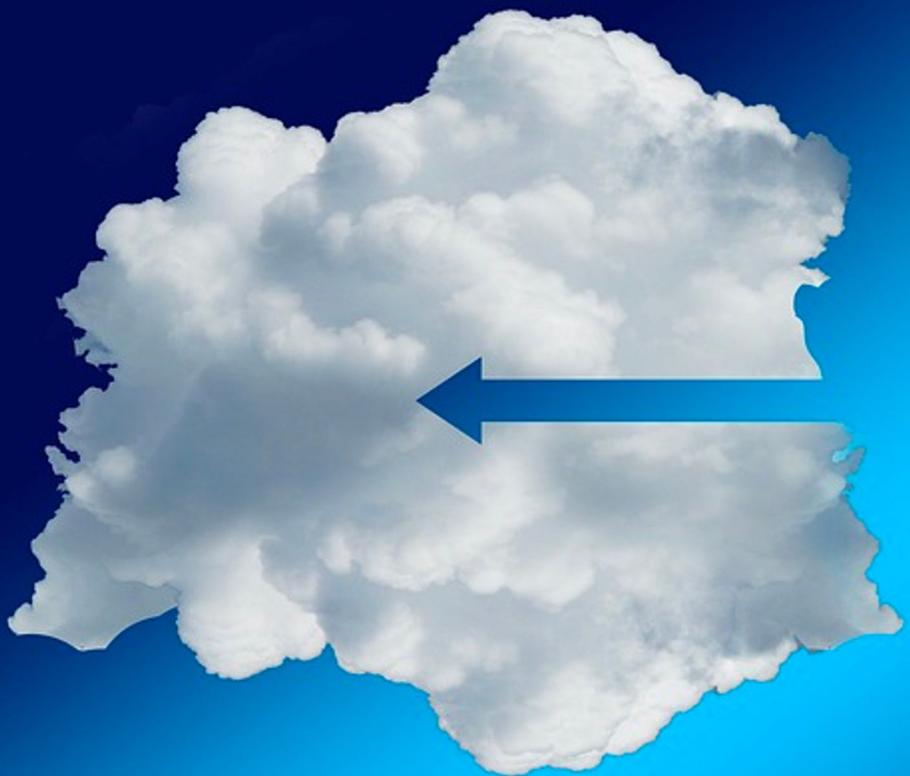


Docencia en la nube

Despliegue y gestión de plataformas Moodle con IaC sobre AWS

Cristina Jerez
UPCnet - Universitat Politècnica de Catalunya

David Garcia
UPCnet - Universitat Politècnica de Catalunya



CLOUD

ESCALABILIDAD



Evolución Ops en Entornos Cloud



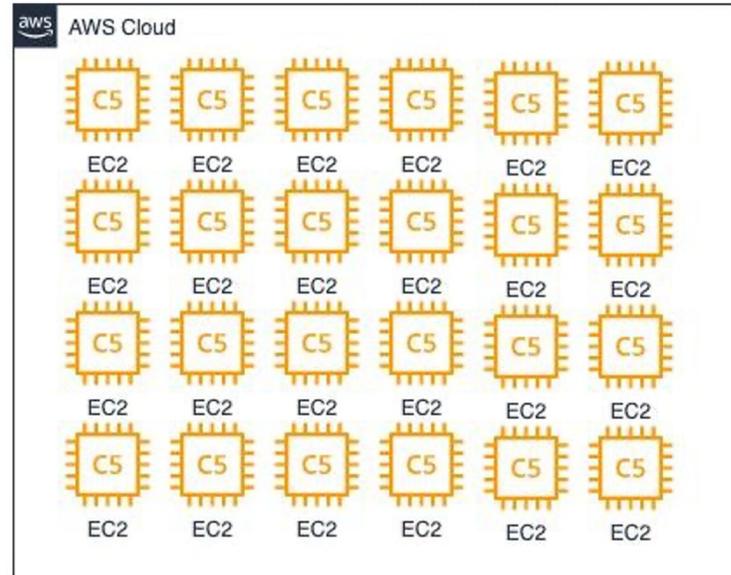
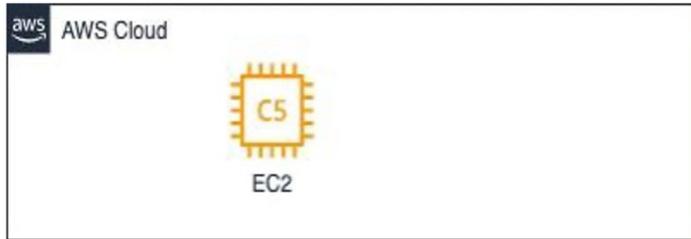
OPS TEAM



OPS TEAM



OPS TEAM

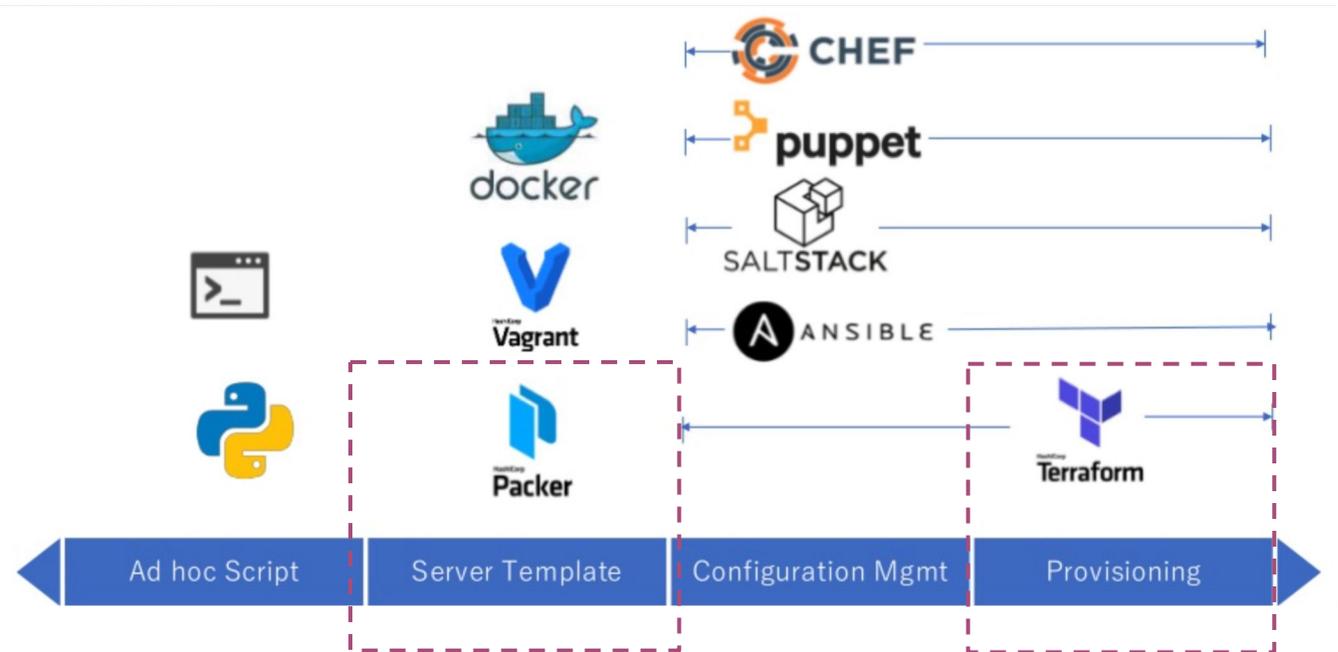


Herramientas de IaC

Necesitamos herramientas que puedan ayudar a desplegar entornos de formación de manera automática y agilizar la administración de los entornos.

Beneficios

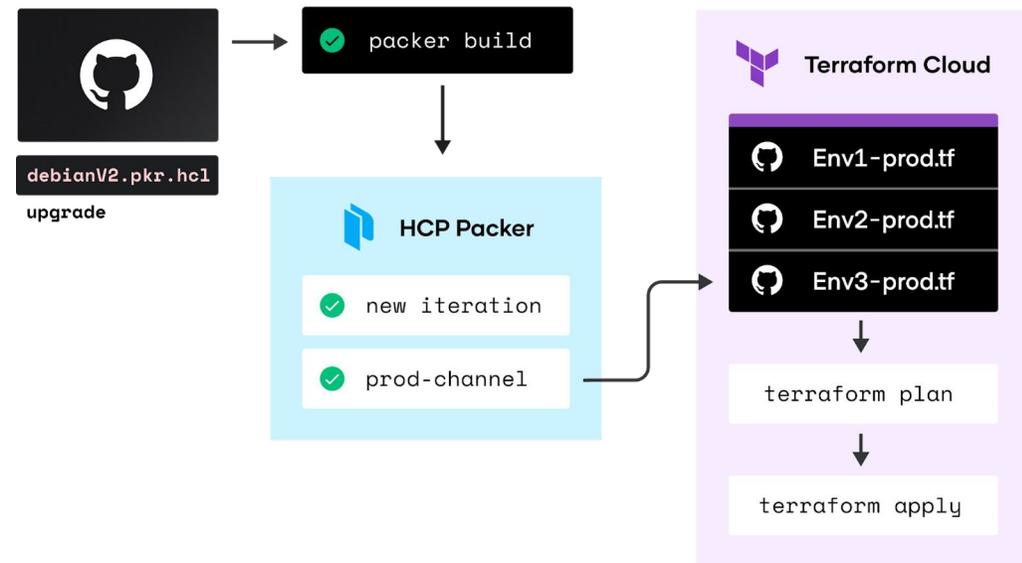
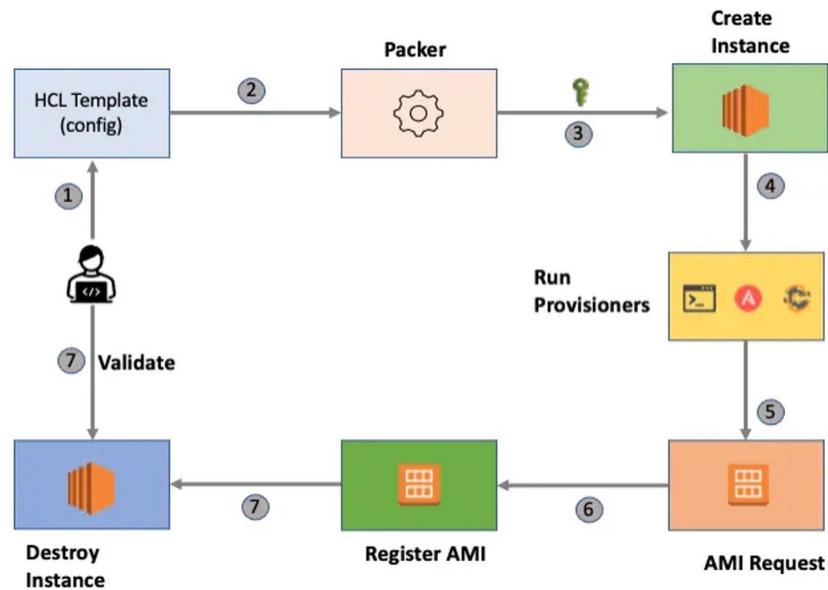
- **Aprovisionamiento rápido y repetible** de todo tipo de infraestructura
- **Agilidad en la escalabilidad**
- **Reducción de errores** gracias a la eliminación de configuraciones manuales
- **Coherencia** de los diferentes entornos (dev, test, prod...)



Cómo utilizamos las herramientas de IaC

Packer

Herramienta Open Source que nos permite crear imágenes de servidores y contenedores para múltiples plataformas desde un único punto de configuración.

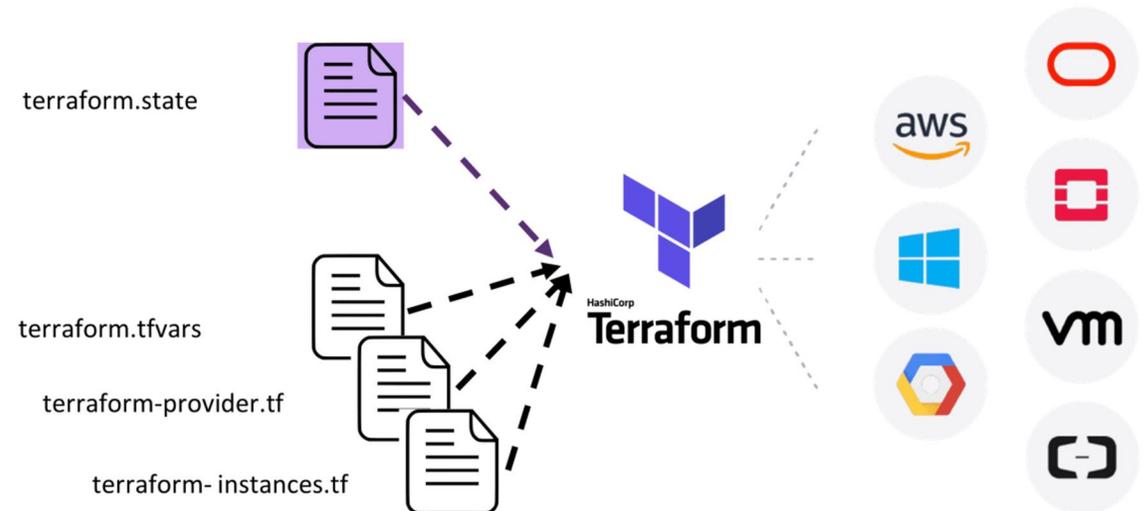


Cómo utilizamos las herramientas de IaC

Terraform

Terraform es un software Open Source que proporciona un modo de configuración para desplegar infraestructura. Una vez desplegada, terraform cambia de forma segura y eficiente la infraestructura a medida que evoluciona la configuración.

- **Describir** la infraestructura de un modo declarativo.
- **Mantener** el estado de la infraestructura.
- **Evolucionar** la infraestructura de manera sencilla
- **Reutilizar** código para diferentes proyectos.

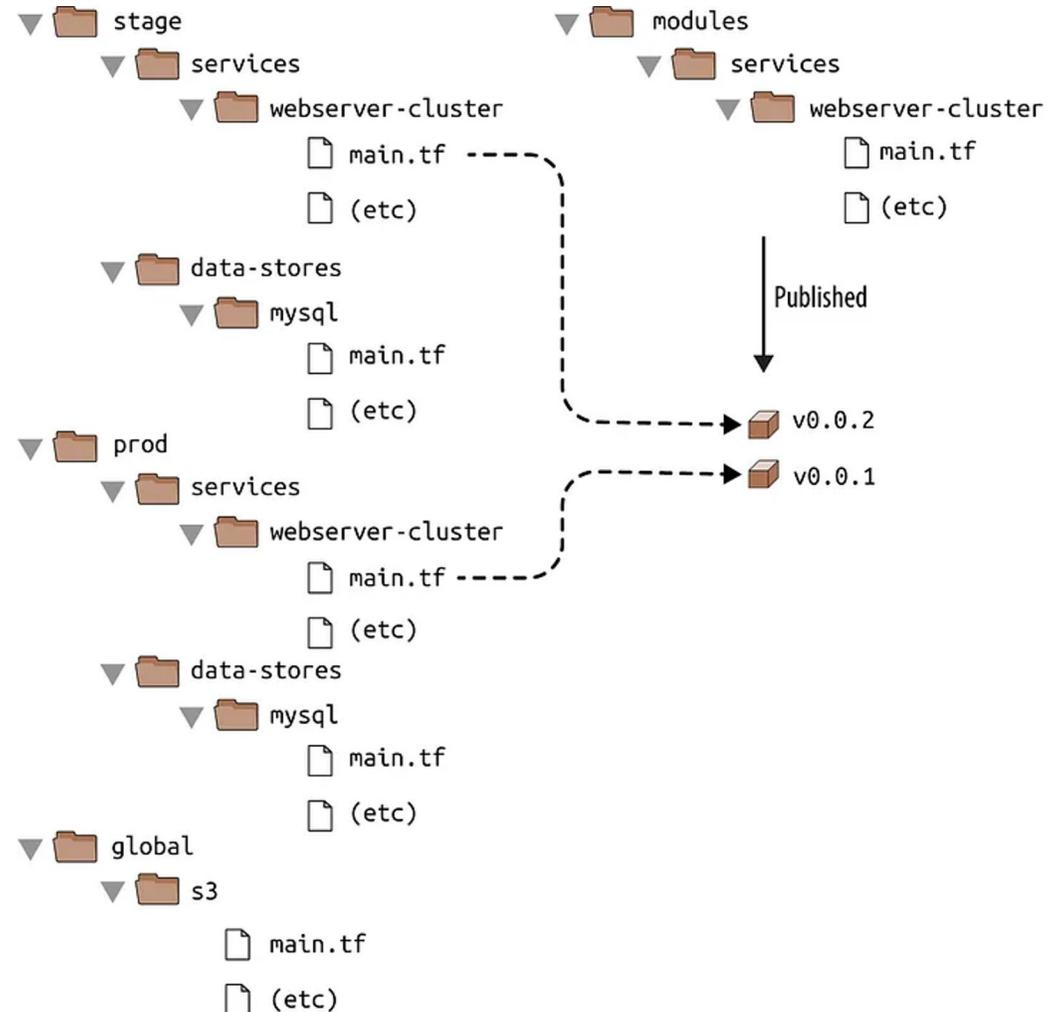


Cómo utilizamos las herramientas de IaC

Terraform

```
variable "public_subnet_cidrs" {  
  type      = list(string)  
  description = "Public Subnet CIDR values"  
  default   = ["10.0.1.0/24", "10.0.2.0/24", "10.0.3.0/24"]  
}  
  
variable "private_subnet_cidrs" {  
  type      = list(string)  
  description = "Private Subnet CIDR values"  
  default   = ["10.0.4.0/24", "10.0.5.0/24", "10.0.6.0/24"]  
}
```

```
resource "aws_subnet" "public_subnets" {  
  count      = length(var.public_subnet_cidrs)  
  vpc_id     = aws_vpc.main.id  
  cidr_block = element(var.public_subnet_cidrs, count.index)  
  
  tags = {  
    Name = "Public Subnet ${count.index + 1}"  
  }  
}  
  
resource "aws_subnet" "private_subnets" {  
  count      = length(var.private_subnet_cidrs)  
  vpc_id     = aws_vpc.main.id  
  cidr_block = element(var.private_subnet_cidrs, count.index)  
  
  tags = {  
    Name = "Private Subnet ${count.index + 1}"  
  }  
}
```



Cómo utilizamos las herramientas de IaC

Rundeck

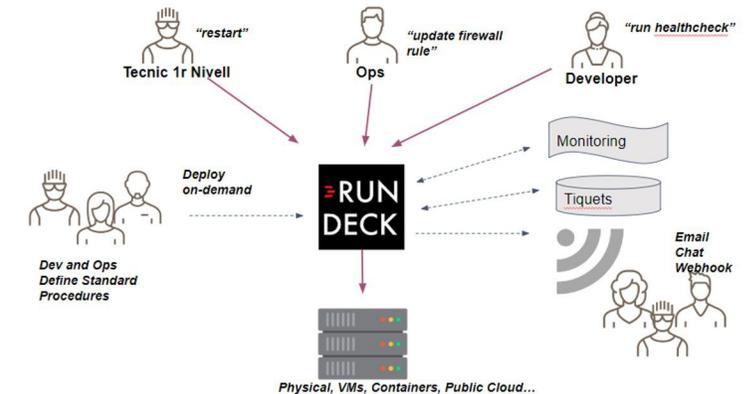
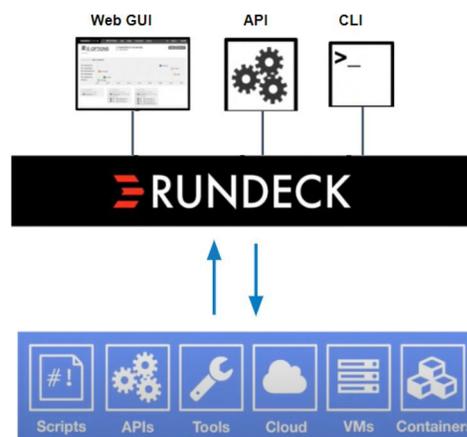
Rundeck es una consola centralizada para gestionar la automatización de su entorno.

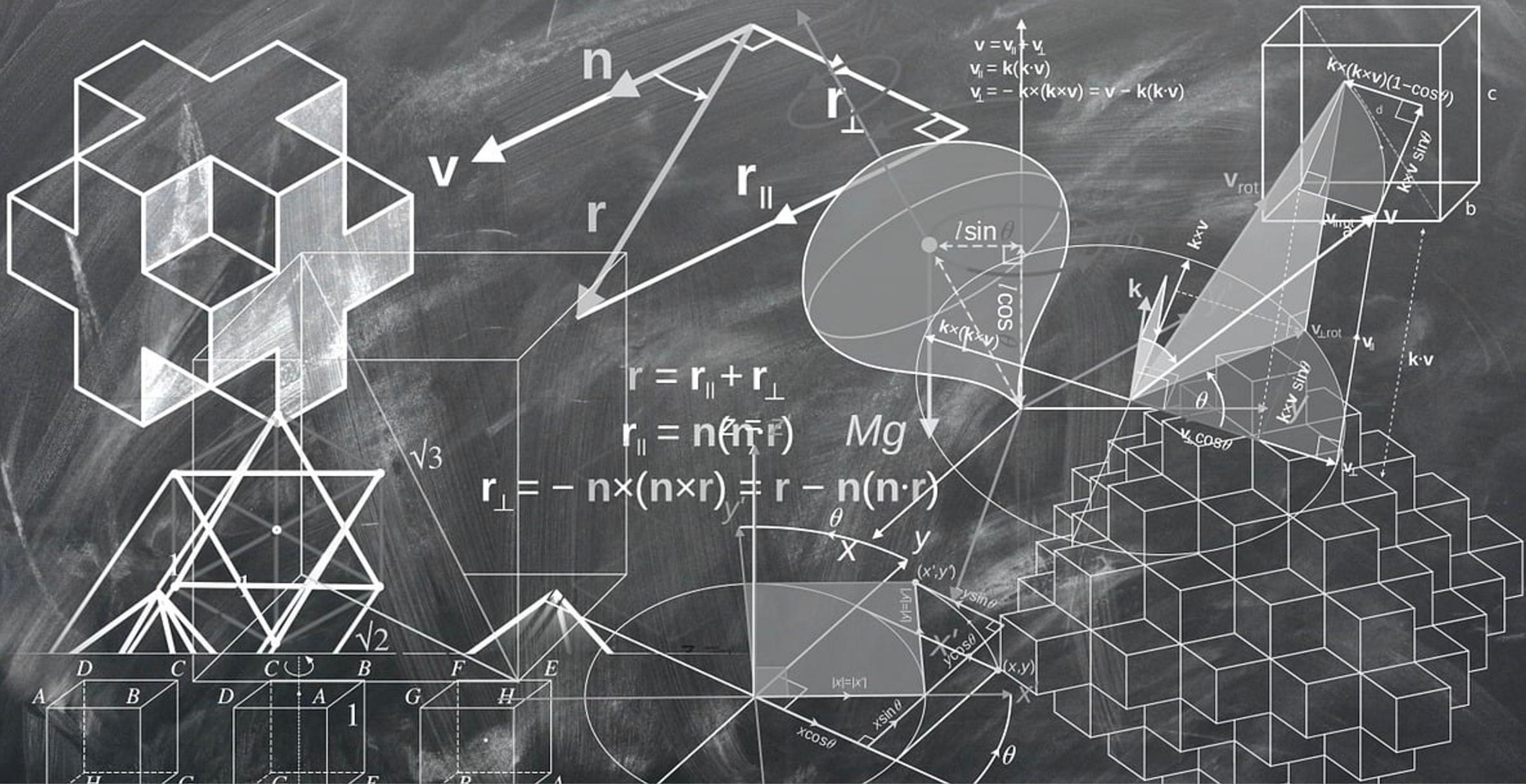
FUNCIONALIDADES

- Transforma un procedimiento o tarea **ejecutable bajo demanda o tipo cron**
- Permite **delegar la ejecución** a un usuario final, técnico no administrador o API

CASOS DE USO

- Despliegues (CI/CD)
- Soporte operacional
- Automatización Tareas





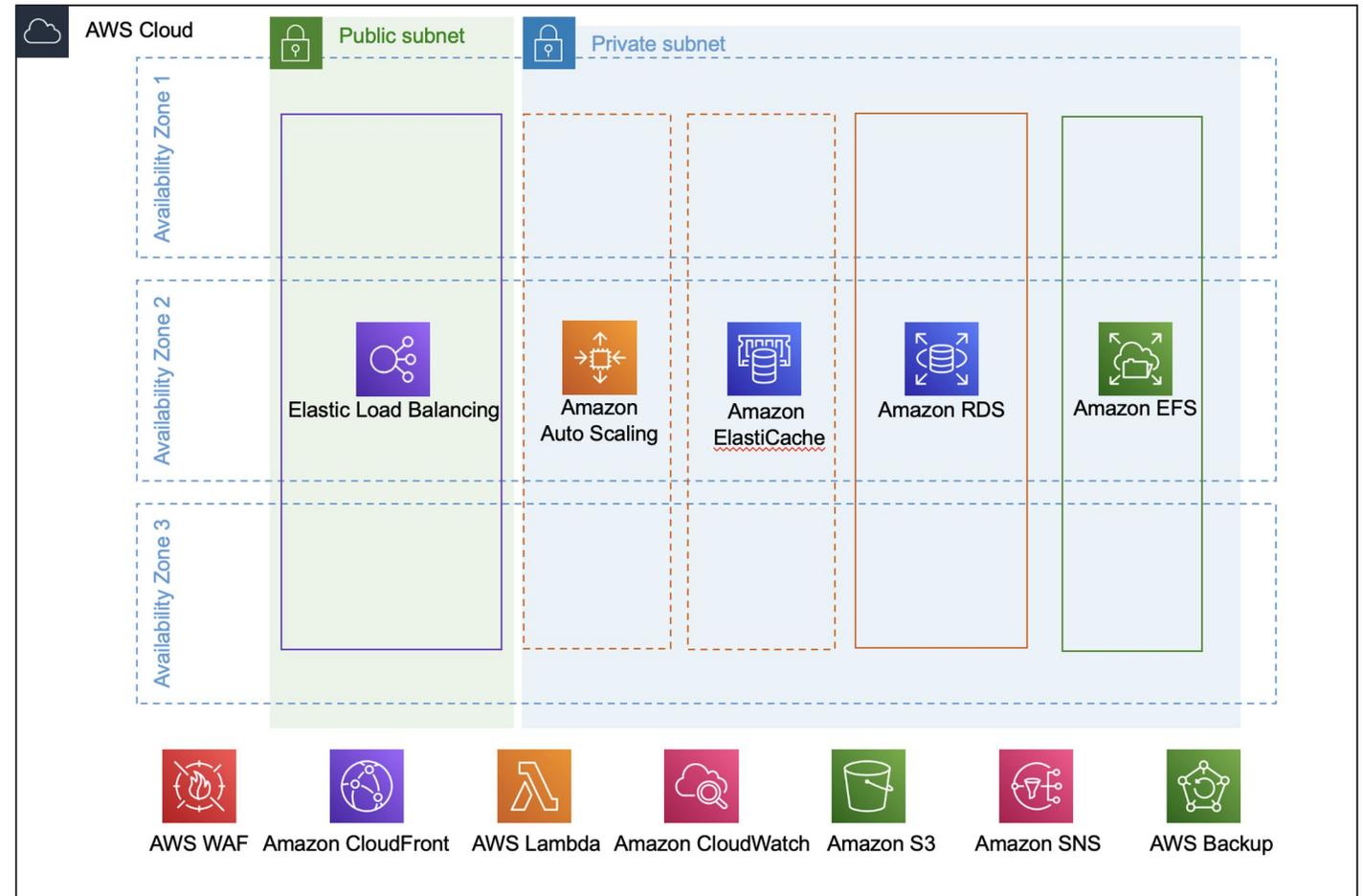
Arquitectura Plataforma Moodle sobre AWS

Priorizamos el uso de servicios gestionados

- RDS
- ElastiCache
- ALB
- CloudFront
- ASG
- Route53

Características & Ventajas

- Disponibilidad
- Resiliencia
- Seguridad
- Integración entre servicios gestionados



Cómo trabajamos en entornos

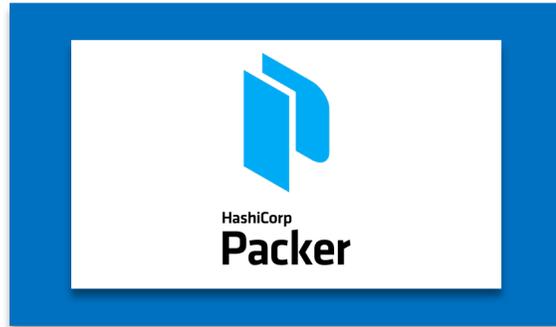


Imagen base

Instalación de software

Sistema Operativo

Discos adicionales

Posibilidad de copiar
archivos locales

Gestión de usuarios,
grupos y permisos



Arquitectura

Definición de los recursos
que forman parte de la
arquitectura

Aplicar modificaciones a
recursos existentes

Módulos reutilizables

Disaster Recovery



Automatismos

Despliegues de nuevos
campus

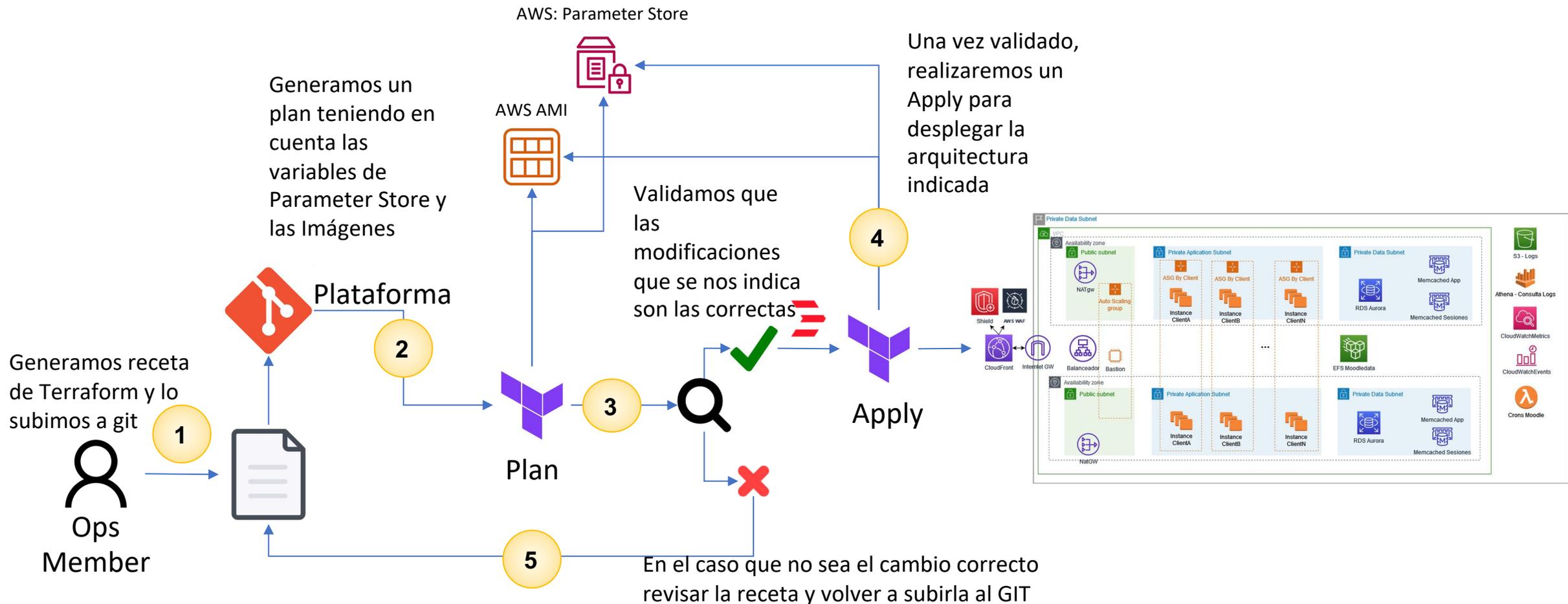
Subidas de versión

Banner de mantenimiento

Tareas programadas

Notificaciones para la
gestión de costes

Despliegue Campus virtual - Flujo Terraform

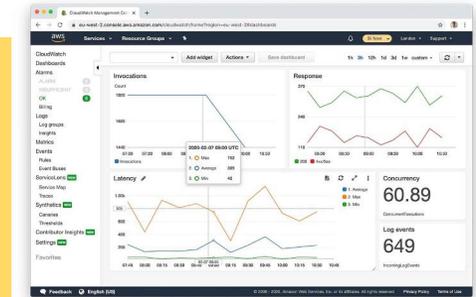


Cómo controlamos el estado de la plataforma

Monitorización

1

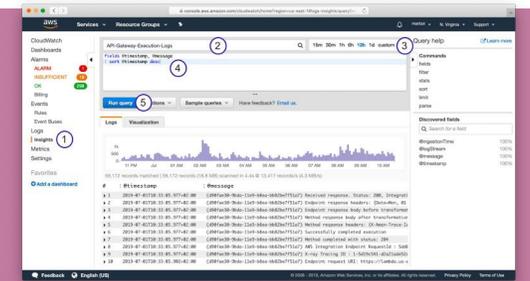
Agentes en EC2
Métricas en **Cloudwatch**
Generación de **Dashboards**



Logs

2

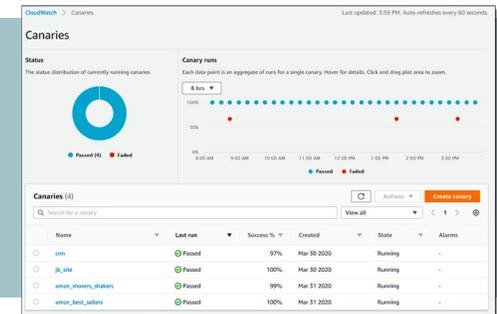
CloudWatch Logs
Queries
Insights



Calidad de Servicio

3

Synthetic AWS
Simulación Navegaciones
Vinculación con alarmas



Conclusiones y Recomendaciones

La **experiencia** en despliegue y administración de entornos **Moodle** con **Terraform** en entornos **Cloud** hace que podamos exponer las siguientes **conclusiones** y **recomendaciones**:



Potenciación
DEVOPS



Mejor control de
EVOLUCIÓN ENTORNO



Integración con CI/CD



Menor Coste
ADMINISTRACIÓN



Pruebas de Carga para validar evolución



Más
DISPONIBILIDAD

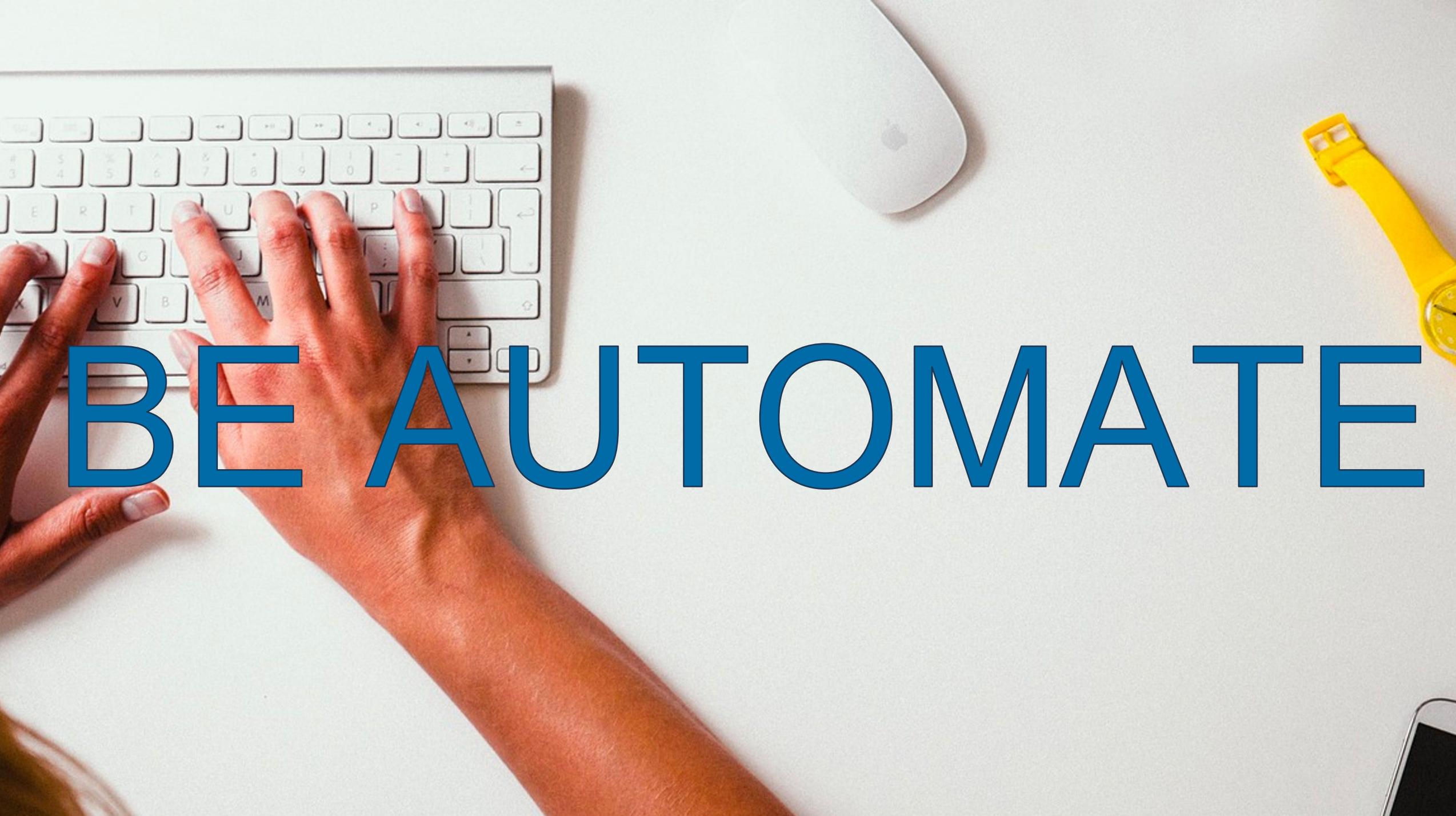


Utilización de Módulos de IaC (Terraform) comunes
Validación de cambios automática, marcha atrás simple



Uso de Servicios WAF y revisión seguridad entorno





BE AN AUTOMATE





Cristina Jerez Alvarez
cristina.jerez-alvarez@upcnet.es



David Garcia Domenech
david.garcia.domenech@upcnet.es