

Console Home [Info](#)

Recently visited [Info](#)

[EC2](#)[Billing and Cost Management](#)[Resource Groups & Tag Editor](#)

1. Entramos a EC2

Launch a virtual server

To get started, launch an Amazon EC2 instance, which is a virtual server in the cloud.

[Launch instance](#)[View dashboard](#)

2. Launch instance

Name and tags [Info](#)

Name

[Add additional tags](#)

▼ Application and OS Images (Amazon Machine Image) [Info](#)

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below

[Recents](#)[Quick Start](#)

 aws	 Mac	 ubuntu	 Microsoft	 Red Hat	 SUSE	 debian
---------	---------	------------	---------------	-------------	----------	------------

[Browse more AMIs](#)

Including AMIs from
AWS, Marketplace and
the Community

Amazon Machine Image (AMI)

Ubuntu Server 24.04 LTS (HVM), SSD Volume Type
ami-04f167a56786e4b09 (64-bit (x86)) / ami-0ae6f07ad3a8ef182 (64-bit (Arm))
Virtualization: hvm ENA enabled: true Root device type: ebs

[Free tier eligible](#)

3. Elegimos el nombre, el sistema operativo de la instancia y una free tier elegible

▼ Instance type [Info](#) | [Get advice](#)

Instance type

t2.micro

Free tier eligible

Family: t2 1 vCPU 1 GiB Memory Current generation: true

On-Demand Ubuntu Pro base pricing: 0.0134 USD per Hour On-Demand Linux base pricing: 0.0116 USD per Hour

On-Demand SUSE base pricing: 0.0116 USD per Hour On-Demand Windows base pricing: 0.0162 USD per Hour

On-Demand RHEL base pricing: 0.026 USD per Hour

☒ All generations

[Compare instance types](#)

Additional costs apply for AMIs with pre-installed software

4. Elegimos el tipo de instancia gratis también

▼ Key pair (login) [Info](#)

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name - required

Select



Create new key pair

5. Creamos la llave, Guardamos el archivo .pem en una ubicación para acceder a ella después

▼ Key pair (login) [Info](#)

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name - required

proyecto2



Create new key pair

6. Seleccionamos lo máximo permitido gratuito que son 30GB y gp3 (gp3 es el tipo más reciente de volumen EBS SSD de uso general en AWS. Es una mejora directa de gp2, y tiene más rendimiento y más control)

▼ Configure storage [Info](#)

[Advanced](#)

1x

30

GiB

gp3

Root volume, 3000 IOPS, Not encrypted



Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage



[Add new volume](#)

The selected AMI contains instance store volumes, however the instance does not allow any instance store volumes. None of the instance store volumes from the AMI will be accessible from the instance



Click refresh to view backup information

The tags that you assign determine whether the instance will be backed up by any Data Lifecycle Manager policies.



0 x File systems

[Edit](#)

7. Creamos la instancia

▼ Summary

Number of instances [Info](#)

1

Software Image (AMI)

Canonical, Ubuntu, 24.04, amd64...[read more](#)
ami-04f167a56786e4b09

Virtual server type (instance type)

t2.micro

Firewall (security group)

New security group

Storage (volumes)

1 volume(s) - 30 GiB

Free tier: In your first year of opening an AWS account, you get 750 hours per month of t2.micro instance usage (or t3.micro where t2.micro isn't available) when used with free tier AMIs, 750 hours per month of public IPv4 address usage, 30 GiB of EBS storage, 2 million I/Os, 1 GB of snapshots, and 100 GB of bandwidth to the internet.



[Cancel](#)

[Launch instance](#)

[Preview code](#)

8. Launch instance

✓ Success

Successfully initiated launch of instance (i-0cef2fbaaa4b74f31)



▼ Launch log

Initializing requests	✓ Succeeded
Creating security groups	✓ Succeeded
Launch initiation	✓ Succeeded

9. Click en la instancia

Instances (1) [Info](#)

Last updated
less than a minute ago



[Connect](#)

Find Instance by attribute or tag (case-sensitive)

All states ▼

Instance ID = i-0cef2fbaaa4b74f31



[Clear filters](#)

<input type="checkbox"/>	Name ↗	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone
<input type="checkbox"/>	proyecto2	i-0cef2fbaaa4b74f31	Running	t2.micro	Initializing	View alarms +	us-east-2b

10. Click en security

Instance summary for i-0cef2fbaaa4b74f31 (proyecto2) Info

Updated less than a minute ago

Instance ID

i-0cef2fbaaa4b74f31

IPv6 address

—

Hostname type

IP name: ip-172-31-25-154.us-east-2.compute.internal

Answer private resource DNS name

IPv4 (A)

Auto-assigned IP address

3.144.210.123 [Public IP]

IAM Role

—

IMDSv2

Required

Operator

—

Public IPv4 address

3.144.210.123 | [open address](#)

Instance state

Running

Private IP DNS name (IPv4 only)

ip-172-31-25-154.us-east-2.compute.internal

Instance type

t2.micro

VPC ID

vpc-0fc98990dd31bdef1

Subnet ID

subnet-01076899858d55d36

Instance ARN

arn:aws:ec2:us-east-2:495801657802:instance/i-0cef2fbaaa4b74f31

Pr

Pr

El

Al

Al

M

fa

Details

Status and alarms

Monitoring

Security

Networking

Storage

Tags

11. Security Groups

Details | Status and alarms | Monitoring | Security | Networking | Storage | Tags

▼ Security details

IAM Role

—

Security groups

sg-02103dcee6c0b2756 (launch-wizard-6)

▼ Inbound rules

Filter rules

Name

Security group rule ID

Port range

Protocol

Source

12. Editar los Inbound Rules

Inbound rules

Manage tags Edit inbound rules

Search

Name

Security group rule ID

IP version

Type

Protocol

Port range

Source

Description

No security group rules found

sg-0860b9c60a5d8f031 - launch-wizard-5

Actions

Details

Security group name

launch-wizard-5

Security group ID

sg-0860b9c60a5d8f031

Description

launch-wizard-5 created 2025-04-22T19:58:10.791Z

VPC ID

vpc-0fc98990dd31bdef1

Owner

495801657802

Inbound rules count

1 Permission entry

Outbound rules count

1 Permission entry

Inbound rules

Outbound rules

Sharing - new

VPC associations - new

Tags

Inbound rules (1)

Manage tags

Edit inbound rules

Search

< 1 >

	Name	Security group rule ID	IP version	Type	Protocol	Port range	Source	Description
	-	sgr-0b76b9ee9d99fc95	IPv4	SSH	TCP	22	0.0.0.0/0	-

13. Add Rule y agregamos el puerto 5000 y 0.0.0.0

Edit inbound rules

Inbound rules control the incoming traffic that's allowed to reach the instance.

Inbound rules

Security group rule ID

Type

Protocol

Port range

Source

Description - optional

Delete

-

SSH

TCP

22

Anywh...

0.0.0.0/0

Delete

-

Custom TCP

TCP

5000

Anywh...

0.0.0.0/0

Delete

Add rule

Rules with source of 0.0.0.0/0 or ::/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.

Cancel

Preview changes

Save rules

14. Save Rules, ya esta instancia permite recibir request a través del puerto 5000

Cancel

Preview changes

Save rules

15. Click en conect para conectarse a la instancia

Instance summary for i-0cef2fbaaa4b74f31 (proyecto2)

Connect

Instance state

Actions

Updated less than a minute ago

Instance ID

i-0cef2fbaaa4b74f31

IPv6 address

-

Hostname type

IP name: ip-172-31-25-154.us-east-2.compute.internal

Answer private resource DNS name

IPv4 (A)

Auto-assigned IP address

3.144.210.123 Public IP

Public IPv4 address

3.144.210.123 | open address

Instance state

Running

Private IP DNS name (IPv4 only)

ip-172-31-25-154.us-east-2.compute.internal

Instance type

t2.micro

VPC ID

vpc-0b-0800dd421bdef1

Private IPv4 addresses

172.31.25.154

Public IPv4 DNS

ec2-3-144-210-123.us-east-2.compute.amazonaws.com | open address

Elastic IP addresses

-

AWS Compute Optimizer finding

Get into AWS Compute Optimizer for recommendations. Learn more

Connect to instance [Info](#)

Connect to your instance i-0cef2fbaaa4b74f31 (proyecto2) using any of these options

EC2 Instance Connect | Session Manager | SSH client | EC2 serial console

Instance ID
i-0cef2fbaaa4b74f31 (proyecto2)

Connection Type

☒ Connect using EC2 Instance Connect
Connect using the EC2 Instance Connect browser-based client, with a public IPv4 or IPv6 address.

☐ Connect using EC2 Instance Connect Endpoint
Connect using the EC2 Instance Connect browser-based client, with a private IPv4 address and a VPC endpoint.

☒ Public IPv4 address
3.144.210.123

☐ IPv6 address

Username
Enter the username defined in the AMI used to launch the instance. If you didn't define a custom username, use the default username, ubuntu.

Note: In most cases, the default username, ubuntu, is correct. However, read your AMI usage instructions to check if the AMI owner has changed the default AMI username.

[Cancel](#) [Connect](#)

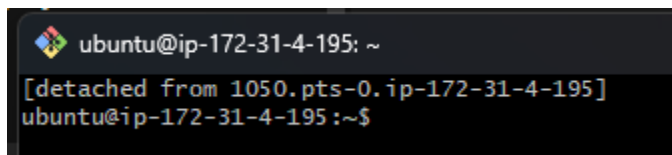
16. sudo apt update (para actualizar el indice de paquetes)
17. sudo apt install python3-pip
18. sudo apt install python3-venv -y (permite crear entornos virtuales)
19. python3 -m venv venv (creamos el entorno virtual)
20. source venv/bin/activate (activamos el entorno virtual)
21. pip install flask joblib scikit-learn torch pandas gunicorn
22. cd ~ (asegurarnos que estamos en home)
23. Dentro de la consola de nuestra maquina usar Git Bash:
24. cd "/c/Users/b_o_x/Desktop/Uniandes/Ciclo 4/Machine Learning y Procesamiento de Lenguaje Natural/Proyecto 2/AWS"
25. chmod 400 proyecto2.pem
26. ssh -i proyecto2.pem [ubuntu@3.144.210.123](#)
27. desde la consola de AWS
28. nano main.py (pegar el contenido del codigo de la api para crear el archivo main.api)
29. Guarda con Ctrl + O, luego Enter, y sal con Ctrl + X
30. nano entrenar_modelo.py (creamos los archivos del modelo ver el archivo entrenar_modelo)
31. Guarda con Ctrl + O, luego Enter, y sal con Ctrl + X
32. python3 entrenar_modelo.py (para que los archivos se creen con las instancias del entorno virtual de la maquina AWS)
33. uvicorn main:app --host 0.0.0.0 --port 5000 (para ejecutar la API)
34. sudo apt install uvicorn
35. uvicorn main:app --host 0.0.0.0 --port 5000 (para ejecutar la API)
36. pip install fastapi uvicorn
37. uvicorn main:app --host 0.0.0.0 --port 5000
38. pip install flask
39. pip install flasgger

- 40. python main.py
- 41. <http://3.144.210.123:5000/docs>

Conectarse SSH:

- 1- Ubicamos Git Bash en la dirección donde se encuentra el archivo .pem
- 2- `chmod 400 proyecto_1.pem` (cambiar permisos de archivo)
- 3- `ssh -i proyecto_1.pem ubuntu@<IP_PUBLICA>` (Conectarse usando SSH)
- 4- `ssh -i proyecto_1.pem ubuntu@3.142.12.195`
- 5- yes (escribir yes)
- 6- Ya estamos dentro de nuestra instancia creada EC2, estamos dentro de nuestra maquina virtual en AWS, ya aquí podemos scribir:
- 7- Screen
- 8- `source venv/bin/actíivate` (activar entorno virtual)
- 9- `python3 main.py`
- 10- Dejar la API corriendo y desconectarte de screen
- 11- Ahora haz lo siguiente:
- 12- **Presiona** Ctrl + A (manteniendo Ctrl, presiona A)
- 13- Luego **suelta todo** y presiona **D**
- 14- Esto no cierra tu API.

Esto solamente te desconecta de screen, pero tu API sigue corriendo en segundo plano.



```
ubuntu@ip-172-31-4-195: ~  
[detached from 1050.pts-0.ip-172-31-4-195]  
ubuntu@ip-172-31-4-195:~$
```

- 15-
- 16- Volver más tarde a tu sesión de screen
- 17- `screen -r`
- 18- <http://3.142.12.195:5000/docs>
- 19- Perfecto, si ya hiciste Ctrl + A luego D, tu API **sigue corriendo en segundo plano** dentro de la sesión screen. Ahora puedes acceder al punto final desde cualquier navegador o herramienta (como Swagger, Postman o curl).

<http://3.144.210.123:5000/docs>