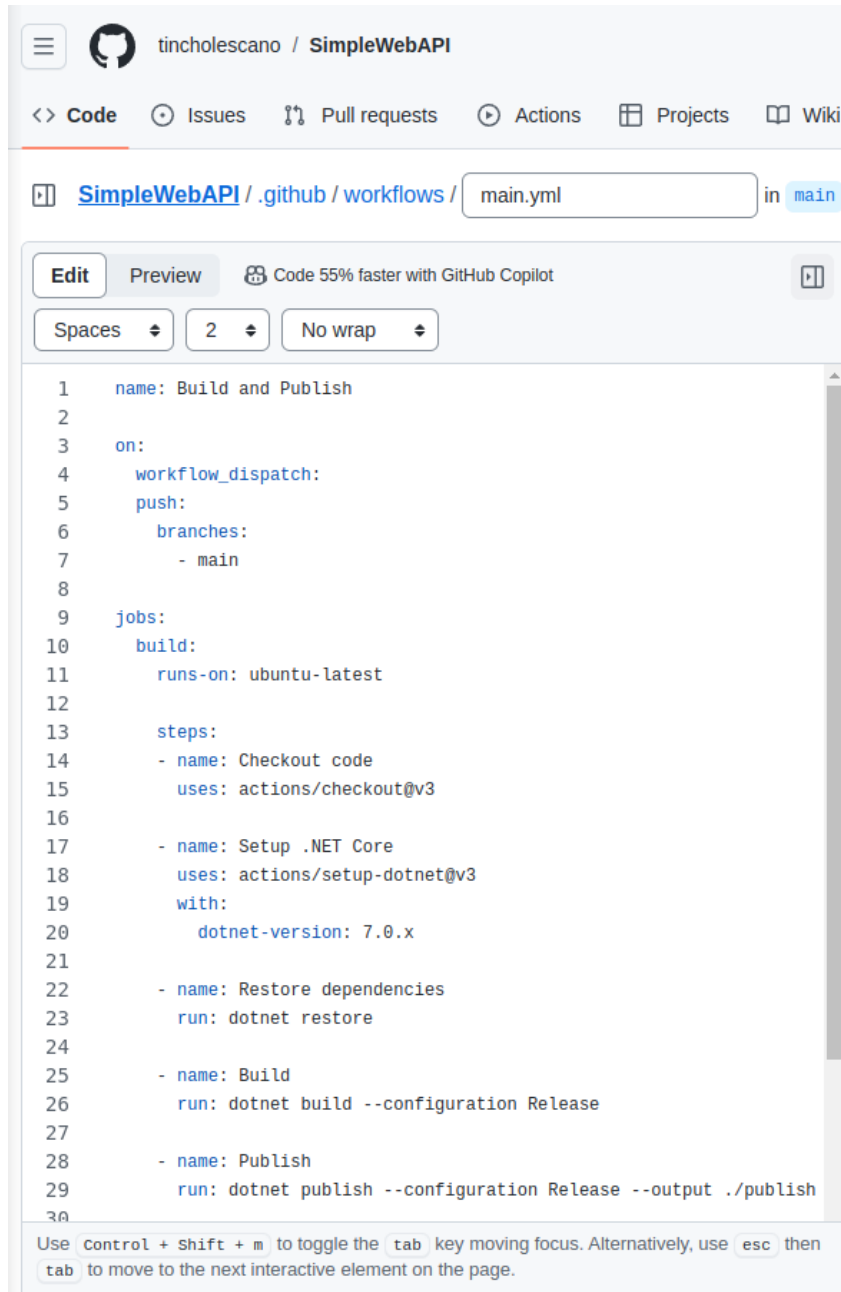


Trabajo Práctico 8 - Herramientas de construcción de software en la nube

2- Configurando GitHub Actions



The screenshot shows the GitHub Actions workflow editor for the repository 'tincholescano / SimpleWebAPI'. The breadcrumb navigation indicates the file path: 'SimpleWebAPI / .github / workflows / main.yml' in the 'main' branch. The editor interface includes tabs for 'Edit' and 'Preview', a 'Code 55% faster with GitHub Copilot' badge, and formatting controls for 'Spaces' (set to 2) and 'No wrap'. The workflow file content is as follows:

```
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
23         run: dotnet restore
24
25       - name: Build
26         run: dotnet build --configuration Release
27
28       - name: Publish
29         run: dotnet publish --configuration Release --output ./publish
30
```

At the bottom of the editor, a tip states: 'Use Control + Shift + m to toggle the tab key moving focus. Alternatively, use esc then tab to move to the next interactive element on the page.'

Commit changes



Commit message

Create main.yml

Extended description

Add an optional extended description..

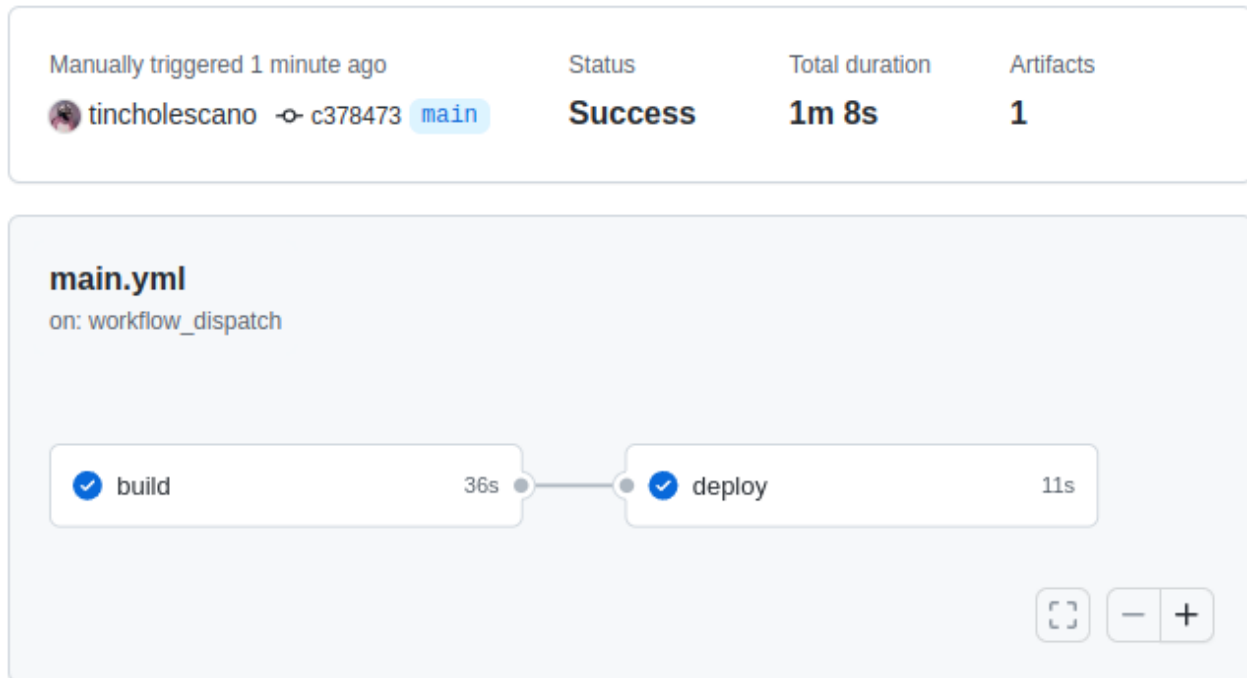
Commit Email

lescanomartin.ml@gmail.com ↕

- ☒ Commit directly to the `main` branch
- ☐ Create a **new branch** for this commit and start a pull request
[Learn more about pull requests](#)

Cancel

Commit changes



El pipeline de acciones se utiliza para construir y publicar una aplicación .NET Core y, posteriormente, desplegarla en un servidor. Este pipeline se encuentra definido en un archivo de flujo de trabajo de GitHub y consta de dos trabajos: "build" y "deploy". A continuación, explicaré paso a paso lo que realiza este pipeline:

Trabajo "build":







- Entorno de Ejecución: Este trabajo se ejecuta en una instancia de Ubuntu.
- Pasos del Trabajo "build":
 - Checkout Code: Se utiliza la acción `actions/checkout` para clonar el código fuente del repositorio.
 - Setup .NET Core: Se utiliza la acción `actions/setup-dotnet` para configurar la versión de .NET Core. En este caso, se configura para usar la versión 7.0.x.
 - Restore Dependencies: Se utiliza el comando `dotnet restore` para restaurar las dependencias del proyecto.
 - Build: Se utiliza el comando `dotnet build` para compilar la aplicación en modo Release.
 - Publish: Se utiliza el comando `dotnet publish` para crear una versión publicada de la aplicación en el directorio `./publish`.
 - Upload Artifacts: Se utiliza la acción `actions/upload-artifact` para cargar (subir) los archivos publicados al artefacto llamado "app".

Trabajo "deploy":

- Dependencia de "build": Este trabajo depende de que "build" se complete con éxito antes de ejecutarse.
- Entorno de Ejecución: Este trabajo también se ejecuta en una instancia de Ubuntu.
- Pasos del Trabajo "deploy":
 - Download Artifacts: Se utiliza la acción `actions/download-artifact` para descargar los archivos del artefacto "app" creado en el trabajo "build".

- Output Contents: Se muestra el contenido del directorio de trabajo actual utilizando el comando "ls".
- Deploy to Server: En este paso, se puede agregar lógica personalizada para el despliegue de la aplicación en un servidor. En el ejemplo proporcionado, se imprime "Deploy", pero este paso generalmente se utilizaría para realizar tareas de despliegue específicas, como copiar los archivos publicados a un servidor de destino.

3- Configurar un workflow en GitHub Actions para generar una imagen de Docker de SimpleWebApi y subirla a DockerHub

Repository secrets			
	PASSWORD	Updated now	 
	USER_NAME	Updated now	 


SimpleWebAPI / .github / workflows /

imgDH.yml

in main

Edit

Preview

 Code 55% faster with GitHub Copilot



Spaces



2



No wrap



```
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.USER_NAME }/simple-
17
18     - name: Log in to Docker Hub
19       run: docker login -u ${ secrets.USER_NAME } -p ${ secrets.PASSWORD }
20
21     - name: Push Docker image to Docker Hub
22       run: |
23         docker push ${ secrets.USER_NAME }/simple-web-api-gh:latest
24
25     - name: Clean up
26       run: docker logout
27       if: always() # Se ejecutará incluso si un paso anterior falla
28
```

Commit changes



Commit message

Create imgDH.yml

Extended description

Add an optional extended description..

Commit Email

lescanomartin.ml@gmail.com ↕

- ☒ Commit directly to the main branch
- ☐ Create a **new branch** for this commit and start a pull request
- [Learn more about pull requests](#)

Cancel

Commit changes

martinlescano / **simple-web-api-gh**

Contains: Image | Last pushed: a few seconds ago

Inactive

☆ 0

↓ 0

Public

```
tincho@IdeaPad-5-14ALC05:~/Documents/Ingenieria de Software III/SimpleWebAPI$ docker pull martinlescano/simple-web-api-gh:latest
latest: Pulling from martinlescano/simple-web-api-gh
e67fdae35593: Already exists
0ab66724116f: Already exists
14ccddebb1bc: Already exists
5e265b51b431: Already exists
3bda6efdfc5b: Already exists
5f09473a413e: Pull complete
4f4fb700ef54: Pull complete
eb423430718d: Pull complete
Digest: sha256:20ac3af014874d5f1d9b187f0117affb2104564fcb7954d83c679e3cfe1775ce
Status: Downloaded newer image for martinlescano/simple-web-api-gh:latest
docker.io/martinlescano/simple-web-api-gh:latest
```

What's Next?

View a summary of image vulnerabilities and recommendations → `docker scout quickview martinlescano/simple-web-api-gh:latest`

```
tincho@IdeaPad-5-14ALC05:~/Documents/Ingenieria de Software III/SimpleWebAPI$ docker run --name myapi -d -p 8080:80 martinlescano/simple-web-api-gh
26ad3e709022cfa6103a35744fd18b77331fe1c90b1beabd500dcab38a492bf2
```

```
[{"date": "2023-10-18", "temperatureC": 50, "temperatureF": 121, "summary": "Sweltering"}, {"date": "2023-10-19", "temperatureC": 37, "temperatureF": 98, "summary": "Sweltering"}, {"date": "2023-10-20", "temperatureC": -7, "temperatureF": 20, "summary": "Sweltering"}, {"date": "2023-10-21", "temperatureC": 9, "temperatureF": 48, "summary": "Calido"}, {"date": "2023-10-22", "temperatureC": 4, "temperatureF": 39, "summary": "Scorching"}]
```