

Parte I - Desarrollo de código:

El siguiente documento describe el diseño y justificación de un pipeline CI/CD utilizando GitHub Actions para automatizar el desarrollo, pruebas, y despliegue de una aplicación. Este pipeline garantiza la integridad del código y permite un despliegue rápido y seguro en un entorno de producción.

Diseño del Pipeline

El pipeline consta de tres etapas principales:

1. Integración de Pruebas:

- a. Se ejecuta pruebas unitarias para validar la funcionalidad de los componentes individuales.
- b. Se ejecuta pruebas de integración para garantizar que los módulos interactúen correctamente.

2. Automatización del Despliegue:

- a. Se genera los archivos estáticos de la aplicación.
- b. Se implementa la aplicación en un entorno de producción, como Netlify, usando configuraciones de despliegue automatizadas.

3. Rollback Automatizado:

- a. Si el despliegue falla, el pipeline revierte a la última versión funcional, garantizando la continuidad del servicio.

¿Por qué utilizar Github Actions?

1. Integración Nativa con GitHub:

- a. GitHub Actions está integrado directamente en los repositorios de GitHub, lo que elimina la necesidad de herramientas externas para la integración.
- b. Permite disparar flujos de trabajo basados en eventos del repositorio, como push o pull requests.

2. Facilidad de Uso y Configuración:

- a. La configuración del pipeline se realiza mediante archivos YAML simples y declarativos.
- b. GitHub Actions ofrece una amplia variedad de acciones reutilizables de la comunidad, facilitando la implementación.

Ejemplo de configuración del Pipeline

```
name: CI/CD Pipeline
```

```
on:
```

```
  push:
```

```
    branches:
```

```
      - main
```

```
jobs:
```

```
  build-and-test:
```

runs-on: ubuntu-latest

steps:

- name: Checkout code

uses: actions/checkout@v3

- name: Set up Node.js

uses: actions/setup-node@v3

with:

node-version: '16'

- name: Install dependencies

run: npm install

- name: Run unit tests

run: npm run test:unit

- name: Run integration tests

run: npm run test:integration

deploy:

needs: build-and-test

runs-on: ubuntu-latest

steps:

- name: Build project

run: npm run build

- name: Deploy to Netlify

uses: nwtgck/actions-netlify@v1.1.3

with:

publish-dir: ./dist

production-deploy: true

netlify-auth-token:\${{secrets.NETLIFY_AUTH_TOK

EN }}

netlify-site-id: \${{ secrets.NETLIFY_SITE_ID }}