

LABORATORY NO.05 - Cloud



---

UNIVERSIDAD

ELABORADO POR:

ESTEBAN AGUILERA CONTRERAS  
JUAN DAVID RODRIGUEZ RODRIGUEZ

PROFESOR(ES):

JOHN PACHON

RECO  
2025-1

## **OBJECTIVE**

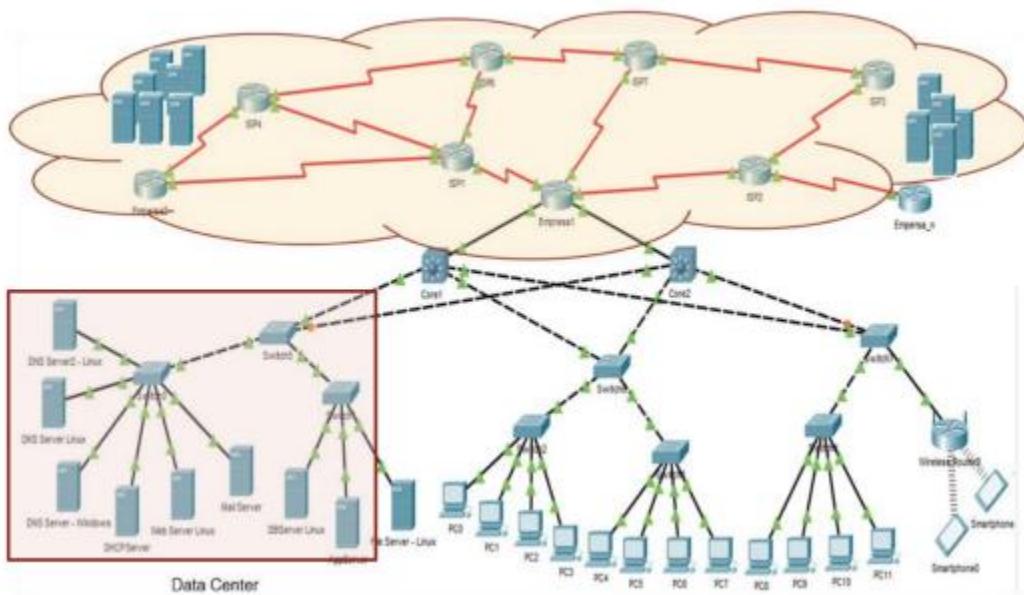
Install and configure base software - Web Servers

## **TOOLS TO BE USED**

- Computers
- Internet access

## CONTEXT

We are continuing to work within a company's infrastructure, which typically includes various IT services. This infrastructure features both wired and wireless user workstations and servers (both physical and virtualized), all connected through switches (Layer 2 and Layer 3), wireless equipment, and routers connecting them to the Internet. Cloud infrastructures are also common, provisioning resources as needed by the organization. The servers may host services such as web, DNS, email, databases, storage, and applications, among others.



In this part of the lab, we will focus on continuing to set up our servers.

## **INTRODUCTION**

En este laboratorio se configuró y desplegó un entorno de servidores en la nube utilizando AWS. Se creó una instancia EC2, se diseñó y configuró una VPC con sus respectivas subredes y reglas de seguridad, y se instaló un servidor web que se conectó a una base de datos MySQL alojada en Amazon RDS. El objetivo fue integrar de forma práctica los conceptos de infraestructura en la nube, demostrando cómo desplegar servicios web y gestionar recursos de forma flexible y escalable.

## THEORETICAL FRAMEWORK

- Amazon EC2 y VPC
  - Amazon EC2 es un servicio que permite ejecutar servidores virtuales en la nube, ofreciendo flexibilidad en recursos y configuraciones. La Virtual Private Cloud (VPC) es una red virtual aislada dentro de AWS que permite definir rangos de direcciones IP, subredes y reglas de acceso, garantizando la seguridad y el control del tráfico.
- Security Groups y Reglas de Acceso
  - Los Security Groups actúan como cortafuegos virtuales que controlan el tráfico entrante y saliente de las instancias. Se configuran para permitir solo el acceso necesario (por ejemplo, puertos HTTP/HTTPS para servidores web y restringir SSH a IPs específicas) y así proteger la infraestructura.
- Amazon RDS
  - RDS es un servicio gestionado de bases de datos relacionales que facilita el despliegue, escalado y administración de bases de datos como MySQL. Permite concentrarse en el desarrollo y funcionamiento de la aplicación sin preocuparse por la gestión de la infraestructura subyacente.
- Integración de Servicios
  - La integración entre la instancia EC2 y RDS demuestra cómo se pueden conectar diferentes servicios en la nube para crear aplicaciones completas. En este laboratorio se utilizó un servidor web que accede a datos almacenados en una base de datos, ejemplificando la comunicación entre componentes distribuidos en una infraestructura en la nube.

## RESEARCH

a. What is an EC2 instance and what is it used for?

- Es un servidor virtual en la nube que se utiliza para ejecutar aplicaciones. Ofrece flexibilidad en la elección del hardware, sistema operativo, almacenamiento y red, permitiendo nos a nosotros adaptar los recursos a nuestras necesidades. Se usa para alojar sitios web, aplicaciones, bases de datos, análisis de datos y más.

b. What is a VPC, how should it be configured, and what best practices should be considered?

- Una VPC (Virtual Private Cloud) es una red virtual aislada dentro de la nube de AWS donde se pueden desplegar instancias EC2 y otros recursos.

Para configurarla:

- Se debe definir un rango de direcciones IP.
- Crear subredes en diferentes zonas de disponibilidad.
- Configurar tablas de rutas y puertas de enlace para permitir o restringir el tráfico.
- Establecer grupos de seguridad y listas de control de acceso (ACLs) para la seguridad de la red.

c. How can I run multiple systems within an Amazon EC2 environment?

- Múltiples instancias EC2, cada una ejecutando un sistema diferente.
- Instancias con contenedores, utilizando Docker o Kubernetes.
- Uso de Elastic Load Balancer (ELB) para distribuir tráfico entre múltiples instancias.
- Autoscaling Groups para ajustar automáticamente la cantidad de instancias según la demanda.

d. How quickly can I scale the capacity (both up and down) of an EC2 instance?

- Vertical scaling (cambiar el tamaño de la instancia) requiere detenerla y cambiar el tipo de instancia, lo cual toma pocos minutos.
- Horizontal scaling (añadir o quitar instancias) se puede hacer automáticamente con Auto Scaling Groups, agregando instancias en segundos o minutos según la demanda.

e. How does this service differ from standard hosting services?

- Escalabilidad flexible, permitiendo ajustar recursos según la demanda.
- Personalización total, eligiendo sistema operativo, almacenamiento y red.
- Pago por uso, sin necesidad de pagar una tarifa fija mensual.
- Alta disponibilidad, ya que permite distribuir cargas en múltiples regiones y zonas de disponibilidad.

f. What is Amazon RDS?

- Amazon RDS (Relational Database Service) es un servicio administrado de bases de datos relacionales en AWS. Permite desplegar, escalar y administrar bases de datos como MySQL, PostgreSQL, SQL Server, Oracle y Amazon Aurora sin preocuparse por la infraestructura.

## CONFIGURATION

1. Log in to the AWS Management Console at [awsacademy.instructure.com](https://awsacademy.instructure.com) and locate Lab
  - a. Creamos nuestra cuenta en AWS Management Console con el correo de la escuela y una contraseña asignada por nosotros

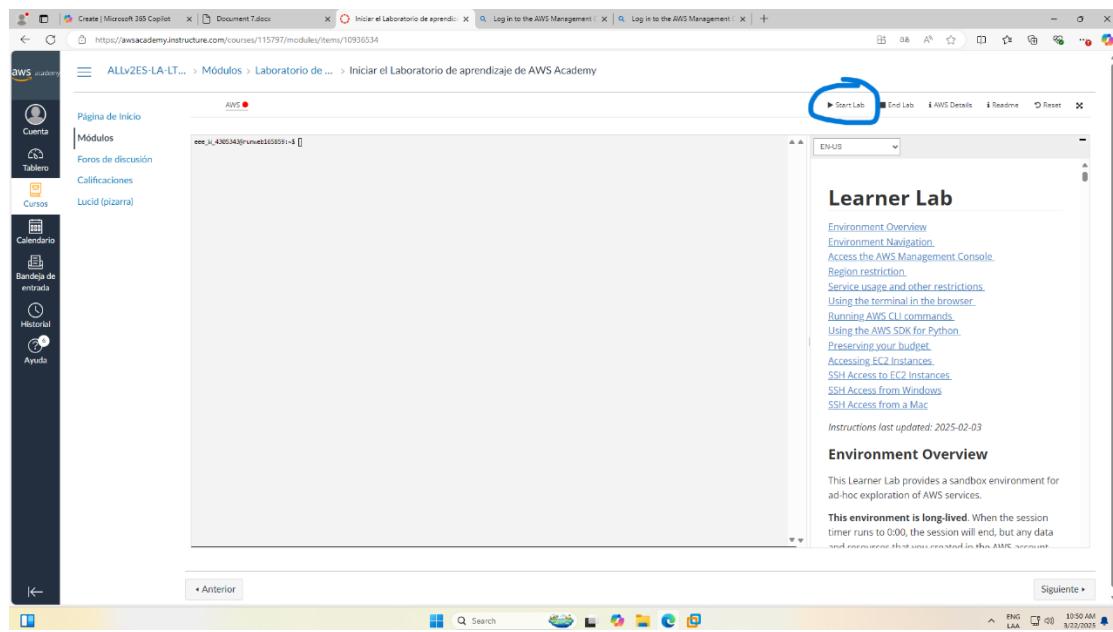
The screenshot shows the AWS Academy dashboard with a sidebar on the left containing links like 'Cuenta', 'Tablero', 'Cursos', 'Calendario', 'Bandeja de entrada', 'Historial', and 'Ayuda'. The main area displays a list of notifications under 'Notificaciones'. One notification is expanded, showing 'información general sobre el curso' and several sub-items related to AWS Academy learning, such as 'e aprendizaje de AWS Academy: cumplimiento y seguridad', 'e aprendizaje de AWS Academy', 'Laboratorio de aprendizaje de AWS Academy', and 'Los laboratorios de aprendizaje de AWS Academy'. A 'Complete todos los ítems' button is visible.

2. Navigate to the Modules section, click on "Learner Lab", accept the terms and conditions, and then click on "Start Lab."

- a. Damos click en Iniciar la laboratorio de aprendizaje de AWS Academy

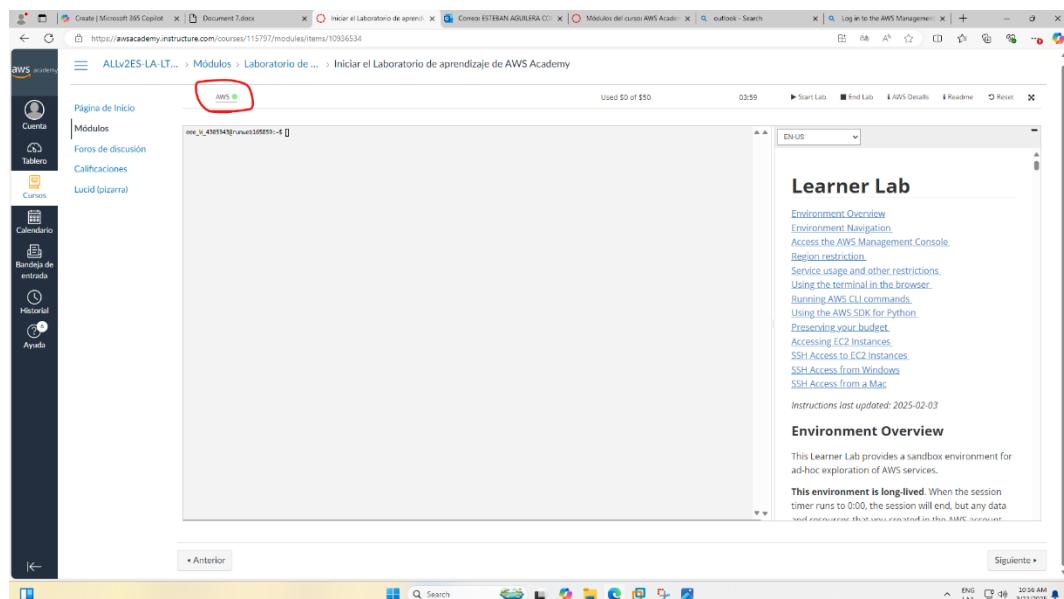
The screenshot shows the 'Módulos' (Modules) page of the AWS Academy course. The sidebar includes 'Página de Inicio', 'Modulos', 'Foros de discusión', 'Calificaciones', and 'Lucid (pizarra)'. The main content area lists course modules: 'Bienvenida e Información general sobre el curso', 'Laboratorio de aprendizaje de AWS Academy: cumplimiento y seguridad', 'Laboratorio de aprendizaje de AWS Academy' (which is circled in red), and 'Iniciar el Laboratorio de aprendizaje de AWS Academy'. Below these are sections for 'Recursos de los laboratorios de aprendizaje de AWS Academy' and 'Demostración: cómo acceder al Laboratorio de aprendizaje'. A 'Complete todos los ítems' button is also present.

- b. Damos click en Start Lab

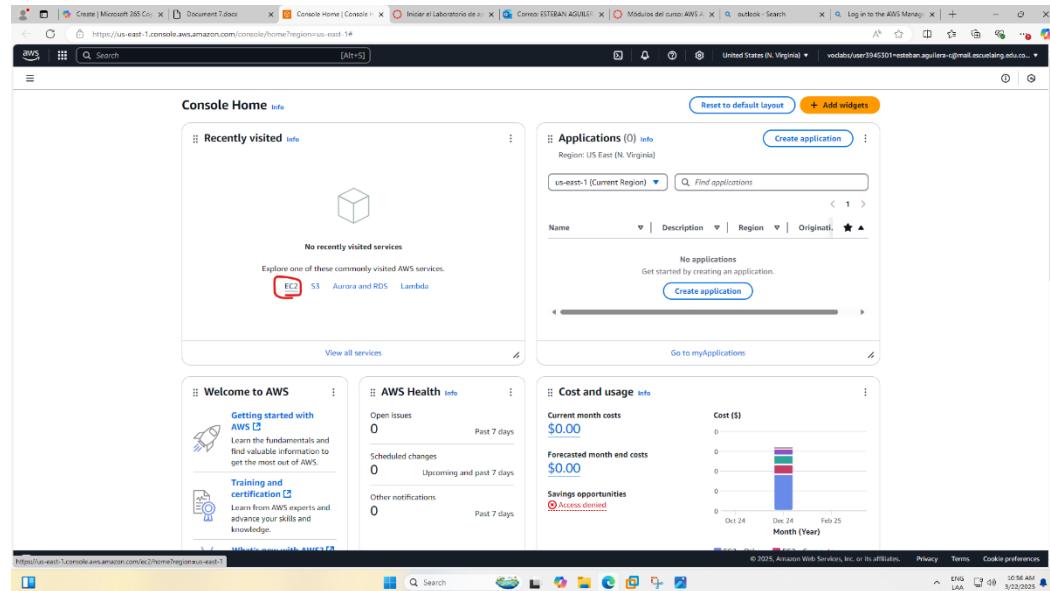


3. Once the lab loads, click the “AWS” button in the upper left corner to be redirected to the AWS Console.

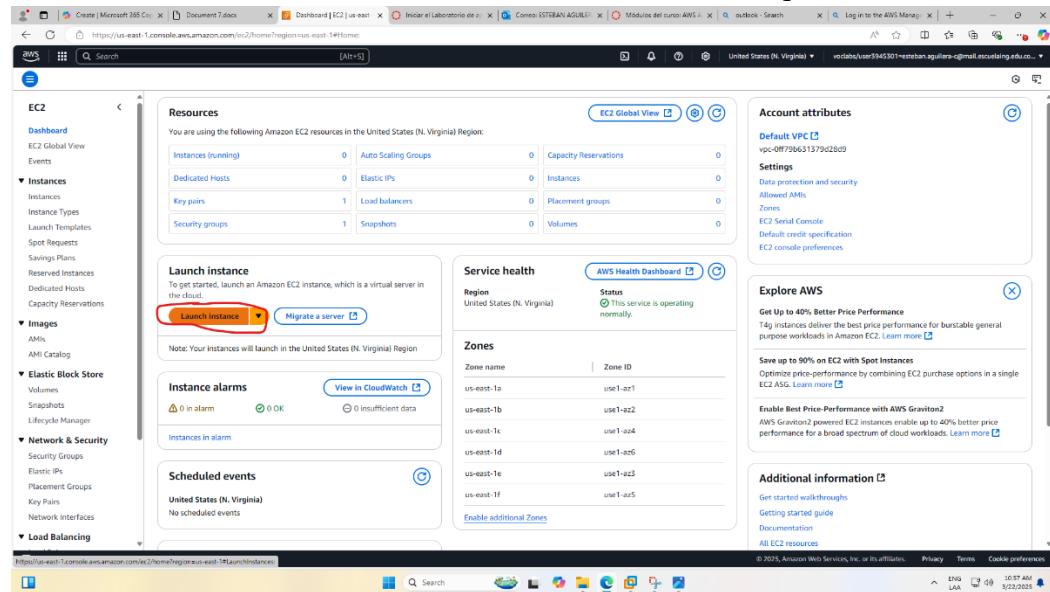
- a. Damos click en AWS cuando este el símbolo en Verde



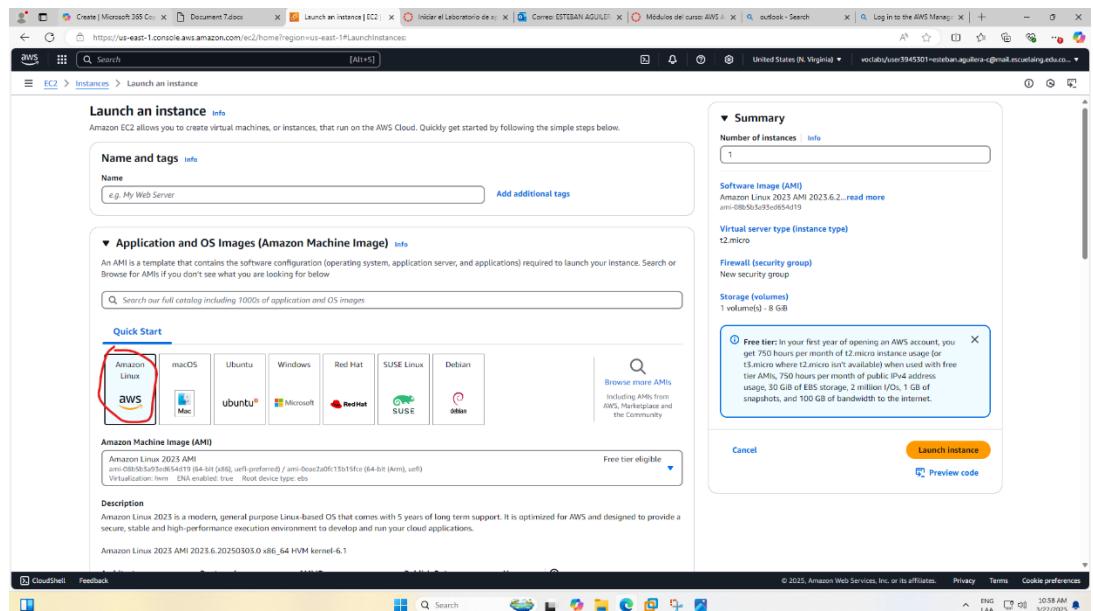
4. Click on EC2.



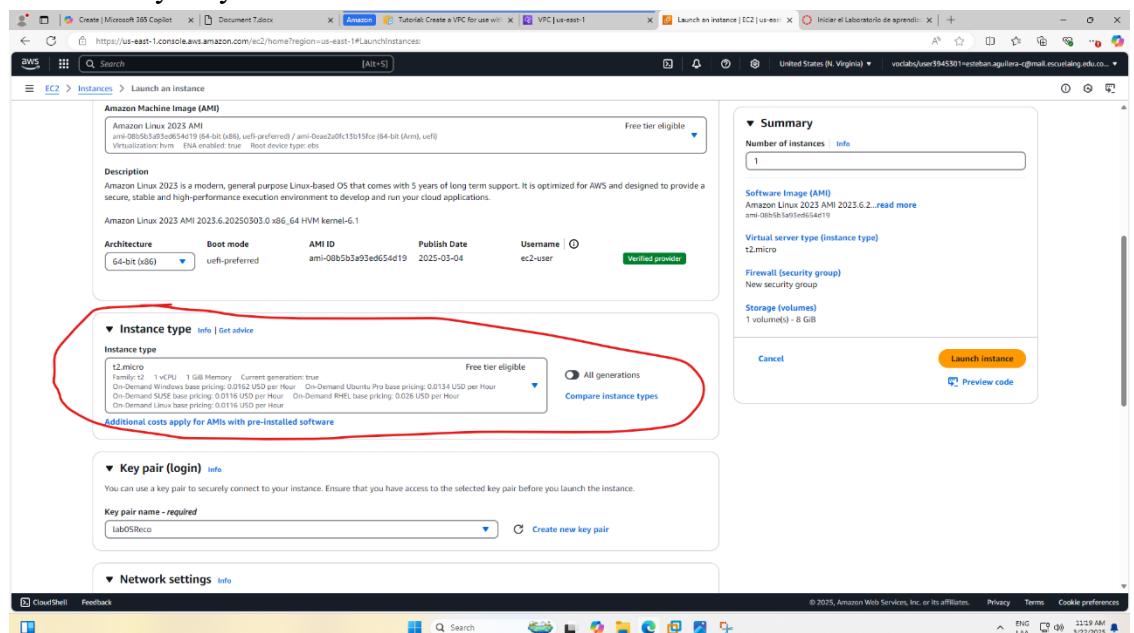
5. Select the EC2 Dashboard and then choose the “Launch Instance” option.



6. For this exercise, select the Amazon Linux 2 AMI.



7. Choose the instance type t2.micro. What are the different instance types available in Amazon EC2? Why do you think we chose t2.micro?



- a. Amazon EC2 ofrece varios tipos de instancias según su propósito:

i. Instancias de Uso General

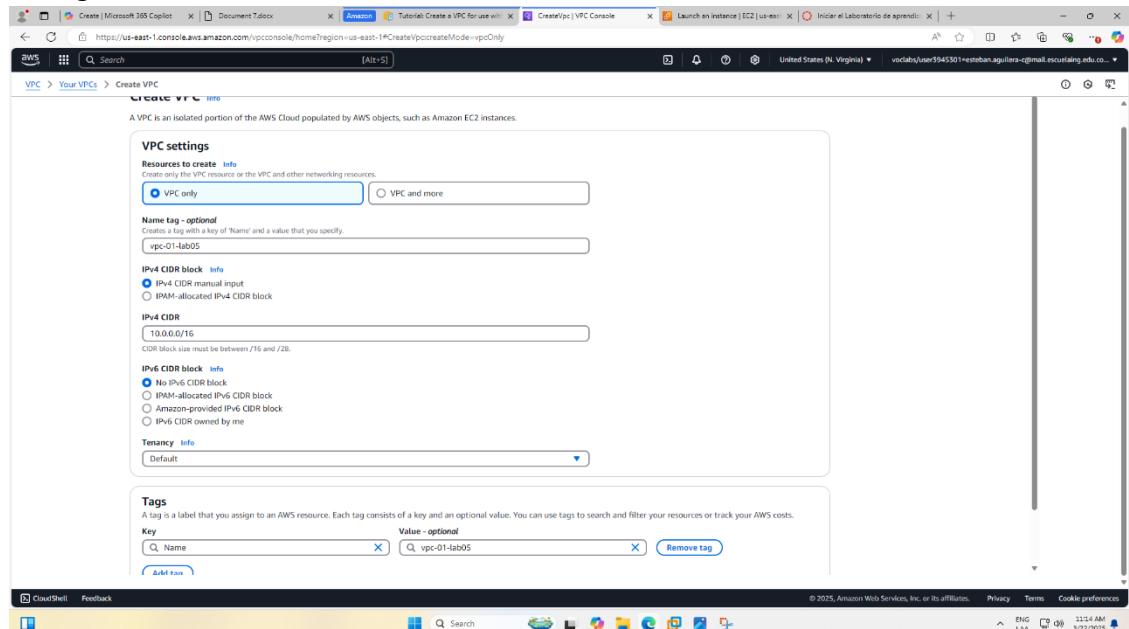
1. Equilibradas en potencia, memoria y red.
2. Ideales para aplicaciones web, desarrollo y pequeñas bases de datos.

ii. Instancias de Alto Cómputo

1. Más capacidad de procesamiento

2. Usadas en análisis de datos y cálculos intensivos.
- iii. Instancias con Mayor Memoria*
1. Diseñadas para aplicaciones que requieren mucha RAM
  2. Perfectas para bases de datos grandes y análisis de big data.
- iv. Instancias de Almacenamiento Optimizado*
1. Diseñadas para manejar grandes volúmenes de datos.
  2. Ofrecen acceso rápido y eficiente a la información.
- v. Instancias con Aceleración de Hardware*
1. Incluyen GPUs y otros procesadores especializados
  2. Utilizadas en IA, gráficos avanzados y simulaciones
- vi. Instancias de Alto Rendimiento*
1. Diseñadas para cálculos científicos y técnicos complejos.
  2. Ofrecen máxima potencia para simulaciones y modelado avanzado.
- b. Se eligió porque opción económica y está incluida en el AWS Free Tier. Su rendimiento es suficiente para aplicaciones ligeras y permite aumentar temporalmente la capacidad si es necesario.
8. Consult the following documentation related to creating a VPC and create the VPC using the addresses provided by your instructor.

- a. Consultando la documentación de como crear una VPC se coloca la siguiente configuración



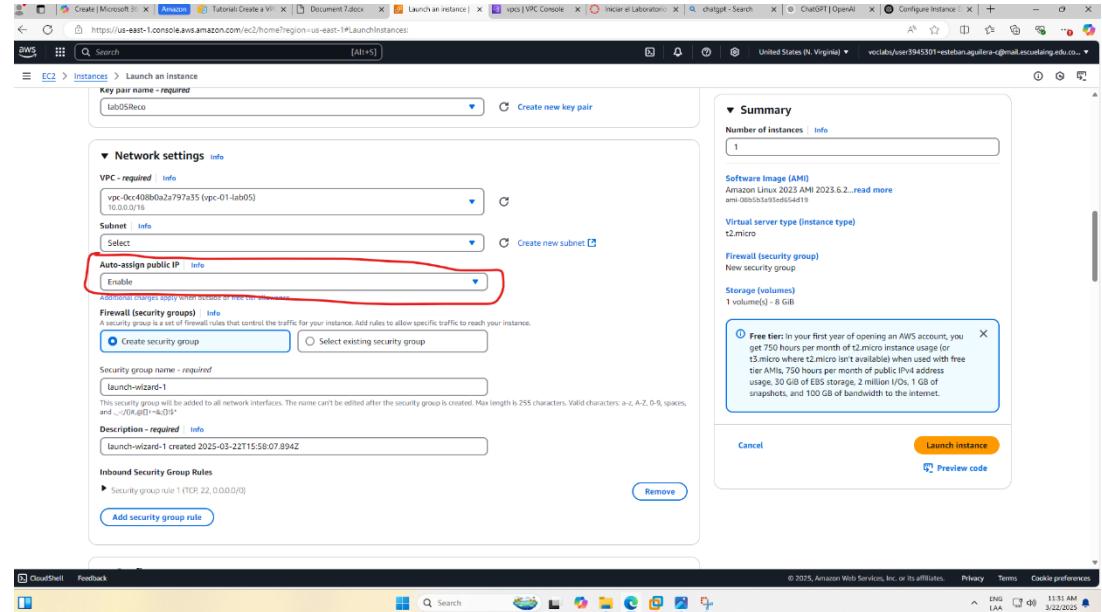
- b. Luego de esto, la VPC quedo creada correctamente

The screenshot shows the AWS VPC console with a success message: "You successfully created vpc-0cc408b0a2a797a35 / vpc-01-lab05". The VPC details pane includes fields for VPC ID, State, DNS resolution, Main network ACL, IPv4 CIDR, and Network Address Usage metrics. The Resource map pane displays the VPC, Subnets (0), Route tables (1), and Network connections (0).

9. On the "Configure Instance Details" page, set the values as specified in the previous step.
- a. Agregamos la VPC creada anteriormente en la instancia que estamos creando

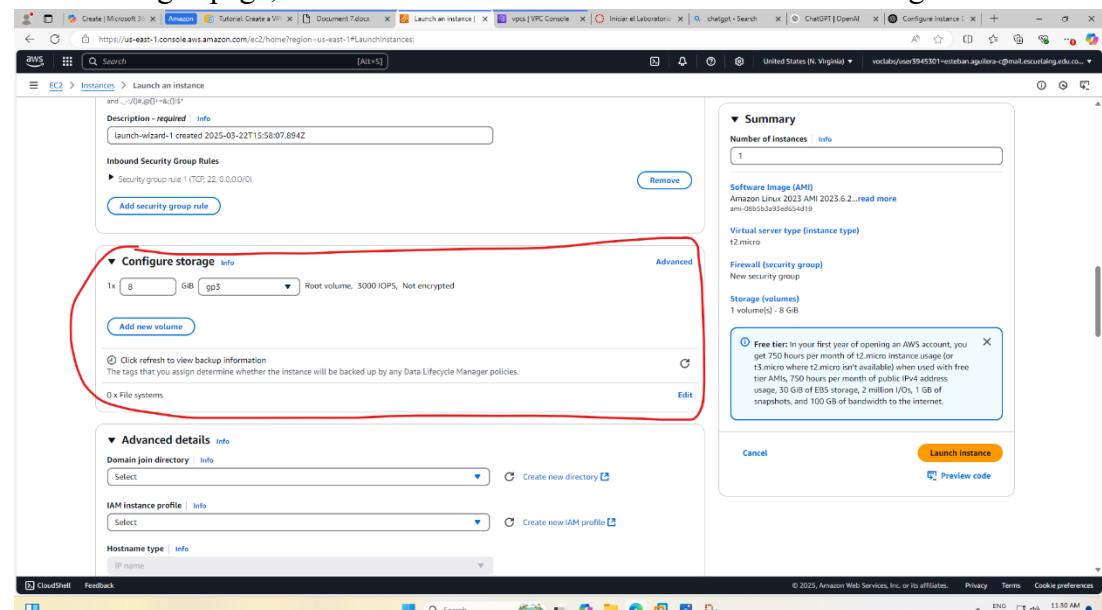
The screenshot shows the AWS EC2 "Launch an instance" configuration page. The "Network settings" section is highlighted with a red box around the "VPC - required" dropdown, which is set to "vpc-0cc408b0a2a797a35 (vpc-01-lab05)". Other settings shown include "Subnet - required", "Auto-assign public IP", "Firewall (security group)", "Security group name - required", and "Description - required". The right side of the screen displays the "Summary" section with 1 instance, software image (Amazon Linux 2023.6.2...), virtual server type (t2.micro), firewall (New security group), storage (1 volume(s) - 8 GB), and a free tier note.

## 10. Under “Auto-assign Public IP,” select Enable.



## 11. Click on “Next: Add Storage.”

## 12. On the ”Add Storage” page, leave the default values and click “Next: Add Tags.”



## 13. On the ”Add Tags” page, click “Add Tag,” then enter Name for the Key and tutorial-web-server for the Value.

- Agregamos una etiqueta con los valores presentados a continuación

### Launch an instance Info

Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.

▼ Name and tags Info

Key <small>Info</small>	Value <small>Info</small>	Resource types <small>Info</small>
<input type="text" value="Name"/> <span>X</span>	<input type="text" value="tutorial-web-server"/> <span>X</span>	Select resource types <span>▼</span>
<span>Instances X</span>		
<span>Add new tag</span>		

You can add up to 49 more tags.

14. On the "Configure Security Group" page, choose "Select an existing security group." What are security groups, and what should be considered when creating a security group for a public web server?

a. Preguntas

- i. Los Security Groups en AWS funcionan como un cortafuegos para las instancias EC2. Básicamente, controlan qué tipo de tráfico puede entrar y salir de un servidor. Se configuran con reglas que permiten o bloquean conexiones según las necesidades del sistema.
- ii. Al configurar un Security Group para un servidor web accesible desde internet, hay que tener en cuenta lo siguiente:
  1. Permitir acceso HTTP/HTTPS
    - a. Ambos deben estar abiertos para todo el mundo, ya que cualquier usuario puede visitar un sitio web.
  2. Restringir el acceso por SSH
    - a. El acceso por SSH (para conectarse al servidor) debe estar limitado a direcciones IP específicas, como la de la oficina o un VPN, para evitar ataques externos.
  3. No dejar puertos innecesarios abiertos
    - a. Solo se deben abrir los puertos estrictamente necesarios.
  4. Configurar las reglas de salida
    - a. Por defecto, AWS permite que el servidor envíe tráfico a cualquier dirección.
  5. Separar Security Groups por funciones
    - a. Es recomendable tener un Security Group para el servidor web y otro para la base de datos o servicios internos, en lugar de permitirlo todo en uno solo.
- b. Vamos al apartado de Security Group

The screenshot shows the AWS VPC Dashboard. On the left, there's a sidebar with sections for Virtual private cloud (Your VPCs, Subnets, Route tables, Internet gateways, Egress-only Internet gateways, Carrier gateways, DHCP option sets, Elastic IPs, Managed prefix lists, NAT gateways, Peering connections), Security (Network ACLs, Security groups), and PrivateLink and Lattice (Getting started, Endpoints, Endpoint services, Service networks, Lattice services, Resource configurations, Resource gateways). The main area displays 'Resources by Region' with links to 'See all regions' for each category. A red box highlights the 'See all regions' link under the 'Security Groups' section.

- c. Agregamos la VPC creada anteriormente , asignamos un nombre , descripción y permitimos todo el trafico en las reglas de entrada (solo para temas del laboratorio se asigna all traffic, pero es mala práctica)

The screenshot shows the 'Create security group' wizard. It starts with the 'Basic details' step, where the security group name is 'tutorial-securitygroup' and the description is 'Tutorial Security Group'. Under 'VPC info', it shows the selected VPC is 'vpc-0c408b0a2a797a35 (vpc-01-lab05)'. The 'Inbound rules' section has a table with one row: Type: All traffic, Protocol: All, Port range: All, Source: Anywhere..., and Description: optional. A note at the bottom says: 'Rules with source of 0.0.0.0 or ::/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.' The 'Outbound rules' section is also shown below.

- d. Agregamos el security Group y dejamos la subnet la cual se asigna por defecto ( No cambiarla ya que, si no , no permite conectarla creando una subnet personalizada)

**Network settings** [Info](#)

**VPC - required** | [Info](#)

vpc-0cc408b0a2a797a35 (vpc-01-lab05)  
10.0.0.0/16

**Subnet** | [Info](#)

subnet-0e750ee47da19a99a tutorial-db-subnet-group  
VPC: vpc-0cc408b0a2a797a35 Owner: 741586629460 Availability Zone: us-east-1f  
Zone type: Availability Zone IP addresses available: 4091 CIDR: 10.0.0.0/20

**Create new subnet** [Create new subnet](#)

**Auto-assign public IP** | [Info](#)

Enable

Additional charges apply when outside of free tier allowance

**Firewall (security groups)** | [Info](#)

A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

Create security group  Select existing security group

**Common security groups** | [Info](#)

Select security groups

tutorial-securitygroup sg-09af879bb022a7006 [X](#)  
VPC: vpc-0cc408b0a2a797a35

**Compare security group rules**

Security groups that you add or remove here will be added to or removed from all your network interfaces.

**Advanced network configuration**

15. On the "Review Instance Launch" page, verify your configuration and click "Launch."

Note: To launch an EC2 instance, click "Launch Instances." On the "Launch Status" page, note the identifier of the new EC2 instance (e.g., i-0288d65fd4470b6a9).

- Verificamos el resumen que todo este correcto y lanzamos la instancia

**Summary**

**Number of instances** | [Info](#)

1

**Software Image (AMI)**  
Amazon Linux 2023 AMI 2023.6.2... [read more](#)  
ami-08b5b3a93ed654d19

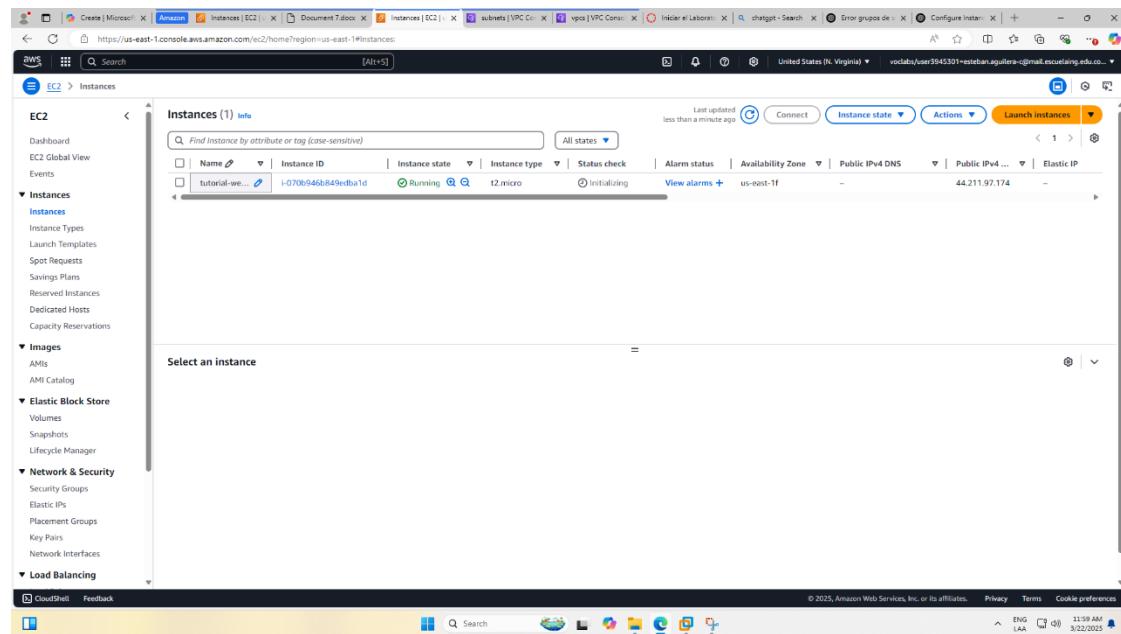
**Virtual server type (instance type)**  
t2.micro

**Firewall (security group)**  
tutorial-securitygroup

**Storage (volumes)**  
1 volume(s) - 8 GiB

**Cancel** **Launch instance** [Preview code](#)

16. To locate the created instance, click on "View Instances."



17. Install a web server on the newly created instance. Note: Refer to the instructions on how to connect to the instance in order to install the web service.

- Nos conectamos a la instancia desde nuestro cmd y con el comando ssh -i “idInstancia”

```
C:\Users\Esteban Aguilera\Downloads>ssh -i "EstebanKey.pem" ec2-user@ec2-54-89-247-237.compute-1.amazonaws.com
The authenticity of host 'ec2-54-89-247-237.compute-1.amazonaws.com (54.89.247.237)' can't be established.
ED25519 key fingerprint is SHA256:isDNeNhx8AlJbpdR373d+HoLAX2y4RWmD2vLb8sc5dk.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'ec2-54-89-247-237.compute-1.amazonaws.com' (ED25519) to the list of known hosts.
Last login: Sat Mar 29 00:58:34 2025 from ec2-18-206-107-27.compute-1.amazonaws.com
#
# Amazon Linux 2
~~ \_#####
~~ \_#####
~~ \|#
~~ \|# AL2 End of Life is 2026-06-30.
~~ \|# -->
~~ \|# A newer version of Amazon Linux is available!
~~ \|# / Amazon Linux 2023, GA and supported until 2028-03-15.
~~ \|# / https://aws.amazon.com/linux/amazon-linux-2023/
~/`
```

- Actualizamos todos los paquetes instalados en el sistema a sus últimas versiones

```
[ec2-user@ip-10-0-30-18 ~]$ sudo yum update -y
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
No packages marked for update
[ec2-user@ip-10-0-30-18 ~]$ |
```

- Instalamos el paquete de Apache HTTP Server en el sistema

```

No packages marked for update
[ec2-user@ip-10-0-30-18 ~]$ sudo yum install -y httpd
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Resolving Dependencies
--> Running transaction check
--> Package httpd.x86_64 0:2.4.62-1.amzn2.0.2 will be installed
--> Processing Dependency: httpd-filesystem = 2.4.62-1.amzn2.0.2 for package: httpd-2.4.62-1.amzn2.0.2.x86_64
--> Processing Dependency: httpd-tools = 2.4.62-1.amzn2.0.2 for package: httpd-2.4.62-1.amzn2.0.2.x86_64
--> Processing Dependency: /etc/mime.types for package: httpd-2.4.62-1.amzn2.0.2.x86_64
--> Processing Dependency: httpd-filesystem for package: httpd-2.4.62-1.amzn2.0.2.x86_64
--> Processing Dependency: mod_http2 for package: httpd-2.4.62-1.amzn2.0.2.x86_64
--> Processing Dependency: system-logos-httpd for package: httpd-2.4.62-1.amzn2.0.2.x86_64
--> Processing Dependency: libapr-1.so.0()(64bit) for package: httpd-2.4.62-1.amzn2.0.2.x86_64
--> Processing Dependency: libaprutil-1.so.0()(64bit) for package: httpd-2.4.62-1.amzn2.0.2.x86_64
--> Running transaction check
--> Package apr.x86_64 0:1.7.2-1.amzn2.0.1 will be installed
--> Package apr-util.x86_64 0:1.6.3-1.amzn2.0.1 will be installed
--> Processing Dependency: apr-util-bdb(x86-64) = 1.6.3-1.amzn2.0.1 for package: apr-util-1.6.3-1.amzn2.0.1.x86_64
--> Package generic-logos-httpd.noarch 0:18.0.0-4.amzn2 will be installed
--> Package httpd-filesystem.noarch 0:2.4.62-1.amzn2.0.2 will be installed
--> Package httpd-tools.x86_64 0:2.4.62-1.amzn2.0.2 will be installed
--> Package mailcap.noarch 0:2.1.41-2.amzn2 will be installed
--> Package mod_http2.x86_64 0:1.15.19-1.amzn2.0.2 will be installed
--> Running transaction check
--> Package apr-util-bdb.x86_64 0:1.6.3-1.amzn2.0.1 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

=====


| Package                                               | Arch   | Version             | Repository | Size            |
|-------------------------------------------------------|--------|---------------------|------------|-----------------|
| <b>Installing:</b>                                    |        |                     |            |                 |
| httpd                                                 | x86_64 | 2.4.62-1.amzn2.0.2  | amzn2-core | 1.4 M           |
| <b>Installing for dependencies:</b>                   |        |                     |            |                 |
| apr                                                   | x86_64 | 1.7.2-1.amzn2.0.1   | amzn2-core | 130 k           |
| apr-util                                              | x86_64 | 1.6.3-1.amzn2.0.1   | amzn2-core | 101 k           |
| apr-util-bdb                                          | x86_64 | 1.6.3-1.amzn2.0.1   | amzn2-core | 22 k            |
| generic-logos-httpd                                   | noarch | 18.0.0-4.amzn2      | amzn2-core | 19 k            |
| httpd-filesystem                                      | noarch | 2.4.62-1.amzn2.0.2  | amzn2-core | 25 k            |
| httpd-tools                                           | x86_64 | 2.4.62-1.amzn2.0.2  | amzn2-core | 89 k            |
| mailcap                                               | noarch | 2.1.41-2.amzn2      | amzn2-core | 31 k            |
| mod_http2                                             | x86_64 | 1.15.19-1.amzn2.0.2 | amzn2-core | 149 k           |
| <b>Transaction Summary</b>                            |        |                     |            |                 |
| Install 1 Package (+8 Dependent packages)             |        |                     |            |                 |
| Total download size: 1.9 M                            |        |                     |            |                 |
| Installed size: 5.3 M                                 |        |                     |            |                 |
| Downloading packages:                                 |        |                     |            |                 |
| (1/9): apr-1.7.2-1.amzn2.0.1.x86_64.rpm               |        |                     |            | 130 kB 00:00:00 |
| (2/9): apr-util-1.6.3-1.amzn2.0.1.x86_64.rpm          |        |                     |            | 101 kB 00:00:00 |
| (3/9): generic-logos-httpd-18.0.0-4.amzn2.noarch.rpm  |        |                     |            | 19 kB 00:00:00  |
| (4/9): apr-util-bdb-1.6.3-1.amzn2.0.1.x86_64.rpm      |        |                     |            | 22 kB 00:00:00  |
| (5/9): httpd-filesystem-2.4.62-1.amzn2.0.2.noarch.rpm |        |                     |            | 25 kB 00:00:00  |
| (6/9): httpd-2.4.62-1.amzn2.0.2.x86_64.rpm            |        |                     |            | 1.4 MB 00:00:00 |
| (7/9): mailcap-2.1.41-2.amzn2.noarch.rpm              |        |                     |            | 31 kB 00:00:00  |
| (8/9): httpd-tools-2.4.62-1.amzn2.0.2.x86_64.rpm      |        |                     |            | 89 kB 00:00:00  |
| (9/9): mod_http2-1.15.19-1.amzn2.0.2.x86_64.rpm       |        |                     |            | 149 kB 00:00:00 |
| Total                                                 |        |                     |            |                 |
| Running transaction check                             |        |                     |            |                 |
| Running transaction test                              |        |                     |            |                 |
| Transaction test succeeded                            |        |                     |            |                 |
| Running transaction                                   |        |                     |            |                 |
| Installing : apr-1.7.2-1.amzn2.0.1.x86_64             |        |                     |            |                 |
| Installing : apr-util-bdb-1.6.3-1.amzn2.0.1.x86_64    |        |                     |            |                 |
|                                                       |        |                     |            |                 |



```

1/9
2/9

```


```

d. Habilitamos el servicio de apache

```

[ec2-user@ip-10-0-30-18 ~]$ sudo systemctl enable httpd
Created symlink from /etc/systemd/system/multi-user.target.wants/httpd.service to /usr/lib/systemd/system/httpd.service.
[ec2-user@ip-10-0-30-18 ~]$

```

e. Iniciamos el servicio y luego verificamos que este funcionando con el comando systemctl status

```
[ec2-user@ip-10-0-30-18 ~]$ sudo systemctl start httpd
[ec2-user@ip-10-0-30-18 ~]$ sudo systemctl status httpd
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; enabled; vendor preset: disabled)
   Active: active (running) since Sat 2025-03-29 01:02:13 UTC; 8s ago
     Docs: man:httpd.service(8)
 Main PID: 4057 (httpd)
    Status: "Processing requests..."
      CGroupl: /system.slice/httpd.service
           └─4057 /usr/sbin/httpd -DFOREGROUND
              ├─4058 /usr/sbin/httpd -DFOREGROUND
              ├─4059 /usr/sbin/httpd -DFOREGROUND
              ├─4060 /usr/sbin/httpd -DFOREGROUND
              ├─4061 /usr/sbin/httpd -DFOREGROUND
              └─4062 /usr/sbin/httpd -DFOREGROUND

Mar 29 01:02:13 ip-10-0-30-18.ec2.internal systemd[1]: Starting The Apache HTTP Server...
Mar 29 01:02:13 ip-10-0-30-18.ec2.internal systemd[1]: Started The Apache HTTP Server.
[ec2-user@ip-10-0-30-18 ~]$ |
```

- f. Verificamos que la web este funcionando abriendo el web link de la instancia

**Instance summary for i-01271739849fd4ee2 (Server-Esteban-Def)**

- Public IPv4 address: 54.89.247.237 (open address)
- Instance state: Running
- Private IP4 DNS: ip-10-0-30-18.ec2.internal
- Instance type: t2.micro
- VPC ID: vpc-091863f312562b452 (VPC-Esteban-Def-vpc)
- Subnet ID: subnet-0d4af12d8cfbd16ee (VPC-Esteban-Def-subnet-public2-us-east-1b)
- Instance ARN: arn:aws:ec2:us-east-1:741586629460:instance/i-01271739849fd4ee2

- g. Nos mostrara la configuración básica del web server de APACHE

This page is used to test the proper operation of the Apache HTTP server after it has been installed. If you can read this page, it means that the Apache HTTP server installed at this site is working properly.

If you are a member of the general public:  
The fact that you are seeing this page indicates that the website you just visited is either experiencing problems, or is undergoing routine maintenance.  
If you would like to let the administrators of this website know that you've seen this page instead of the page you expected, you should send them e-mail. In general, mail sent to the name "webmaster" and directed to the website's domain should reach the appropriate person.  
For example, if you experienced problems while visiting www.example.com, you should send e-mail to "webmaster@example.com".

If you are the website administrator:  
You may now add content to the directory /var/www/html/. Note that until you do so, people visiting your website will see this page, and not your content. To prevent this page from ever being used, follow the instructions in the file /etc/httpd/conf.d/welcome.conf.  
You are free to use the image below on web sites powered by the Apache HTTP Server:

**Powered by APACHE 2.4**

18. Finally, add content to the web server that connects to your Amazon database instance.

a. Creamos la base de datos

The screenshot shows the 'Aurora and RDS' interface with the 'Databases' section selected. A blue banner at the top right suggests creating a blue/green deployment. The main area displays a table titled 'Databases (0)' with columns for DB identifier, Status, Role, Engine, Region, and Size. A message indicates 'No instances found'.

b. Seleccionamos MySQL

The screenshot shows the 'Create database' page. In the 'Engine options' section, the 'MySQL' engine type is selected, highlighted with a blue border. Other options include Aurora (MySQL Compatible), Aurora (PostgreSQL Compatible), PostgreSQL, Oracle, MariaDB, Microsoft SQL Server, and IBM Db2.

c. Seleccionamos en el apartado Template, la opción Free Tier

**Edition**  
 MySQL Community

**Engine version** [Info](#)  
View the engine versions that support the following database features.

**Hide filters**

Show only versions that support the Multi-AZ DB cluster [Info](#)  
Create a Multi-AZ DB cluster with one primary DB instance and two readable standby DB instances. Multi-AZ DB clusters provide up to 2x faster transaction commit latency and automatic failover in typically under 35 seconds.

Show only versions that support the Amazon RDS Optimized Writes [Info](#)  
Amazon RDS Optimized Writes improves write throughput by up to 2x at no additional cost.

**Engine version**  
MySQL 8.0.40

Enable RDS Extended Support [Info](#)  
Amazon RDS Extended Support is a paid offering. By selecting this option, you consent to being charged for this offering if you are running your database major version past the RDS end of standard support date for that version. Check the end of standard support date for your major version in the [RDS for MySQL documentation](#).

**Templates**  
Choose a sample template to meet your use case.

Production  
Use defaults for high availability and fast, consistent performance.

Dev/Test  
This instance is intended for development use outside of a production environment.

Free tier  
Use RDS Free Tier to develop new applications, test existing applications, or gain hands-on experience with Amazon RDS. [Info](#)

- d. Agregamos un identificador a la instancia de nuestra db y una contraseña (estebitanA123)

**Settings**

**DB instance identifier** [Info](#)  
Type a name for your DB instance. The name must be unique across all DB instances owned by your AWS account in the current AWS Region.

The DB instance identifier is case-insensitive, but is stored as all lowercase (as in "mydbinstance"). Constraints: 1 to 63 alphanumeric characters or hyphens. First character must be a letter. Can't contain two consecutive hyphens. Can't end with a hyphen.

**Credentials Settings**

**Master username** [Info](#)  
Type a login ID for the master user of your DB instance.

1 to 16 alphanumeric characters. The first character must be a letter.

**Credentials management**  
You can use AWS Secrets Manager or manage your master user credentials.

Managed in AWS Secrets Manager - most secure  
RDS generates a password for you and manages it throughout its lifecycle using AWS Secrets Manager.

Self managed  
Create your own password or have RDS create a password that you manage.

Auto generate password  
Amazon RDS can generate a password for you, or you can specify your own password.

**Master password** [Info](#)

Minimum constraints: At least 8 printable ASCII characters. Can't contain any of the following symbols: / \ ^ @

**Password strength** [Very strong](#)

**Confirm master password** [Info](#)

- e. Seleccionamos la opción Connect to an EC2 Resource , seleccionamos nuestra instancia creada anteriormente en EC2 instance y seleccionamos el Security Group en add VPC Security group

**Connectivity** [Info](#)

**Compute resource**

Choose whether to set up a connection to a compute resource for this database. Setting up a connection will automatically change connectivity settings so that the compute resource can connect to this database.

**Don't connect to an EC2 compute resource**  
Don't set up a connection to a compute resource for this database. You can manually set up a connection to a compute resource later.

**Connect to an EC2 compute resource**  
Set up a connection to an EC2 compute resource for this database.

**EC2 instance** [Info](#)

Choose the EC2 instance to add as the compute resource for this database. A VPC security group is added to this EC2 instance. A VPC security group is also added to the database with an inbound rule that allows the EC2 instance to access the database.

I-01271739849f4dee2  
Server-Esteban-Def

**Some VPC settings can't be changed when a compute resource is added**  
Adding an EC2 compute resource automatically selects the VPC, DB subnet group, and public access settings for this database. To allow the EC2 instance to access the database, a VPC security group rds-ec2-X is added to the database and another called ec2-rds-X to the EC2 instance. You can remove the new security group for the database only by removing the compute resource.

**Virtual private cloud (VPC)** [Info](#)

Choose the VPC. The VPC defines the virtual networking environment for this DB instance.

VPC-Esteban-Def-vpc (type:091863f312562b452)  
4 Subnets, 2 Availability Zones

Only VPCs with a corresponding DB subnet group are listed.

**After a database is created, you can't change its VPC.**

**DB subnet group** [Info](#)

Choose the DB subnet group. The DB subnet group defines which subnets and IP ranges the DB instance can use in the VPC that you selected.

**Choose existing**  
Choose existing DB subnet group

**Automatic setup**  
RDS creates a new subnet group for you or reuses an existing subnet group

**DB subnet group name**  
rds-ec2-db-subnet-group-1

New DB subnet group created.

**Public access** [Info](#)

**Yes**  
RDS assigns a public IP address to the database. Amazon EC2 instances and other resources outside of the VPC can connect to your database. Resources inside the VPC can also connect to the database. Choose one or more VPC security groups that specify which resources can connect to the database.

**No**  
RDS doesn't assign a public IP address to the database. Only Amazon EC2 instances and other resources inside the VPC can connect to your database. Choose one or more VPC security groups that specify which resources can connect to the database.

**VPC security group (firewall)** [Info](#)

Choose one or more VPC security groups to allow access to your database. Make sure that the security group rules allow the appropriate incoming traffic.

**Choose existing**  
Choose existing VPC security groups

**Create new**  
Create new VPC security group

**Additional VPC security group**

Choose one or more options

default  Group-EstebanReco

**Availability Zone** [Info](#)

**Amazon RDS will add a new VPC security group rds-ec2-1 to allow connectivity with your compute resource.**

- f. Ya llenada toda la configuración, creamos la base de datos y esperamos unos minutos a que se inicialice

Databases (1)											<input checked="" type="checkbox"/> Group resources	<a href="#">Modify</a>	<a href="#">Actions</a>	<a href="#">Restore from S3</a>	<a href="#">Create database</a>	
											<a href="#">Filter by databases</a>	< 1 >				
	DB identifier	Status	Role	Engine	Region ...	Size	Recommendations	CPU	Current activity	Mail						
<input type="radio"/>	database-esteban	Available	Instance	MySQL Co...	us-east-1b	db.t4g.micro		29.80%	0 Connections	non						

- g. Ahora entramos a la instancia e instalamos las dependencias de mysql

```
[ec2-user@ip-10-0-30-18 ~]$ sudo yum install -y --nogpgcheck mysql mysql-client
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
63 packages excluded due to repository priority protections
No package mysql-client available.
Resolving Dependencies
--> Running transaction check
--> Package mysql-community-client.x86_64 0:8.0.41-1.el7 will be installed
--> Processing Dependency: mysql-community-client-plugins = 8.0.41-1.el7 for package: mysql-community-client-8.0.41-1.el7.x86_64
--> Processing Dependency: mysql-community-libs(x86-64) >= 8.0.11 for package: mysql-community-client-8.0.41-1.el7.x86_64
--> Processing Dependency: libncurses.so.5()(64bit) for package: mysql-community-client-8.0.41-1.el7.x86_64
--> Processing Dependency: libtinfo.so.5()(64bit) for package: mysql-community-client-8.0.41-1.el7.x86_64
--> Running transaction check
--> Package mariadb-libs.x86_64 1:5.5.68-1.amzn2.0.1 will be obsoleted
--> Processing Dependency: libmysqclient.so.18()(64bit) for package: 2:postfix-2.10.1-6.amzn2.0.4.x86_64
--> Processing Dependency: libmysqclient.so.18(libmysqlclient_18)(64bit) for package: 2:postfix-2.10.1-6.amzn2.0.4.x86_64
--> Package mysql-community-client-plugins.x86_64 0:8.0.41-1.el7 will be installed
--> Processing Dependency: mysql-community-common(x86-64) >= 8.0.11 for package: mysql-community-libs-8.0.41-1.el7.x86_64
--> Package ncurses-compat-libs.x86_64 0:6.0-8.20170212.amzn2.1.8 will be installed
--> Running transaction check
--> Package mysql-community-common.x86_64 0:8.0.41-1.el7 will be installed
--> Package mysql-community-libs-compat.x86_64 0:8.0.41-1.el7 will be obsoleting
--> Finished Dependency Resolution

Dependencies Resolved

=====
| Package           | Arch   | Version        | Repository      | Size |
|=====             | =====  | =====          | =====           | ===== |
| Installing:      |         |                |                 |       |
|   mysql-community-client | x86_64 | 8.0.41-1.el7  | mysql80-community | 16 M |
|   mysql-community-libs | x86_64 | 8.0.41-1.el7  | mysql80-community | 1.5 M |
|     replacing mariadb-libs.x86_64 1:5.5.68-1.amzn2.0.1 |          |                |       |
|   mysql-community-libs-compat | x86_64 | 8.0.41-1.el7  | mysql80-community | 669 k |
|     replacing mariadb-libs.x86_64 1:5.5.68-1.amzn2.0.1 |          |                |       |
| Installing for dependencies: |         |                |                 |       |
|   mysql-community-client-plugins | x86_64 | 8.0.41-1.el7  | mysql80-community | 3.5 M |
|   mysql-community-common | x86_64 | 8.0.41-1.el7  | mysql80-community | 666 k |
|   ncurses-compat-libs | x86_64 | 6.0-8.20170212.amzn2.1.8 | amzn2-core | 309 k |
| Transaction Summary |         |                |                 |       |
|=====             | =====  | =====          | =====           | ===== |
| Install 3 Packages (+3 Dependent packages) |         |                |                 |       |
| Total size: 23 M |         |                |                 |       |
| Downloading packages: |         |                |                 |       |
| Running transaction check |         |                |                 |       |
| Running transaction test |         |                |                 |       |
| Transaction test succeeded |         |                |                 |       |
| Running transaction |         |                |                 |       |
|   Installing : mysql-community-client-plugins-8.0.41-1.el7.x86_64 | 1/7 |
|   Installing : mysql-community-common-8.0.41-1.el7.x86_64 | 2/7 |
|   Installing : mysql-community-libs-8.0.41-1.el7.x86_64 | 3/7 |
|   Installing : ncurses-compat-libs-6.0-8.20170212.amzn2.1.8.x86_64 | 4/7 |
|   Installing : mysql-community-client-8.0.41-1.el7.x86_64 | 5/7 |
|   Installing : mysql-community-libs-compat-8.0.41-1.el7.x86_64 | 6/7 |
|   Erasing : 1:mariadb-libs-5.5.68-1.amzn2.0.1.x86_64 | 7/7 |
|   Verifying : mysql-community-client-plugins-8.0.41-1.el7.x86_64 | 1/7 |
|   Verifying : mysql-community-libs-compat-8.0.41-1.el7.x86_64 | 2/7 |
|   Verifying : ncurses-compat-libs-6.0-8.20170212.amzn2.1.8.x86_64 | 3/7 |
|   Verifying : mysql-community-common-8.0.41-1.el7.x86_64 | 4/7 |
|   Verifying : mysql-community-libs-8.0.41-1.el7.x86_64 | 5/7 |
|   Verifying : mysql-community-client-8.0.41-1.el7.x86_64 | 6/7 |
|   Verifying : 1:mariadb-libs-5.5.68-1.amzn2.0.1.x86_64 | 7/7 |
| Installed: |         |                |                 |       |
|   mysql-community-client.x86_64 0:8.0.41-1.el7 | mysql-community-libs.x86_64 0:8.0.41-1.el7 | mysql-community-libs-compat.x86_64 0:8.0.41-1.el7 |
```

## h. Ingresamos como admin a la base de datos con el Endpoint creado en aws

The screenshot shows the AWS RDS console for the database 'database-esteban'. The 'Connectivity & security' tab is active. Key details shown include:

- Endpoint:** database-esteban.cdr79fxwduge.us-east-1.rds.amazonaws.com
- Port:** 3306
- Status:** Available
- Role:** Instance
- Engine:** MySQL Community
- Region & AZ:** us-east-1b

The 'Networking' section lists the availability zone as us-east-1b and the VPC as VPC-Esteban-Def-vpc (vpc-091865f312562b452). The 'Security' section shows the VPC security group rds-ec2-1 (sg-0075e3581076cf94f) with an active rule for the default security group.

```
[ec2-user@ip-10-0-30-18 ~]$ mysql -h database=esteban.cdr79fxwduge.us-east-1.rds.amazonaws.com -u admin -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 44
Server version: 8.0.40 Source distribution

Copyright (c) 2000, 2025, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> |
```

- i. Creamos la base de datos

```
mysql> CREATE DATABASE esteban_db;
Query OK, 1 row affected (0.02 sec)
```

- j. Ingresamos a la base de datos

```
mysql> use esteban_db
Database changed
```

- k. Creamos una tabla la cual va guardar información de los usuarios

```
mysql> CREATE TABLE usuarios (
    >     id INT AUTO_INCREMENT PRIMARY KEY,
    >     nombre VARCHAR(100) NOT NULL,
    >     correo VARCHAR(100) UNIQUE NOT NULL,
    >     contraseña VARCHAR(255) NOT NULL,
    >     fecha_creacion TIMESTAMP DEFAULT CURRENT_TIMESTAMP
    > );
Query OK, 0 rows affected (0.05 sec)
```

- l. Ingresamos un ejemplo de inserción

```
mysql> INSERT INTO usuarios (nombre, correo, contraseña)
->
-> VALUES ('Esteban', 'estebanac0903@gmail.com', 'estebitan');
Query OK, 1 row affected (0.01 sec)
```

- m. Verificamos que hayan ingresado los valores a la tabla

```
mysql> SELECT * FROM usuarios;
+----+-----+-----+-----+-----+
| id | nombre | correo          | contraseña | fecha_creacion |
+----+-----+-----+-----+-----+
| 1  | Esteban | estebanac0903@gmail.com | estebitan  | 2025-03-29 02:00:05 |
+----+-----+-----+-----+-----+
1 row in set (0.00 sec)
```

- n. Ahora vamos a conectarlo con la web
- o. Instalamos dependencias de Python

```
[ec2-user@ip-10-0-30-18 ~]$ sudo yum install -y python3-pip
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core
63 packages excluded due to repository priority protections
Package python3-pip-20.2.2-1.amzn2.0.9.noarch already installed and latest version
Nothing to do
[ec2-user@ip-10-0-30-18 ~]$ pip3 install flask mysql-connector-python
Defaulting to user installation because normal site-packages is not writeable
Collecting Flask
  Downloading Flask-2.2.5-py3-none-any.whl (101 kB)
Collecting mysql-connector-python
  Downloading mysql_connector_python-8.0.33-cp37-cp37m-manylinux1_x86_64.whl (27.4 MB)
Collecting click>=8.0
  Downloading click-8.1.8-py3-none-any.whl (98 kB)
Collecting itsdangerous>=2.0
  Downloading itsdangerous-2.1.2-py3-none-any.whl (15 kB)
Collecting Werkzeug<2.2.2
  Downloading Werkzeug-2.2.3-py3-none-any.whl (233 kB)
Collecting importlib-metadata>=3.6.0; python_version < "3.10"
  Downloading importlib_metadata-6.7.0-py3-none-any.whl (22 kB)
Collecting Jinja2>=3.0
  Downloading jinja2-3.1.6-py3-none-any.whl (134 kB)
Collecting protobuf<=3.20.3,>=3.11.0
  Downloading protobuf-3.20.3-cp37-cp37m-manylinux1_x86_64.whl (1.0 MB)
Collecting MarkupSafe>=2.1.1
  Downloading MarkupSafe-2.1.5-cp37-cp37m-manylinux_2_17_x86_64_manylinux2014_x86_64.whl (25 kB)
Collecting typing_extensions>=3.6.4; python_version < "3.8"
  Downloading typing_extensions-4.7.1-py3-none-any.whl (33 kB)
Collecting zipp>=0.5
  Downloading zipp-3.15.0-py3-none-any.whl (6.8 kB)
Installing collected packages: typing_extensions, zipp, importlib-metadata, click, itsdangerous, MarkupSafe, Werkzeug, Jinja2, flask, protobuf, mysql-connector-python
```

- p. Agregamos un archivo .py para crear un programa que arranca un mini sitio web que se conecta a una base de datos en la nube y muestra la lista de usuarios registrados cuando visitas la dirección del servidor.

```
GNU nano 2.9.8
from flask import Flask, jsonify
import mysql.connector

app = Flask(__name__)

# Configuración de la base de datos
db_config = {
    "host": "database-esteban.cdr79fxwduge.us-east-1.rds.amazonaws.com",
    "user": "admin",
    "password": "estebitan",
    "database": "esteban_db"
}

@app.route("/")
def home():
    return "Servidor Flask corriendo correctamente!"

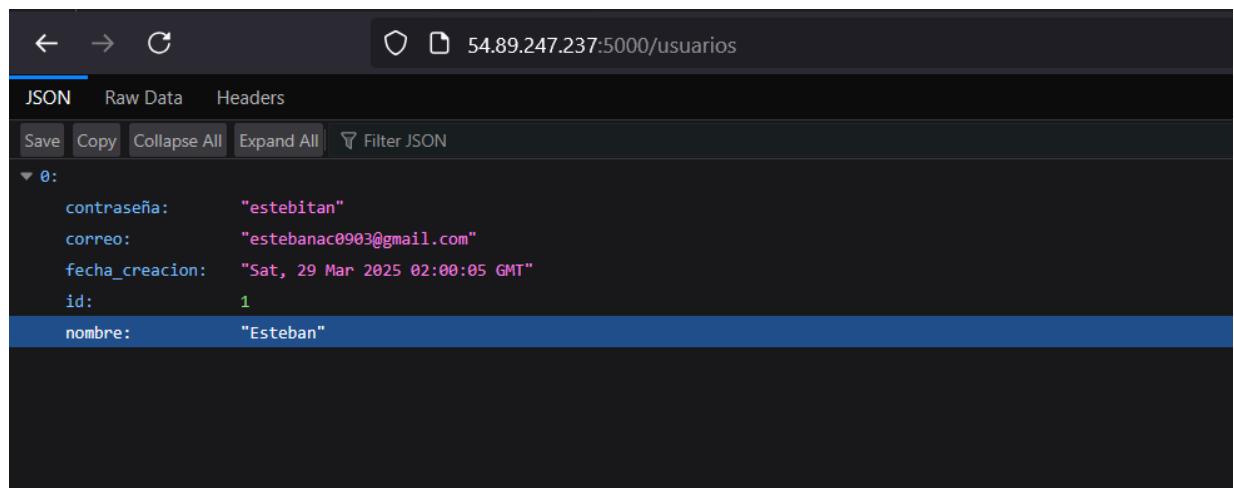
@app.route('/usuarios', methods=['GET'])
def get_usuarios():
    try:
        conn = mysql.connector.connect(**db_config)
        cursor = conn.cursor(dictionary=True)
        cursor.execute("SELECT * FROM usuarios")
        usuarios = cursor.fetchall()
        cursor.close()
        conn.close()
        return jsonify(usuarios)
    except Exception as e:
        return jsonify({"error": str(e)}), 500

if __name__ == '__main__':
    app.run(host='0.0.0.0', port=5000, debug=True)
```

- q. Ejecutamos el código en segundo plano

```
[ec2-user@ip-10-0-30-18 ~]$ nohup python3 app.py > flask.log 2>&1 &
[2] 1366
[ec2-user@ip-10-0-30-18 ~]$ |
```

- r. Ahora abrimos la pagina web desde el puerto 5000 en la ruta que definimos la cual es usuarios



A screenshot of a JSON viewer interface, likely from a browser extension or a dedicated tool like Postman. The URL in the address bar is 54.89.247.237:5000/usuarios. The interface has tabs for "JSON", "Raw Data", and "Headers", with "JSON" selected. Below the tabs are buttons for "Save", "Copy", "Collapse All", "Expand All", and "Filter JSON". The main area displays a single object under the key "0:".

```
0:
  contraseña: "estebitan"
  correo: "estebanac0903@gmail.com"
  fecha_creacion: "Sat, 29 Mar 2025 02:00:05 GMT"
  id: 1
  nombre: "Esteban"
```

## **CONCLUSIONS**

- Se logró desplegar y configurar correctamente una instancia EC2, una VPC y un servidor web, demostrando la versatilidad y escalabilidad de la infraestructura en la nube de AWS.
- La integración del servidor web con Amazon RDS permitió evidenciar cómo se pueden construir aplicaciones distribuidas que aprovechan servicios gestionados, reduciendo la complejidad en la administración de recursos.
- El laboratorio reforzó conceptos clave de redes y seguridad en la nube, como la configuración de Security Groups y la creación de subredes, fundamentales para asegurar el acceso y la integridad de los servicios implementados.