

TP 8 - Herramientas de construcción de software en la nube

Github actions para SimpleWebAPI

The image shows two screenshots from the GitHub Actions interface. The top screenshot displays the workflow configuration file `main.yml` for the repository `SimpleWebAPI`. The workflow is named "Build and Publish" and is triggered on a `workflow_dispatch` event. It runs on the `ubuntu-latest` runner and consists of several steps: checking out the code, setting up the .NET Core environment (version 7.0.x), and restoring dependencies. The bottom screenshot shows the execution history of the workflow. It was triggered via a push 1 minute ago, resulting in a successful run with a total duration of 56 seconds and 1 artifact generated. The workflow diagram shows two steps: `build` (33s) and `deploy` (3s).

```
1 name: Build and Publish
2
3 on:
4   workflow_dispatch:
5     push:
6       branches:
7         - main
8
9 jobs:
10  build:
11    runs-on: ubuntu-latest
12
13    steps:
14      - name: Checkout code
15        uses: actions/checkout@v3
16
17      - name: Setup .NET Core
18        uses: actions/setup-dotnet@v3
19        with:
20          dotnet-version: 7.0.x
21
22      - name: Restore dependencies
```

Triggered via push 1 minute ago

	Status	Total duration	Artifacts
sofiacersofios pushed → 9dd61e1 main	Success	56s	1

main.yml
on: push

```
graph LR
    build[build 33s] --> deploy[deploy 3s]
```

Este pipeline automatiza dos tareas principales: hacer el **build** y **deploy** de una aplicación. El pipeline tiene un flujo de trabajo con varios pasos que se ejecutan en respuesta a eventos específicos.

Realiza los siguientes pasos:

- Clona el repositorio.
- Configura el entorno para .NET Core.
- Restaura las dependencias del proyecto.
- Compila el proyecto.
- Publica la aplicación en un directorio específico.
- Sube los artefactos generados.
- Descarga los artefactos para el despliegue
- Muestra el contenido de los artefactos.

- Simula un paso de despliegue.

Workflow en GitHub Actions para generar una imagen de Docker de SimpleWebApi y subirla a DockerHub

The screenshot shows the GitHub Actions interface for a workflow named 'Create dockerimage.yml' in the repository 'SimpleWebAPI'. The workflow is triggered on a push to the 'main' branch. The configuration file 'dockerimage.yml' is shown with the following steps:

```

1  name: Docker Image CI
2
3  on:
4    workflow_dispatch:
5    push:
6      branches: [ "main" ]
7
8  jobs:
9    build:
10     runs-on: ubuntu-latest
11     steps:
12       - name: Checkout code
13         uses: actions/checkout@v2
14
15       - name: Build the Docker image
16         run: docker build . --file Dockerfile --tag ${{ secrets.DOCKERHUB_USERNAME }}/simple-web-api-gh:latest
17
18       - name: Log in to Docker Hub
19         run: docker login -u ${{ secrets.DOCKERHUB_USERNAME }} -p ${{ secrets.DOCKERHUB_PASSWORD }}
20
21       - name: Push Docker image to Docker Hub
22         run: |
23           docker push ${{ secrets.DOCKERHUB_USERNAME }}/simple-web-api-gh:latest

```

The workflow run is shown as successful, triggered via push now, with a status of 'Success', a total duration of 52s, and 1 artifact. The workflow steps are visualized as 'build' (27s) and 'deploy' (7s).

Imagen publicada en DockerHub

The screenshot shows the DockerHub page for the image 'sofiacersofios/simple-web-api-gh'. The image is published by 'sofiacersofios' and was updated 4 minutes ago. The image is available for pulling from DockerHub.

```

PS C:\Users\sofia\OneDrive\Escritorio\IdS3\tp8> docker pull sofiacersofios/simple-web-api-gh:latest
latest: Pulling from sofiacersofios/simple-web-api-gh
e67fdae35593: Pull complete
0ab66724116f: Pull complete
14ccddeb1bc: Pull complete
5e265b51b431: Pull complete
3bda6efdfc5b: Pull complete
44efb2de2c71: Pull complete
4f4fb700ef54: Pull complete
c1d9c6729c24: Pull complete
Digest: sha256:599695f7ef1d4baff805c7ccf0a530a8bbdbb1a4a0cde12cc01bfbf83861857
Status: Downloaded newer image for sofiacersofios/simple-web-api-gh:latest
docker.io/sofiacersofios/simple-web-api-gh:latest

```

```
PS C:\Users\sofia\OneDrive\Escritorio\IdS3\tp8> docker run --name myapi -d -p 8080:80 sofiacersofios/simple-web-api-gh-e6789a3a5f57b868ca9e813b418307ca2cf8ae8448fb4fab438311b84598906e
PS C:\Users\sofia\OneDrive\Escritorio\IdS3\tp8> |
```

sofiacersofios/simple-w...

IN USE

latest

9c3623bd8b66

9 minutes ago

215.55 MB

⋮

RUN ▶

localhost:8080/weatherforecast

```
[{"date": "2023-10-17", "temperatureC": 41, "temperatureF": 105, "summary": "Scorching"}, {"date": "2023-10-18", "temperatureC": 31, "temperatureF": 87, "summary": "Calido"}, {"date": "2023-10-19", "temperatureC": 36, "temperatureF": 96, "summary": "Scorching"}, {"date": "2023-10-20", "temperatureC": 12, "temperatureF": 53, "summary": "Caloron"}, {"date": "2023-10-21", "temperatureC": 46, "temperatureF": 114, "summary": "Sweltering"}]
```

Crear una GitHub Action que genere los artefactos para el proyecto React

name: Build React App

on:

push:

branches:

- main

jobs:

build:

runs-on: ubuntu-latest

steps:

- name: Checkout code

uses: actions/checkout@v2

- name: Set up Node.js

uses: actions/setup-node@v2

with:

node-version: 14

- name: Install dependencies

run: npm install

- name: Build the React app

run: npm run build

- name: Upload Artifacts

uses: actions/upload-artifact@v2

with:

name: build

path: build

main.yml

on: push

✓ build

41s

⌵

⌵

⌵

build
succeeded 1 minute ago in 41s

Search logs

- > ✓ Set up job 2s
- > ✓ Checkout code 1s
- > ✓ Set up Node.js 0s
- > ✓ Install dependencies 24s
- > ✓ Build the React app 9s
- > ✓ Upload Artifacts 2s
- > ✓ Post Set up Node.js 0s
- > ✓ Post Checkout code 0s
- > ✓ Complete job 0s

Crear una GitHub Action que genere una imagen de Docker para el proyecto React y lo suba a DockerHub

```
1  name: Build and Upload Docker Image
2
3  on:
4    push:
5      branches:
6        - main
7
8  jobs:
9    build:
10     runs-on: ubuntu-latest
11
12     steps:
13       - name: Checkout Repository
14         uses: actions/checkout@v2
15
16       - name: Build Docker Image
17         run: docker build . --file Dockerfile --tag ${ secrets.DOCKERHUB_USERNAME }}/my-react-app:latest
18
19       - name: Log in to Docker Hub
20         run: docker login -u ${ secrets.DOCKERHUB_USERNAME } -p ${ secrets.DOCKERHUB_PASSWORD }
21
22       - name: Push Docker Image to Docker Hub
23         run: docker push ${ secrets.DOCKERHUB_USERNAME }}/my-react-app:latest
24
25       - name: Clean up
26         run: docker logout
27       if: always() # Se ejecutará incluso si un paso anterior falla
```

main.yml
on: push

✓ build 53s

[- +]

build
succeeded 1 minute ago in 53s


Search logs


>	✓ Set up job	1s
>	✓ Checkout Repository	1s
>	✓ Build Docker Image	37s
>	✓ Log in to Docker Hub	0s
>	✓ Push Docker Image to Docker Hub	11s
>	✓ Clean up	0s
>	✓ Post Checkout Repository	0s
>	✓ Complete job	0s


Imagen subida a DockerHub


sofiacersofios / my-react-app

Contains: Image | Last pushed: a minute ago

 Inactive

 0

 0

 Public