

ATENCIÓN: La prueba técnica completa debe ser entregada en un único repositorio

Prueba 1 - Diagrama de Red Produzca un diagrama de red (puede utilizar lucidchart) de una aplicación web en GCP o AWS y escriba una descripción de texto de 1/2 a 1 página de sus elecciones y arquitectura.

El diseño debe soportar:

- Cargas variables
- Contar con HA (alta disponibilidad)
- Frontend en Js
- Backend con una base de datos relacional y una no relacional
- La aplicación backend consume 2 microservicios externos

El diagrama debe hacer un mejor uso de las soluciones distribuidas.

Prueba 2 - Despliegue de una aplicación Django y React.js Elaborar el deployment dockerizado de una aplicación en django (backend) con frontend en React.js **contenida en el repositorio**. Es necesario desplegar todos los servicios en un solo docker-compose.

Se deben entregar los Dockerfiles pertinentes para elaborar el despliegue y justificar la forma en la que elabora el deployment (supervisor, scripts, docker-compose, kubernetes, etc)

Subir todo lo elaborado a un repositorio (github, gitlab, bitbucket, etc). En el repositorio se debe incluir el código de la aplicación y un archivo README.md con instrucciones detalladas para compilar y desplegar la aplicación, tanto en una PC local como en la nube (AWS o GCP).

Prueba 3 - CI/CD Dockerizar un nginx con el index.html default. Elaborar un pipeline que ante cada cambio realizado sobre el index.html buildee la nueva imagen y la actualize en la plataforma elegida. (docker-compose, swarm, kubernetes, etc.) Para la creación del CI/CD se puede utilizar cualquier plataforma (CircleCI, Gitlab, Github, Bitbucket.)

Requisitos y deseables:

La solución al ejercicio debe mostrarnos que usted puede:

Automatizar la parte del proceso de despliegue. usar conceptos de CI para aprovisionar el software necesario para que los entregables se ejecuten use cualquier herramienta de CI de su elección para implementar el entregable