



Módulo 4: Front-end

Sesión 3: Despliegue en AWS

Equipo de desarrolladores:

Cristian David Ríos MSc
Daniel Escobar Grisales MSc
Nestor Rafael Calvo MSc

Coordinador del proyecto:

Prof. Dr.-Ing. Juan Rafael Orozco Arroyave



Hola!

Mi nombre es Cristian Ríos

Puedes encontrarme como:

 @cdavidrios

 @cdavid-rios

Agenda

- Servicios que se usarán en el despliegue en AWS
- Codepipeline
 - ¿Qué es Codepipeline?
 - Implementación
- CloudFront
 - ¿Qué es CloudFront?
 - Implementación
 - Validación de visualización del proyecto

A decorative network diagram in the top-left corner, featuring a complex web of interconnected nodes and lines, with some nodes highlighted in blue and others in grey.

1.

Servicios AWS

Servicios AWS



CodeCommit

CodeCommit es un servicio de control de código fuente seguro que aloja repositorios de Git privados en AWS.



CodePipeline

CodePipeline es un servicio de entrega continua que permite automatizar canalizaciones de lanzamiento



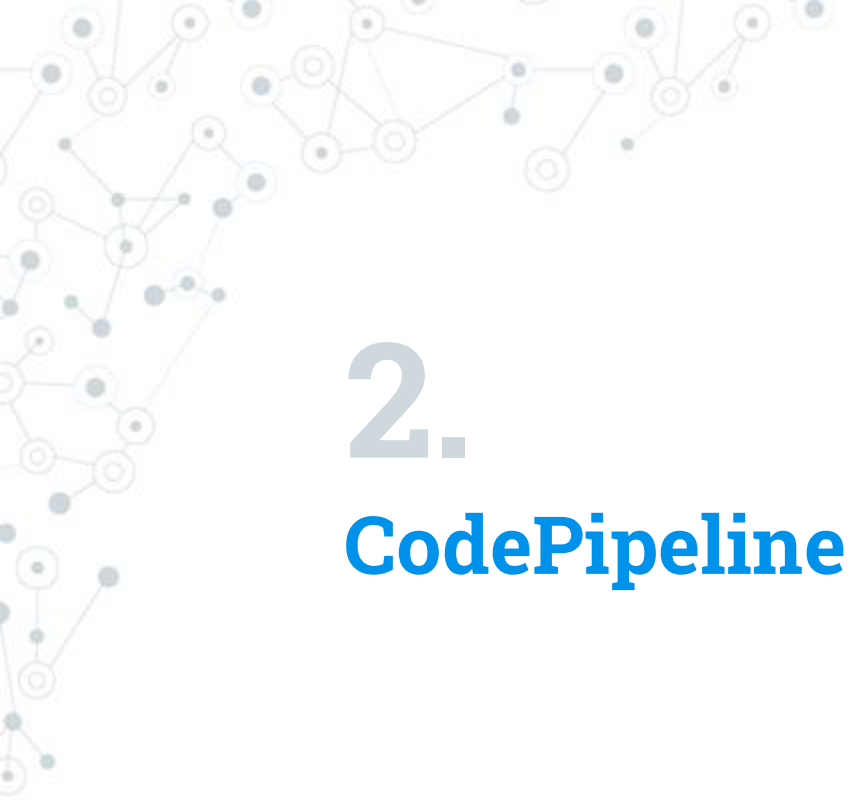
Simple Storage Service

S3 es un servicio de almacenamiento de objetos que ofrece escalabilidad, disponibilidad de datos y seguridad.



CloudFront

CloudFront es un servicio de entrega de contenido que distribuye datos, vídeos, aplicaciones y API a clientes de todo el mundo de forma segura

A decorative network diagram in the top-left corner, featuring a complex web of interconnected nodes and lines. The nodes are represented by small circles, some of which are highlighted with concentric circles, and the lines are thin and grey.

2.

CodePipeline

¿Qué es CodePipeline?

- CodePipeline es un servicio de entrega continua que permite automatizar canalizaciones de lanzamiento.
- Este servicio automatiza las fases de compilación, prueba e implementación del proceso de lanzamiento cada vez que se realiza una modificación en el repositorio
- CodePipeline será el servicio encargado de construir y desplegar el código fuente almacenado en CodeCommit y llevarlo a S3 para su correcta visualización



Creación y configuración del pipeline

Inicialmente, nos debemos dirigir al servicio CodePipeline y darle al botón “Create pipeline”. La siguiente figura muestra la configuración inicial del pipeline donde se le asigna un nombre y se crea un nuevo rol (automáticamente).

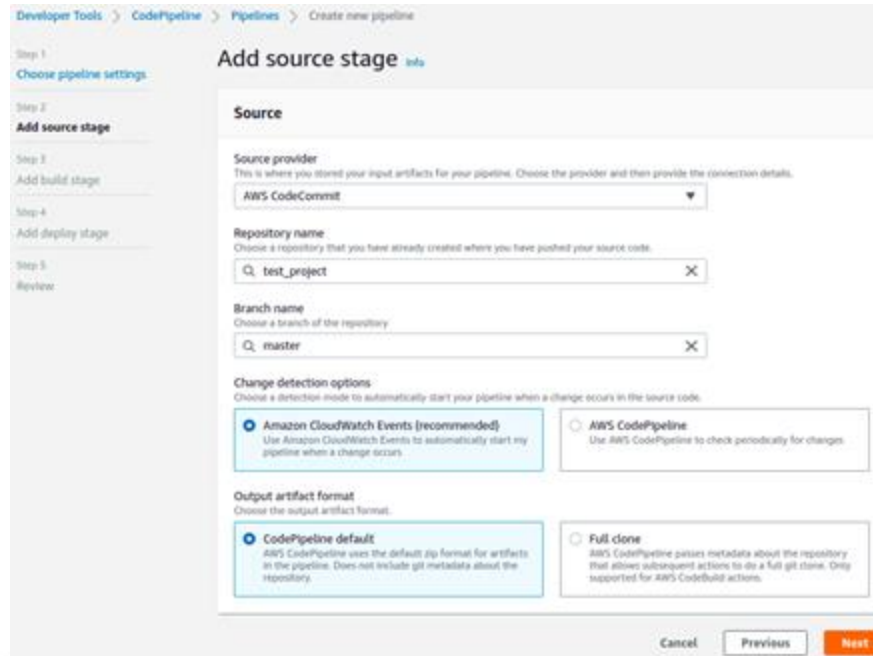
The screenshot shows the 'Create new pipeline' wizard in the AWS CodePipeline console. The breadcrumb trail at the top reads: 'Developer Tools > CodePipeline > Pipelines > Create new pipeline'. On the left, a sidebar lists the steps: 'Step 1: Choose pipeline settings', 'Step 2: Add source stage', 'Step 3: Add build stage', 'Step 4: Add deploy stage', and 'Step 5: Review'. The main content area is titled 'Choose pipeline settings' with an 'Info' link. It contains the following sections:

- Pipeline settings**
 - Pipeline name:** A text input field containing 'test-project-pipeline'. Below it, a note states: 'Enter the pipeline name. You cannot edit the pipeline name after it is created. No more than 100 characters.'
- Service role**
 - ☒ **New service role:** Create a service role in your account.
 - ☐ **Existing service role:** Choose an existing service role from your account.
- Role name:** A text input field containing 'AWSCodePipelineServiceRole-us-east-1-test-project-pipeline'. Below it, a note states: 'Type your service role name.'
- ☒ **Allow AWS CodePipeline to create a service role so it can be used with this new pipeline**

At the bottom, there is a section for 'Advanced settings' with a right-pointing arrow. At the very bottom right, there are 'Cancel' and 'Next' buttons.

Creación y configuración del pipeline

Ahora es necesario agregar el servicio que contiene el código fuente, en nuestro caso CodeCommit.



The screenshot shows the 'Add source stage' configuration page in the AWS CodePipeline console. The left sidebar lists the steps: Step 1: Choose pipeline settings, Step 2: Add source stage (selected), Step 3: Add build stage, Step 4: Add deploy stage, and Step 5: Review. The main content area is titled 'Add source stage' and includes an 'info' link. It contains several sections: 'Source' with a 'Source provider' dropdown set to 'AWS CodeCommit'; 'Repository name' with a text input 'test_project'; 'Branch name' with a text input 'master'; 'Change detection options' with two radio buttons, 'Amazon CloudWatch Events (recommended)' (selected) and 'AWS CodePipeline'; and 'Output artifact format' with two radio buttons, 'CodePipeline default' (selected) and 'Full clone'. At the bottom right are 'Cancel', 'Previous', and 'Next' buttons.

Developer Tools > CodePipeline > Pipelines > Create new pipeline

Step 1
Choose pipeline settings

Step 2
Add source stage

Step 3
Add build stage

Step 4
Add deploy stage

Step 5
Review

Add source stage [info](#)

Source

Source provider
This is where you stored your input artifacts for your pipeline. Choose the provider and then provide the connection details.

AWS CodeCommit

Repository name
Choose a repository that you have already created where you have pushed your source code.

test_project

Branch name
Choose a branch of the repository.

master

Change detection options
Choose a detection mode to automatically start your pipeline when a change occurs in the source code.

☒ Amazon CloudWatch Events (recommended)
Use Amazon CloudWatch Events to automatically start my pipeline when a change occurs.

☐ AWS CodePipeline
Use AWS CodePipeline to check periodically for changes.

Output artifact format
Choose the output artifact format.

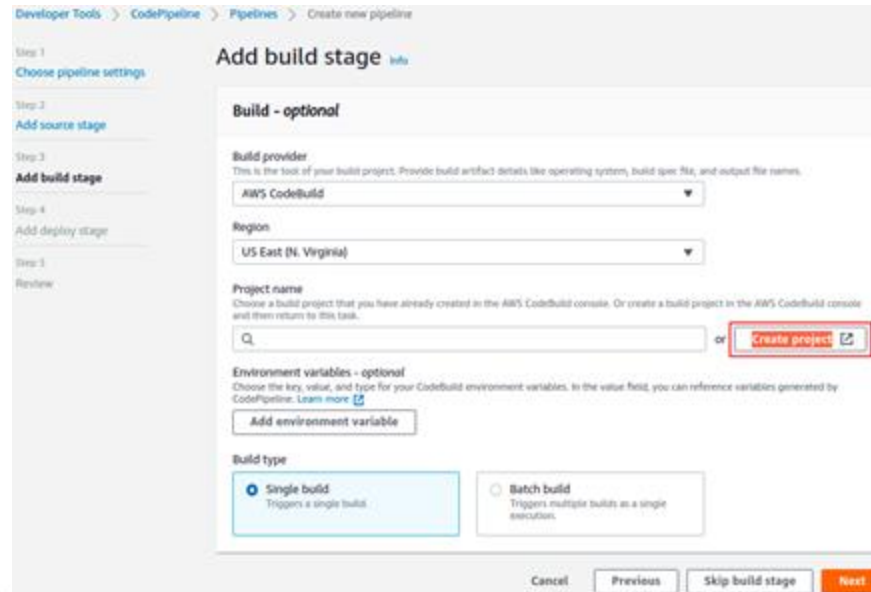
☒ CodePipeline default
AWS CodePipeline uses the default zip format for artifacts in the pipeline. Does not include git metadata about the repository.

☐ Full clone
AWS CodePipeline passes metadata about the repository that allows subsequent actions to do a full git clone. Only supported for AWS CodeBuild actions.

Cancel Previous Next

Creación y configuración del pipeline

El tercer paso de configuración es donde se agrega la etapa de compilación del proyecto y es la encargada de convertir nuestro código fuente en el contenido estático que será almacenado en S3. En esta etapa debemos darle a “create project” como lo muestra la figura



Developer Tools > CodePipeline > Pipelines > Create new pipeline

Step 1
Choose pipeline settings

Step 2
Add source stage

Step 3
Add build stage

Step 4
Add deploy stage

Step 5
Review

Add build stage info


Build - optional

Build provider
This is the tool of your build project. Provide build artifact details like operating system, build queue ID, and output file names.

AWS CodeBuild

Region
US East (N. Virginia)

Project name
Choose a build project that you have already created in the AWS CodeBuild console. Or create a build project in the AWS CodeBuild console and then return to this task.

or **Create project** 

Environment variables - optional
Choose the key, value, and type for your CodeBuild environment variables. In the value field, you can reference variables generated by CodePipeline. [Learn more](#)

Add environment variable

Build type

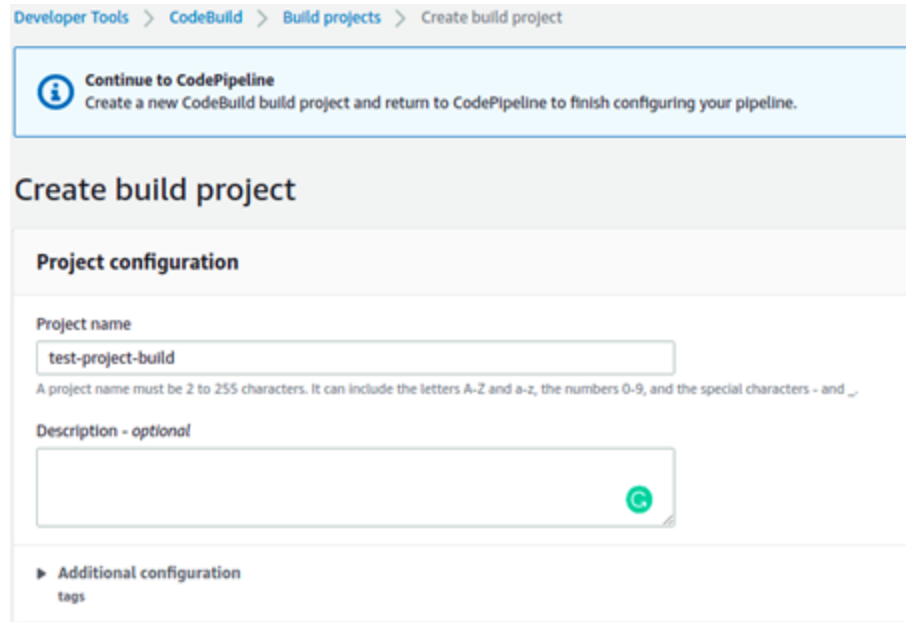
☒ **Single build**
Triggers a single build.

☐ **Batch build**
Triggers multiple builds as a single execution.

Cancel **Previous** **Skip build stage** **Next**

Creación y configuración del pipeline

La siguiente figura muestra el primer paso para la creación del “build project”, en este caso se pide el nombre del proyecto y una descripción breve del mismo.



The screenshot shows the 'Create build project' page in the AWS CodeBuild console. At the top, a breadcrumb trail reads 'Developer Tools > CodeBuild > Build projects > Create build project'. Below this is a light blue banner with an information icon and the text: 'Continue to CodePipeline. Create a new CodeBuild build project and return to CodePipeline to finish configuring your pipeline.' The main heading is 'Create build project'. Under the 'Project configuration' section, there is a 'Project name' field containing 'test-project-build'. Below the field, a note states: 'A project name must be 2 to 255 characters. It can include the letters A-Z and a-z, the numbers 0-9, and the special characters - and _.' There is also a 'Description - optional' field, which is currently empty. At the bottom, there is a section for 'Additional configuration' with a sub-label 'tags'.

Developer Tools > CodeBuild > Build projects > Create build project

Continue to CodePipeline
Create a new CodeBuild build project and return to CodePipeline to finish configuring your pipeline.

Create build project

Project configuration

Project name
test-project-build

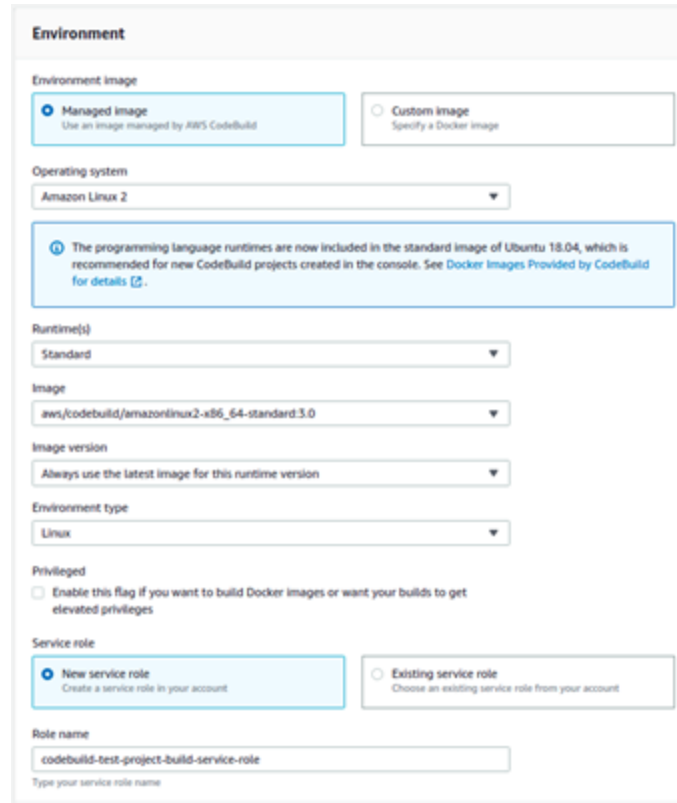
A project name must be 2 to 255 characters. It can include the letters A-Z and a-z, the numbers 0-9, and the special characters - and _.

Description - optional

Additional configuration
tags

Creación y configuración del pipeline

Ahora debemos elegir el environment para el proyecto creado, en nuestro caso nosotros usamos una imagen de Linux 2 creada por Amazon. En la parte final, nos pone el nombre del rol asignado al proyecto, que está relacionado con el nombre del proyecto asignado en la parte 1.



Environment

Environment image

☒ Managed image
Use an image managed by AWS CodeBuild

☐ Custom image
Specify a Docker image

Operating system

Amazon Linux 2

The programming language runtimes are now included in the standard image of Ubuntu 18.04, which is recommended for new CodeBuild projects created in the console. See [Docker Images Provided by CodeBuild for details](#).

Runtime(s)

Standard

Image

aws/codebuild/amazonlinux2-x86_64-standard.3.0

Image version

Always use the latest image for this runtime version

Environment type

Linux

Privileged

☐ Enable this flag if you want to build Docker images or want your builds to get elevated privileges

Service role

☒ New service role
Create a service role in your account

☐ Existing service role
Choose an existing service role from your account

Role name

codebuild-test-project-build-service-role

Type your service role name

Creación y configuración del pipeline

Ahora se debe asignar el archivo ejecutable que construirá el proyecto alojado en CodeCommit, en nuestro caso, debemos escoger la opción “Use a buildspec file”. Automáticamente ejecutará el archivo “buildspec.yml” que se encuentra en el repositorio

Buildspec

Build specifications

☒ Use a buildspec file
Store build commands in a YAML-formatted buildspec file

☐ Insert build commands
Store build commands as build project configuration

Buildspec name - optional
By default, CodeBuild looks for a file named buildspec.yml in the source code root directory. If your buildspec file uses a different name or location, enter its path from the source root here (for example, buildspec-two.yml or configuration/buildspec.yml).

Batch configuration
You can run a group of builds as a single execution. Batch configuration is also available in advanced option when starting build.

☐ Define batch configuration - optional
You can also define or override batch configuration when starting a build batch.

Logs

CloudWatch

☒ CloudWatch logs - optional
Checking this option will upload build output logs to CloudWatch.

Group name

Stream name

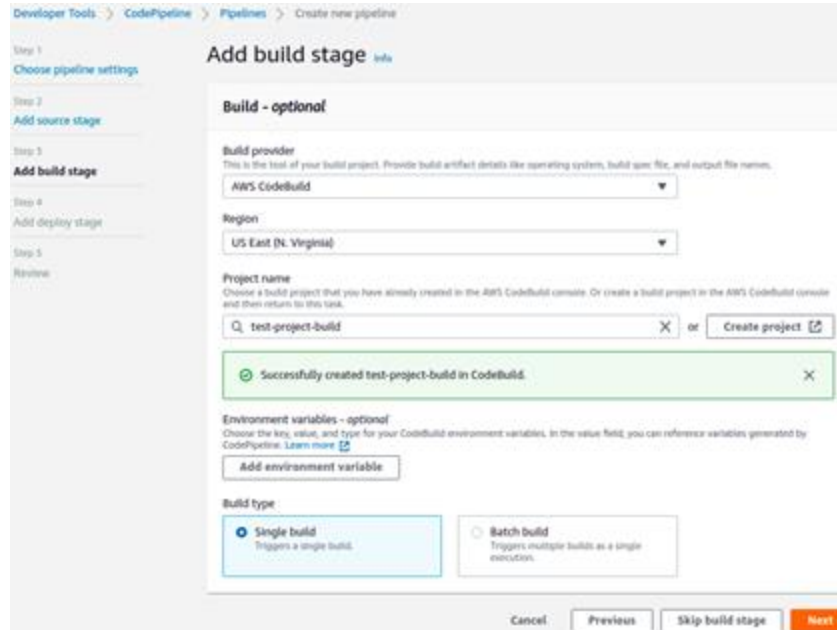
S3

☐ S3 logs - optional
Checking this option will upload build output logs to S3.

Cancel **Continue to CodePipeline**

Creación y configuración del pipeline

Finalmente, el “build project” será creado exitosamente y se debe continuar con la configuración del pipeline



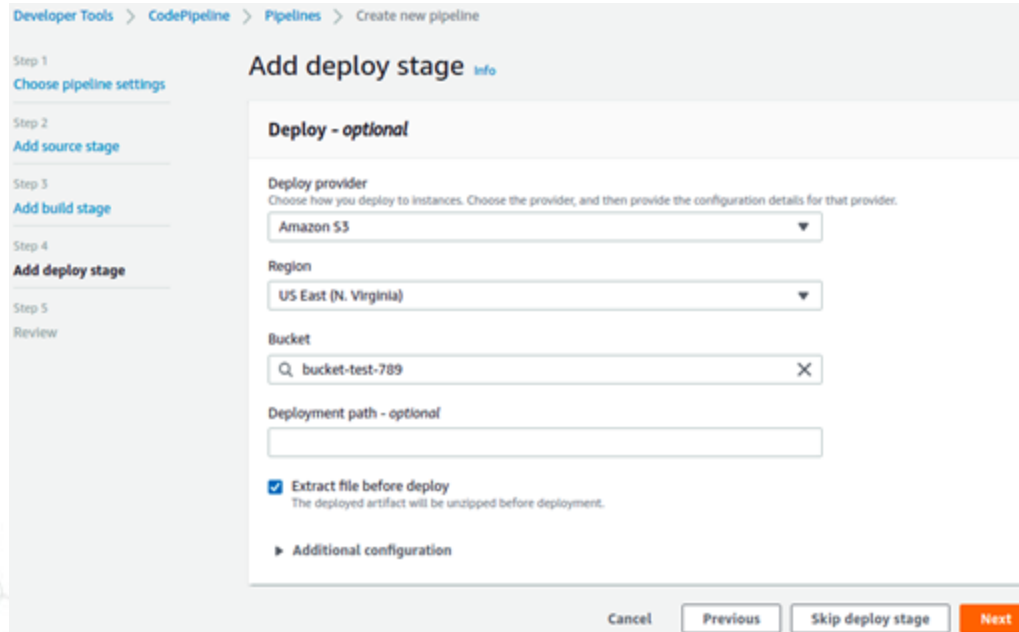
The screenshot displays the 'Add build stage' configuration interface in the AWS CodePipeline console. The left sidebar shows the setup steps: Step 1 (Choose pipeline settings), Step 2 (Add source stage), Step 3 (Add build stage), Step 4 (Add deploy stage), and Step 5 (Review). The main panel is titled 'Add build stage' and contains the following sections:

- Build - optional**: A section header for the build stage configuration.
- Build provider**: A dropdown menu set to 'AWS CodeBuild'.
- Region**: A dropdown menu set to 'US East (N. Virginia)'.
- Project name**: A text input field containing 'test-project-build', followed by a search icon, an 'or' separator, and a 'Create project' button with a plus icon.
- Environment variables - optional**: A section with a description and a link to 'Learn more'. Below it is an 'Add environment variable' button.
- Build type**: Two radio button options: 'Single build' (selected) and 'Batch build'.

At the bottom of the form, there are four buttons: 'Cancel', 'Previous', 'Skip build stage', and 'Next'.

Creación y configuración del pipeline

Ahora, la etapa 4 de la configuración del pipeline es donde se realizará el despliegue luego de compilar el proyecto. En nuestro caso, en el bucket creado en S3



The screenshot shows the AWS CodePipeline console interface for creating a new pipeline. The breadcrumb navigation at the top reads: Developer Tools > CodePipeline > Pipelines > Create new pipeline. On the left sidebar, the steps are listed: Step 1: Choose pipeline settings, Step 2: Add source stage, Step 3: Add build stage, Step 4: Add deploy stage (which is the active step), and Step 5: Review. The main content area is titled 'Add deploy stage' with an 'Info' link. Below this, the configuration for a 'Deploy - optional' stage is shown. The 'Deploy provider' is set to 'Amazon S3'. The 'Region' is 'US East (N. Virginia)'. The 'Bucket' is 'bucket-test-789'. The 'Deployment path - optional' field is empty. There is a checked checkbox for 'Extract file before deploy' with the note 'The deployed artifact will be unzipped before deployment.' Below this is a section for 'Additional configuration'. At the bottom right, there are four buttons: 'Cancel', 'Previous', 'Skip deploy stage', and 'Next'.

Developer Tools > CodePipeline > Pipelines > Create new pipeline

Step 1
Choose pipeline settings

Step 2
Add source stage

Step 3
Add build stage

Step 4
Add deploy stage

Step 5
Review

Add deploy stage [Info](#)

Deploy - optional

Deploy provider
Choose how you deploy to instances. Choose the provider, and then provide the configuration details for that provider.

Amazon S3

Region
US East (N. Virginia)

Bucket
bucket-test-789

Deployment path - optional

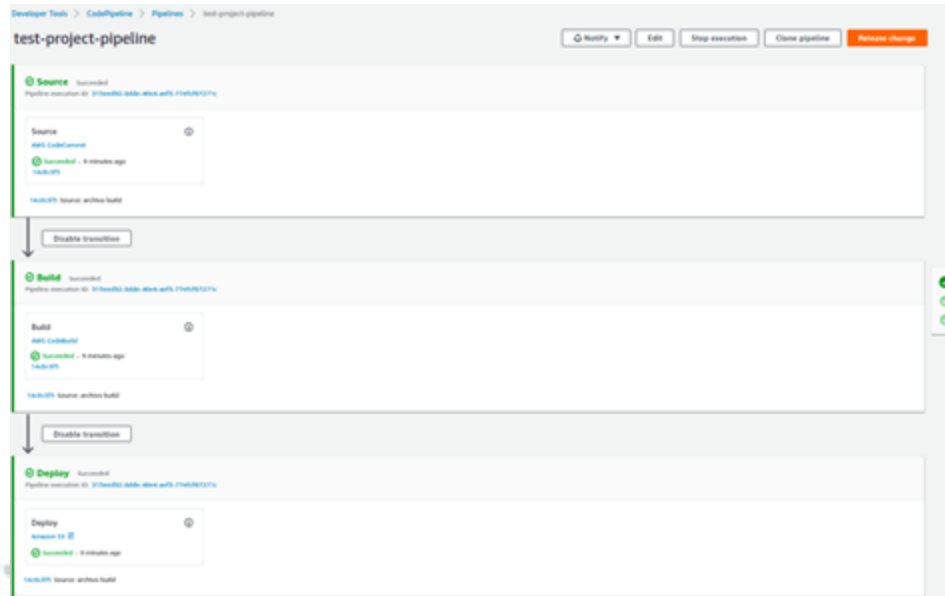
☒ **Extract file before deploy**
The deployed artifact will be unzipped before deployment.

► **Additional configuration**

Cancel Previous Skip deploy stage Next

Creación y configuración del pipeline

Finalmente, en la última etapa se muestra un resumen de las configuraciones realizadas. En estos momentos, el pipeline comenzará a ejecutarse automáticamente y luego de algunos minutos nos mostrará que pudo realizar el despliegue satisfactoriamente



Validación del despliegue

Ahora solo queda consultar la URL obtenida en S3 y debería mostrarnos el contenido de nuestro servicio web.

En este caso, ya quedó configurado el pipeline, por lo tanto, cualquier cambio realizado en el repositorio de AWS (CodeCommit) será compilado y mostrado en la URL de S3 en cuestión de minutos.

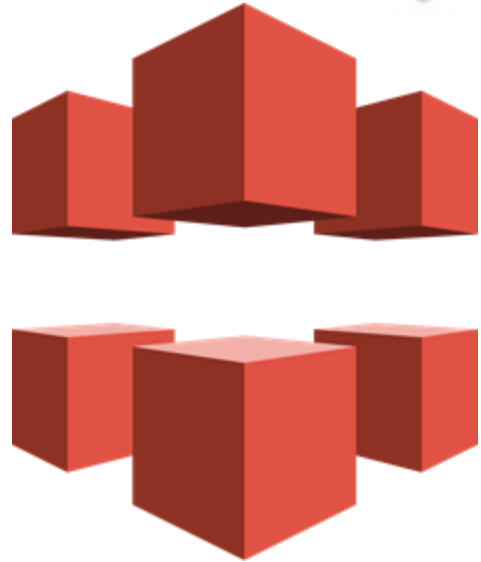
A decorative network diagram in the top-left corner, featuring a complex web of interconnected nodes and lines, with some nodes highlighted in blue and green.

2.

CloudFront

¿Qué es CloudFront?

- ⦿ CloudFront es un servicio de entrega de contenido que distribuye datos, vídeos, aplicaciones y API a clientes de todo el mundo de forma segura, con baja latencia, altas velocidades de transferencia y dentro de un entorno intuitivo para desarrolladores
- ⦿ En nuestro caso, nosotros usamos CloudFront para asociar un certificado web (Secure Sockets Layer, SSL) a nuestro servicio web



Asignación de certificado SSL a S3

Inicialmente, debemos ir al servicio de CloudFront en AWS y darle click a “Create Distribution”. La siguiente figura muestra lo que nos despliega AWS y la configuración que se debe elegir

Nota: En esta etapa inicialmente se debe tener un certificado asociado a su dominio, este se obtiene en el servicio “AWS Certificate Manager”

Edit Distribution

Distribution Settings

Price Class	<div>Use All Edge Locations (Best Performance) ▾</div>	?
AWS WAF Web ACL	<div>None ▾</div>	?
Alternate Domain Names (CNAMEs)	<div>biometria.udea.edu.co</div>	?
SSL Certificate	<div><div><input type="radio"/> Default CloudFront Certificate (*.cloudfront.net)</div><div>Choose this option if you want your users to use HTTPS or HTTP to access your content with the CloudFront domain name (such as https://d1111111abcdetf.cloudfront.net/logo.jpg). Important: If you choose this option, CloudFront requires that browsers or devices support TLSv1 or later to access your content.</div><div><input checked="" type="radio"/> Custom SSL Certificate (example.com):</div><div>Choose this option if you want your users to access your content by using an alternate domain name, such as https://www.example.com/logo.jpg. You can use a certificate stored in AWS Certificate Manager (ACM) in the US East (N. Virginia) Region, or you can use a certificate stored in IAM.</div><div><div>biometria.udea.edu.co (5671e958-3b99-)</div><div>?</div><div>Request or Import a Certificate with ACM</div></div><div><div>Learn more about using custom SSL/TLS certificates with CloudFront.</div><div>Learn more about using ACM.</div></div></div>	
Custom SSL Client Support	<div><div><input checked="" type="radio"/> Clients that Support Server Name Indication (SNI) - (Recommended)</div><div>CloudFront serves your content over HTTPS to clients that support SNI. SNI is supported by browsers and clients released after 2010. There is no additional charge for this option. Learn More</div><div><input type="radio"/> Legacy Clients Support (\$600/month prorated charge applies. Learn about pricing)</div><div>CloudFront allocates dedicated IP addresses at each CloudFront edge location to serve your content over HTTPS. Learn More</div></div>	

Asignación de certificado SSL a S3

La siguiente figura es la continuación de la mostrada en la figura anterior y se deben mantener las opciones que se encuentran por defecto.

Security Policy

- ☐ TLSv1
- ☐ TLSv1_2016
- ☐ TLSv1.1_2016
- ☐ TLSv1.2_2018
- ☒ TLSv1.2_2019 (recommended)

See the [list of protocols and ciphers](#) that CloudFront uses for each security policy.

Supported HTTP Versions

- ☒ HTTP/2, HTTP/1.1, HTTP/1.0
- ☐ HTTP/1.1, HTTP/1.0

Default Root Object

Standard Logging

- ☐ On
- ☒ Off

S3 Bucket for Logs

Log Prefix

Cookie Logging

- ☐ On
- ☒ Off

Enable IPv6

☒

[Learn more](#)

Comment

Distribution State

- ☒ Enabled
- ☐ Disabled

Asignación de certificado SSL a S3

En esta etapa es necesario que usted elija el origen del servicio web (S3-bucket). Además, es necesario configurar los métodos HTTP que serán usados en nuestro servicio web

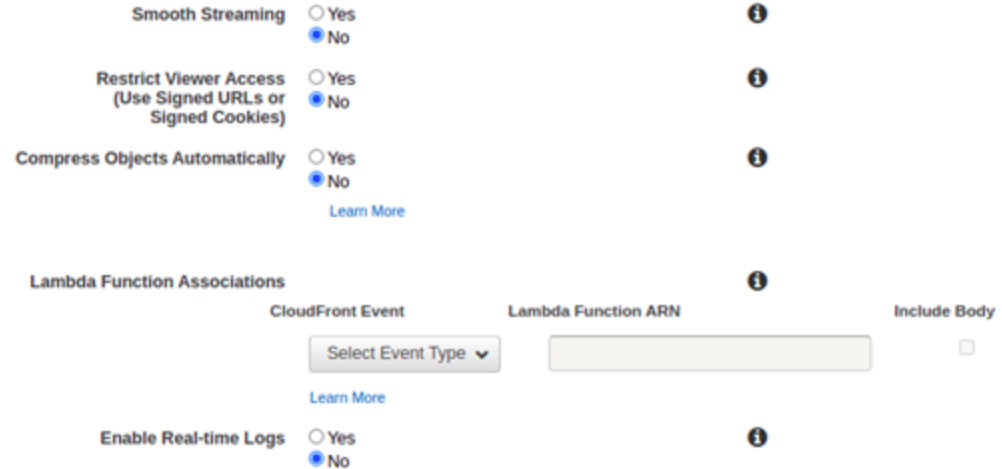
Edit Behavior

Default Cache Behavior Settings

Path Pattern	Default (*)		
Origin or Origin Group	S3-Website-biometria-register.s3-website-us-east-1		
Viewer Protocol Policy	<input type="radio"/> HTTP and HTTPS <input checked="" type="radio"/> Redirect HTTP to HTTPS <input type="radio"/> HTTPS Only		
Allowed HTTP Methods	<input type="radio"/> GET, HEAD <input type="radio"/> GET, HEAD, OPTIONS <input checked="" type="radio"/> GET, HEAD, OPTIONS, PUT, POST, PATCH, DELETE		
Field-level Encryption Config			
Cached HTTP Methods	GET, HEAD (Cached by default) <input type="checkbox"/> OPTIONS		
Cache and origin request settings	<input checked="" type="radio"/> Use a cache policy and origin request policy <input type="radio"/> Use legacy cache settings		
Cache Policy	Managed-CachingDisabled	Create a new policy	
	View policy details		
	Learn More		
Origin Request Policy		Create a new policy	
	View policy details		
	Learn More		

Asignación de certificado SSL a S3

El último paso es mostrado en la siguiente figura. En este caso podemos dejar las configuraciones que trae CloudFront por defecto. Ahora solo queda darle click a “Create Distribution”.



The screenshot shows the configuration page for a new CloudFront distribution. The settings are as follows:

- Smooth Streaming:** ☒ No
- Restrict Viewer Access (Use Signed URLs or Signed Cookies):** ☒ No
- Compress Objects Automatically:** ☒ No
- Lambda Function Associations:**
 - CloudFront Event:** Select Event Type (dropdown menu)
 - Lambda Function ARN:** (empty text box)
 - Include Body:** ☐
- Enable Real-time Logs:** ☒ No

Information icons (i) are present next to the Smooth Streaming, Restrict Viewer Access, Compress Objects Automatically, Lambda Function Associations, and Enable Real-time Logs sections. A "Learn More" link is located below the "Compress Objects Automatically" and "CloudFront Event" settings. A plus icon (+) is visible on the right side of the page.

Asignación de certificado SSL a S3

El último paso es mostrado en la siguiente figura. En este caso podemos dejar las configuraciones que trae CloudFront por defecto. Ahora solo queda darle click a “Create Distribution”.

Smooth Streaming ☐ Yes ☒ No ⓘ

Restrict Viewer Access (Use Signed URLs or Signed Cookies) ☐ Yes ☒ No ⓘ

Compress Objects Automatically ☐ Yes ☒ No ⓘ

[Learn More](#)

Lambda Function Associations ⓘ

CloudFront Event ⓘ

Lambda Function ARN ⓘ

Include Body ☐ ⓘ

[Learn More](#)

Enable Real-time Logs ☐ Yes ☒ No ⓘ

Validación del dominio

Posteriormente, usted podrá consultar su servicio web con el dominio asignado anteriormente, en este caso, el dominio deberá tener un protocolo HTTPS y usted lo puede verificar observando un candado en la parte superior izquierda.

