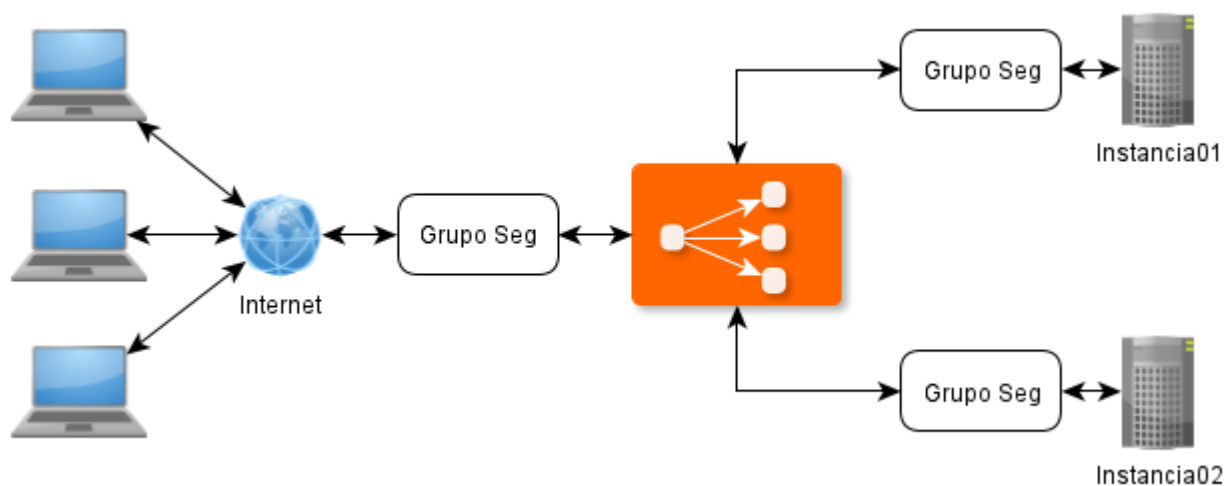




Infraestructura I

Cómo armar un ambiente más complejo en AWS

El objetivo de la clase es que conozcas cuáles son los usos reales que se van a encontrar en las empresas donde se desarrollan. Además, aprenderás buenas prácticas al momento de elegir una arquitectura para tu aplicación y sacarle provecho a lo aprendido en Infraestructura I. El modelo a diseñar es el siguiente:





Vamos a armar el ambiente en 2 clases.

La primera Clase vamos a realizar:

1. Creación de las 2 instancias en la VPC.
2. Deployar el código del trabajo realizado en Front End II.

En la Segunda clase vamos realizar:

3. Creación del load balancer.
4. Configuración del tráfico y verificación del funcionamiento.

Empecemos.

1. Creación de las instancias EC2 en la VPC.


1.a. Acceso a la consola de gestión AWS.

Una vez logueados en la consola de Amazon Educate, seleccionamos la opción **AWS Account**, aparecerá listada la materia y hacemos clic en **Go to Classroom**.

My Classrooms Portfolio Career Pathways Badges Jobs ▾ AWS Account Logout

Consecutive Days: **1** Pathways Completed: **0** Badges Earned: **0** Preferred Language: English ▾

ing over 18 million cloud jobs worldwide
ate introduces you to lucrative cloud-
learning pathways, each with content
:tivities and labs, opportunities to earn
of Completion, and access to the AWS
ses at your school or through online
ne pathway to your dream job in the



If you missed out the "Optimizing your AWS Educate Profile to Help You Find a Cloud Career" webinar and Q&A session, watch it [here](#) !

Suggested Jobs

Entry Level Software Developer
Smoothstack, Inc.
[more about this opportunity](#)

[See More](#)

Seleccionamos la opción **AWS Educate Starter Account**.

AWS Educate Starter Account

Your cloud journey has only just begun. Use your AWS Educate Starter Account to access the AWS Console and resources, and start building in the cloud!



Presionamos el botón de acceso a **AWS Console** y verificamos que el browser no bloquee ventanas emergentes en este sitio.



Vocareum

Welcome to your AWS Educate Account

AWS Educate provides you with access to a wide variety of AWS Services for you to get your hands on and build on AWS! To get started, click on the AWS Console button to log in to your AWS console.

Please read the FAQ below to help you get started on your Starter Account.

- What are the list of services supported?
- What regions are supported with Starter Accounts or Classroom Accounts?
- I can't start any resources. What happened?
- Can I create users within my Starter or Classroom Account for others to access?
- Can I create my own IAM policy within Starter Account or Classroom?

Your AWS Account Status

Active
full access ([introingenieria@gmail.com](#))

\$30
remaining credits (estimated)

2:59
session time

[Account Details](#) [AWS Console](#)

Please use AWS Educate Account responsibly. Remember to shut down your instances when not in use to make the best use of your credits. And, don't forget to logout once you are done with your work!

Nos encontramos con la consola de gestión de la plataforma AWS.

En la consola de gestión de la plataforma AWS hacemos clic en **EC2**.



AWS Management Console

AWS services

► All services

Build a solution

Get started with simple wizards and automated workflows.

Launch a virtual machine

With EC2

2-3 minutes



Build a web app

With Elastic Beanstalk

6 minutes



Build using virtual servers

With Lightsail

1-2 minutes



1.b. Crear una instancia en EC2.

Vale aclarar que este paso lo vamos a repetir para crear también la segunda instancia.

Nos posicionamos en la parte superior derecha de la pantalla y hacemos clic en el botón **Launch instances**.

Launch instances

Elgimos **Ubuntu Server 20.04 LTS**.

Seleccionamos el modelo de máquina **Family T2.micro (capa free)**.



Step 2: Choose an Instance Type

Amazon EC2 provides a wide selection of instance types optimized to fit different use cases. Instances are v for your applications. [Learn more](#) about instance types and how they can meet your computing needs.

Filter by: All instance families Current generation [Show/Hide Columns](#)

Currently selected: t2.micro (- ECUs, 1 vCPUs, 2.5 GHz, -, 1 GiB memory, EBS only)

	Family	Type	vCPUs ⓘ
<input type="checkbox"/>	t2	t2.nano	1
<input checked="" type="checkbox"/>	t2	t2.micro Free tier eligible	1
<input type="checkbox"/>	t2	t2.small	1
<input type="checkbox"/>	t2	t2.medium	2

Hacemos clic en **Next**.

En la interfaz, el Step 3 lo dejamos tal cual está y apretamos **Next**.

En el Step 4, dejamos los discos por defecto de 8 GB, volvemos a presionar **Next**.

En el Step 5, hacemos lo mismo.

COPIAMOS A QUE GRUPO DE SEGURIDAD PERTENECE

sg-0bcec8812b56facd1

En el Step 6 vamos a configurar, por ahora, un grupo de seguridad para el acceso a la instancia.

Step 6: Configure Security Group

A security group is a set of firewall rules that control the traffic for your instance. On this page, you can add rules to allow specific traffic to reach your instance. For example, if you want to set up a web server and allow Internet traffic to reach your instance, add rules that allow unrestricted access to the HTTP and HTTPS ports. You can create a new security group or select from an existing one below. [Learn more](#) about Amazon EC2 security groups.

Assign a security group: ☒ Create a new security group

☐ Select an existing security group

Security group name:

Description:

Type	Protocol	Port Range	Source	Description
SSH	TCP	22	Anywhere 0.0.0.0/0 ::/0	e.g. SSH for Admin Desktop
HTTP	TCP	80	Anywhere 0.0.0.0/0 ::/0	e.g. SSH for Admin Desktop
HTTPS	TCP	443	Anywhere 0.0.0.0/0 ::/0	e.g. SSH for Admin Desktop

[Add Rule](#)

Lo importante es darle un nombre y una descripción que nos ayude a identificarlo y dar acceso a los protocolos:

- ☐ SSH TCP PUERTO 22 ANYWHERE
- ☐ HTTP TCP PUERTO 80 ANYWHERE

Hacemos clic en **Review and Launch**.

Corroboramos la configuración de la instancia y hacemos clic en **Launch instances**.

Select an existing key pair or create a new key pair

A key pair consists of a **public key** that AWS stores, and a **private key file** that you store. Together, they allow you to connect to your instance securely. For Windows AMIs, the private key file is required to obtain the password used to log into your instance. For Linux AMIs, the private key file allows you to securely SSH into your instance.

Note: The selected key pair will be added to the set of keys authorized for this instance. [Learn more about removing existing key pairs from a public AMI.](#)

Key pair name

[Download Key Pair](#)








You have to download the **private key file** (*.pem file) before you can continue. [Store it in a secure and accessible location.](#) You will not be able to download the file again after it's created.

[Cancel](#)
[Launch Instances](#)

Creamos un nuevo key pair, si no tenemos, y descargamos el archivo .pem.

1.c. Repetimos los pasos para crear la segunda instancia.

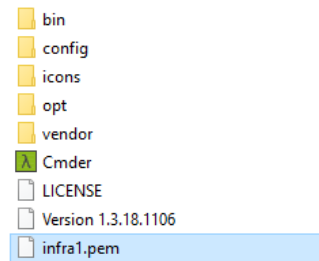
Instance: i-067007f142712d7e1 (Instancia02) =

Details	Security	Networking	Storage	Status checks	Monitoring	Tags
▼ Instance summary Info						
Instance ID  i-067007f142712d7e1 (Instancia02)		Public IPv4 address  34.237.124.200 open address		Private IPv4 addresses  172.31.7.128		
Instance state  Running		Public IPv4 DNS  ec2-34-237-124-200.compute-1.amazonaws.com open address		Private IPv4 DNS  ip-172-31-7-128.ec2.internal		
Instance type t2.micro		Elastic IP addresses -		VPC ID  vpc-4cef8131		

2. Deployar el código del trabajo realizado en Front End II.

Para este apartado vamos a necesitar una consola o terminal BASH para comunicarnos vía SSH. En la actualidad, hay muchos productos disponibles y depende del sistema operativo que estemos utilizando. Por el momento, dejamos a tu criterio cuál te parece más cómodo y agradable a la vista. En este ejemplo, utilizamos windows 10 con CMDER. En caso de no tenerlo, se puede descargar de <https://cmder.net> —recomendamos bajar la versión full que es totalmente portable—.

Copiamos el archivo de claves .pem en la carpeta raíz del cmder, solo por comodidad del ejemplo.



Abrir la carpeta en Bash donde esta la clave

Abrimos la consola. En la parte inferior derecha abrimos un bash como administrador.



Vamos a buscar la IP de la "Instancia01" que está online.

```
chmod 400 ubuntukey.pem
```

```
david@Escritorio ~/Downloads/cmdr  
λ ssh -i infra1.pem ubuntu@3.221.170.223|
```

```
>> ssh -i infra1.pem ubuntu@3.221.170.223
```



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

System information as of Sat Jul 24 00:12:26 UTC 2021

System load:  0.0          Processes:           100
Usage of /:   16.4% of 7.69GB Users logged in:      0
Memory usage: 23%         IPv4 address for eth0: 172.31.2.9
Swap usage:   0%

1 update can be applied immediately.
To see these additional updates run: apt list --upgradable

The list of available updates is more than a week old.
To check for new updates run: sudo apt update

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-172-31-2-9:~$
```

Una vez dentro, tenemos que instalar un servidor Apache para deployar nuestro código. Con este objetivo, ponemos el siguiente comando:

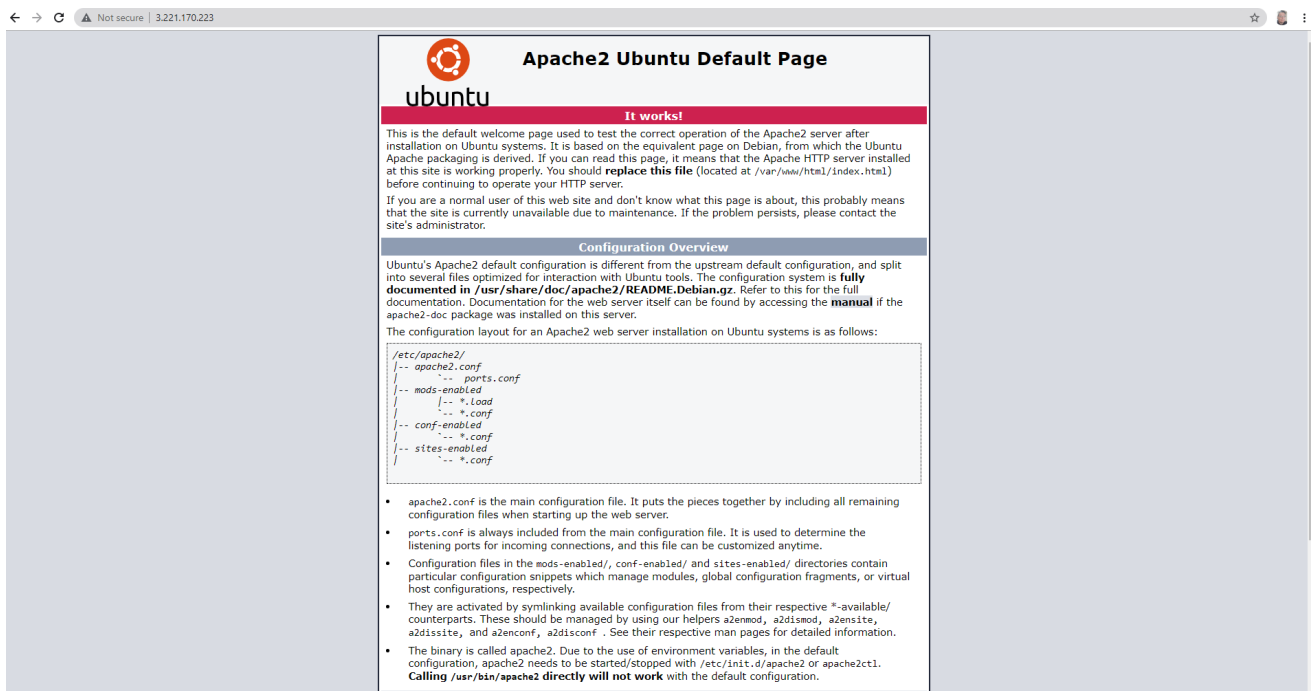
```
>> sudo apt update
```

```
>> sudo apt upgrade -y
```

```
>> sudo apt install apache2 -y
```



Comprobamos que el servicio esté andando. Ingresamos a un explorador y colocamos la IP de nuestra instancia y nos debe contestar: **Apache2 recientemente instalado.**



Luego, clonamos el repositorio del proyecto Front End II. En este caso, lo tenemos en el repositorio público de Github.

```
>> sudo git clone https://github.com/davidroco99/clase25.git
```

```
>> sudo chmod 777 -R clase25/
```

```
>> sudo cp -rf clase25/* /var/www/html/
```



Ingresamos nuevamente a la instancia a través del navegador web (repetimos este procedimiento para la segunda instancia en EC2).

The screenshot shows a web browser window with the address bar displaying '3.221.170.223/login.html'. The main content area features a login form titled 'ToDo' with the subtitle 'Ingresar'. The form includes two input fields: 'Email:' and 'Contraseña:'. Below these fields is a blue button labeled 'Ingresar'. At the bottom of the form, there is a link that reads '¿No tiene una cuenta? Regístrese aquí'.



No es seguro | mibalanceador-866200664.us-east-1.elb.amazonaws.com/login.html

YouTube Maps Gmail Playground Digital... (3) Cómo balancear...

ToDo

Ingresar

Email:

Contraseña:

[Ingresar](#)

[¿No tiene una cuenta? Regístrese aquí](#)

¡Felicitaciones! Ya has llegado hasta acá, en la próxima clase vamos a configurar el load balancer.



Actividades a realizar

1. Intentar ingresar directamente a cada instancia constatar que está corriendo nuestra aplicación. Veremos que para ingresar tenemos que cambiar nuestra url.