

Infraestructura III

Integrando API con microservicios y pipelines

Objetivo:

Crear un pipeline para desplegar una API.

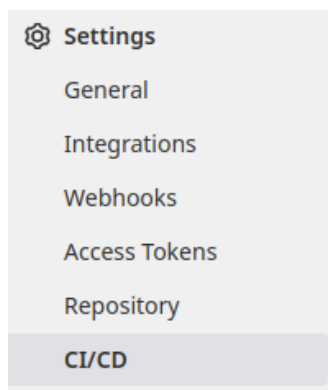
Consigna

Usaremos el ejemplo explicado en la clase como base para generar nuestros scripts de Terraform para poder desplegar una API.

Vamos a desplegar una Lambda, tal cual lo hemos visto en clase. Deberíamos seguir los mismos pasos.

Ahora vamos a crear un nuevo proyecto en GitLab y vamos a configurar las variables de autenticación de AWS:

1. Nos dirigimos a Settings - CI/CD.





2. Nos dirigimos a Variables.

Variables

Variables store information, like passwords and secret keys, that you can use in job scripts. [Learn more.](#)

Variables can be:

- **Protected:** Only exposed to protected branches or tags.
- **Masked:** Hidden in job logs. Must match masking requirements. [Learn more.](#)

Expand

3. Apretamos el botón Add variable.

Add variable

Reveal values

4. Creamos dos variables, AWS_ACCESS_KEY_ID y AWS_SECRET_ACCESS_KEY. Les asignamos los valores que corresponden de acuerdo a los datos que nos pasaron. Tenemos que deseleccionar **Protect variable** y seleccionar **Mask variable**.

Add variable

×

Key

AWS_ACCESS_KEY_ID

Value

EL_VALOR_QUE_NOS_PASARON

Type

Variable

Environment scope

All (default)

Flags

☐ Protect variable ⓘ
Export variable to pipelines running on protected branches and tags only.

☒ Mask variable ⓘ
Variable will be masked in job logs. Requires values to meet regular expression requirements. [More information](#)

Add variable

×

Key

AWS_SECRET_ACCESS_KEY

Value

EL_VALOR_QUE_NOS_PASARON

Type

Variable

Environment scope

All (default)

Flags

☐ Protect variable ⓘ
Export variable to pipelines running on protected branches and tags only.

☒ Mask variable ⓘ
Variable will be masked in job logs. Requires values to meet regular expression requirements. [More information](#)

Ahora vamos a crear los siguientes archivos en el repositorio:



(los archivos son los que están en este repositorio: <https://gl.deitech.online/demonccc/api-deploy>).

Una vez que se haya realizado el primer commit, se dispara un pipeline, el cual consta de 3 jobs:

Update s3.tf

The screenshot shows a GitHub Actions workflow run for the file 's3.tf'. The workflow is named 'Update s3.tf'. It shows a summary of the run: '3 jobs for main in 1 minute and 14 seconds (queued for 2 seconds)'. Below this, there is a commit hash '820f2d46' and a note 'No related merge requests found.' The workflow is divided into three stages: 'Prepare', 'Validate', and 'Build'. Each stage has a job: 'init' under Prepare, 'validate' under Validate, and 'plan' under Build. All jobs are marked with a green checkmark, indicating they passed successfully.

Para poder ejecutar el job de despliegue, será necesario generar un merge request.

Para eso, primero hay que crear un branch con el nombre feature-X (ejemplo: feature-1):

The screenshot shows a GitHub repository interface. At the top, there is a dropdown menu for selecting a branch, currently set to 'main'. Below this, there is a section for 'Update outputs.tf' by 'Claudio Cesar Sanchez Tejeda'. There are buttons for 'Upload File', 'README', and 'CI'. Below these, there are buttons for 'Add Kubernetes cluster' and 'Configure'. A dropdown menu is open, showing options for 'This directory' (New file, Upload file, New directory) and 'This repository' (New branch, New tag). The 'New branch' option is highlighted.



New Branch

Branch name

feature-1

Create from

main

Existing branch name, tag, or commit SHA

Create branch

Cancel

Una vez creado el branch, generamos un cambio en el código (por ejemplo, modificar el archivo README.md o, si nos animamos, modificar algún comportamiento del código).

Después de realizar el commit en dicho branch, hay que generar un merge request (ingresar el título y la descripción que sea más cómoda):

GitLab

Menu

A API Deploy

Project information

Repository

Issues 0

Merge requests 0

CI/CD

Security & Compliance

Deployments

Monitor

Claudio Cesar Sanchez Tejeda > API Deploy > Merge requests

✓ You pushed to **feature-1** just now

Create merge request

Open 0 Merged 1 Closed 0 All 1

Recent searches Search or filter results...



New merge request

From `feature-1` into `main` [Change branches](#)

Title *

Feature 1

Start the title with `Draft:` to prevent a merge request draft from merging before it's ready.
Add [description templates](#) to help your contributors communicate effectively!

Description

Write Preview

Feature 1

Markdown and quick actions are supported

[Attach a file](#)

Assignee

Unassigned

[Assign to me](#)

Reviewer

Unassigned

Milestone

Milestone

Labels

Labels

Merge options

- ☒ Delete source branch when merge request is accepted.
☐ Squash commits when merge request is accepted. [?](#)

Create merge request

Cancel

Una vez creado el merge request, podemos aceptarlo para que se integre al branch master (hay que apretar el botón **Merge**):

Open Created 8 minutes ago by Claudio Cesar Sanchez Tejada Owner Edit Mark as ready ▼

Feature 1

Overview 0 Commits 1 Pipelines 2 Changes 1

Request to merge `feature-1` into `main` Open in Web IDE Check out branch Download ▼

Pipeline #8550 passed for 026d3cbd on feature-1 3 minutes ago Download ▼

Revoke approval Approved by you

1 Terraform report was generated in your pipelines Expand

Merge ☒ Delete source branch ☐ Squash commits [?](#)

> Adds 1 commit and 1 merge commit to main. [Modify merge commit](#)

👍 0 👎 0 😊 Oldest first ▼ Show all activity ▼

Una vez integrado se disparará el pipeline de despliegue:



Feature 1

Overview 0 Commits 1 Pipelines 2 Changes 1

Request to merge feature-1 into main

Pipeline #8550 passed for 026d3cbd on feature-1 3 minutes ago

Approved by you

1 Terraform report was generated in your pipelines

Merged by Claudio Cesar Sanchez Tejada just now
The changes were merged into main with 88396226
The source branch has been deleted

Pipeline #8552 pending for 88396226 on main

0 0

Oldest first Show all activity

Status	Pipeline	Triggerer	Stages
running In progress	Update s3.tf #8566 main -> 833286aa latest		

Cuando terminen los jobs que se ejecutan de forma automática, se habilita el job manual de despliegue:

4 jobs for main (queued for 1 second)

latest

833286aa

No related merge requests found.

Pipeline Needs Jobs 4 Tests 0

Prepare

Validate

Build

Deploy

init

validate

plan

apply

Ahora, para realizar el despliegue, hay que presionar el botón de play.



🕒 4 jobs for `main` (queued for 2 seconds)

🚩 `latest`

🔑 `a44fc4d3` 📄

🔗 No related merge requests found.

Pipeline Needs Jobs 4 Tests 0

Prepare

Validate

Build

Deploy ▶

✅ init ↺

✅ validate ↺

✅ plan ↺

⚙️ apply ▶

Resuelto

El resultado del ejercicio es la ejecución correcta del pipeline con la creación de los recursos de AWS.