



# **Centro de Enseñanza Técnica Industrial**

## **Desarrollo de Software**

### **Entrega Continua**

**Jesús Alberto Aréchiga Carrillo**

**22310439          6N**

**Profesor**

**José Francisco Pérez Reyes**

**Junio 2025**

**Guadalajara, Jalisco**

## Objetivo

Implementar un flujo de entrega continua donde:

1. Jenkins construye la imagen Docker de una aplicación.
2. La imagen se almacena localmente o en un *registry*.
3. Jenkins despliega automáticamente la aplicación como contenedor.

### Asegurarnos de:

- Jenkins instalado.
- Docker instalado y ejecutándose en la máquina de Jenkins.
- Plugin Docker Pipeline en Jenkins.
- Git (local o remoto) con la app de prueba.

### APP.PY (ejemplo)

```
from flask import Flask
```

```
app = Flask(__name__)
```

```
@app.route("/")
```

```
def home():
```

```
    return "Entrega Continua con Jenkins y Docker funcionando "
```

```
if __name__ == "__main__":
```

```
    app.run(host="0.0.0.0", port=5000)
```

## Documento requirements.txt

Flask

## Dockerfile

```
FROM python:3.10-slim
```

```
WORKDIR /app
```

```
COPY . .
```

RUN pip install -r requirements.txt

EXPOSE 5000

CMD ["python", "app.py"]

## Crear el Pipeline en Jenkins

1. Ve a Jenkins → *New Item* → **Pipeline** → MiAppCD.
2. En **Pipeline** → elegir opción *Pipeline script*.
3. Copia este ejemplo de pipeline declarativo:

```
pipeline {
    agent any
    stages {
        stage('Clonar repositorio') {
            steps {
                git 'https://github.com/usuario/mi-aplicacion.git'
            }
        }
        stage('Construir imagen Docker') {
            steps {
                script {
                    dockerImage = docker.build("mi-aplicacion:latest")
                }
            }
        }
        stage('Detener contenedor previo (si existe)') {
            steps {
                sh "docker stop miapp || true && docker rm miapp || true"
            }
        }
        stage('Desplegar contenedor') {
```

```
steps {  
  sh "docker run -d -p 5000:5000 --name miapp mi-aplicacion:latest"  
}  
}  
}  
}
```

## Configurar Jenkins para usar Docker

Verificar Docker en Jenkins

En Jenkins:

- Ve a: Manage Jenkins → Global Tool Configuration.
- Asegúrate que Jenkins tenga acceso a Docker.
- Si necesitas correr Jenkins con permisos de Docker:

### COMANDOS

```
sudo usermod -aG docker jenkins
```

```
sudo systemctl restart Jenkins
```

## Probar el Pipeline

1. Haz un cambio en tu repositorio.
2. Jenkins clonará el repo, creará la imagen, eliminará contenedores anteriores y levantará uno nuevo.
3. Visita: <http://localhost:5000>