

Documentación del Pipeline de CI/CD para el Frontend

Descripción del flujo de trabajo

El flujo de trabajo está definido en el archivo `deploy-frontend.yml`, ubicado en el directorio `.github/workflows/` del repositorio “FrontEnd-Grupo6” de GitHub (donde tenemos almacenado el frontend del proyecto). Este archivo detalla los pasos que se ejecutarán en cada push o merge a la rama master.

Pasos del Pipeline

1. Activación del Flujo de Trabajo

El flujo de trabajo se activa en cada push a la rama master del repositorio.

2. Configuración del Entorno

Se utiliza un entorno `ubuntu-latest` para ejecutar los pasos del flujo de trabajo.

3. Checkout del Repositorio

Se clona el repositorio en la máquina virtual de GitHub Actions para acceder al código fuente.

4. Configuración de Node.js

Se establece la versión de Node.js requerida para el proyecto.

5. Instalación de Dependencias

Se instalan las dependencias del proyecto utilizando `npm install`.

6. Ejecutar el Linter

Se ejecuta el linter configurado (ESLint) para verificar la calidad del código y garantizar que no haya errores de linting.

7. Construir el Proyecto

Se construye el proyecto utilizando `npm run build`, generando los archivos estáticos para el despliegue.

8. Despliegue a S3

Se sincronizan los archivos generados en el directorio `dist/` con el bucket de S3, utilizando las credenciales de AWS almacenadas de manera segura en los secretos del repositorio.

9. Invalidar la Caché de CloudFront

Se invalidan las rutas de la distribución de CloudFront para asegurar que los cambios recientes se reflejen en la entrega del contenido.

En la siguiente página se muestra el código del archivo `deploy-frontend.yml`.

```
1  name: Frontend Deployment
2
3  on:
4    push:
5      branches:
6        - master
7    pull_request:
8      branches:
9        - master
10
11  jobs:
12    deploy:
13      runs-on: ubuntu-latest
14
15      steps:
16        # 1. Checkout del código fuente del repositorio
17        - name: Checkout repository
18          uses: actions/checkout@v3
19
20        # 2. Configuración de Node.js
21        - name: Setup Node.js
22          uses: actions/setup-node@v3
23          with:
24            node-version: '18.x'
25
26        # 3. Instalación de dependencias
27        - name: Install dependencies
28          run: npm install
29
30        # 4. Ejecutar el linter
31        - name: Run linter
32          run: npm run lint
33
34        # 5. Construir el proyecto
35        - name: Build the project
36          run: npm run build
37          env:
38            VITE_AUTH0_DOMAIN: ${ secrets.VITE_AUTH0_DOMAIN }
39            VITE_AUTH0_CLIENT_ID: ${ secrets.VITE_AUTH0_CLIENT_ID }
40            VITE_API_URL: ${ secrets.VITE_API_URL }
41
42        # 6. Desplegar a S3
43        - name: Deploy to S3
44          run: aws s3 sync dist/ s3://${ secrets.S3_BUCKET } --delete
45          env:
46            AWS_ACCESS_KEY_ID: ${ secrets.AWS_ACCESS_KEY_ID }
47            AWS_SECRET_ACCESS_KEY: ${ secrets.AWS_SECRET_ACCESS_KEY }
48            AWS_REGION: ${ secrets.AWS_REGION }
49
50        # 7. Invalidar CloudFront cache
51        - name: Invalidate CloudFront
52          run: aws cloudfront create-invalidation --distribution-id ${ secrets.CLOUDFRONT_DISTRIBUTION_ID } --paths "/*"
53          env:
54            AWS_ACCESS_KEY_ID: ${ secrets.AWS_ACCESS_KEY_ID }
55            AWS_SECRET_ACCESS_KEY: ${ secrets.AWS_SECRET_ACCESS_KEY }
56            AWS_REGION: ${ secrets.AWS_REGION }
```