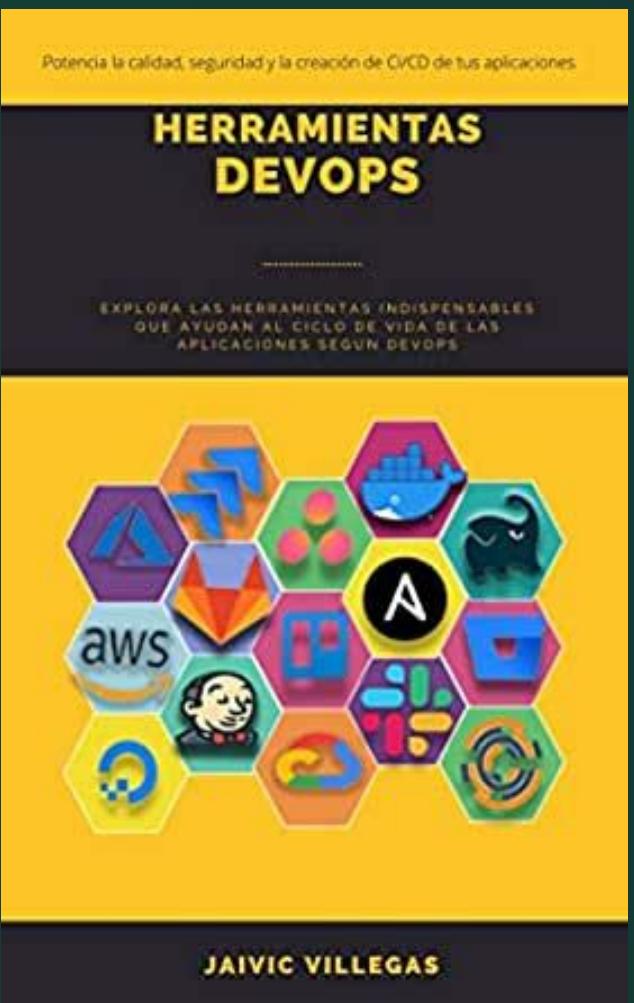


Curso de DevOps

Jaivic Villegas







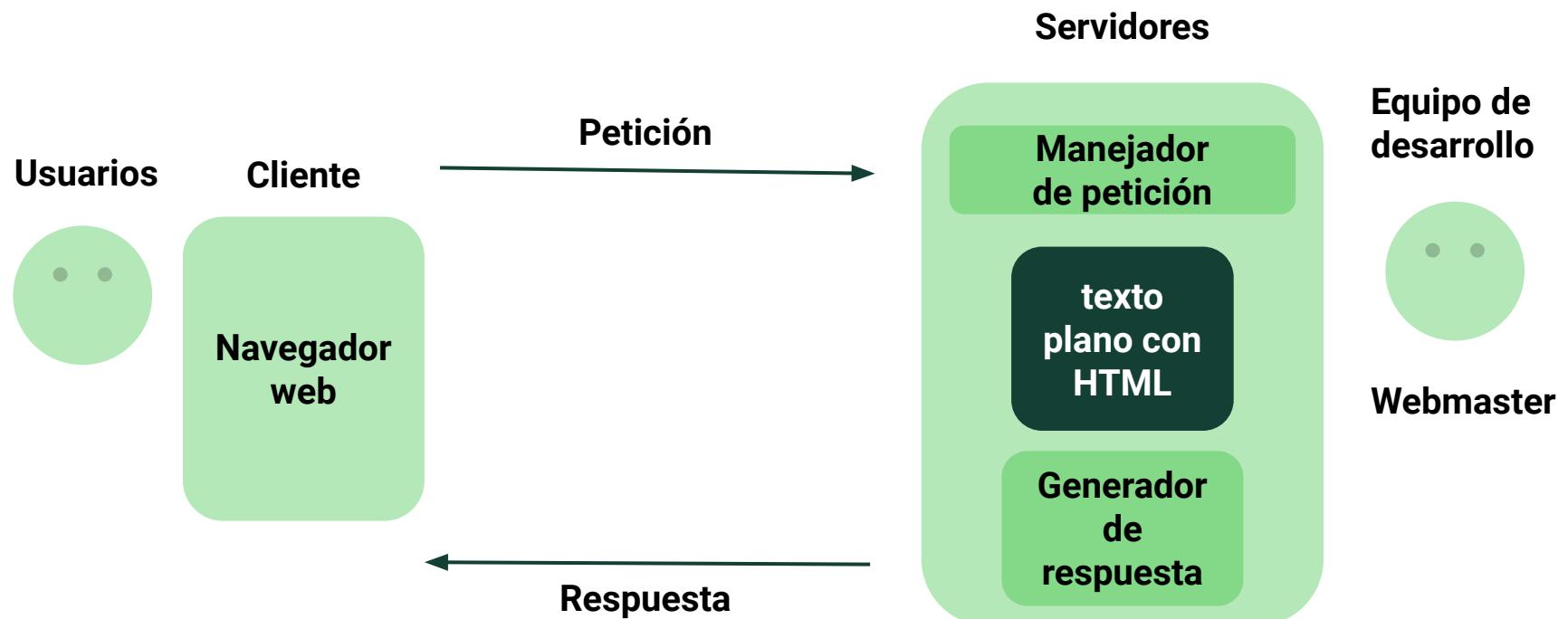
¿Has decidido ser DevOps?

Especialista en DevOps

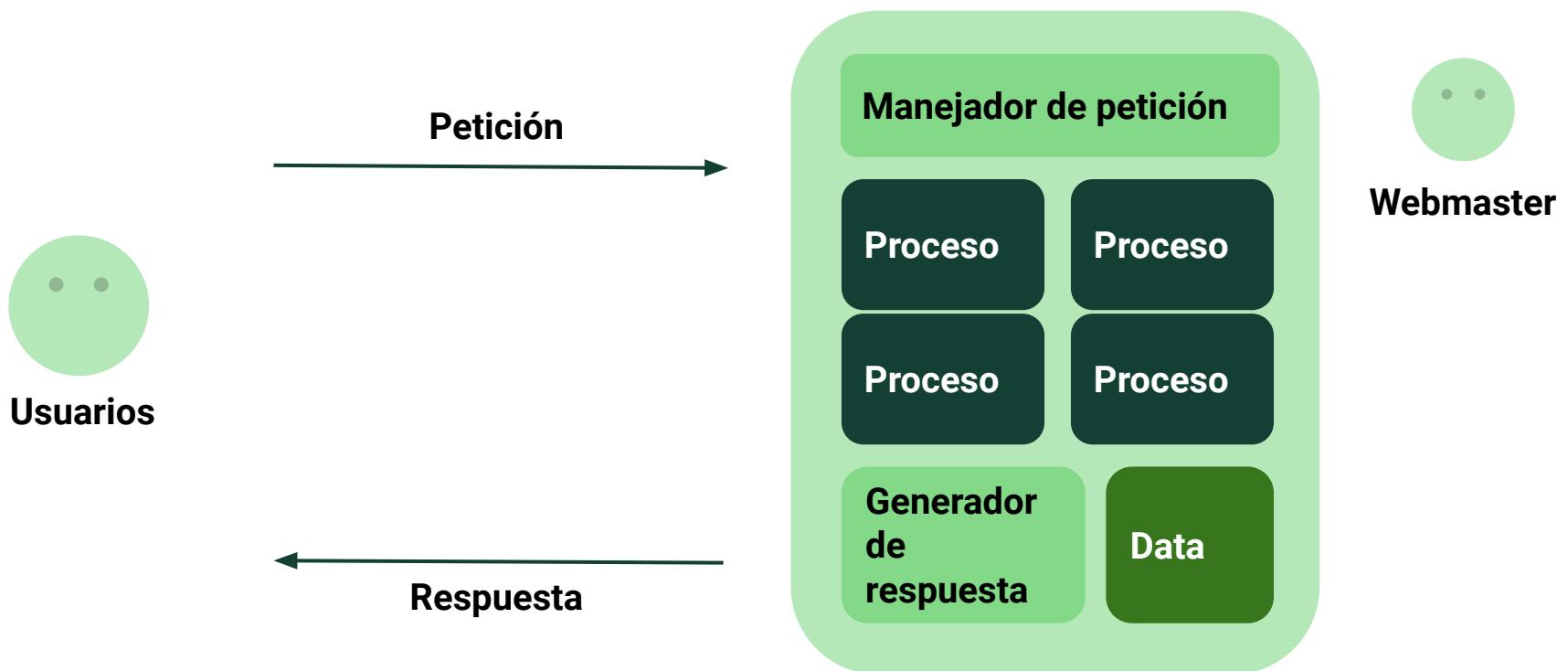
Conceptos base



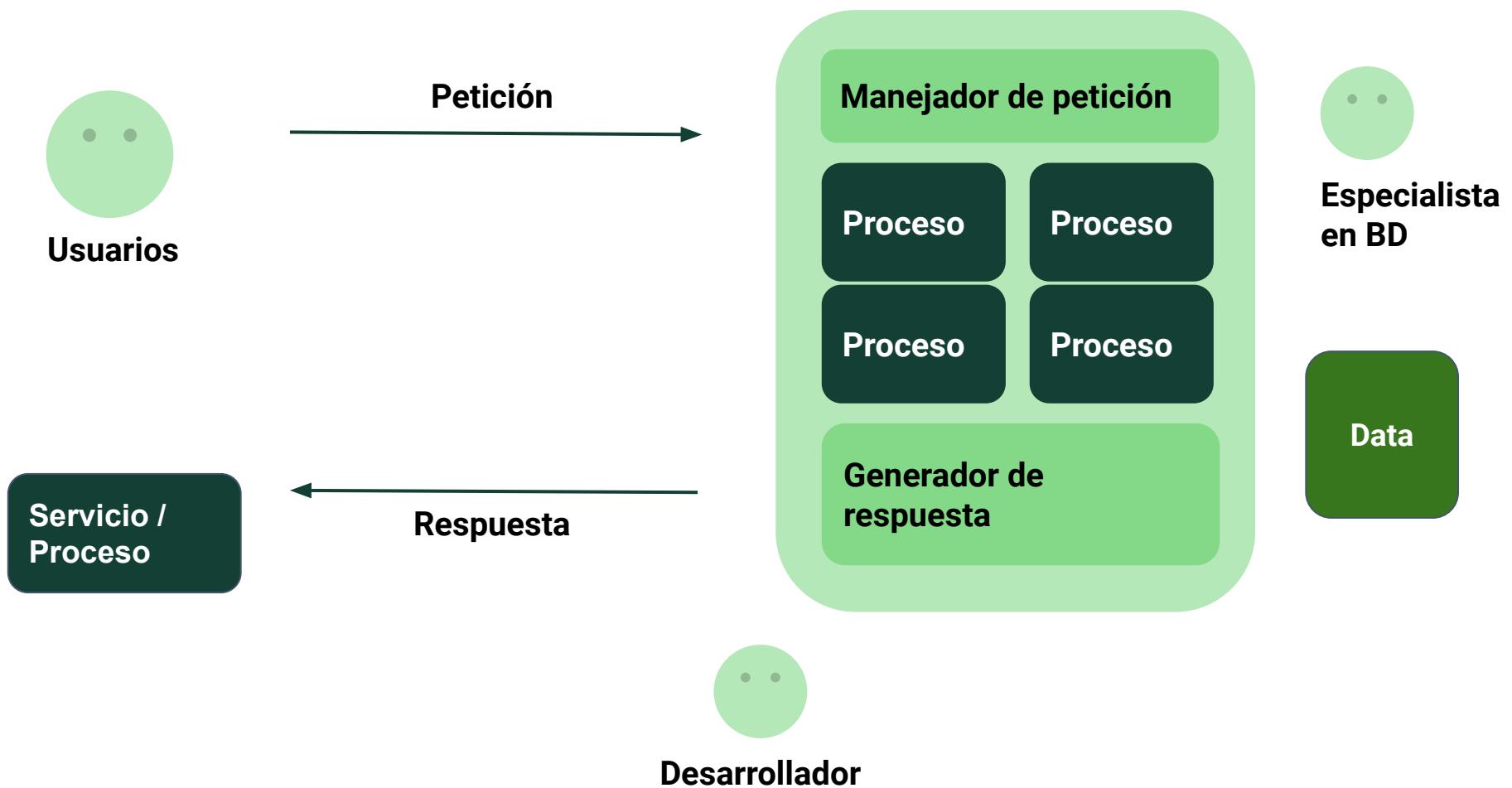
Sistemas en la antigüedad



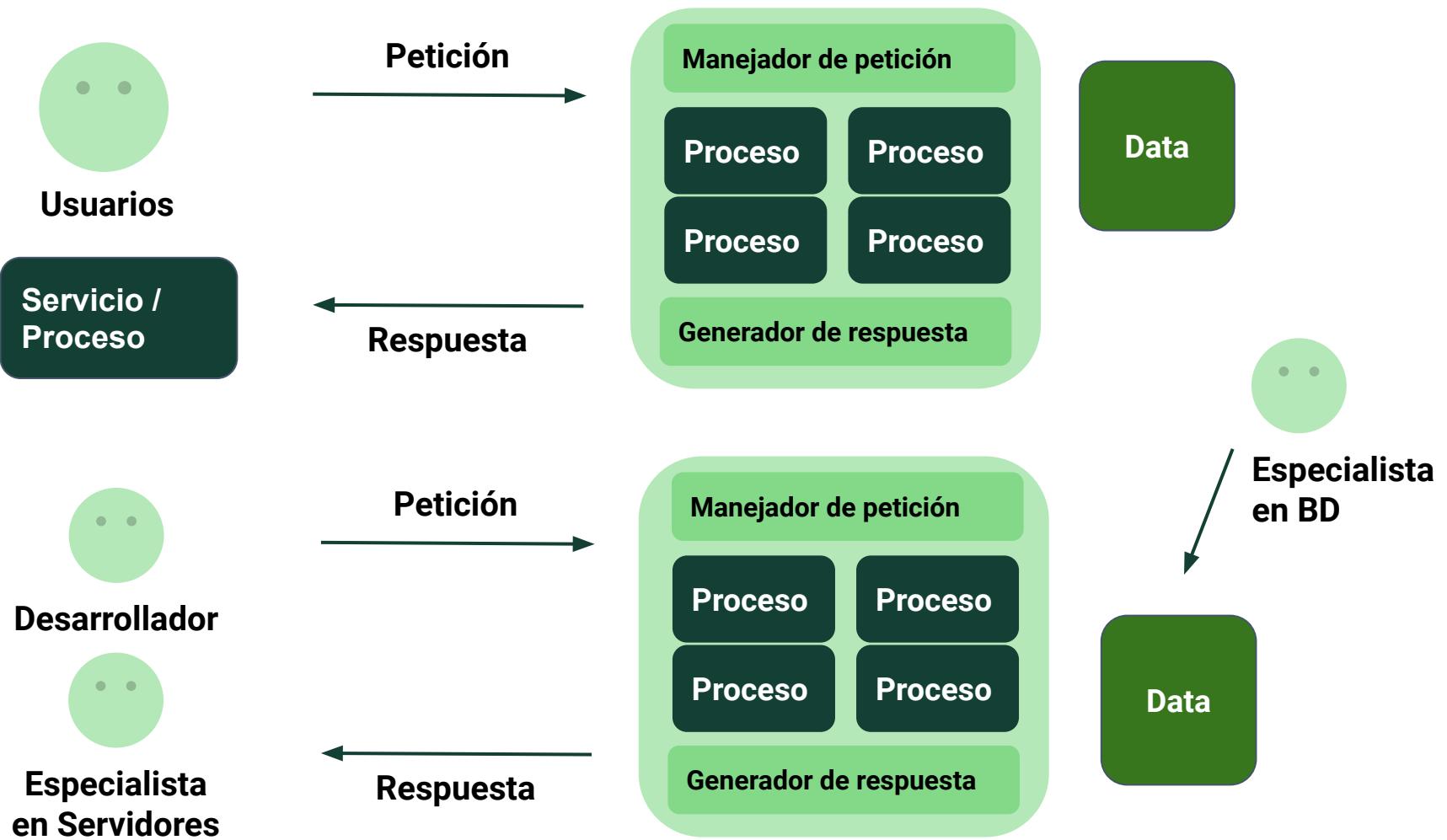
Sistemas en la antigüedad



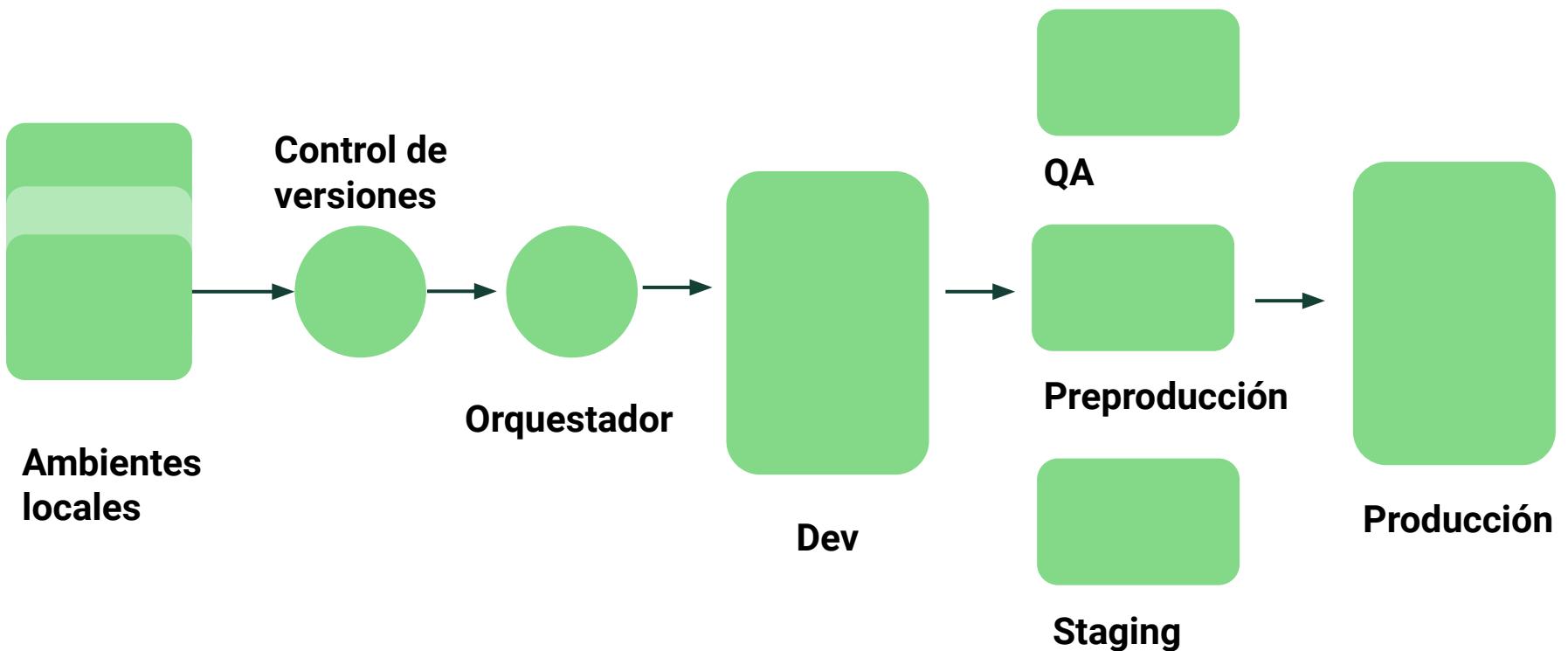
Sistemas en la antigüedad



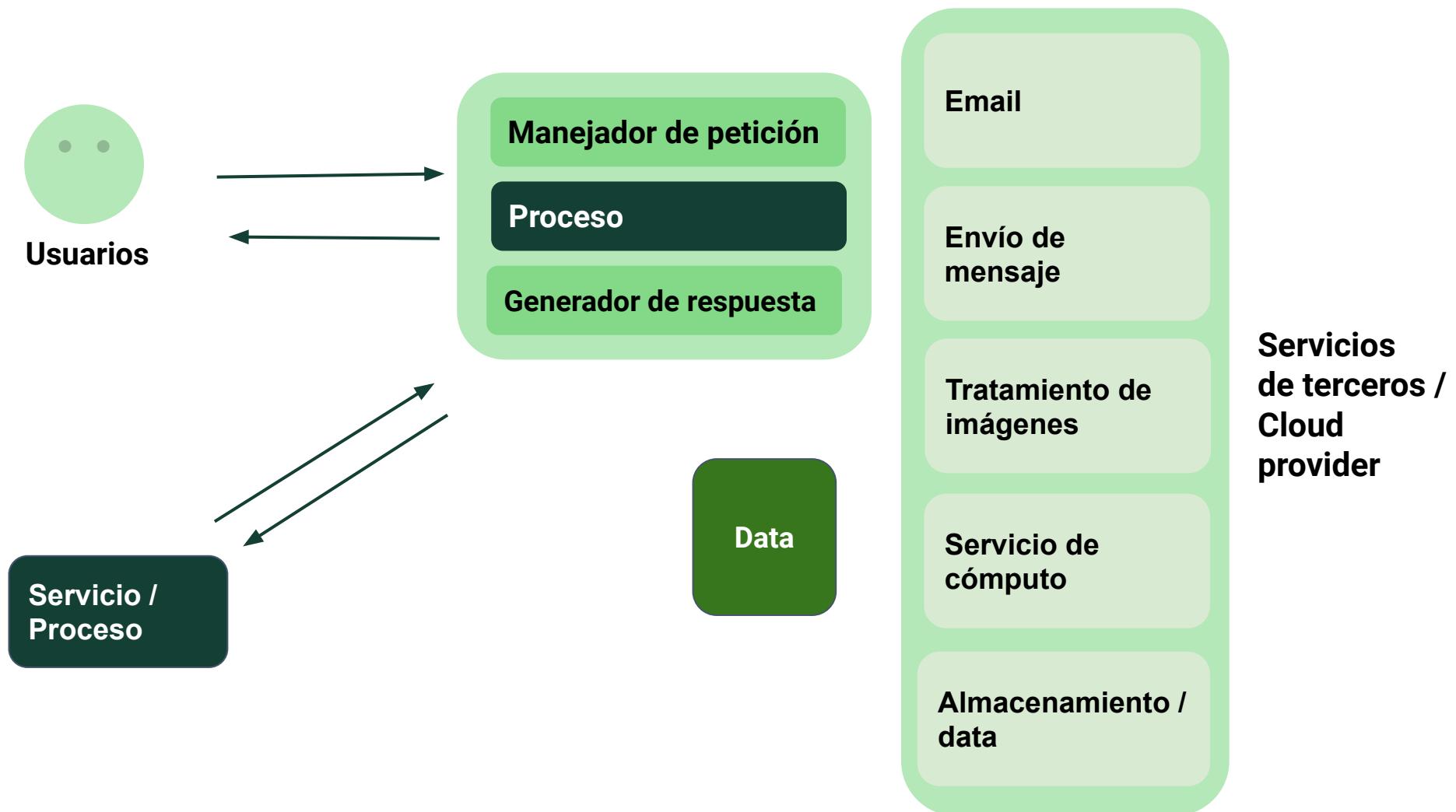
Sistemas en la antigüedad



Ambientes de producción



Sistemas en la antigüedad

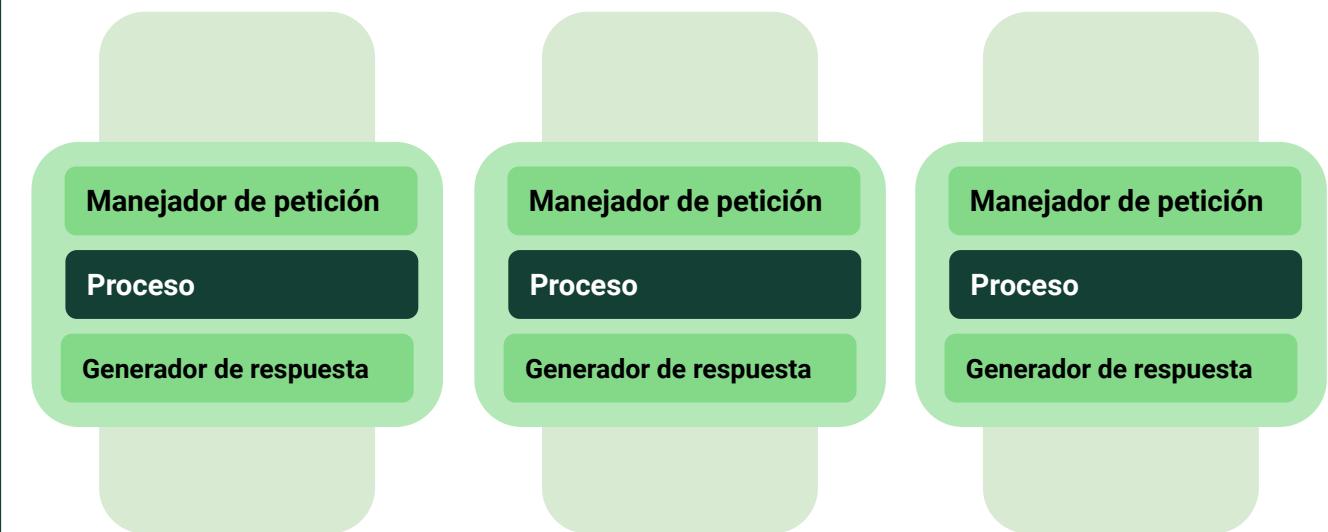


Sistemas en la antigüedad

Monolito

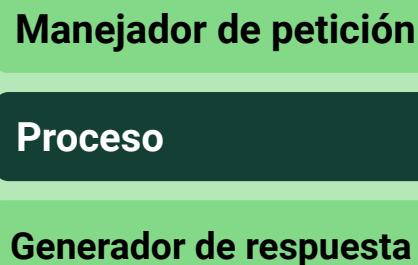
Manejador de petición
Proceso
Generador de respuesta

Microservicios

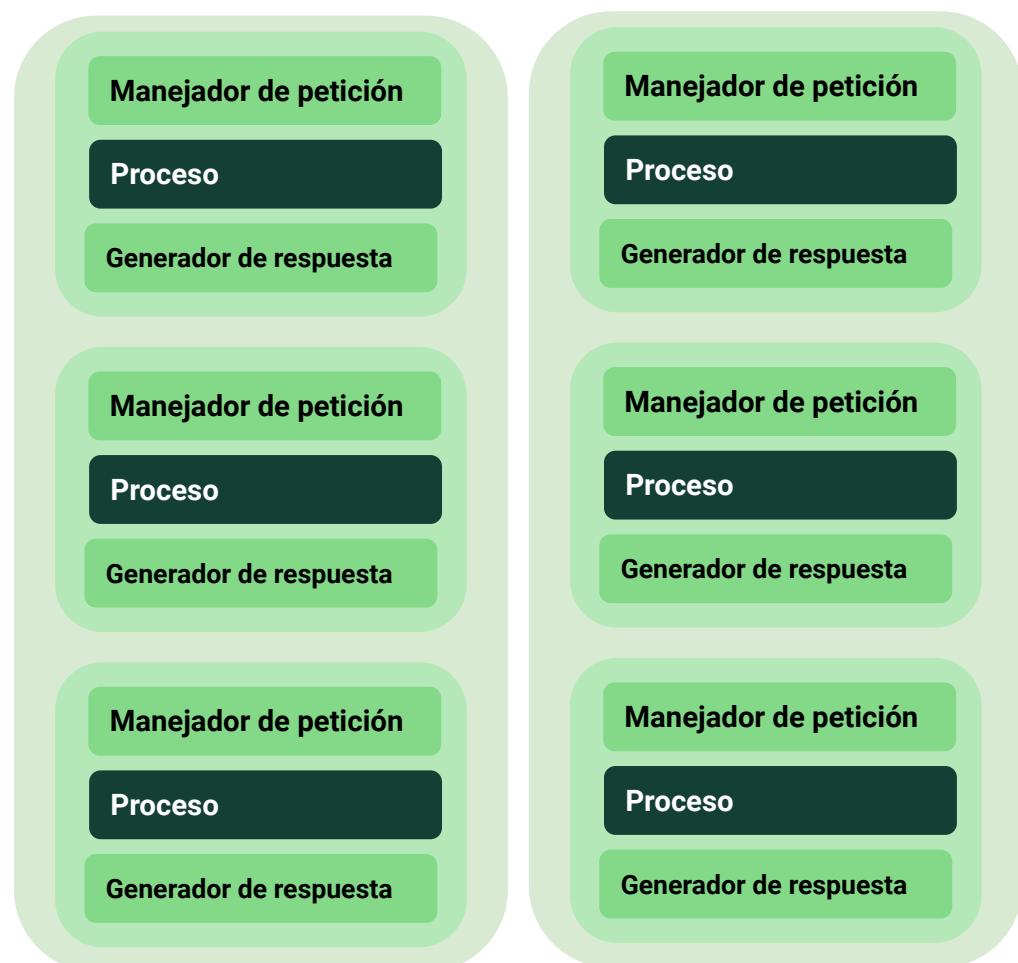


Sistemas en la antigüedad

Monolito



Microservicios con dockers



Resumen de la clase

- Usuarios y desarrolladores
- Monolitos y microservicios
- Ambientes (dev y prod)
- Dockers y Cloud provider

**¿Qué tiene que ver
esto con DevOps?**

¿Qué es DevOps?



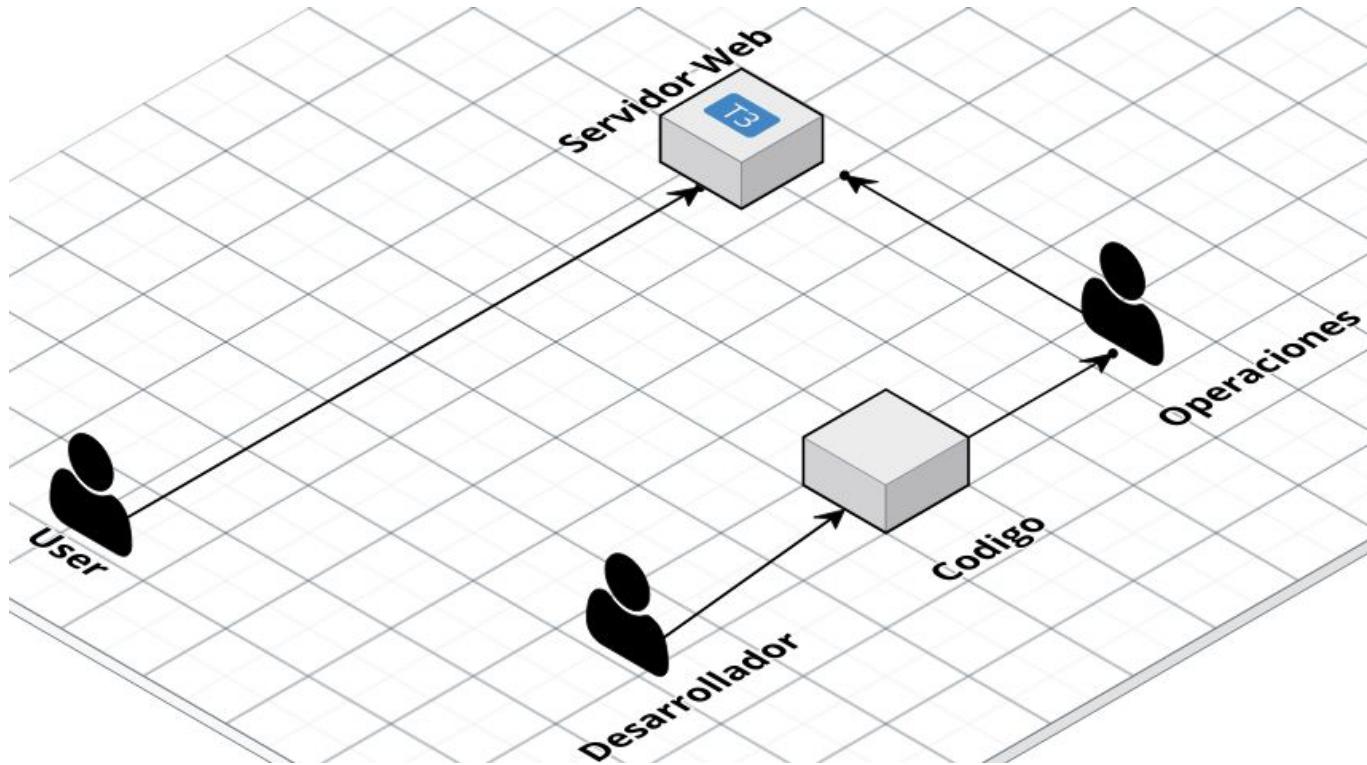
¿Qué es DevOps?

Cultura de mejora continua
en el desarrollo de software.

Origen de DevOps

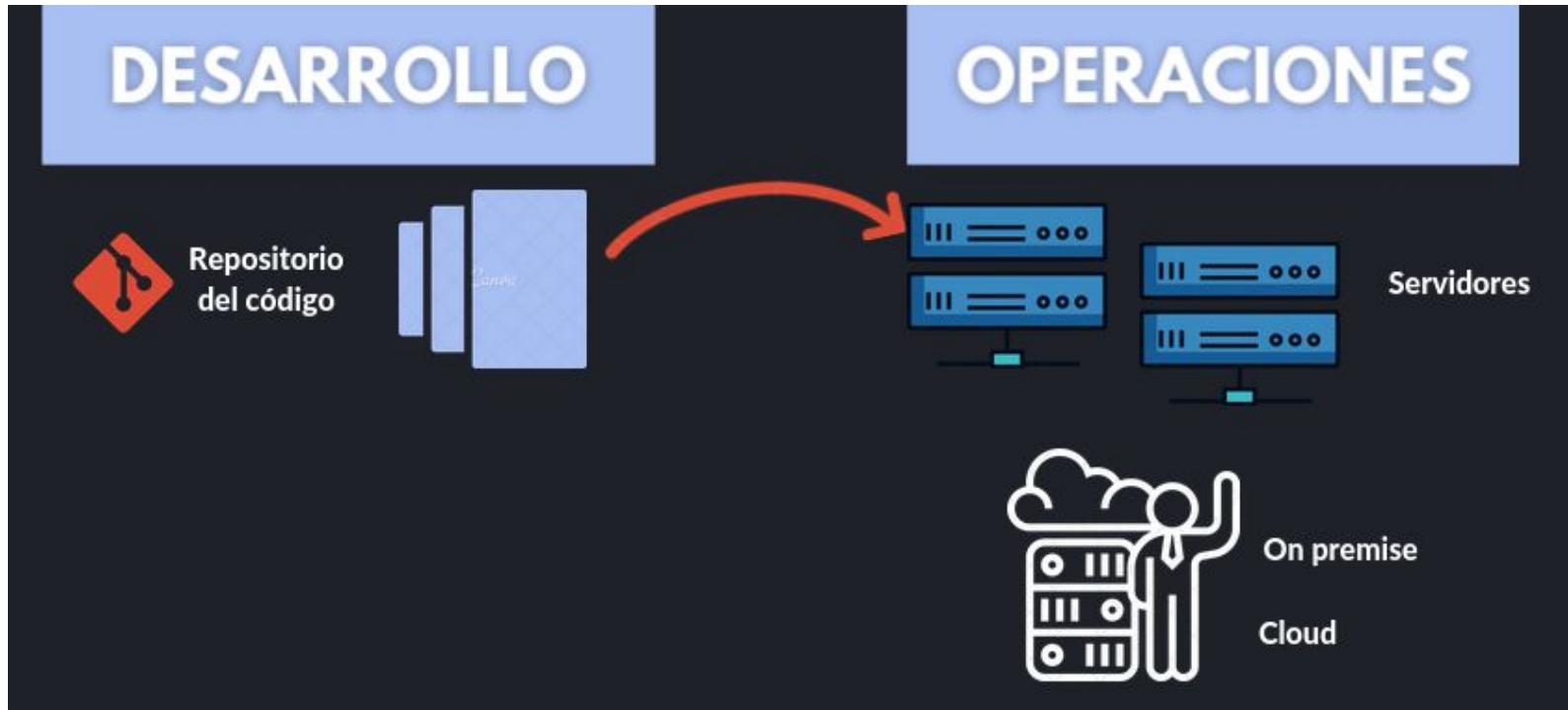
DevOps nace de la intención de ayudar a dos equipos de desarrollo de software a conseguir un objetivo más grande que ellos mismos.

Ejemplo



Servicio web con dos equipos

Ejemplo



DevOps es representado
por la línea roja

DevOps en las empresas

DevOps es una cultura que las empresas transformaron en rol.

Resumen de la clase

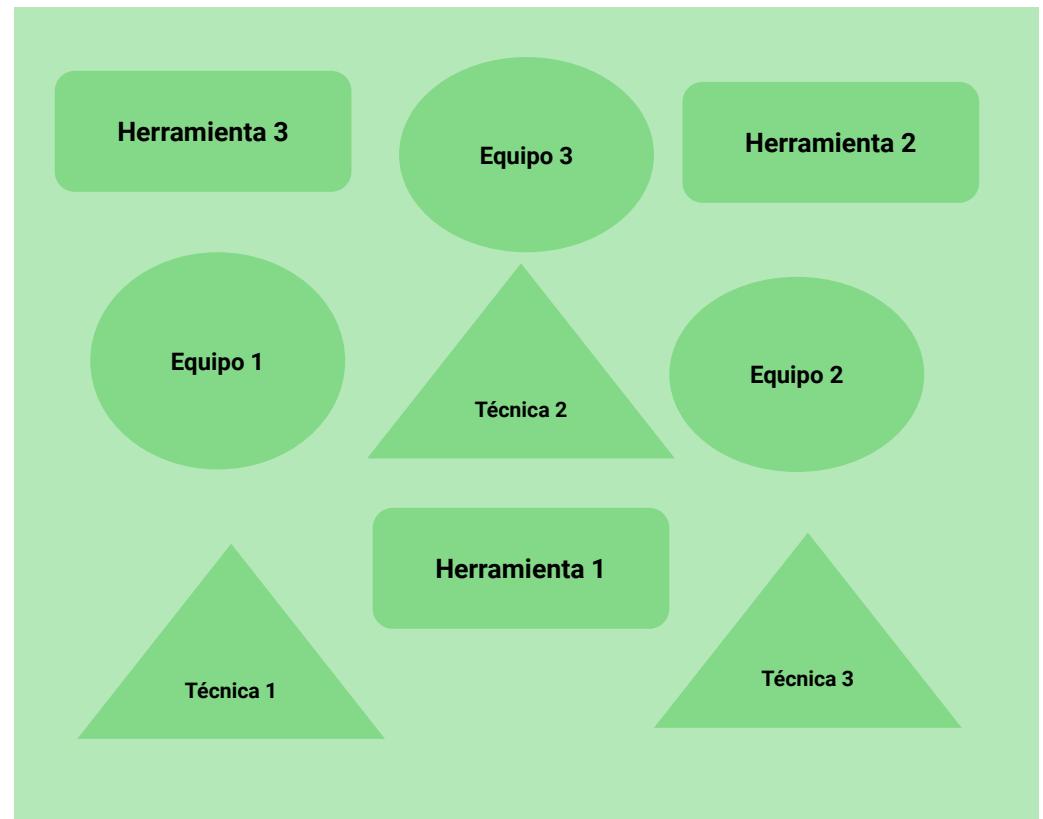
Es un rol que usa herramientas, técnicas y metodologías para ayudar a crear mejora continua en el desarrollo de software.

Niveles de madurez

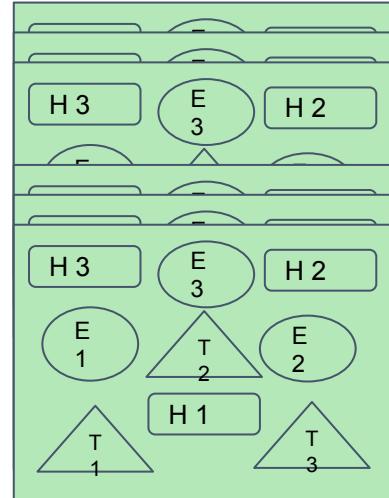


Madurez de la empresa

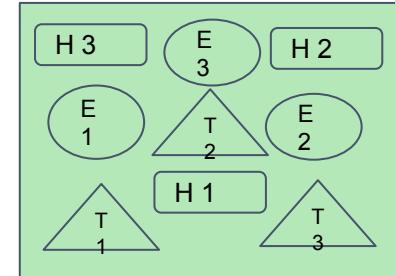
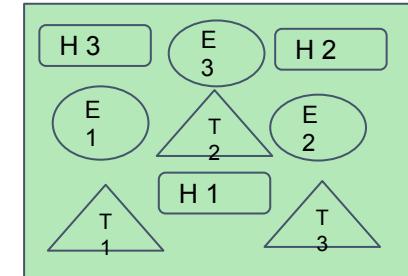
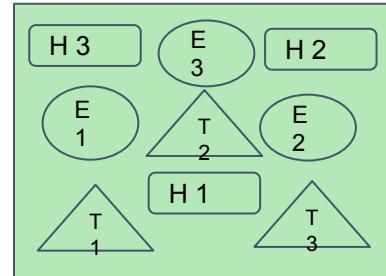
Las empresas son
como un ser vivo



Madurez de la empresa



Nuestro rol es afectado por la cultura de la empresa



Ejemplo de configuraciones

- Empresa iniciando en DevOps.
- Empresa que está migrando a cloud.
- Empresa que le falta asegurar la calidad de su software.
- Empresa iniciando un proyecto nuevo y desea un tener equipo de DevOps.

Resumen de la clase

Las empresas tienen características que las hacen únicas y afectan directamente como se desarrolla y viven los equipos dentro de ella, incluyendo al rol de DevOps.

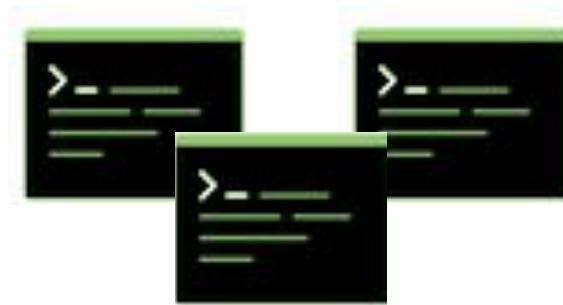
Tipo de empresa por servicios



Tipos de empresas

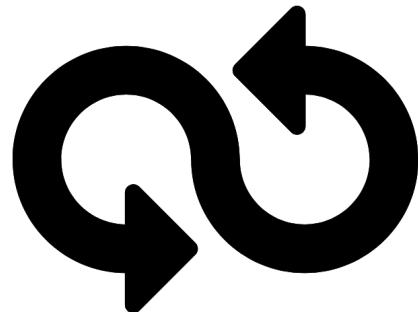


Mantiene un
software

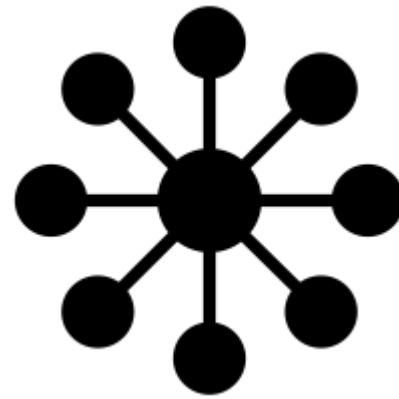


Mantiene muchas
aplicaciones

Tipos de empresas



Da servicios
de DevOps



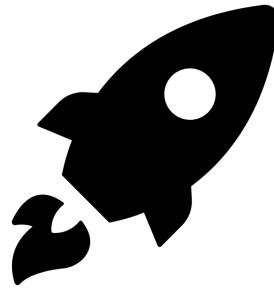
Mantiene muchas
infraestructuras

Resumen de la clase

Las herramientas y tus tareas del día a día dentro de una empresa puede cambiar según el producto o servicio que tenga la empresa.

Tareas principales del rol DevOps

Tareas frecuentes



Automatización
de despliegue



Automatización
de infraestructura

Tareas frecuentes



Implementación de
herramientas de
análisis y monitoreo

Implementación de
pruebas y seguridad

Resumen de la clase

Tus tareas están enfocadas
en ayudar y AUTOMATIZAR.

**¿Qué habilidades
y conocimientos
necesitas para
resolver esas tareas?**

Habilidades y Conocimientos



Recordatorio

- Estás en el medio de muchos equipos y necesitas comunicarte con ellos.
- Tienes que entender y comunicarte con los equipos de seguridad, desarrollo, infraestructura y testing.

Administración de sistema

- Sistema de carpetas.
- Cómo gestionar paquetes y aplicaciones.
- Gestión de variables de entornos.
- Manipulación de archivos.

Programación

- Debes saber programar (nivel básico).
- Conocer comandos de construcción de apps (builds).
- Cómo leer logs de errores.

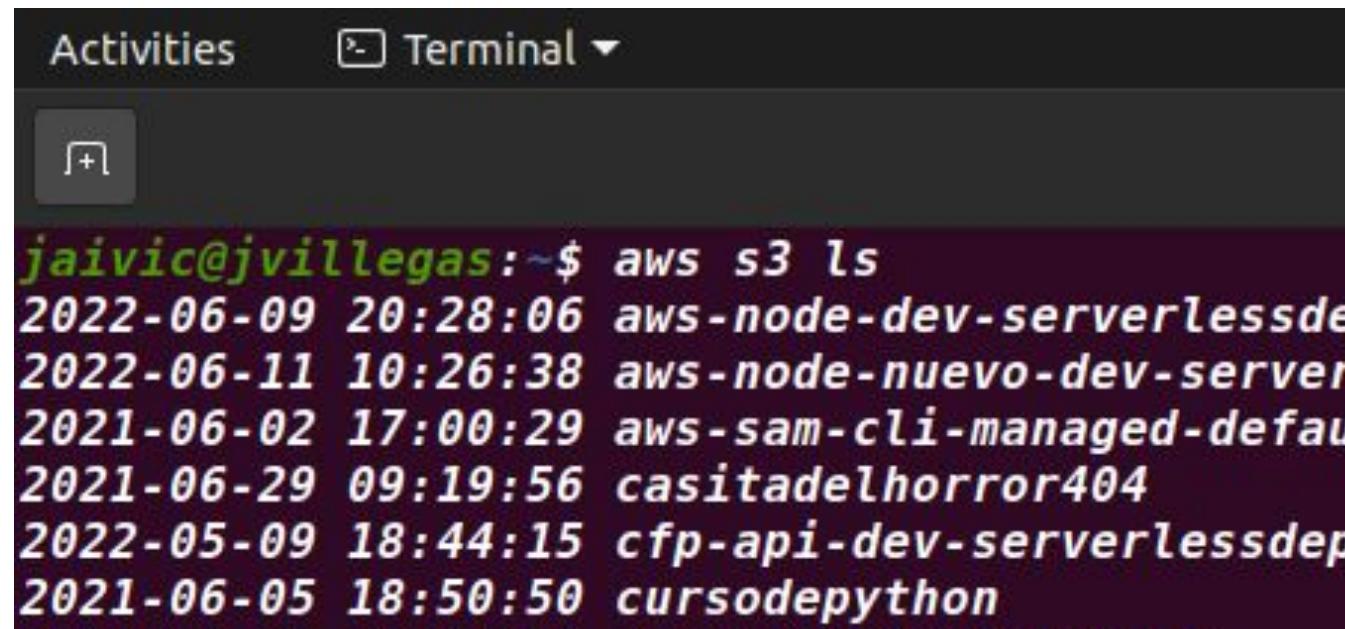
Servidores Web

- Conceptos Básicos (protocolos, puertos).
- Gestión de servidores (crear, leer logs, test).

Servicios de la nube

- Servicios básicos (VM, redes, storage).
- Conocimiento sobre gestión de permisos.
- CLI cloud.

Ejemplo de CLI cloud



A screenshot of a Linux desktop environment showing a terminal window. The terminal title is "Terminal". The user has run the command "aws s3 ls" and is viewing the output which lists several S3 buckets. The output is as follows:

```
jaivic@jvillegas:~$ aws s3 ls
2022-06-09 20:28:06 aws-node-dev-serverlessde...
2022-06-11 10:26:38 aws-node-nuevo-dev-server...
2021-06-02 17:00:29 aws-sam-cli-managed-defau...
2021-06-29 09:19:56 casitadelhorror404
2022-05-09 18:44:15 cfp-api-dev-serverlessdep...
2021-06-05 18:50:50 cursodepython
```

Docker

- Comandos de Docker.
- Crear Dockerfile.
- Saber manipular Docker registry.
- Leer logs de Docker.

Ejemplos de comandos

run

pull

build

push

Proceso de despliegue

- Herramientas de orquestación.
- Manipulación de ambientes.
- Manipulación de variables
de entorno y secretos.
- Leer logs.

¿Qué herramienta aprender?



Resumen de la clase

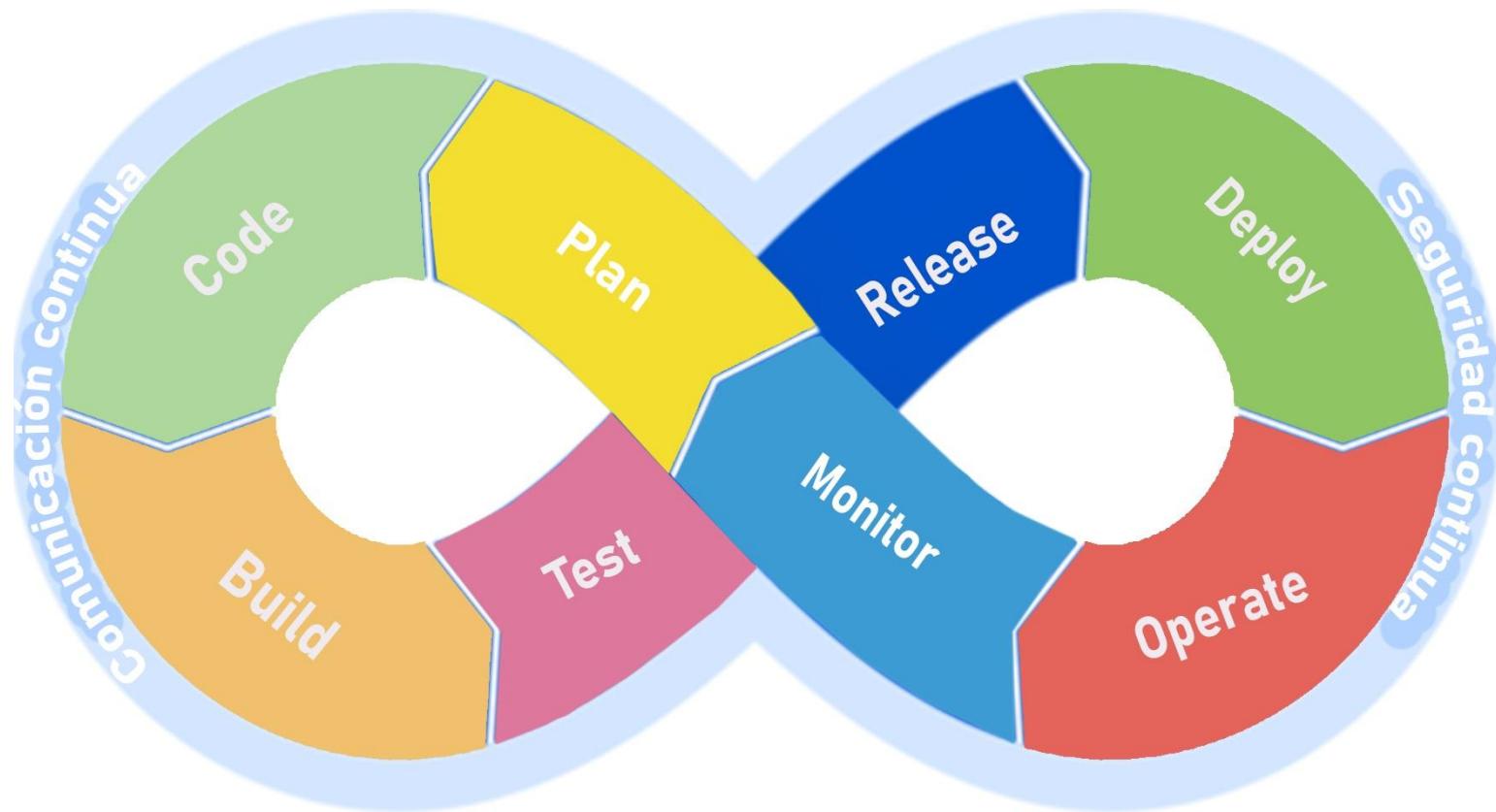
Todo se resume en entender cómo y dónde se desenvuelven las aplicaciones.

Y entender el conjunto de herramientas que la empresa está usando.

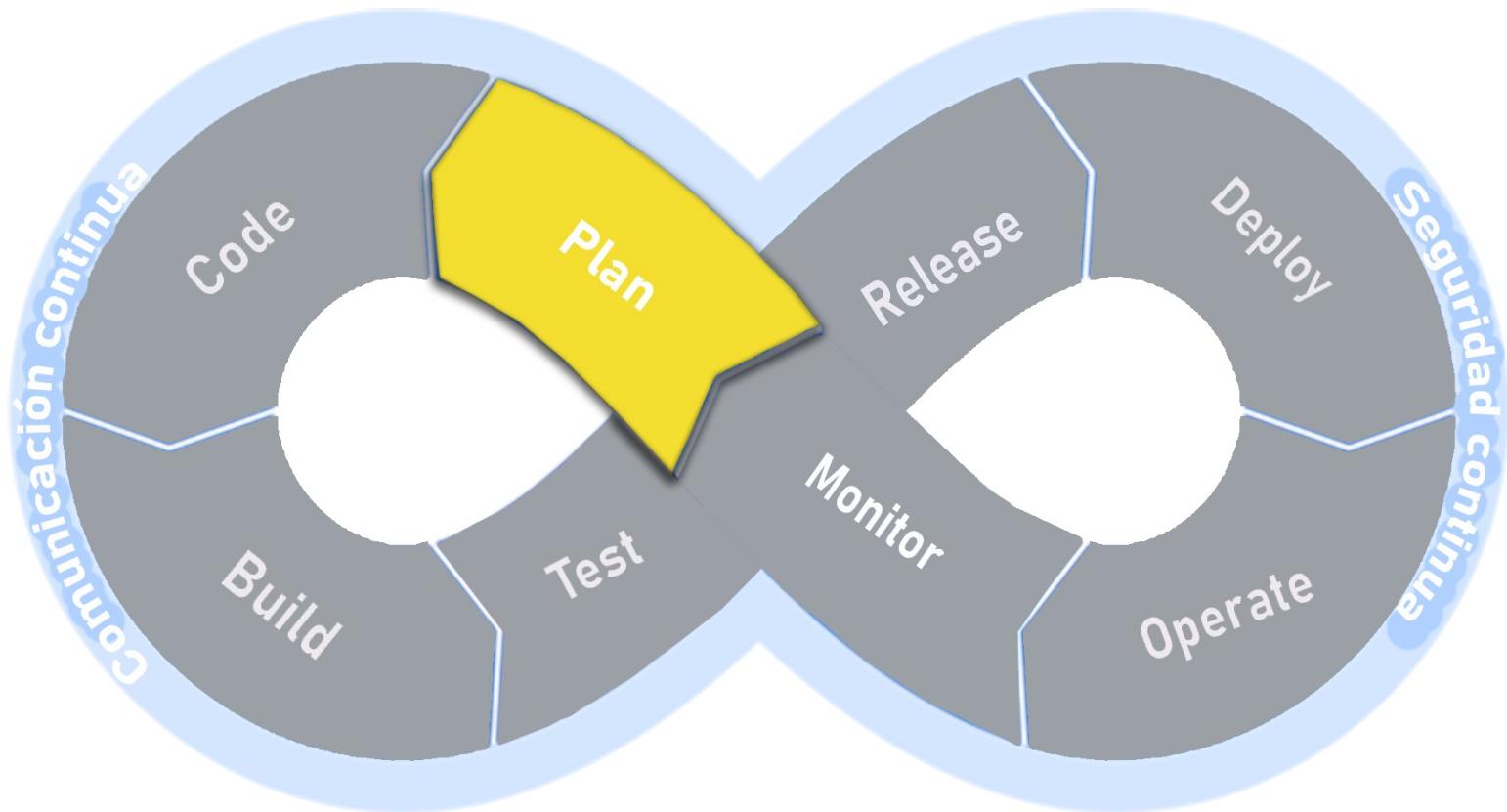
Ciclo de vida del software



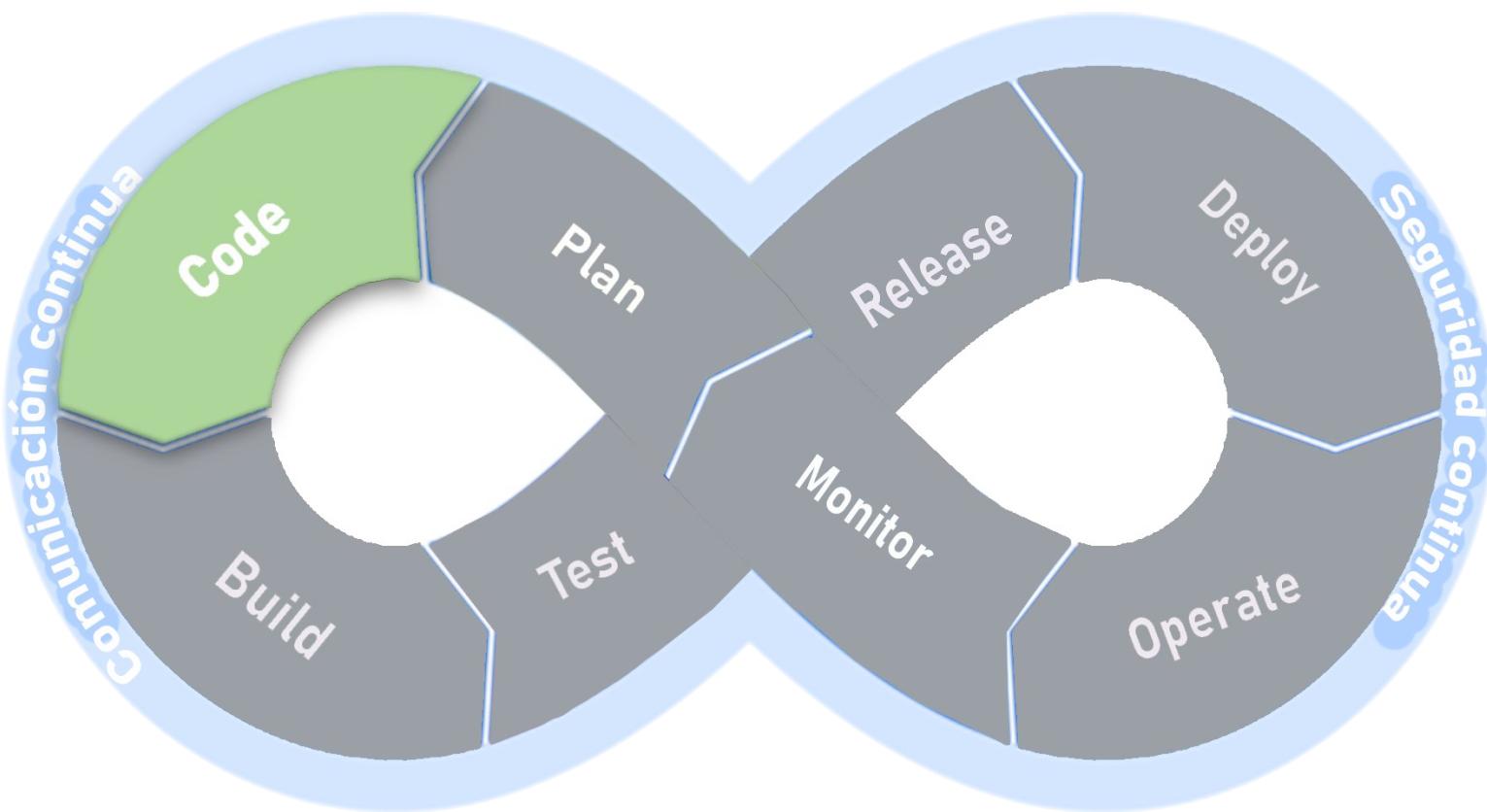
Ciclo de vida



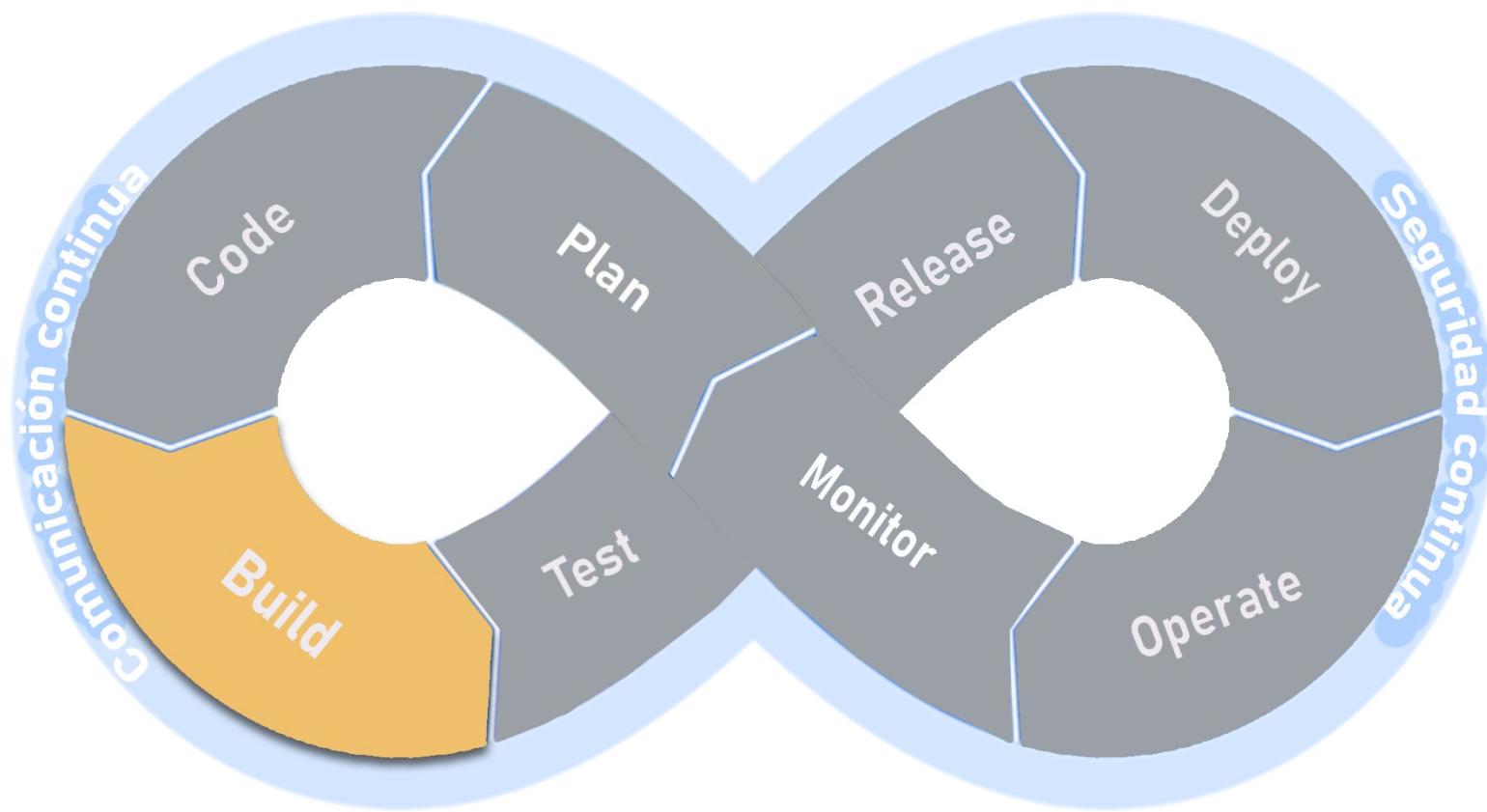
Ciclo de vida



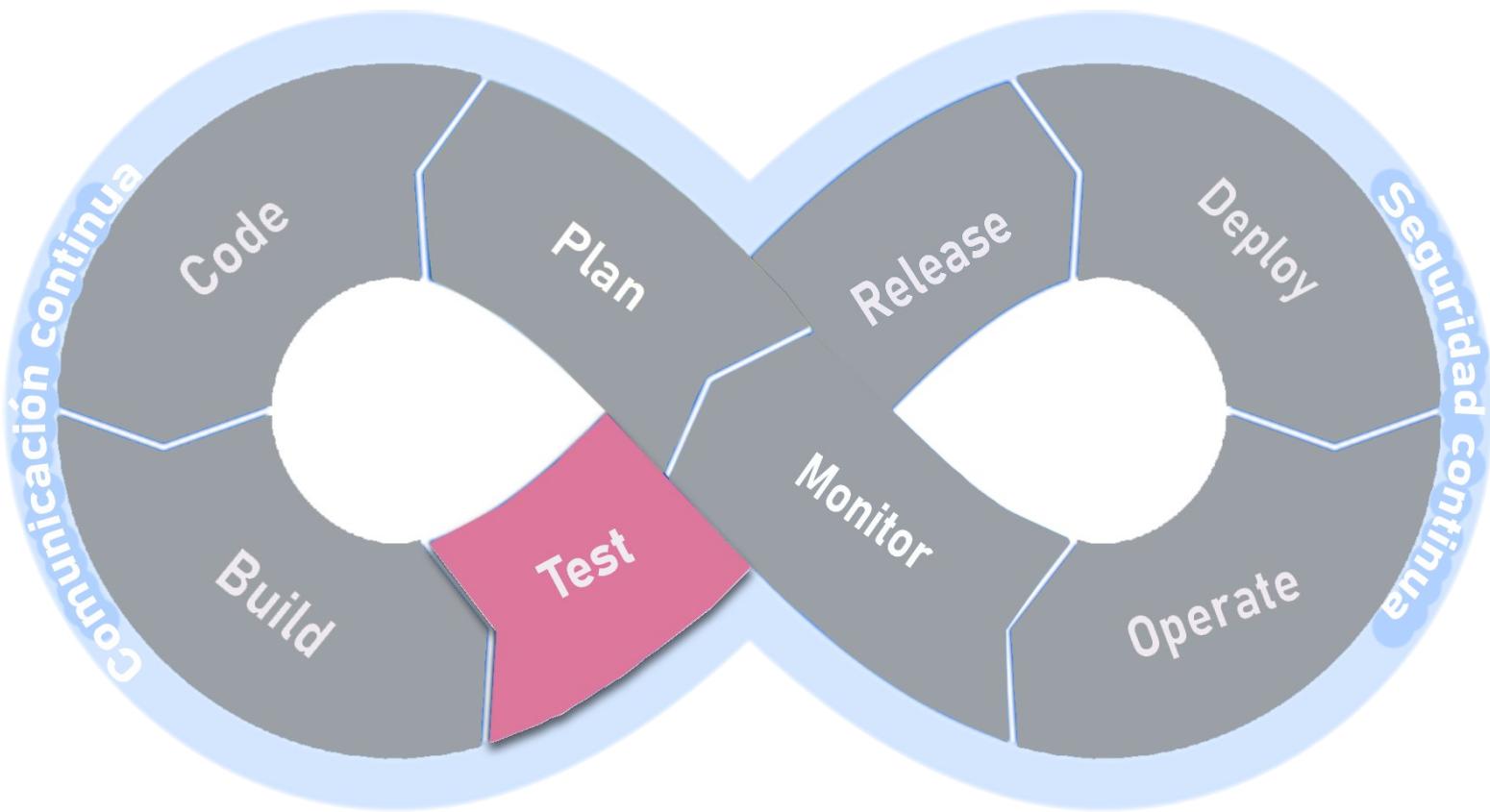
Ciclo de vida



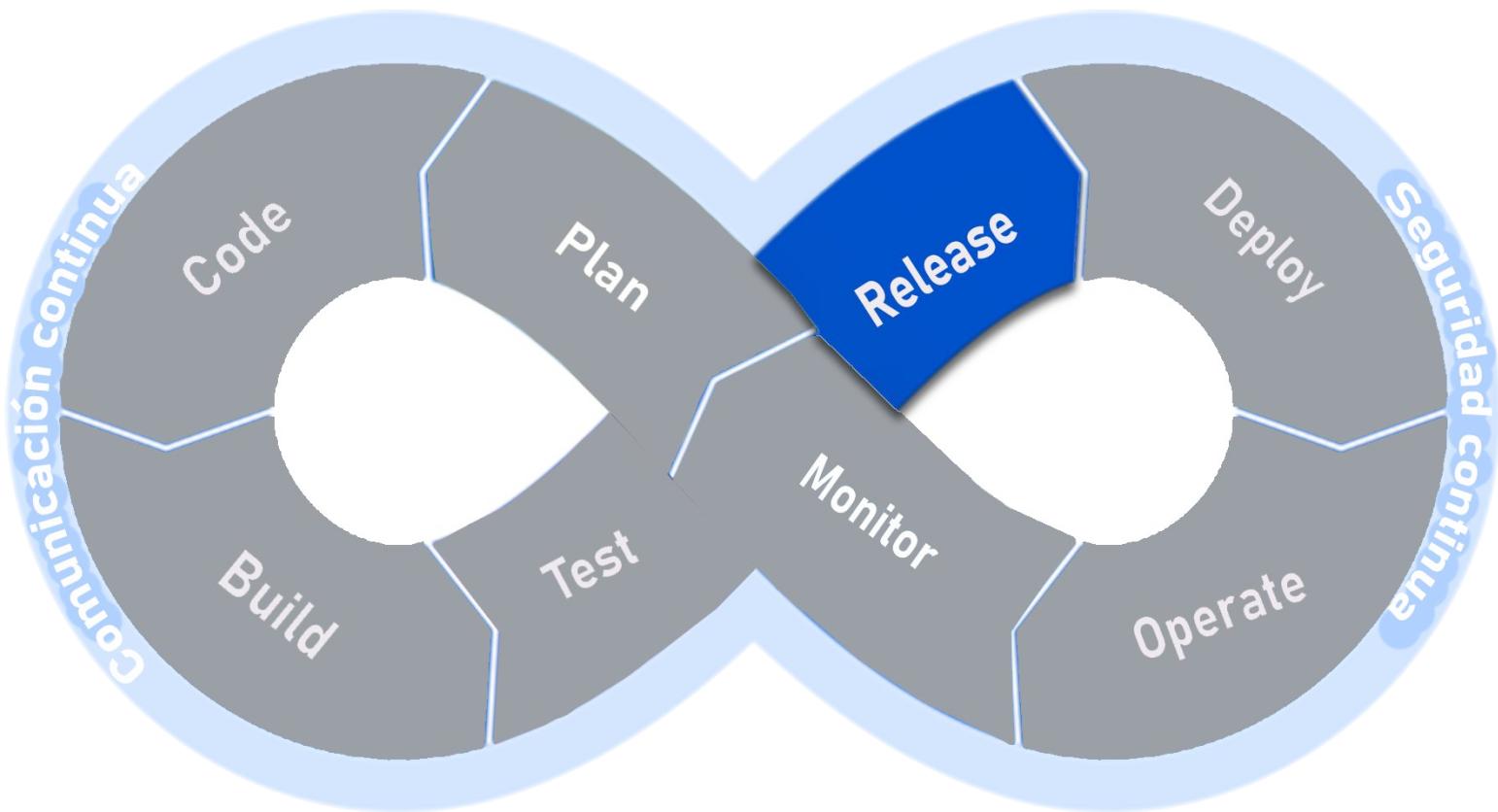
Ciclo de vida



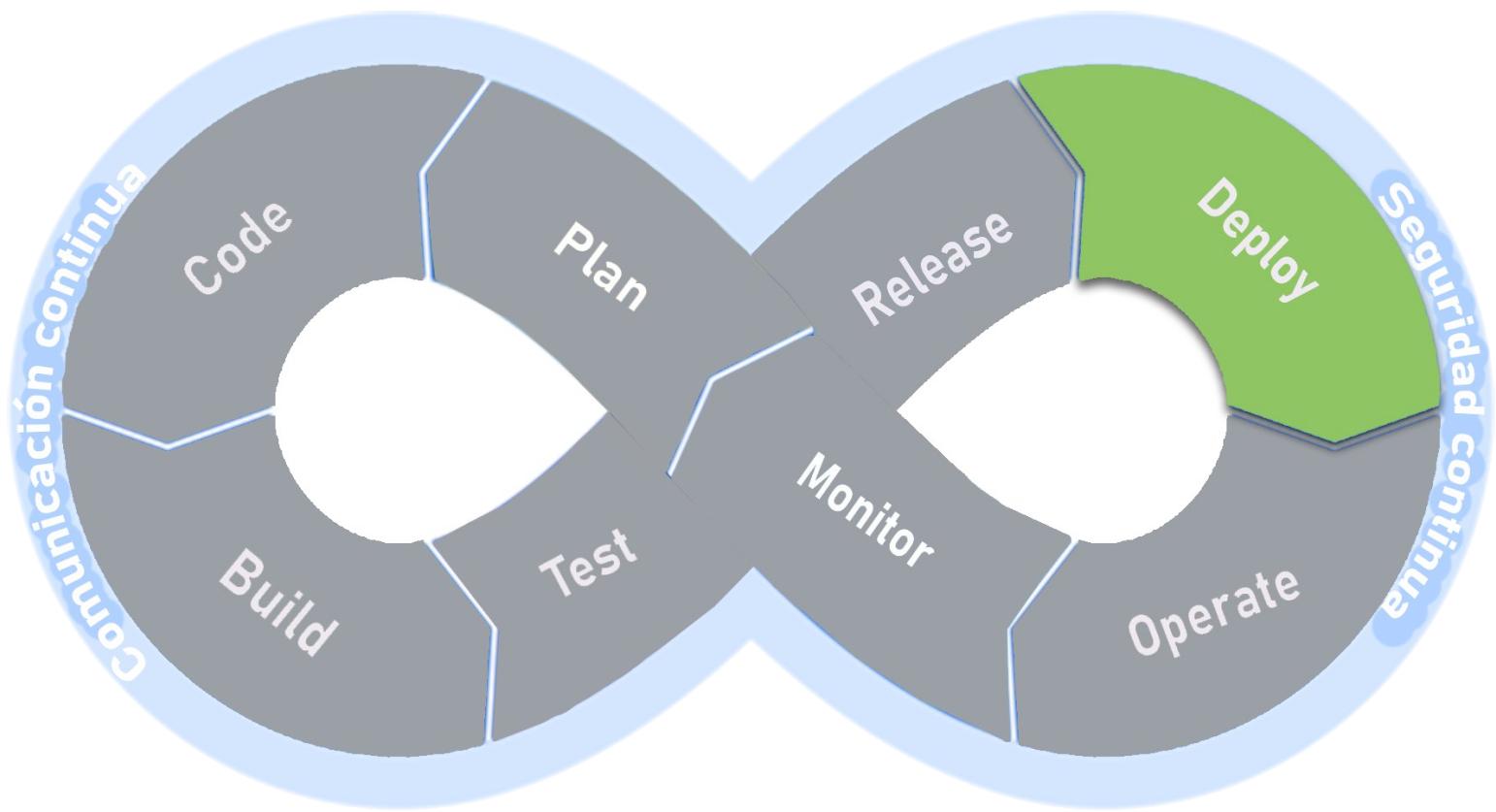
Ciclo de vida



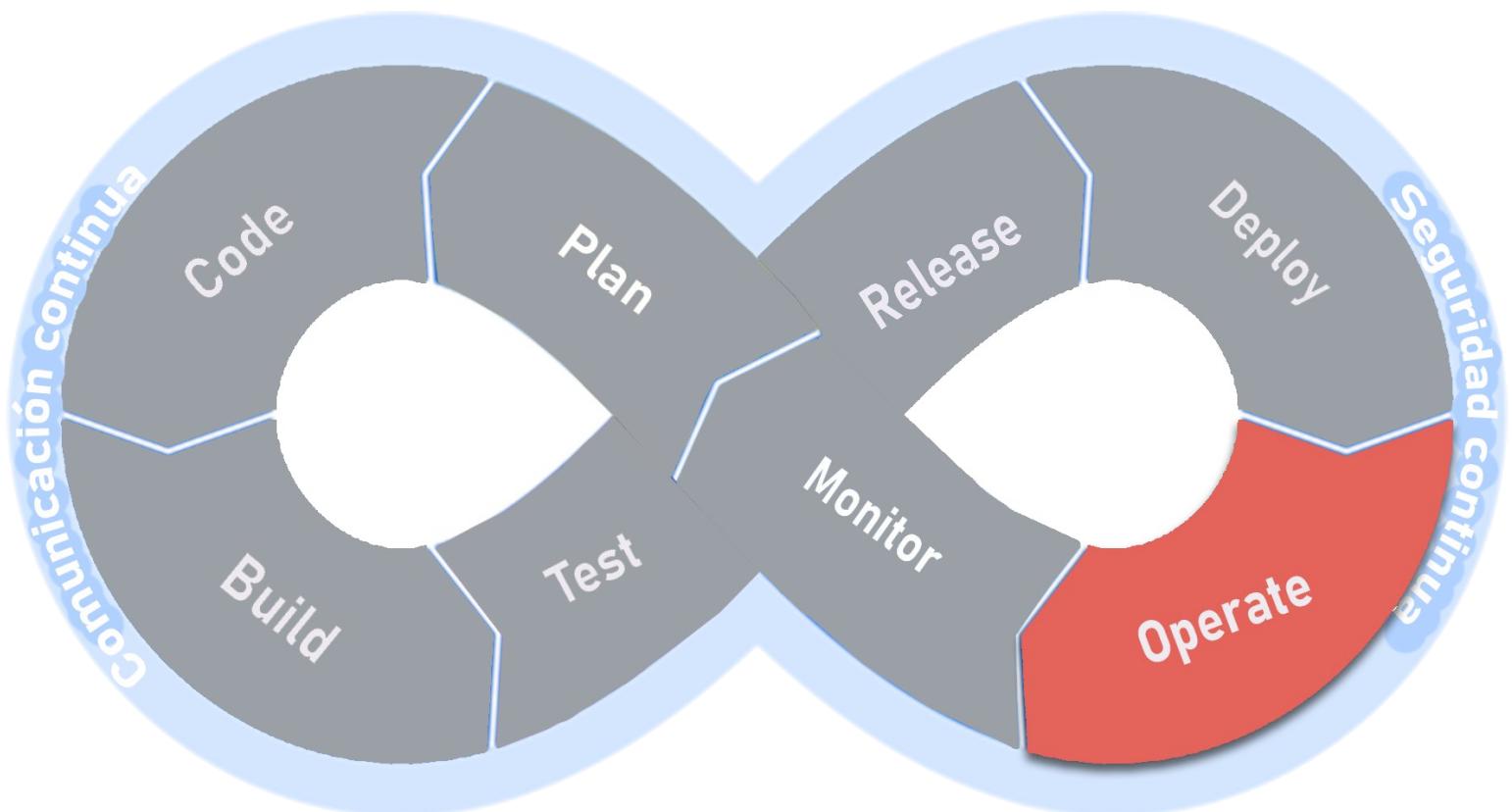
Ciclo de vida



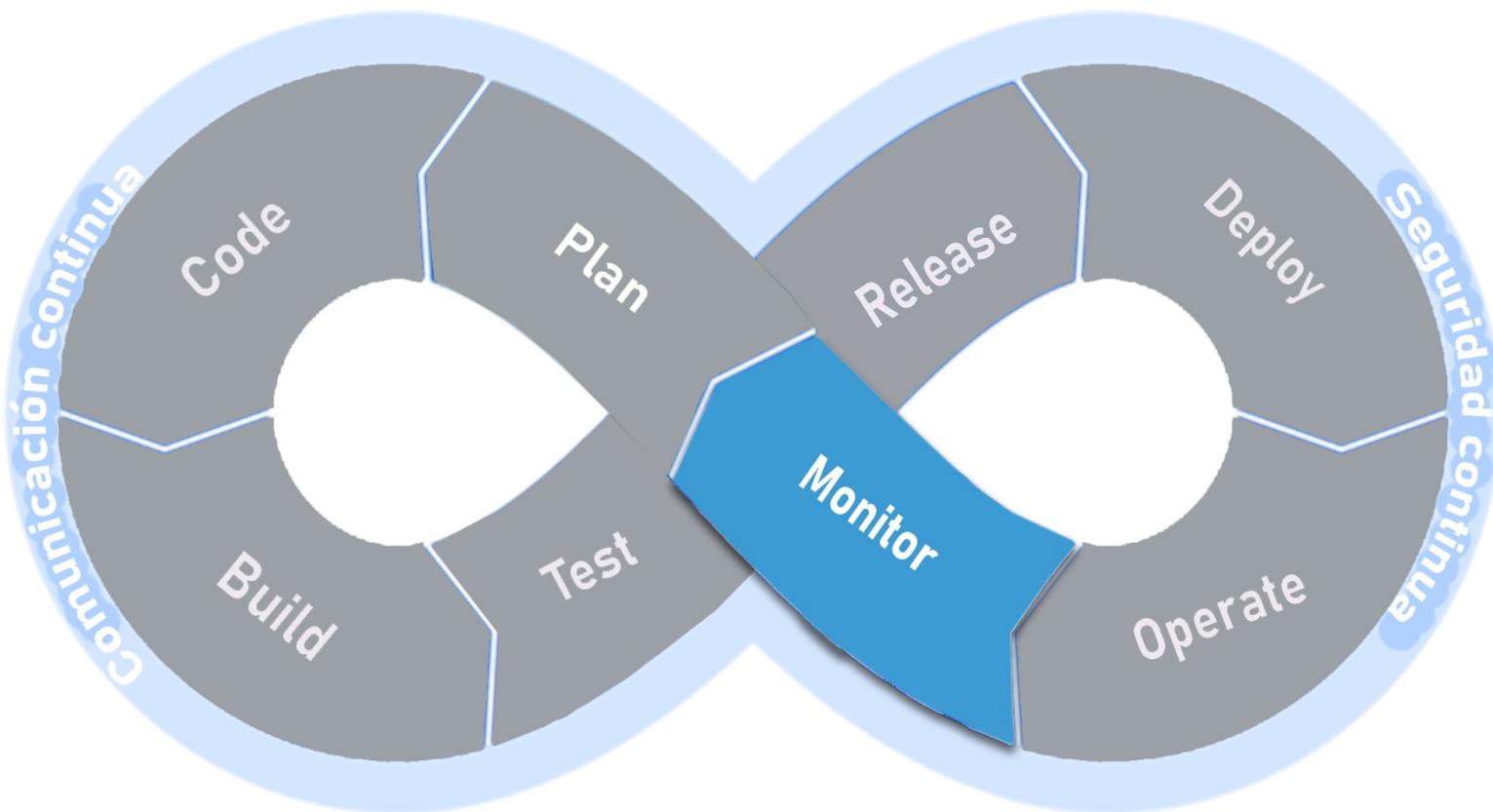
Ciclo de vida



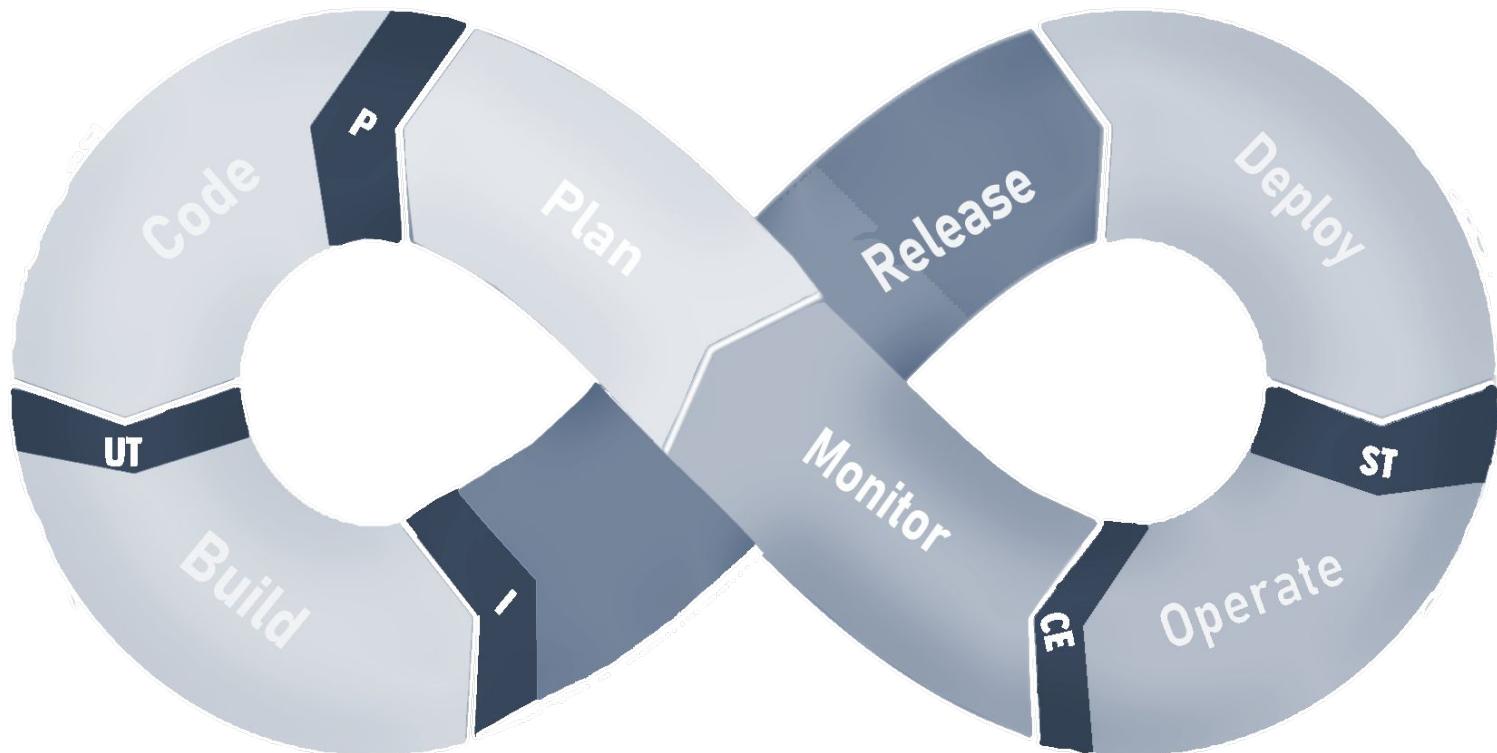
Ciclo de vida



Ciclo de vida



Fases de pruebas



Resumen de la clase

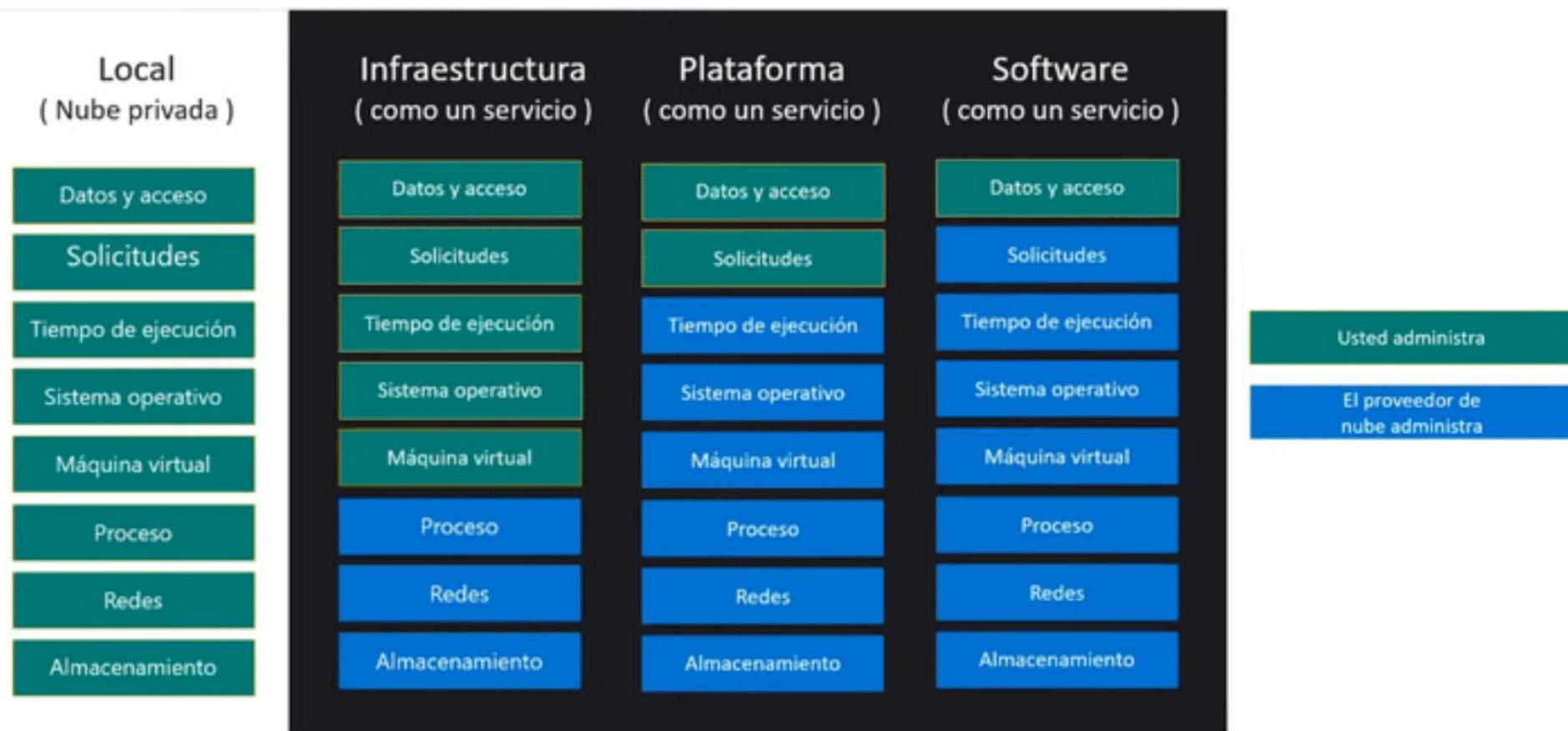
El ciclo es una forma de representar las fases de desarrollo, pero no siempre reflejan la realidad.

Sé flexible.

Cloud



Modelo de responsabilidad



Modelo de responsabilidad

	Compute				Data		
	IaaS	PaaS	CaaS	FaaS	Object Storage	Relational	NoSQL
Amazon Web Services	Elastic Compute Cloud (EC2)	Elastic Beanstalk	Elastic Kubernetes Service (EKS)	Lambda	Simple Storage Service (S3)	Relational Database Service (RDS)	DynamoDB, ...
Microsoft Azure	Virtual Machines	App Service	Azure Kubernetes Service (AKS)	Functions	Blobs	SQL Database	Cosmos DB, ...
Google Cloud Platform	Compute Engine	App Engine	Kubernetes Engine	Cloud Functions	Cloud Storage	Cloud SQL	Cloud Bigtable, ...

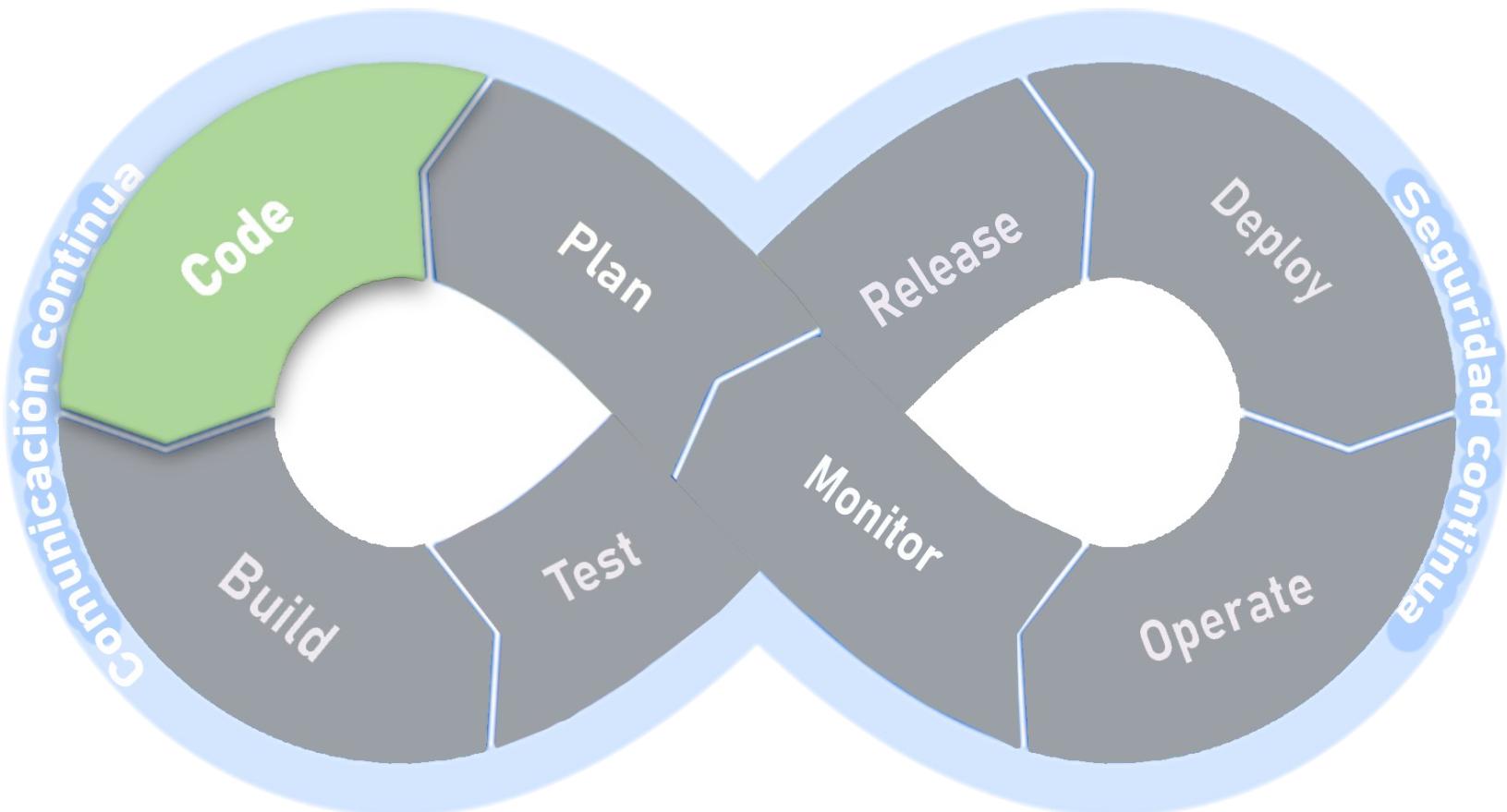
Resumen de la clase

- Recuerda que todos los servicios tienen responsabilidad compartida.
- Recuerda que cada proveedor de la nube tiene sus propios servicios y reglas, pero mantienen similitudes con los demás.

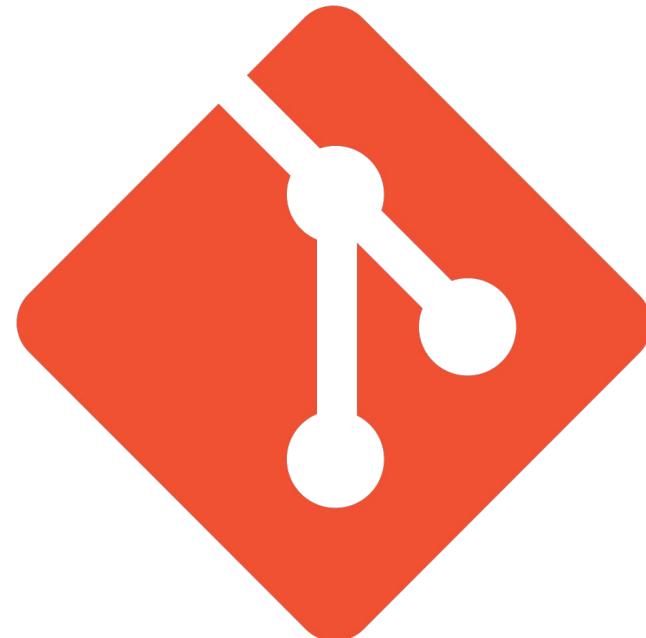
Herramientas de control de versiones



Fase de Code



Control de versiones



Git

Plataformas basadas en GIT

- GitHub
- Bitbucket
- Git Lab
- GPC Cloud Source Repositories
- AWS GitCommit
- Azure repo

Ejemplos de comandos

clone

pull

commit

push

Herramientas de Orquestación



Herramientas de Orquestación



Jenkins



AWS CodePipeline

AWS CodePipeline



AWS CodePipeline



AWS CodePipeline

Source

Build

Test

Production



AWS CodeCommit



AWS CodeBuild

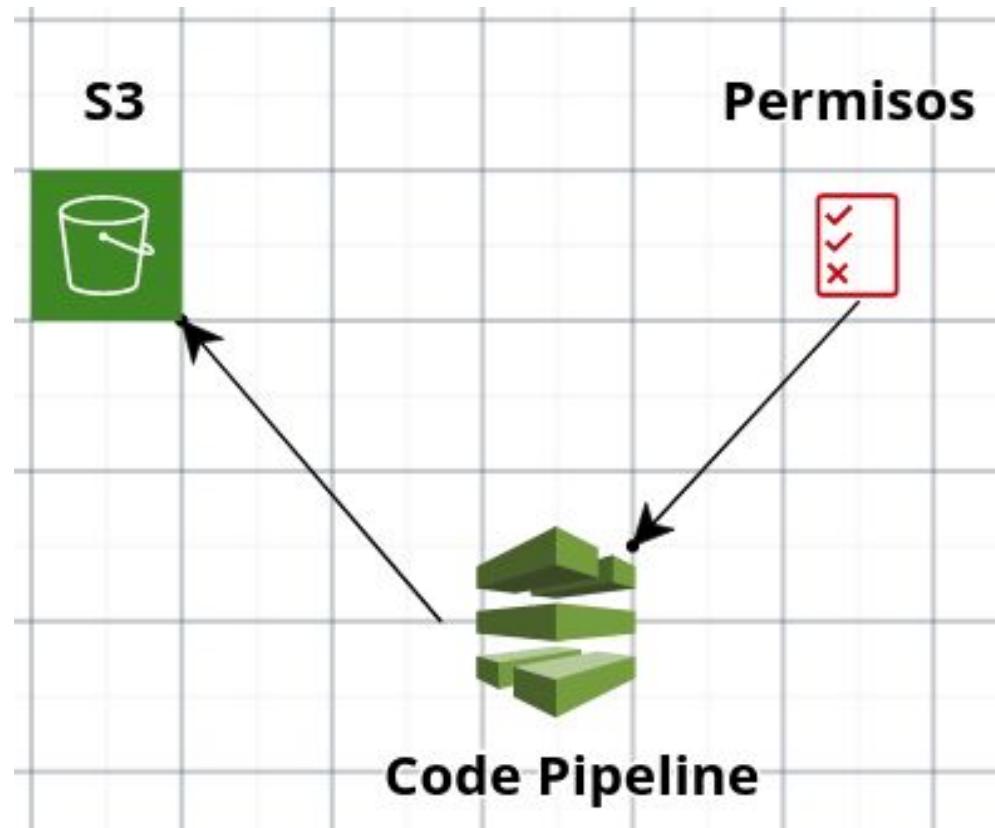


AWS CodeBuild
+ other tools

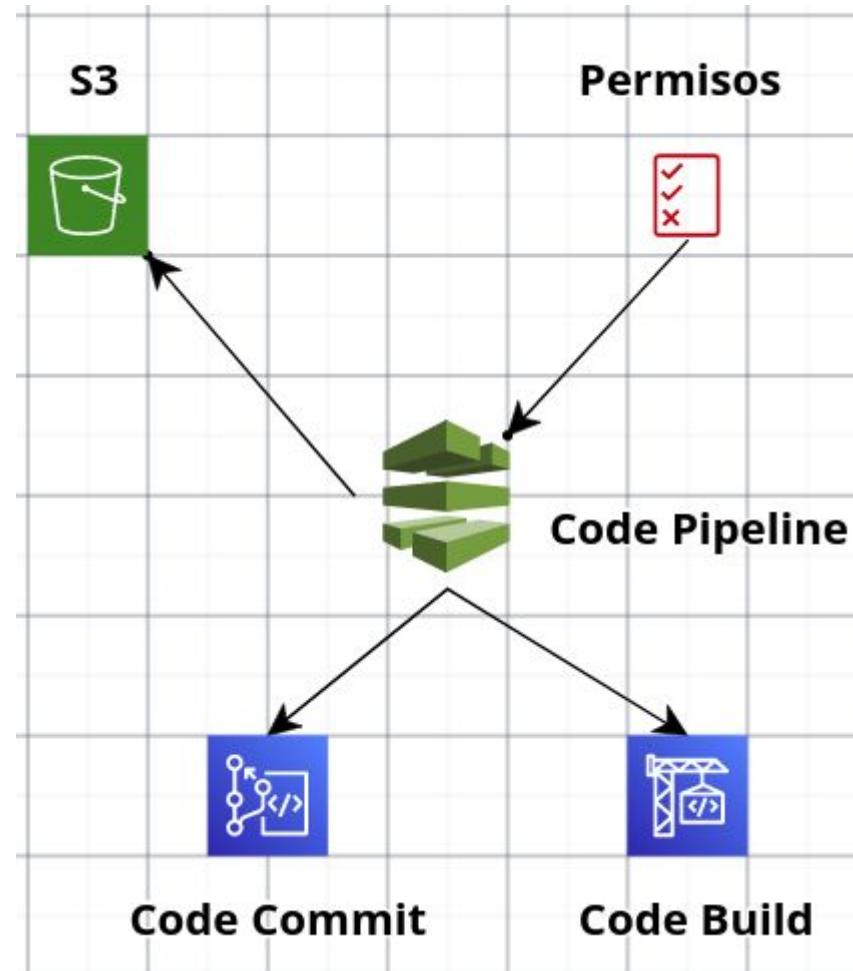


AWS CodeDeploy

AWS CodePipeline

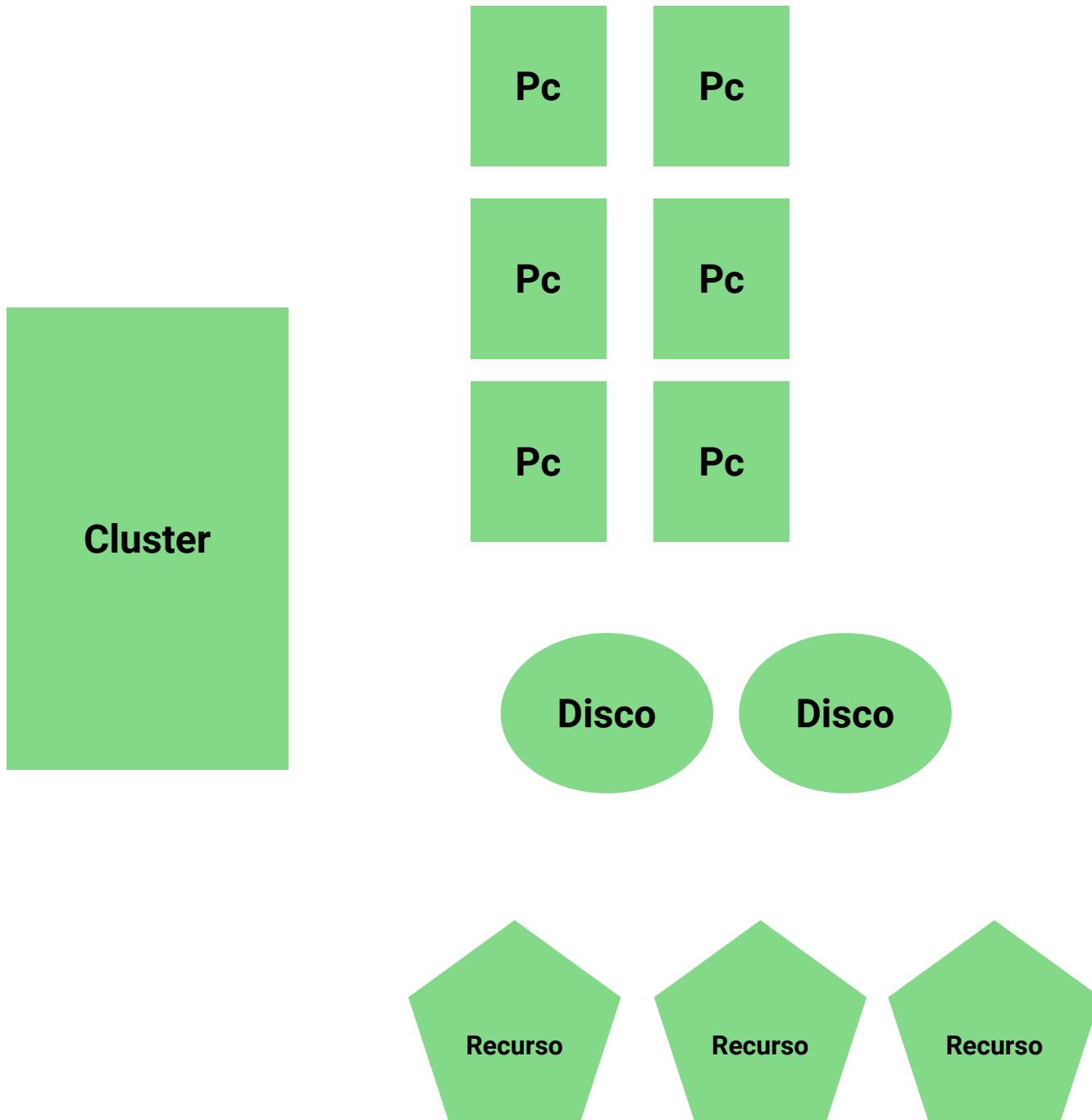


AWS CodePipeline

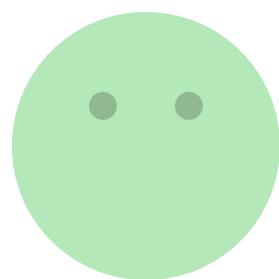


Cluster





Cluster



Usuario

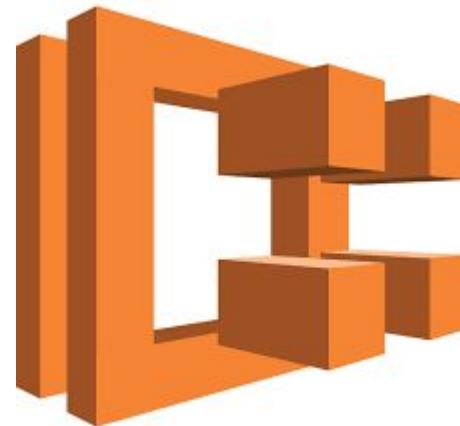


Cluster
Poder de cómputo = 6
Disco 2
Recurso capacidad 3

Clusters

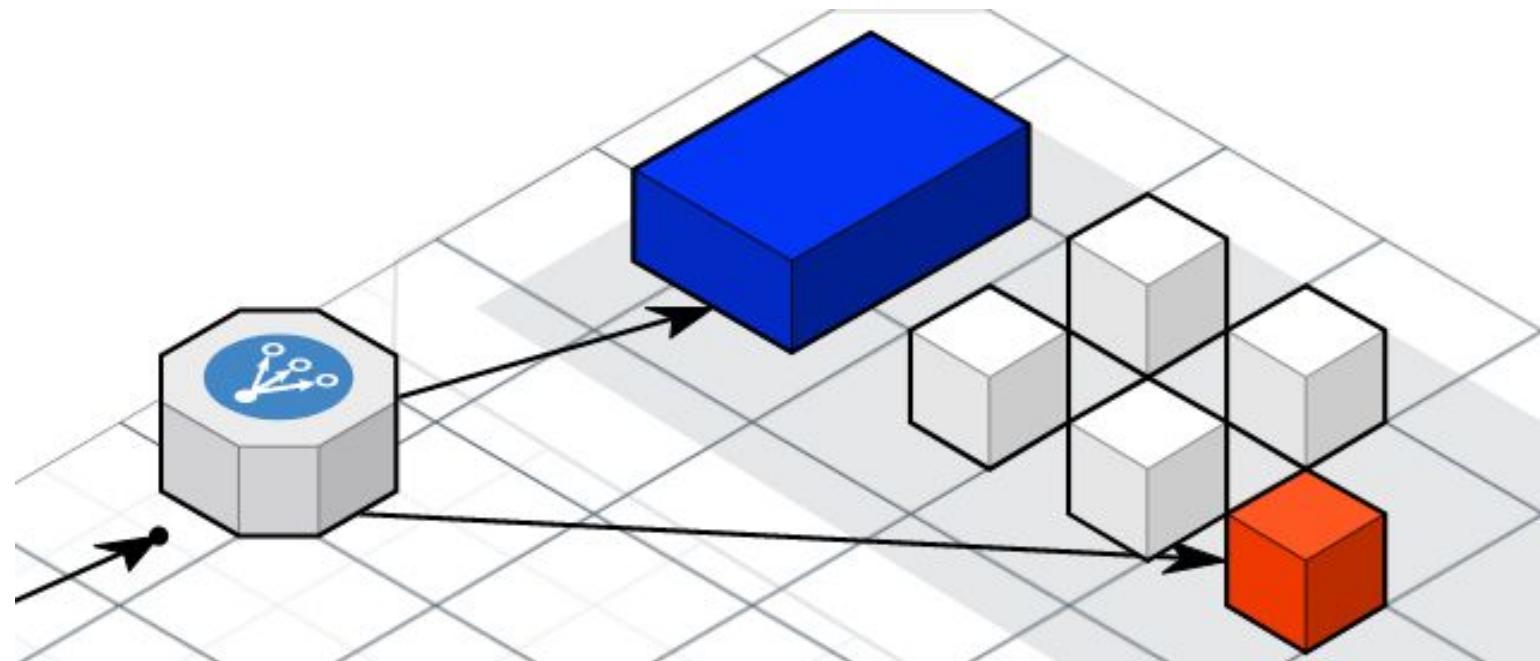


Kubernetes

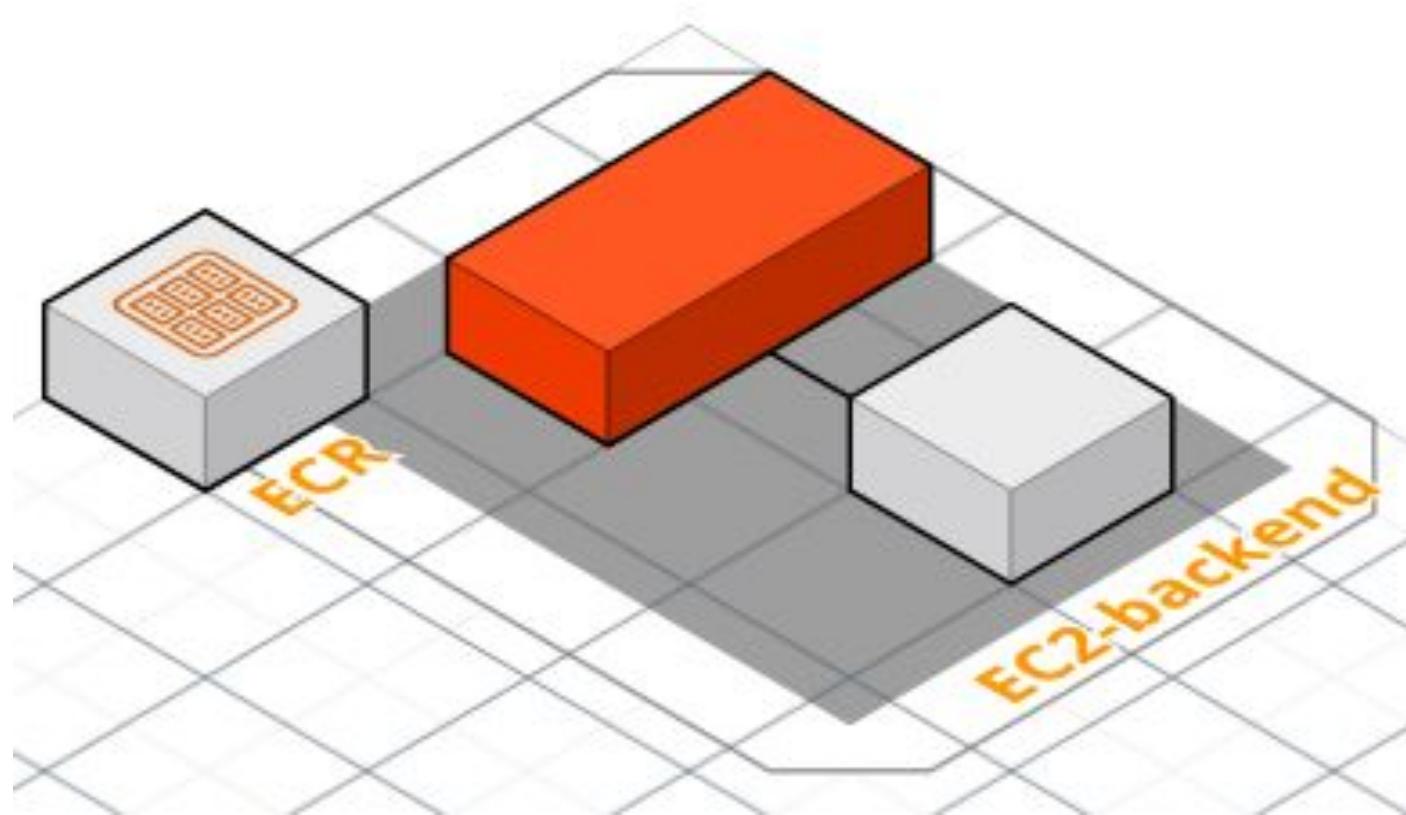


AWS ECS

Kubernetes



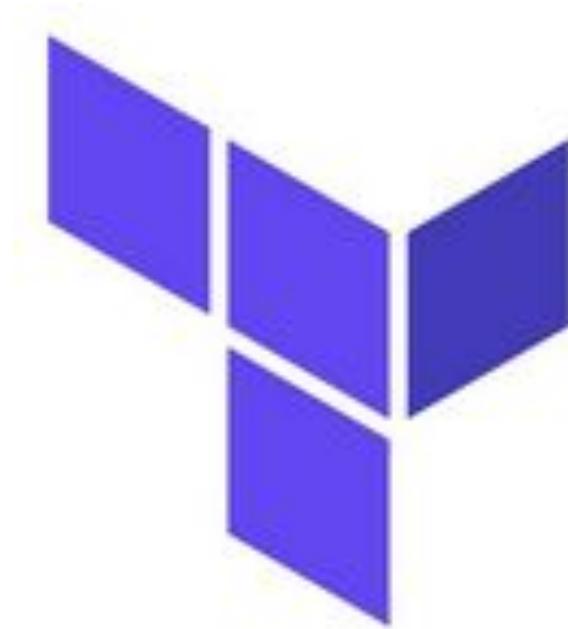
AWS ECS



Herramientas de laC

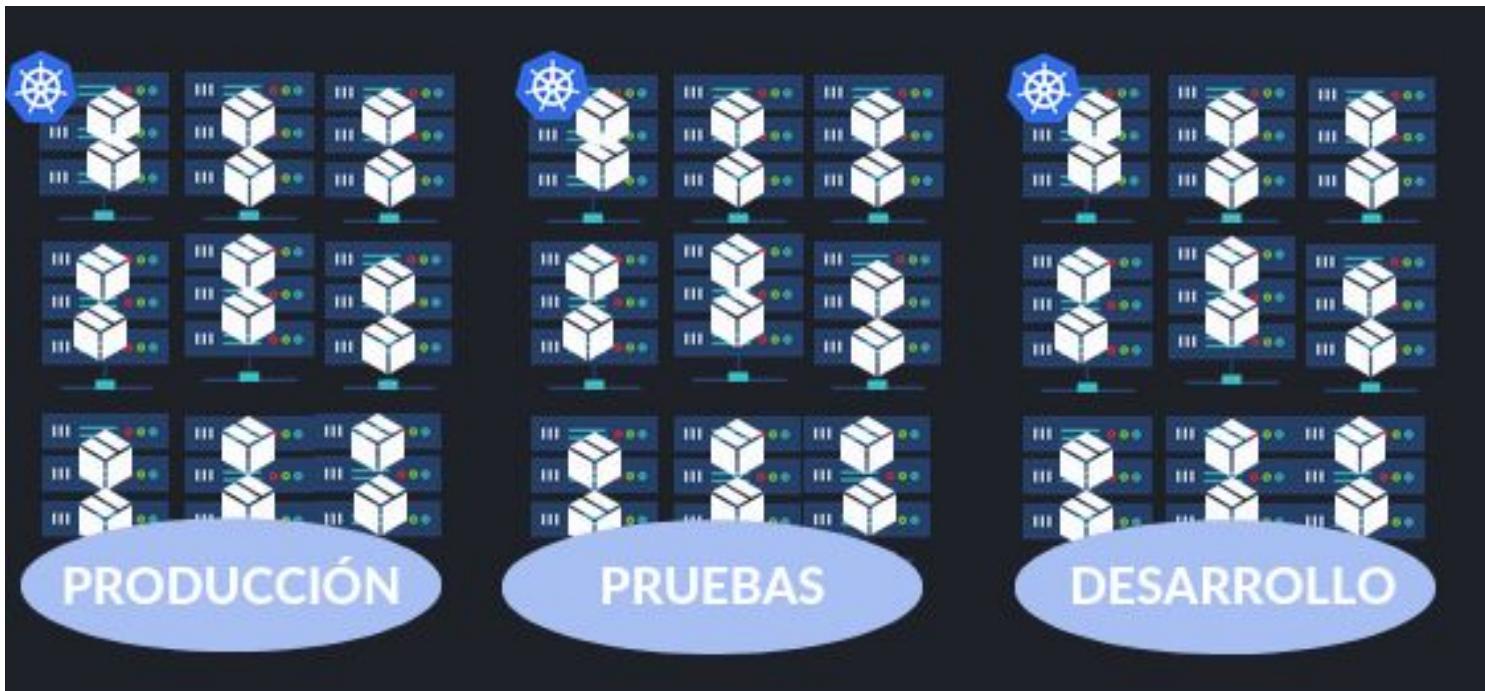


Infraestructura como código



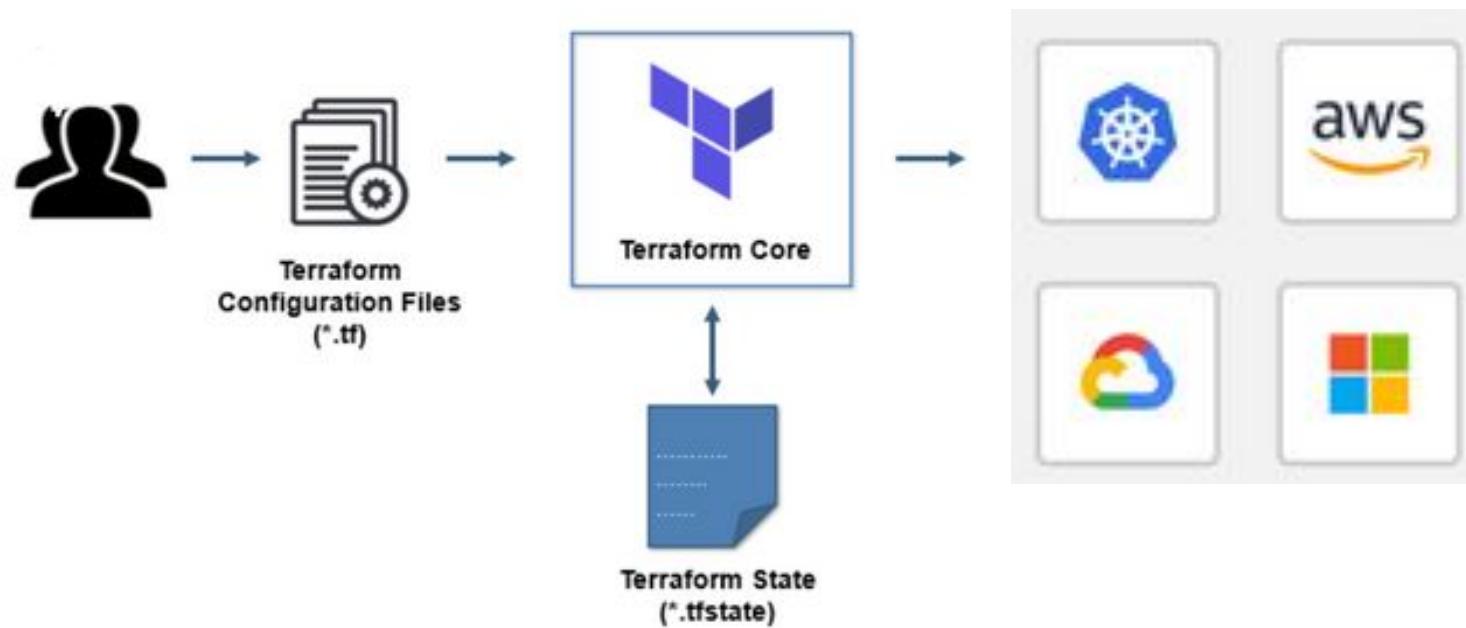
Terraform

Principal tarea



Replicar ambientes

Ejemplo de Terraform



Ejemplo de código

```
automticpipelinebase / Create pipeline / Stateful / Terraform / Dockerfile / .tf
1 terraform{
2   backend "s3" {
3     bucket = "pipeline-artifacts"
4     encrypt = true
5     key = "terraform.tfstate"
6     region = "us-east-1"
7   }
8 }
9 provider "aws" {
10   region = "us-east-1"
11 }
12 #export AWS_ACCESS_KEY_ID=AKIAYFQTFKR6I
```

Ejemplo de código

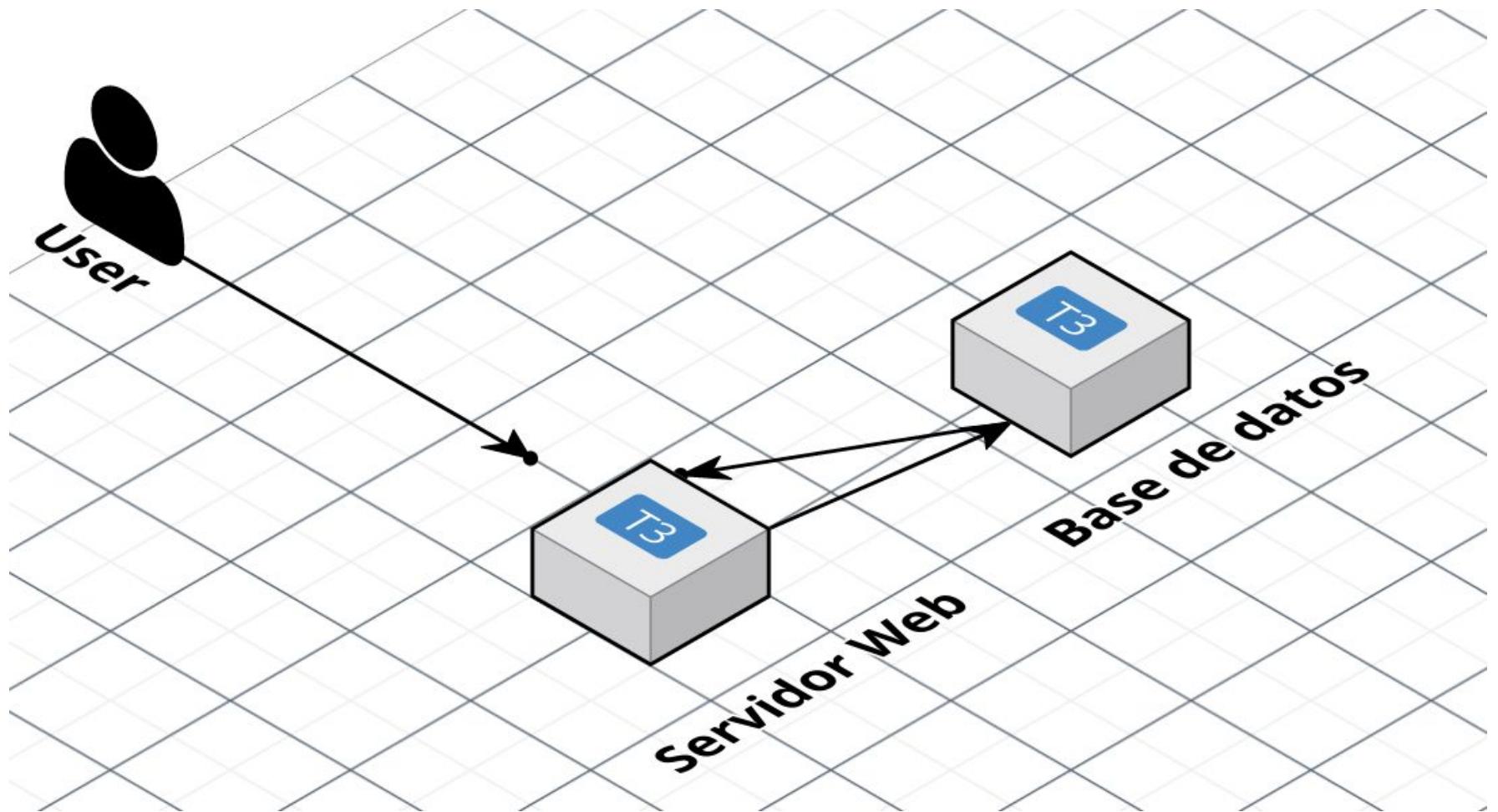
```
resource "aws_codecommit_repository" "iaas-cicd" {
    repository_name = "mi_repo_de_microservicios"
}

resource "aws_codecommit_repository" "paas-cicd" {
    repository_name = "paas-cicd"
}

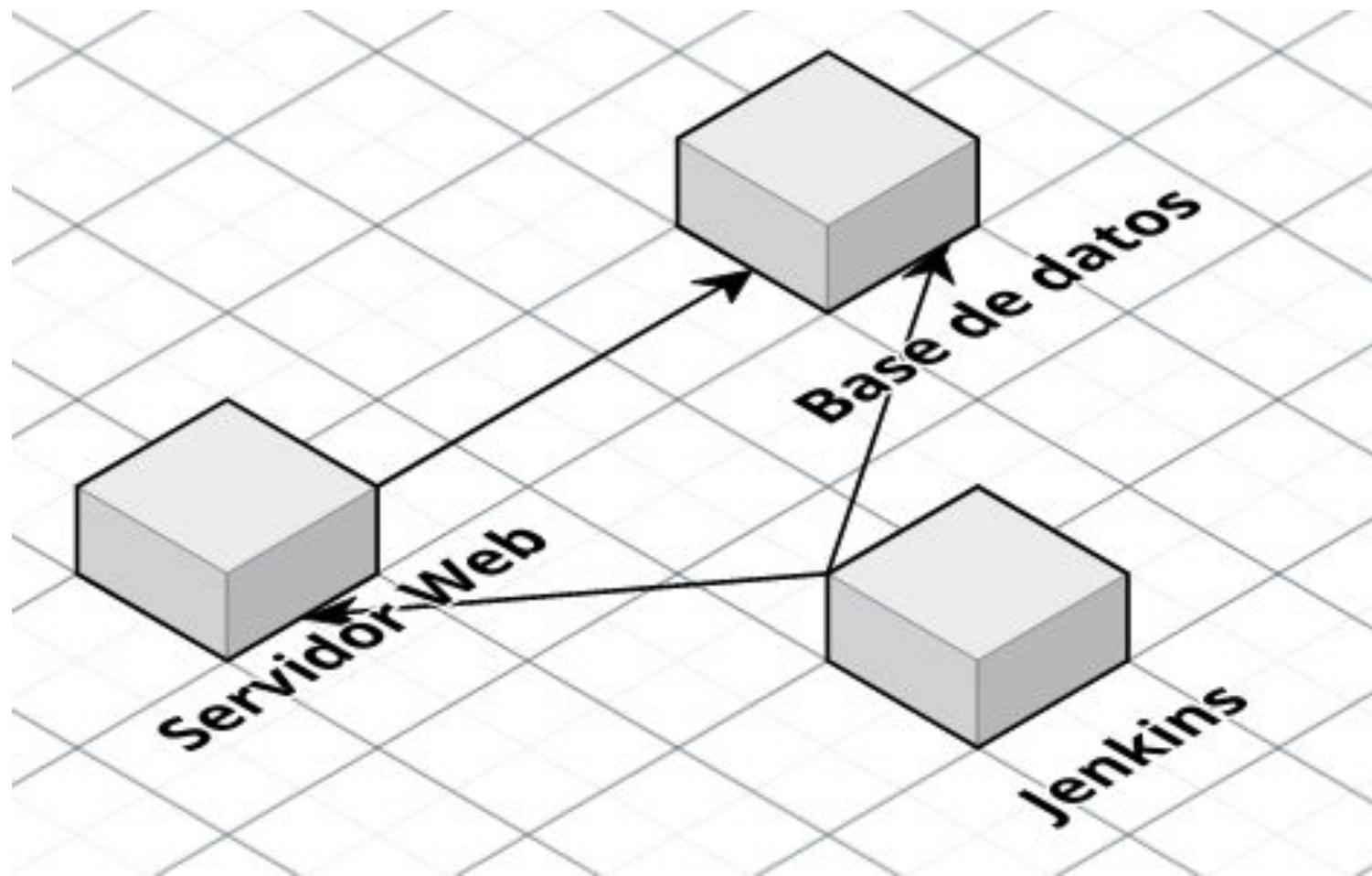
resource "aws_codecommit_repository" "lambdaautho-code" {
    repository_name = "funcion-authorizer"
}
```

Ejercicios

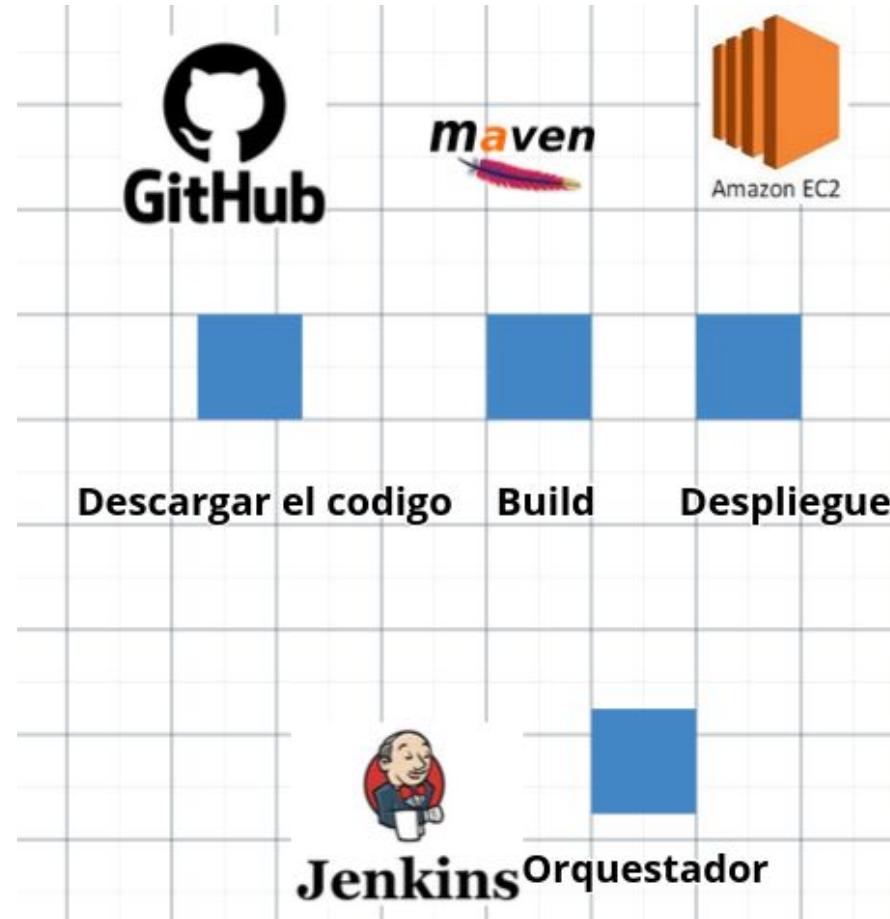
Primer ejemplo



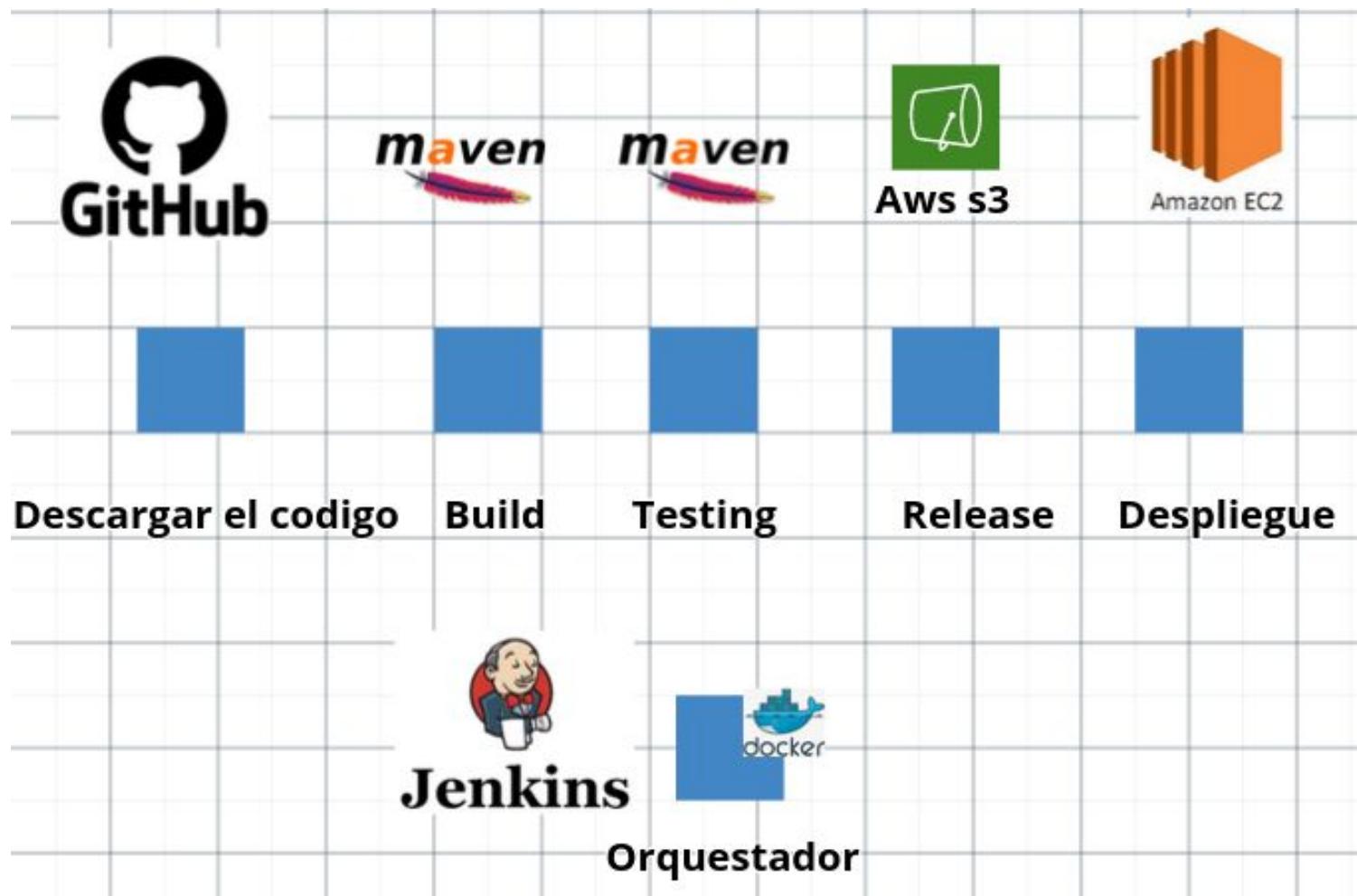
Primer ejemplo



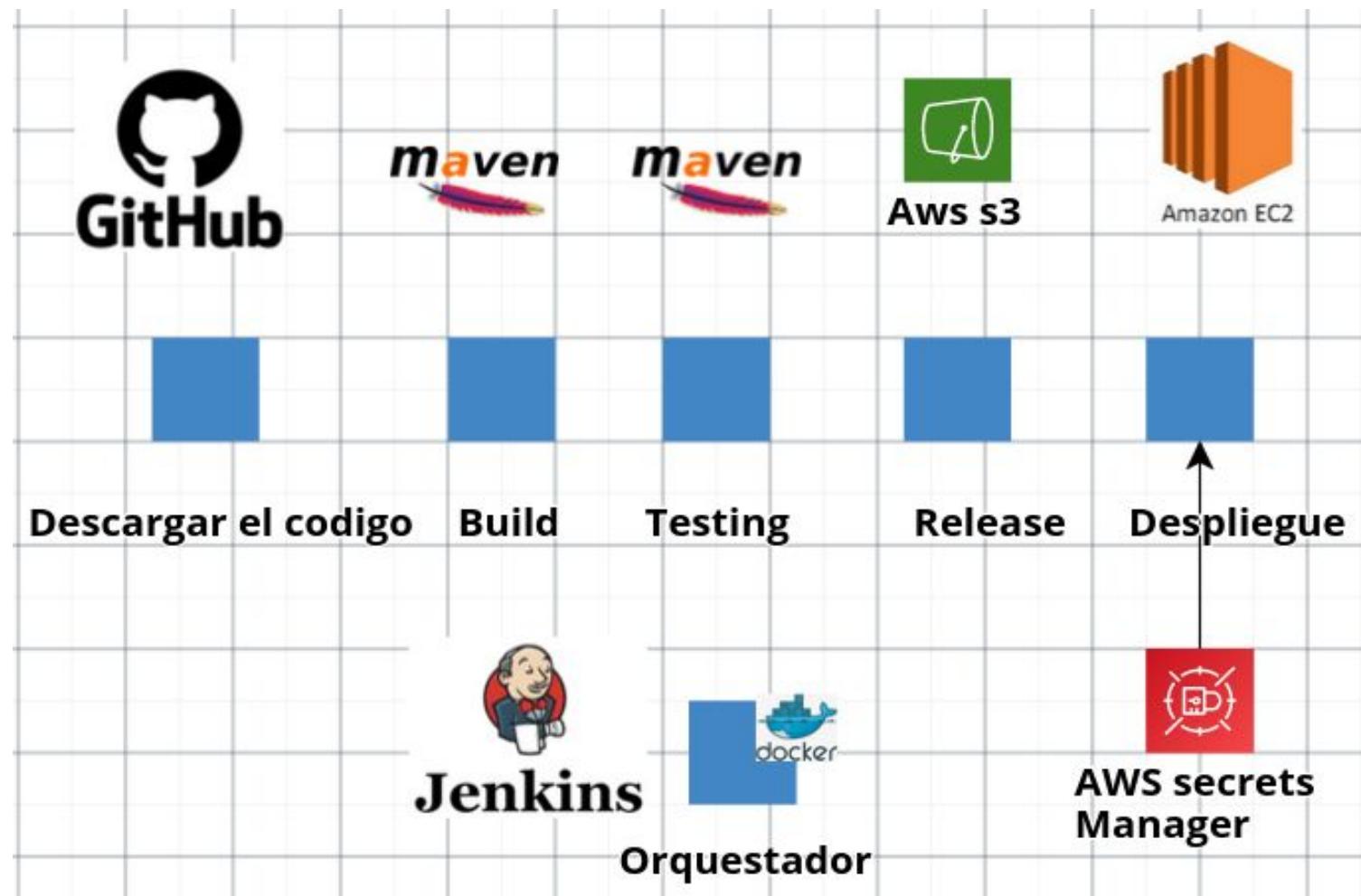
Primera propuesta



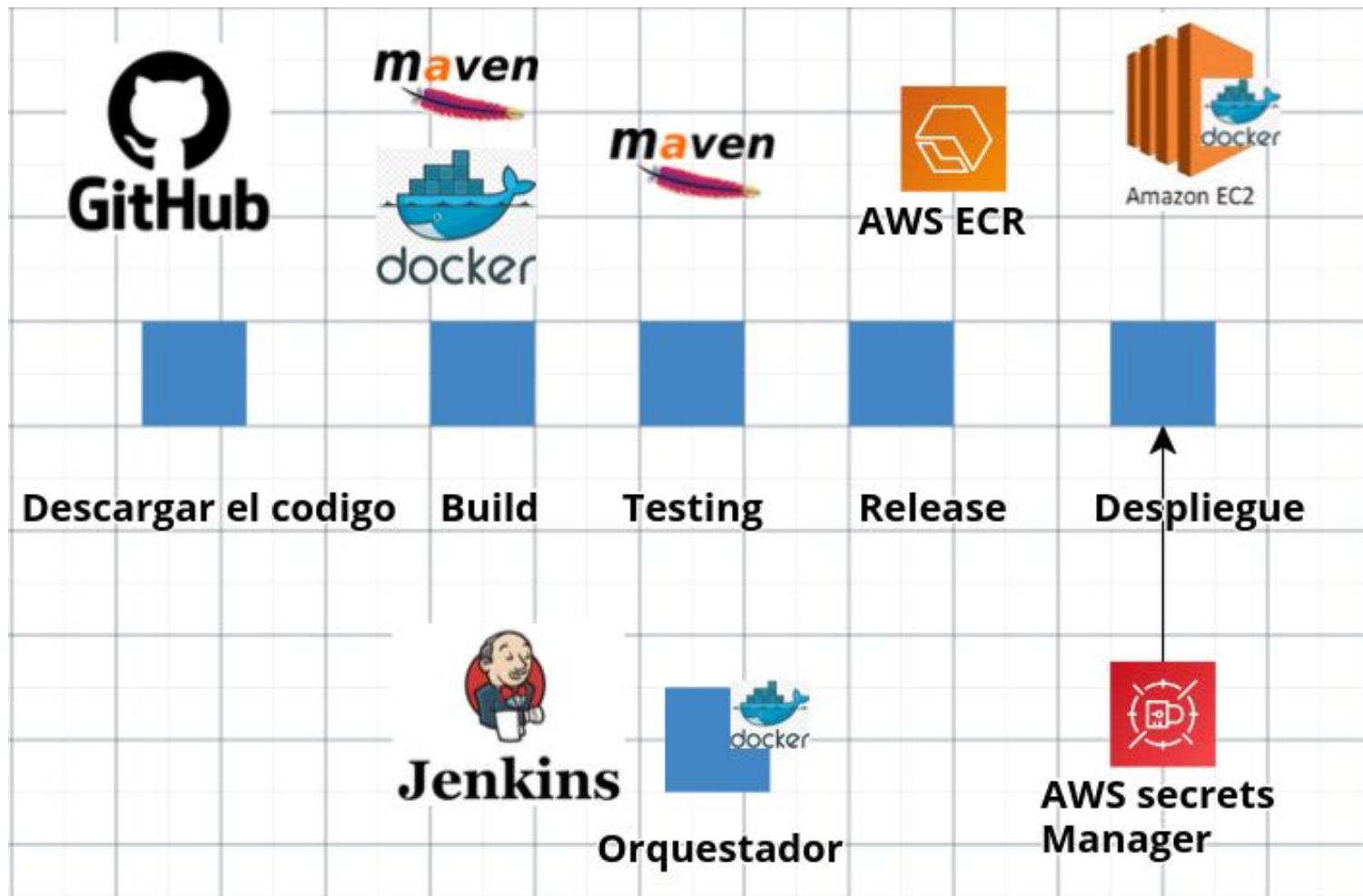
Segunda propuesta



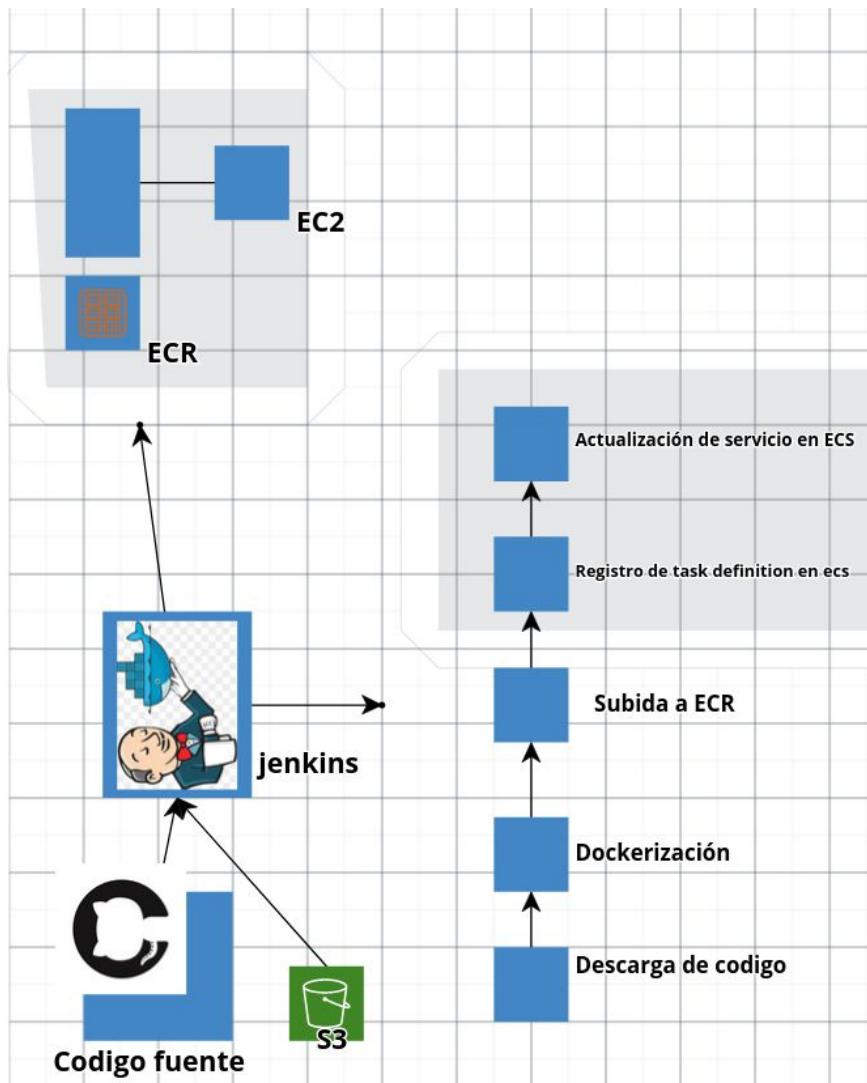
Agregando más servicios



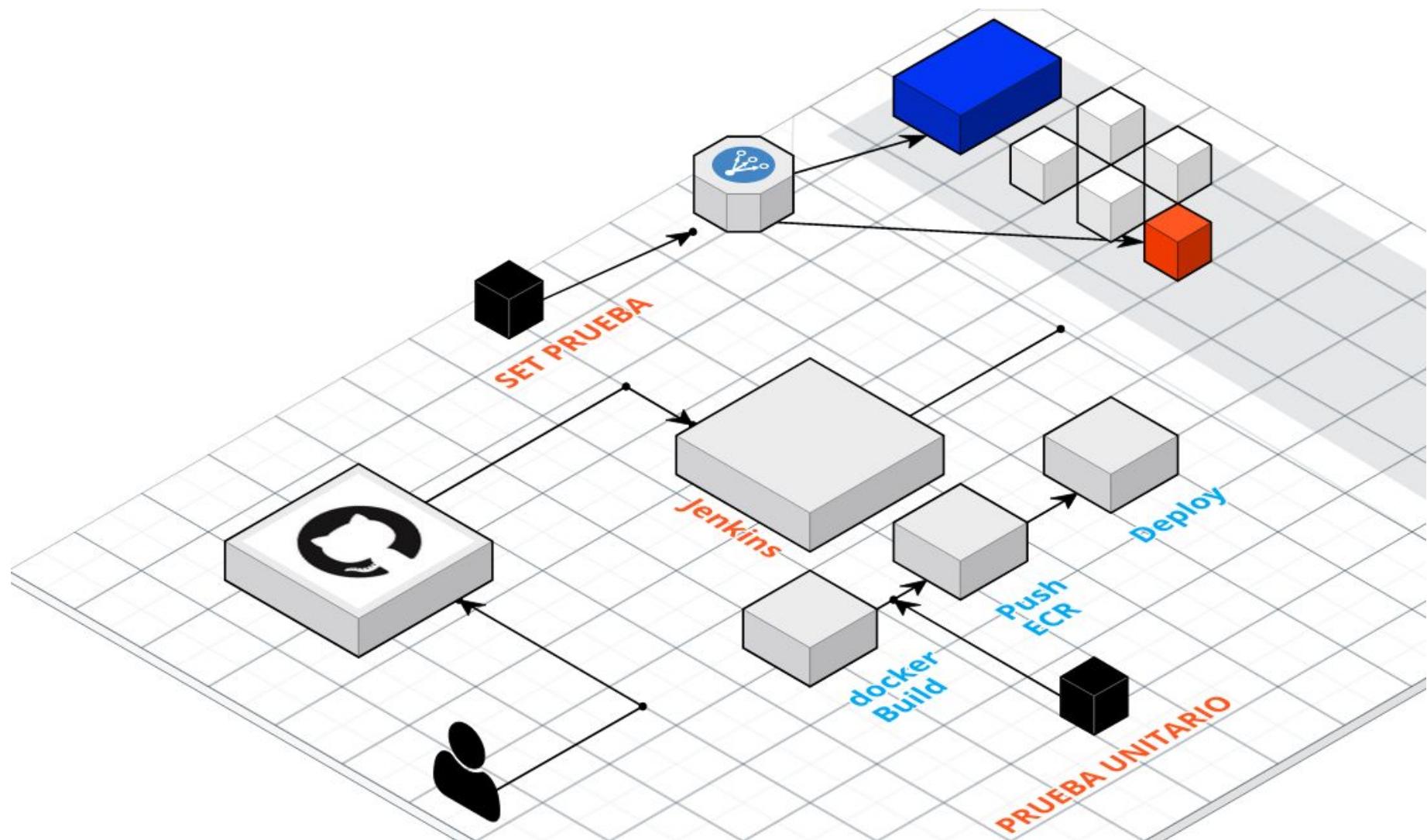
Agregando más servicios



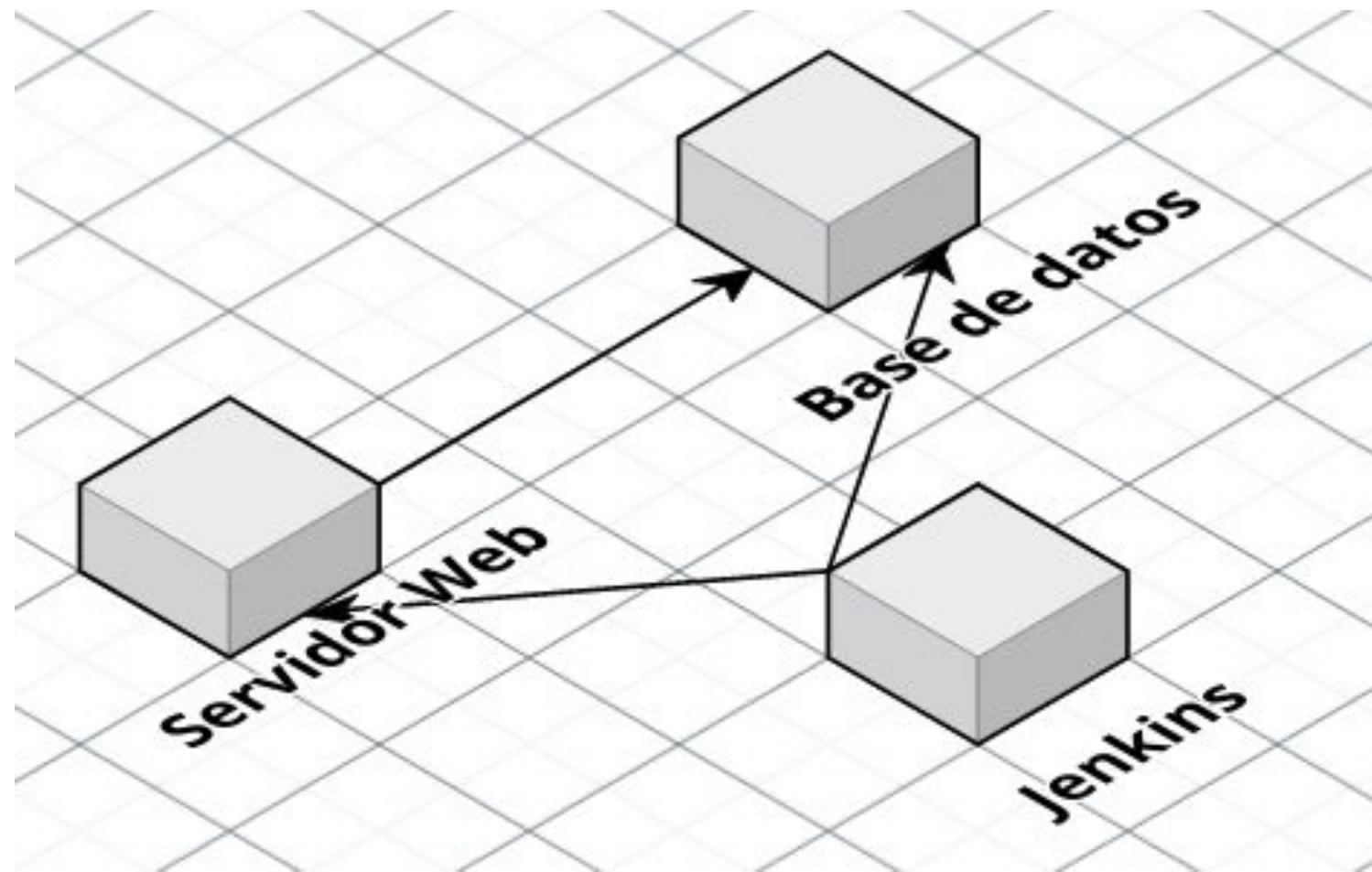
Ejemplo avanzado



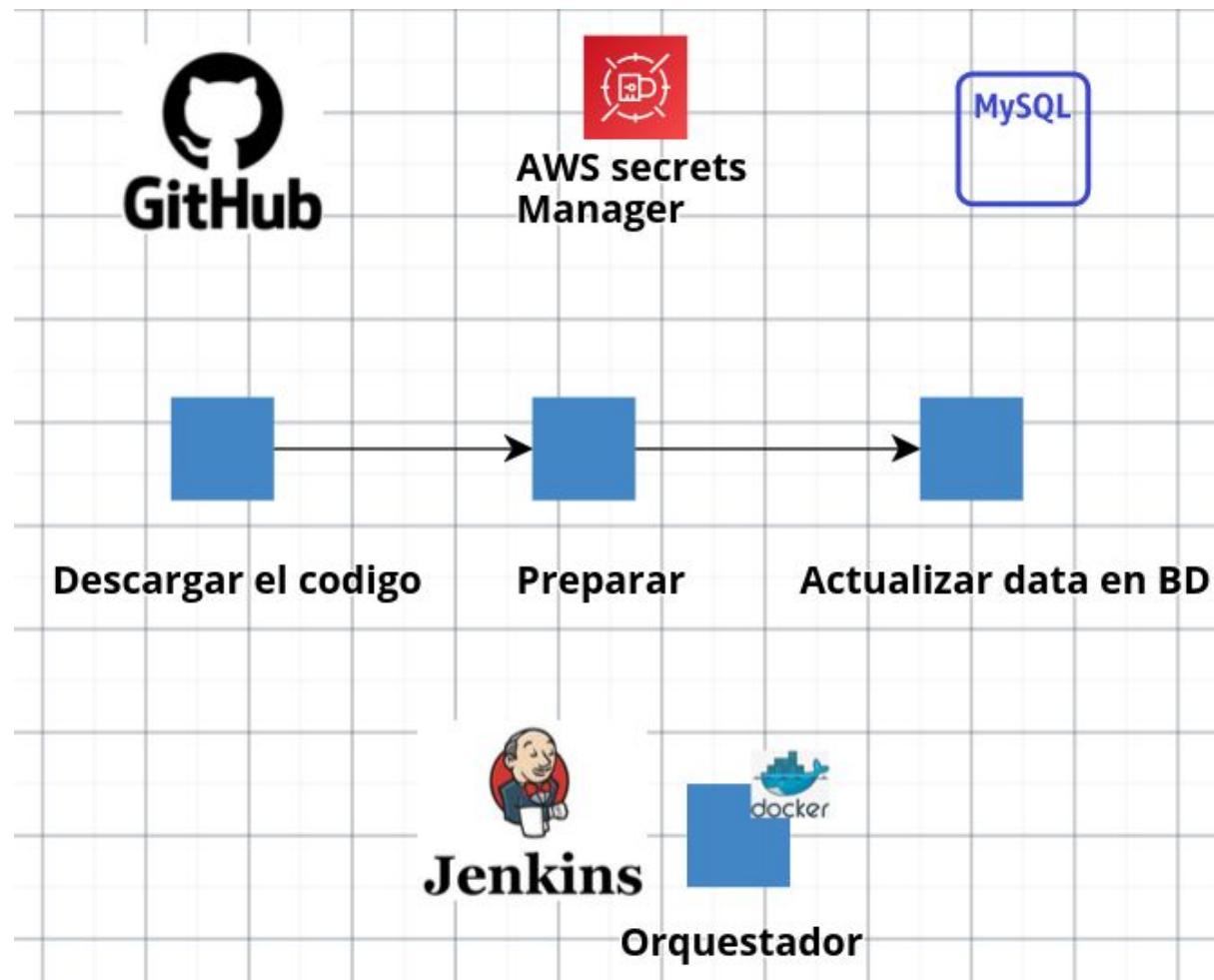
Ejemplo con K8s



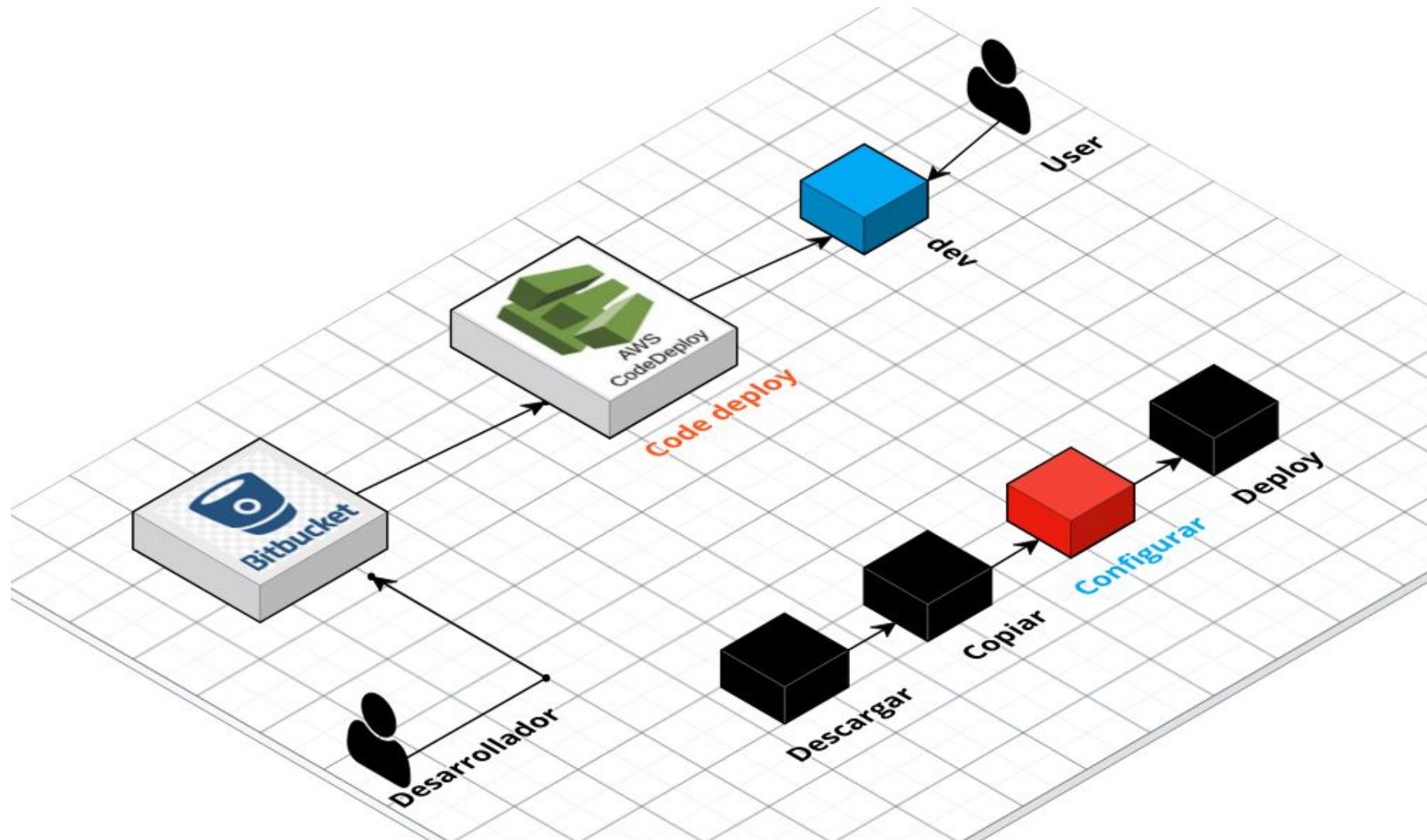
Primer ejemplo



Data automática



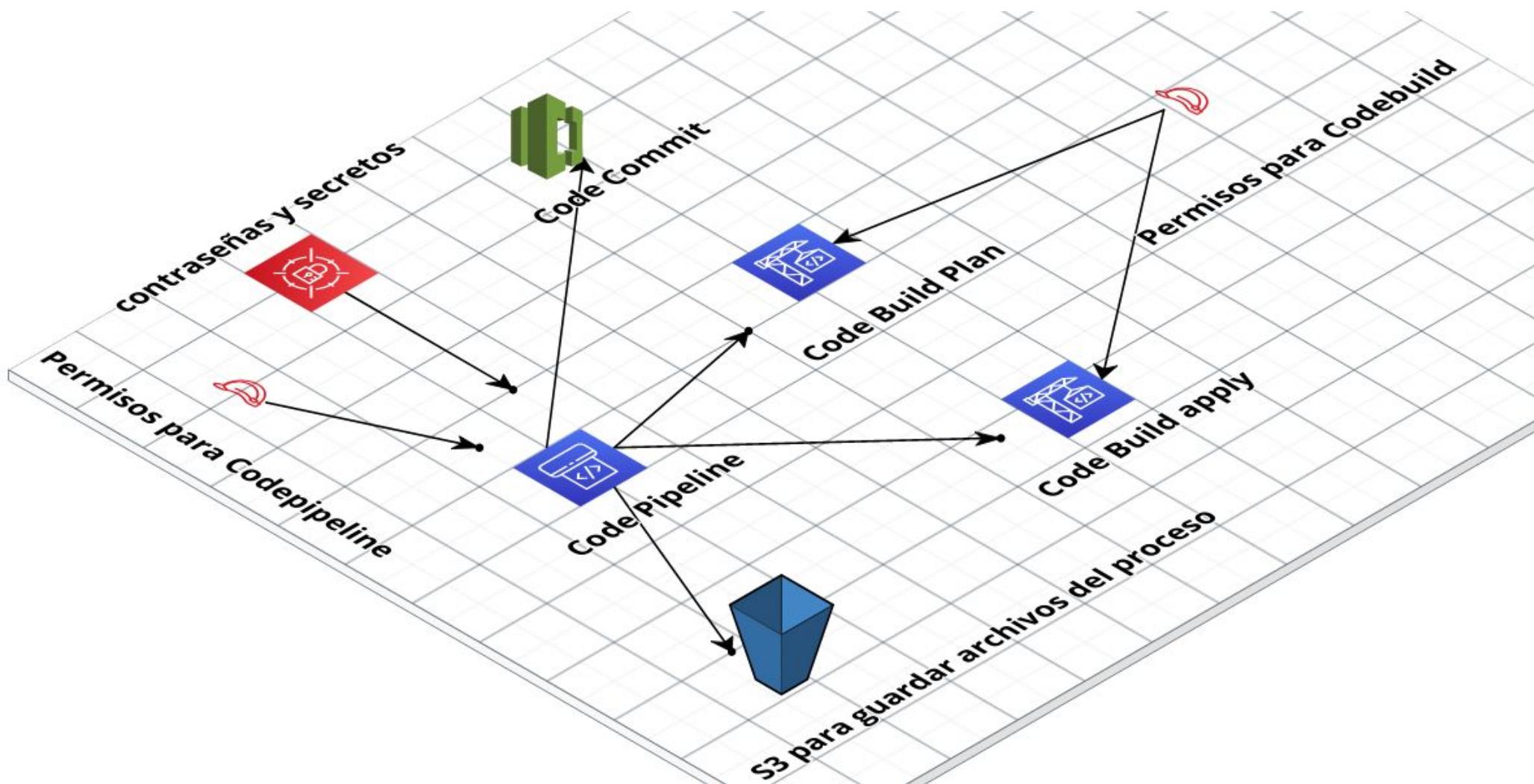
Otras herramientas



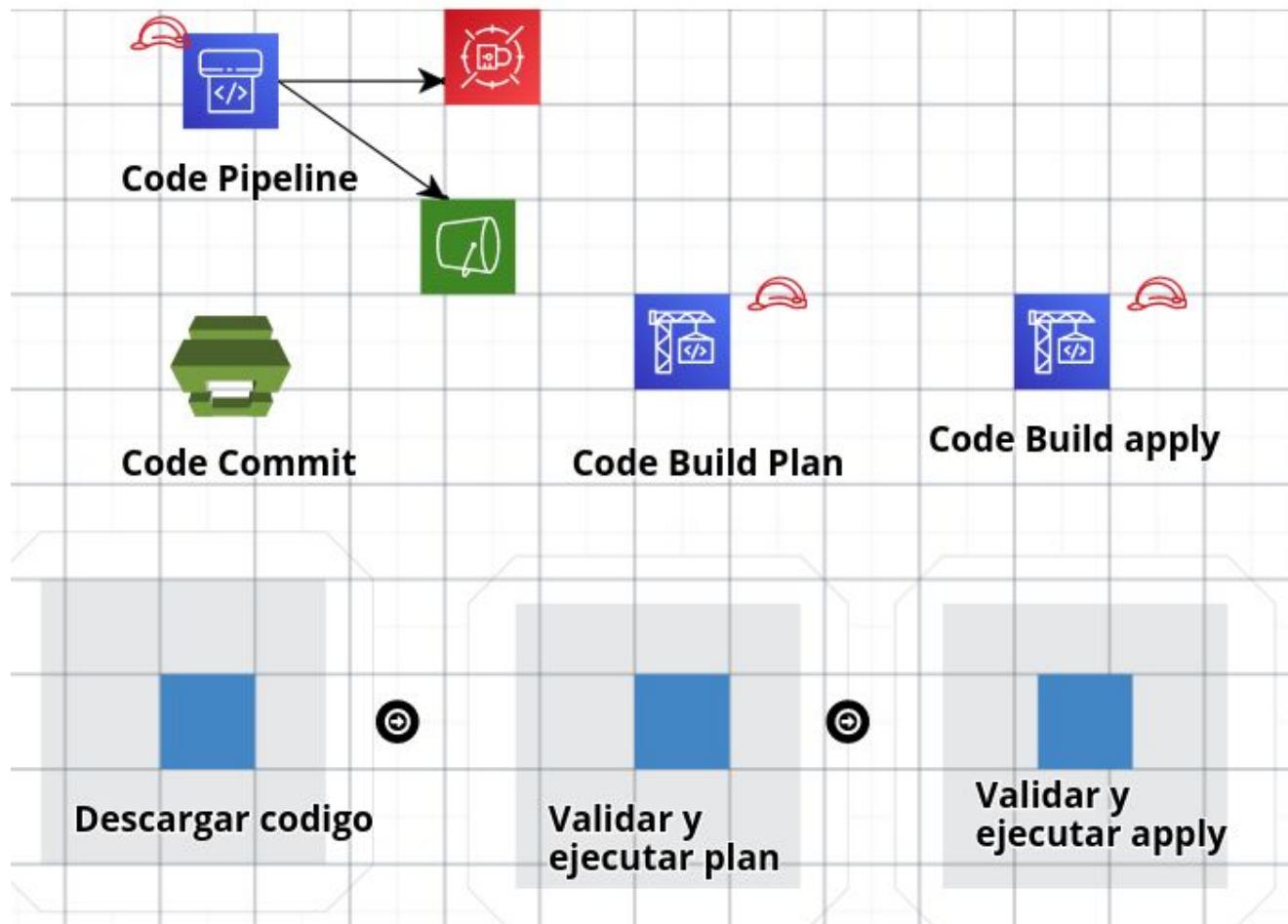
Primer Pipeline Infraestructura Automática



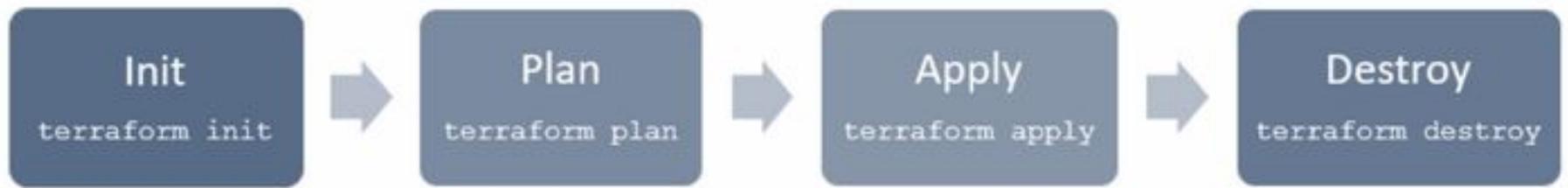
Diagramas de servicios



El proceso



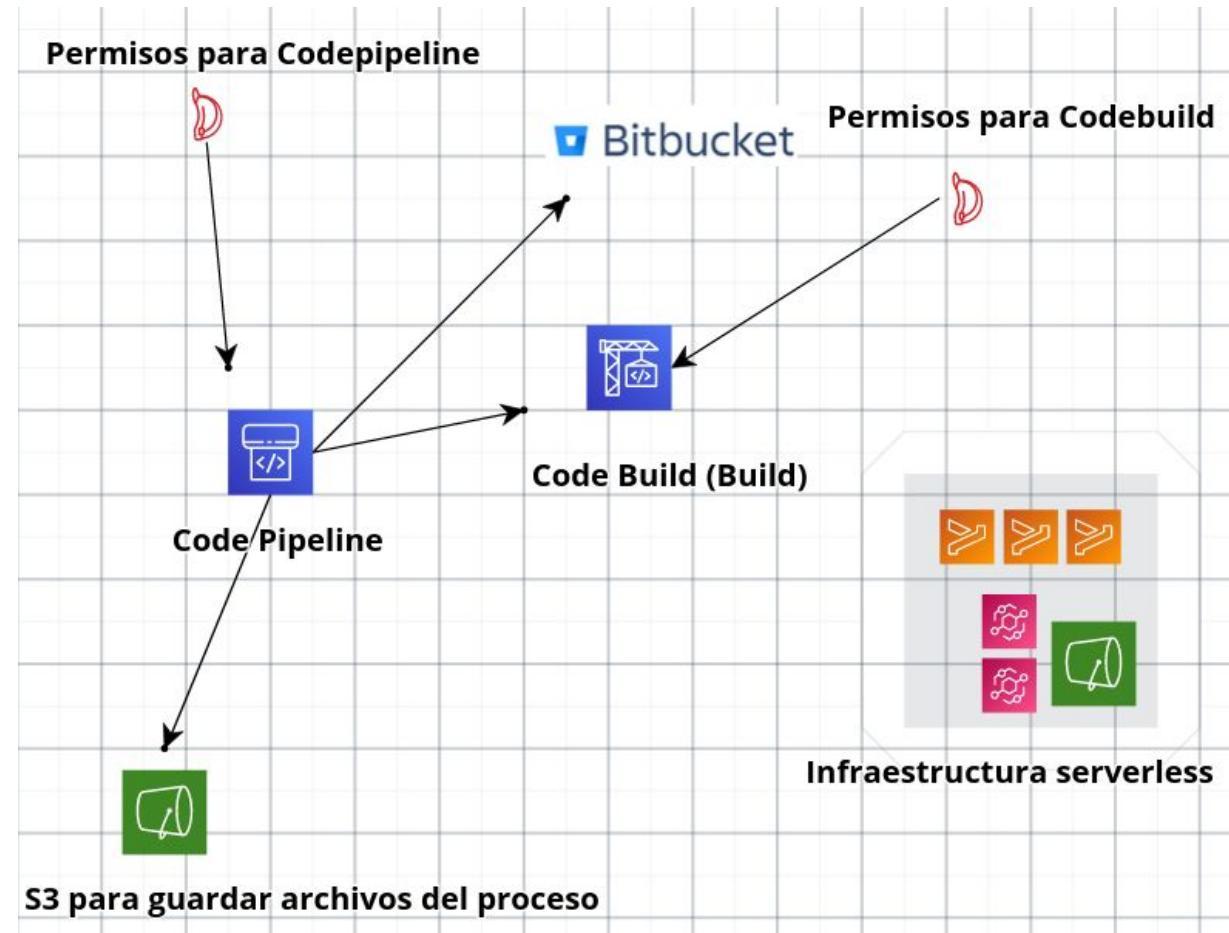
Comandos básicos:



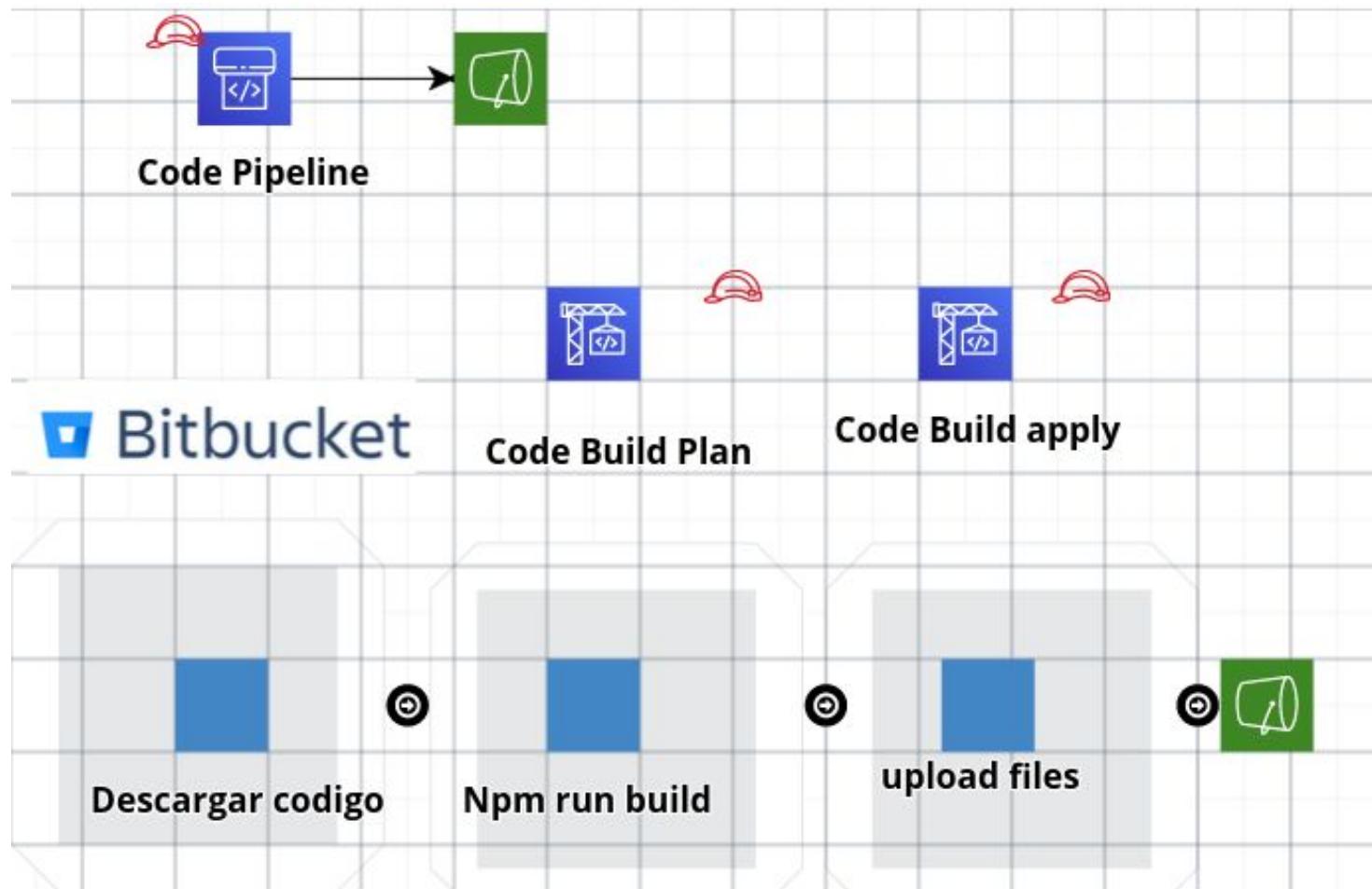
Despliegue de Frontend



Los servicios



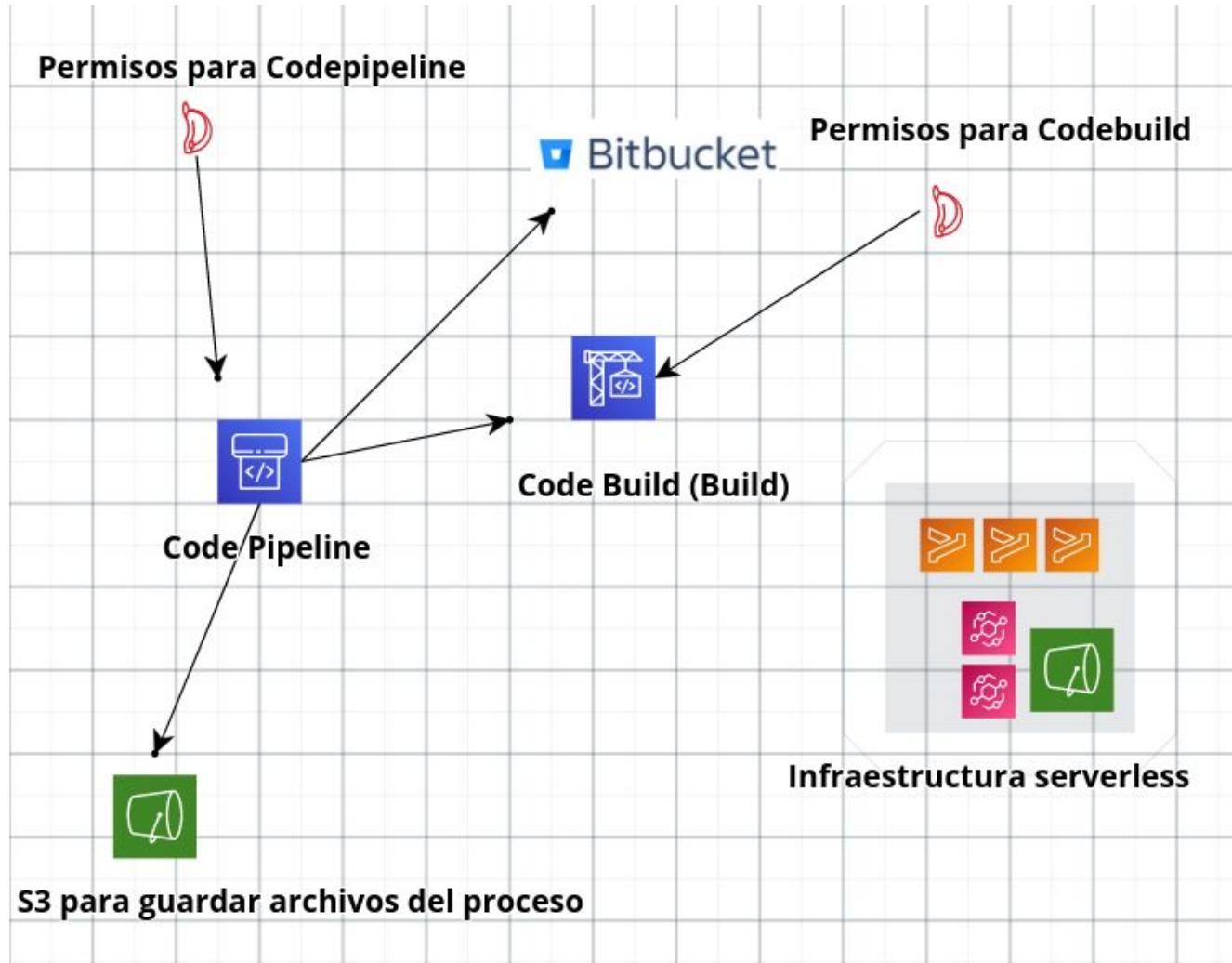
El proceso



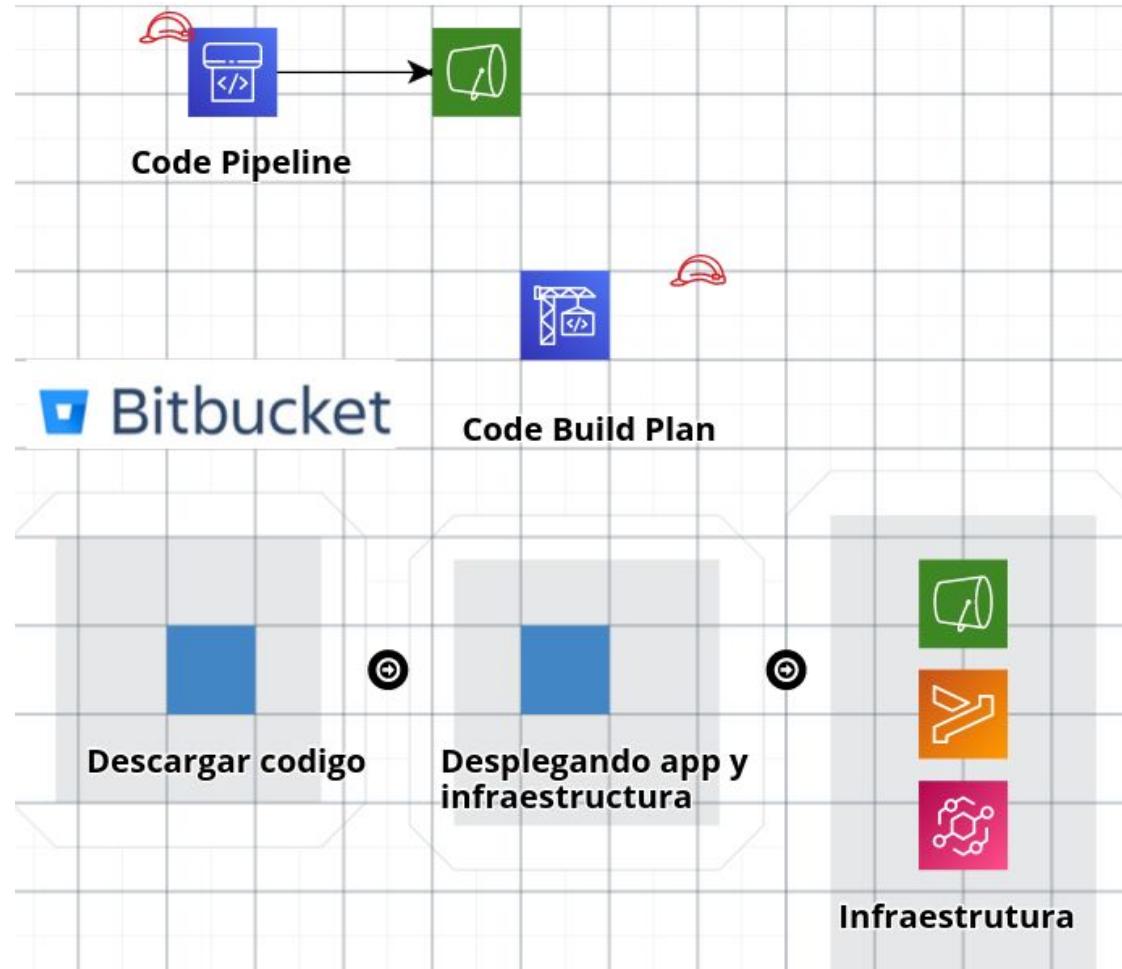
Despliegue de Serverless



Los servicios



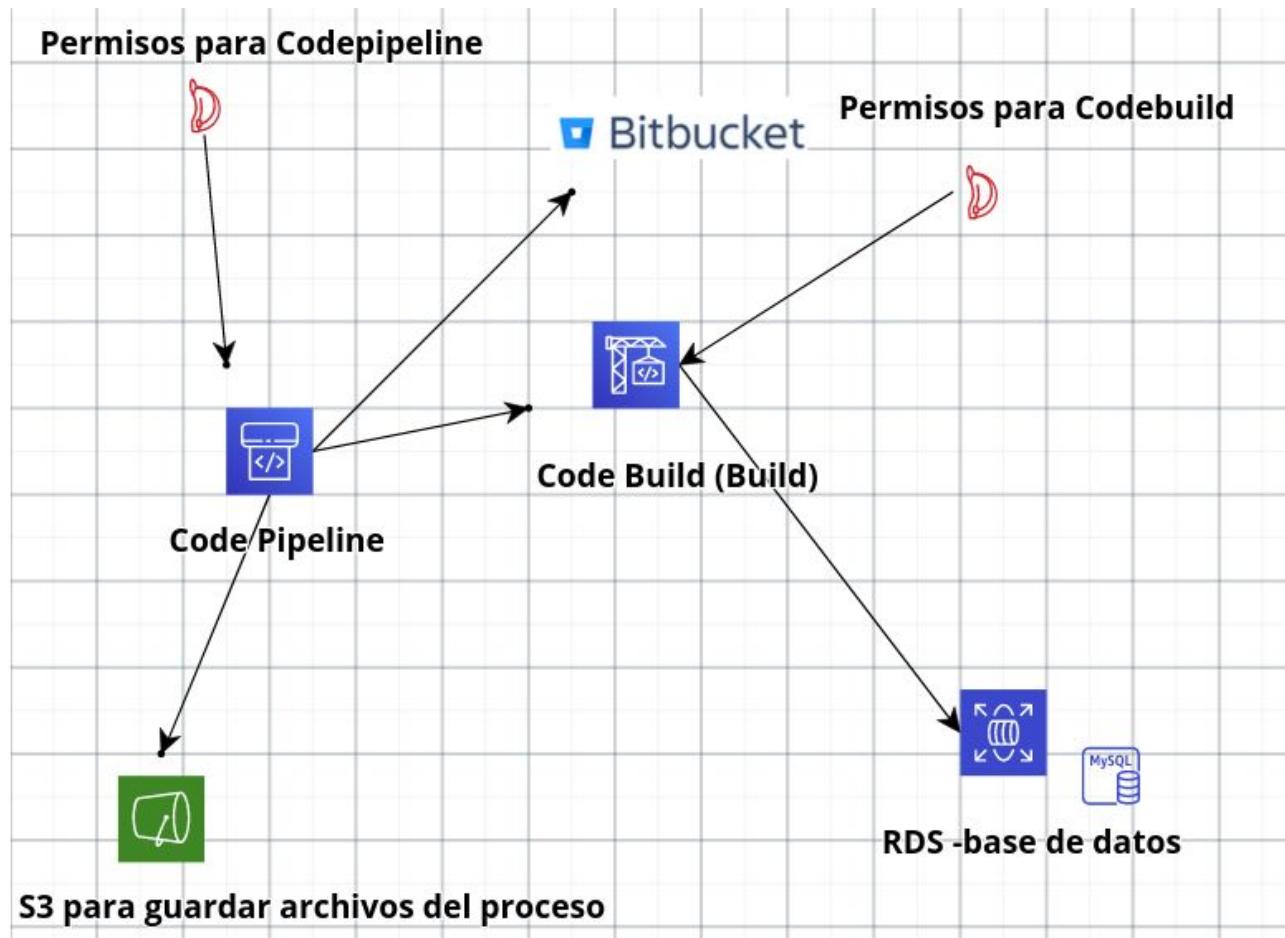
El proceso



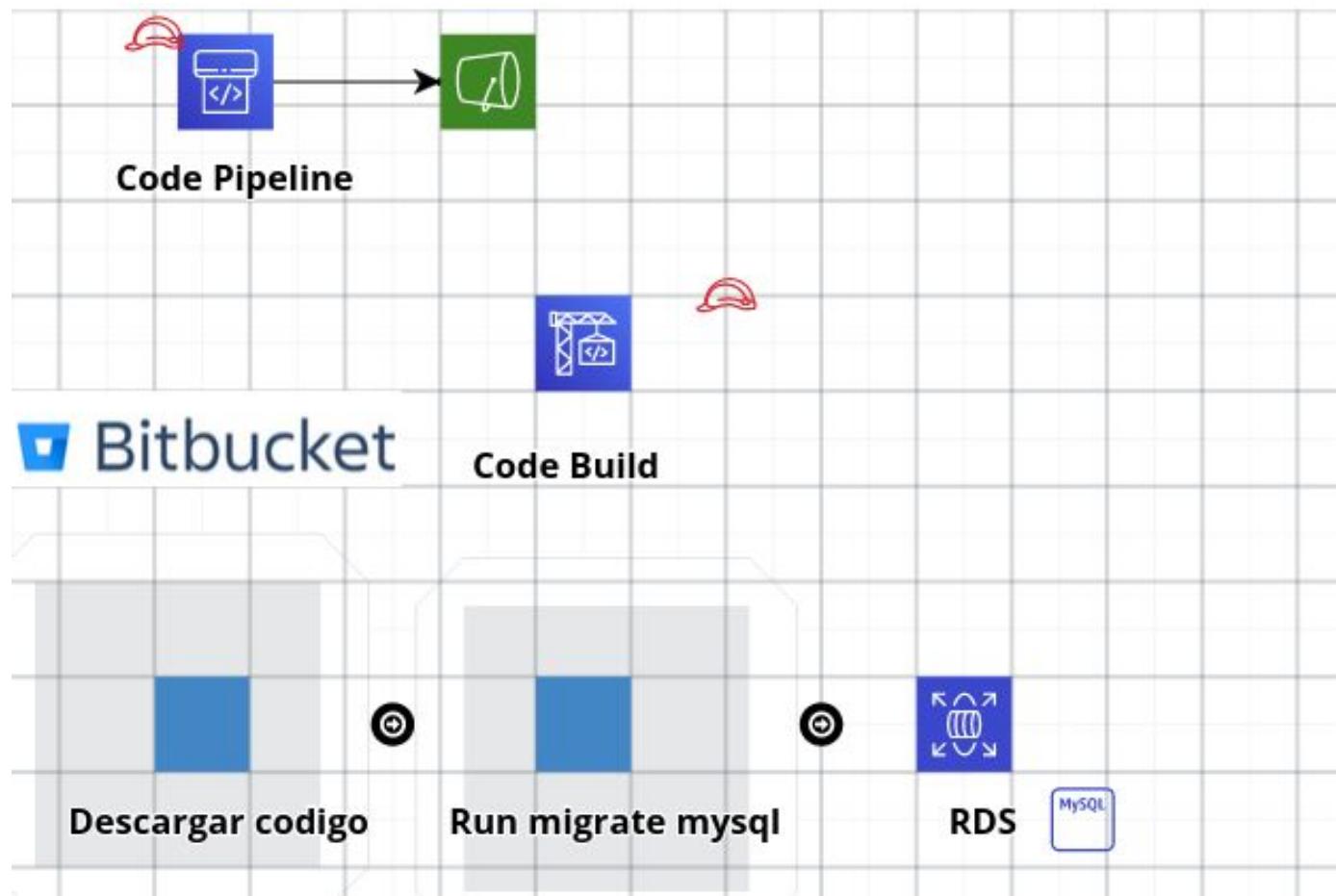
Despliegue de data en base de datos



Los servicios



El proceso



Despliegue en Backend (CI/CD)



El proceso

