetas a los recursos Amazon EC2.
- → Redes virtuales con red propia y aisladas (VPC).

## Para lanzar una instancia



Instancias (en ejecución)	0	Balanceadores de carga	0
Direcciones IP elásticas	0	Grupos de seguridad	1
Grupos de ubicación	0	Hosts dedicados	0
Instancias	0	Instantáneas	0
Pares de claves	0	Volúmenes	0



2

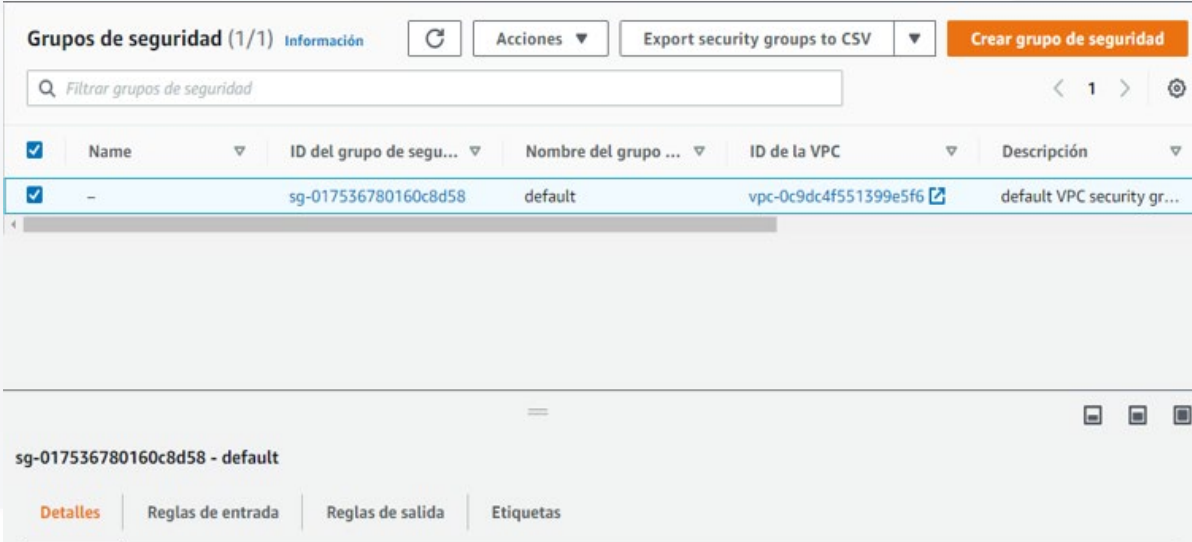
Crear grupo de seguridad.

Sirve de cortafuegos para las instancias asociadas y controlar el tráfico

**Reglas de entrada**  
 Crear reglas que permitan que un tráfico específico llegue a la instancia. Por ejemplo, puedes utilizar reglas para que un servidor web acepte tráfico HTTP y HTTPS.

**Reglas de salida**  
 1. Mantener la regla predeterminada que permite todo el tráfico.  
 2. Seleccionar grupo de seguridad.

Explicación



Grupos de seguridad (1/1) Información 🔄 Acciones ▼ Export security groups to CSV ▼ Crear grupo de seguridad

<input checked="" type="checkbox"/>	Name	ID del grupo de segu...	Nombre del grupo ...	ID de la VPC	Descripción
<input checked="" type="checkbox"/>	-	sg-017536780160c8d58	default	vpc-0c9dc4f551399e5f6 <a href="#">🔗</a>	default VPC security gr...

---

sg-017536780160c8d58 - default

Detalles | Reglas de entrada | Reglas de salida | Etiquetas

3

Iniciar instancia de Windows por medio de la consola de AWS (Amazon Web Services, s.f.-b).

1. Lanzar instancias.
2. Seleccionar una imagen de máquina de Amazon (AMI).
3. Seleccionar el tipo de instancia.
4. Seleccionar, revisar y lanzar.
5. Seleccionar la casilla de verificación de reconocimiento.

### Conexión

1. En el panel EC2, seleccionar la instancia y luego conectar donde se conecta el archivo .pem.
2. Se descifra la contraseña.
3. Se descarga el archivo en el escritorio remoto.
4. Seleccionar Ver certificado (en caso de que sea escritorio remoto).
5. En la consola Amazon EC2 se obtiene el registro del sistema.
6. En el registro se busca RDPCERTIFICATE-THUMBPRINT. Si este coincide con la huella digital del certificado, todo está correcto.
7. En el cuadro de diálogo del certificado, seleccionar Aceptar.

## Balanceo de cargas, elasticidad y escalabilidad

### Balancedador de cargas elástico

El balanceador elástico distribuye automáticamente el tráfico.

Enruta y supervisa el estado de los objetivos registrados que estén en buen estado.

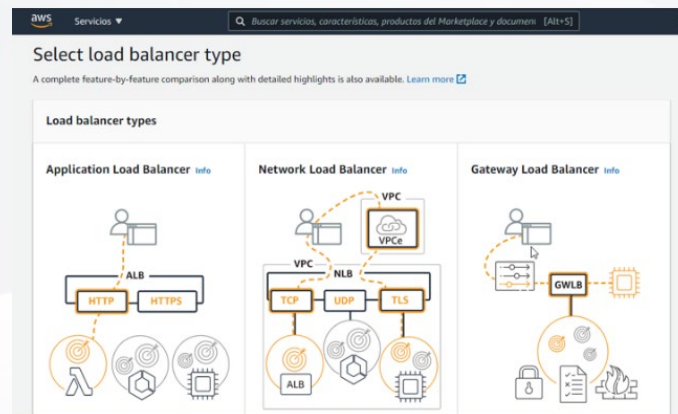
Escala la carga a medida que su tráfico entrante cambia con el tiempo.

Admite balanceadores de aplicación, red de puerta de enlace y clásicos (Amazon Web Services, s.f.-a).

### Balancedador de carga clásico

Se encarga del balanceo básico de las instancias y funciona en los niveles de conexión y solicitud. Este balanceador está diseñado para las aplicaciones creadas en el entorno EC2 clásico.

1. El cliente realiza una solicitud de aplicación.
2. Los receptores del balanceador de carga reciben la solicitud y el puerto que se configuró.
3. El receptor envía solicitudes a instancias registradas que estén en buen estado.



## Ejercicio

Crea una instancia utilizando Amazon EC2. Puedes lanzar la instancia usando el asistente de lanzamiento de instancias o usando el procedimiento manual.

No olvides que para lanzar la instancia primero se tiene que configurar el grupo de seguridad y el par de claves.

Realiza los siguientes pasos:

- a. Haz un inicio de lanzamiento de la instancia.
- b. Elige una Amazon Machine Image (AMI).5. Elige un tipo de instancia.
- c. Configura los detalles de la instancia.
- d. Agrega almacenamiento.
- e. Añade etiquetas.
- f. Configura un grupo de seguridad.
- g. Revisa el lanzamiento de la instancia y selecciona el par de claves.

AWS es una plataforma con múltiples servicios que ayudan a crear una infraestructura sólida para cumplir con todas las actividades ya mencionadas y, de esta manera, implementar servicios tecnológicos en cuestión de minutos y que las empresas puedan marcar una diferencia en las experiencias de los clientes y mejoren la transformación de los negocios.



# Bibliografía



- Amazon Web Services. (s.f.-a). *Introducción a Elastic Load Balancing*. Recuperado de <https://aws.amazon.com/es/elasticloadbalancing/getting-started/?nc=sn&loc=4>
- Amazon Web Services. (s.f.-b). *Amazon Elastic Compute Cloud Documentation*. Recuperado de <https://docs.aws.amazon.com/ec2/index.html>



# Especialización DevOps

Esenciales de AWS. Parte 3

Semana 4



AWS no es más que una plataforma de infraestructura como servicio (IaaS) que se enfoca en el desarrollo de aplicaciones, cumpliendo con servicios de seguridad y confidencialidad. También comprende servicios como plataforma (PaaS) y software como servicio (SaaS) para eliminar la infraestructura de hardware por medio de microservicios.





## Creación de una red

Para las aplicaciones que se ejecutan en EC2, la arquitectura comienza en la red y en AWS se refiere a la VPC (*virtual private cloud*), (Amazon Web Services, s.f.-a).

Lanza recursos de AWS en una red virtual que se defina. Esta red virtual se parece mucho a una red tradicional que operaría en su propio centro de datos.

- Nube privada virtual (VPC).
- Subred.
- Tabla de rutas.
- Puerta de enlace de Internet.
- Punto de enlace de VPC.

Explicación



## Múltiples direcciones IP de la VPC

### Subred pública

Conectada a Internet.

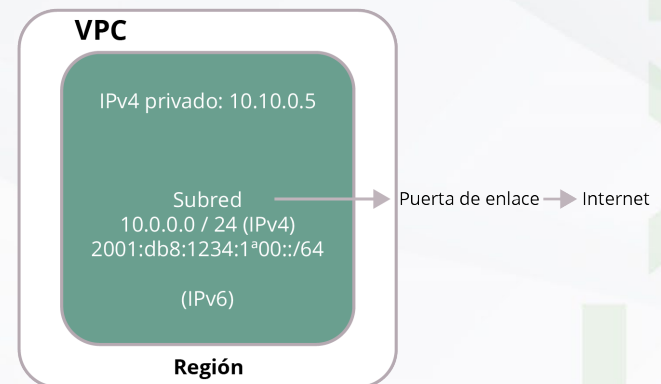
Se especifica IPv4.

### Subred privada

No conectada a Internet.

Se especifica IPv4 en forma de bloque

Cuando se trabaja con instancias EC2, llega un momento donde aumenta la cantidad y es mejor utilizar un bastión.



## Grupos de seguridad

Mantiene la seguridad y se aseguran de que el host de bastión no falle.

### Puerta de enlace NAT

Se utiliza para que las instancias de una subred privada se conecten a los servicios que estén fuera de la nube virtual privada.

Pública.

Privada.

### AWS Network Firewall

Es un servicio de cortafuegos de red que puedes administrar para prevenir y detectar ataques o intrusiones en el perímetro de la VPC.

Cortafuegos.

Políticas de cortafuegos.

Grupo de reglas.

### Amazon Route 53

Es un servicio de DNS (*domain name system*) que se utiliza para redirigir a los usuarios finales a los sitios de Internet mediante la traducción de nombres.

## Servicios de almacenamiento

Amazon Simple Storage Service (Amazon S3)

Seguridad

Rendimiento

Disponibilidad

Escalabilidad

### Tiene varias clases de almacenamiento:

S3 Estándar.

S3 Capas inteligentes.

S3 Estándar-zona única/acceso poco frecuente.

S3 Glacier/Deep archive.

S3 Outposts.



## Servicios de base de datos

Existen cerca de 15 motores de personalización para las bases de datos, incluyendo las bases de datos relacionales, gráficos, de documentos, entre otros (Amazon Web Services, s.f.-b).

Los más importantes son los siguientes:

- Amazon RDS.
- Amazon Aurora.
- Amazon DynamoDB.
- Amazon DynamoDB Accelerator.
- Amazon ElastiCache.



## Ejercicio

En esta actividad crearás una cubeta de Amazon S3 y cargarás un archivo para después recuperarlo y, por último, eliminarlo.

Los siguientes pasos se tienen que explicar y evidenciar con imágenes:

1. Acceso a la consola de Amazon AWS.
2. Creación de una cubeta en Amazon S3.
3. Carga de un archivo.
4. Recuperación del objeto.
5. Eliminación del objeto y de la cubeta.

La nube y la automatización pueden crear infraestructuras tecnológicas en muy poco tiempo. Las tareas pasaron a enfocarse en las áreas de innovación con estrategias flexibles e implementaciones más ágiles y rápidas. AWS es prueba de eso y lo demuestra con los servicios web que ofrece; todos a la mano y con la capacidad de escalar en el desarrollo de los proyectos.



# Bibliografía



- Amazon Web Services. (s.f.-a). *Clases de almacenamiento de Amazon S3*. Recuperado de <https://aws.amazon.com/es/s3/storage-classes/?nc=sn&loc=3>
- Amazon Web Services. (s.f.-b). *What is Amazon VPC?* Recuperado de <https://docs.aws.amazon.com/vpc/latest/userguide/what-is-amazon-vpc.html>