

NOMBRE: Steven Salazar

## PASOS

### 1. Creacion de la aplicacion.

Se crea una ambiente virtual.  
Se instala las librerias para flask  
Se escribe la aplicación.

### 2. GIT

Creacion del repositorio en Github

<https://github.com/SteveOzo/testdevops>

Clonar el repositorio localmente

git clone <https://github.com/SteveOzo/testdevops.git>

Realizar los cambios en el programa y añadirlos al repositorio.

git add  
git commit  
git push

### 3. Implement Test

Para el Test se usó la librería de Python pytest.

Se escribió una prueba básica que valida que se manda el Nombre Juan y se valida el hostname.

### 4. Dockerize the application.

Creamos un archivo Dockerfile con los comandos que tendrá el contenedor.

El contenedor copiara todos los archivos de la carpeta local a una carpeta en el cotenedor

Instalara las librerias necesarias para que la aplicación corra

El contenedor ejecutara la aplicación al ser lanzado.

### 5. Load Balancer with nginx and docker-compose

Creamos un archivo de configuración para nginx el cual haga el nginx funcione como balanceador de carga.

Creamos el contenedor nginx al cual le copiamos la configuración

Creamos un archivo Docker-compose el cual levante dos instancias de la aplicación y la instancia de nginx y exista un vinculo entre el balanceador y las app.

Ejecutarmos el archivo Docker-compose con el comando

docker-compose up --scale app=2

Para ejecutar dos instancias de la app

### 6. Continuos Integration

Utilizando Jenkins escribimos un archivo jenkinsfile que indique las acciones a ejecutar para la tarea de despliegue.

Archivo de ejecución: **Jenkins**

Se debe almacenar en Jenkins las credenciales para acceder al repositorio Docker Hub.