

plain concepts 

Plain Concepts Bilbao Team

27.03.2020

Github Actions

Jorge Turrado Ferrero (Feat. Alberto Gimeno)
Software Development Engineer



Jorge Turrado

Software Development Engineer
Microsoft MVP

Twitter: @JorgeTurrado

Blog: <https://www.fixedbuffer.com>

Cursos:

- Testing de aplicaciones .NET y .NET Core con xUnit y Moq (CampusMVP)

¿Qué es Github Actions?

- Github Actions es una herramienta integrada dentro de Github que nos permite ejecutar flujos de trabajo de manera automática.

¿Cómo trabaja Github Actions?

1. Registramos un flujo de trabajo con un disparador
2. Cuando el disparador se cumple se ejecuta el trabajo
3. Una vez que el trabajo termina se notifica el resultado

Desarrollo asistido

The screenshot displays the GitHub Actions workflow editor interface. The main editor shows a workflow file with the following content:

```
1 name: CI
2
3 on: [push]
4
5 jobs:
6   build:
7
8     strategy:
9       matrix:
10        agent: [ 'windows-latest', 'ubuntu-latest', 'macos-latest' ]
11      runs-on: ${ matrix.agent }
12      name: Integration in ${matrix.agent}
13      steps:
14        - uses: actions/checkout@v2
15        - uses: actions/setup-dotnet@v1
16          with:
17            dotnet-version: '2.2.108' # SDK Version to use.
18        - run: dotnet build "Ejemplo NetCore.sln"
19        - run: dotnet test "Ejemplo NetCore.sln"
20
```

The right sidebar shows the 'Marketplace' tab with a search bar and a list of featured actions:

- Setup Node.js for use with actions** (388 stars)
By actions ✓
Setup a Node.js environment and add it to the PATH, additionally providing proxy support
- Upload artifact** (352 stars)
By actions ✓
Publish files as workflow artifacts
- Download artifact** (109 stars)
By actions ✓
Download workflow artifacts

Modelo de trabajos

```
name: CI

on: [push]

jobs:
  windows:
    runs-on: 'windows-latest'
    steps:
      - uses: actions/checkout@v2
        with:
          submodules: 'recursive'
      - uses: actions/checkout@v1
      - name: configure
        run: mkdir build && cd build && cmake ../..
      - name: build
        run: cmake --build build

  ubuntu:
    runs-on: 'ubuntu-latest'
    steps:
      - uses: actions/checkout@v2
        with:
          submodules: 'recursive'
      - uses: actions/checkout@v1
      - name: configure
        run: mkdir build && cd build && cmake ../..
      - name: build
        run: cmake --build build
```

Search

Features

▶

▶

▶


▶


Marketplace

[Marketplace](#) [Documentation](#)

Search Marketplace for Actions


Featured Actions


**Setup Node.js for use with actions**

By actions 

Setup a Node.js environment and add it to the PATH, additionally providing proxy support


★ 388


**Setup Java JDK**

By actions 

Set up a specific version of the Java JDK and add the command-line tools to the PATH


★ 146


**Close Stale Issues**

By actions 

Action to close stale issues


★ 135

**Setup .NET Core SDK**

By actions 

Set up a specific version of the .NET Core CLI in the PATH and set up authentication to a private NuGet repository

★ 132

**GraphQL query**

By helaili

An action that acts a client for GitHub's GraphQL API

★ 54

Estrategias

```
strategy:
  matrix:
    agent: [ 'windows-latest', 'ubuntu-latest', 'macos-latest' ]
    configuration: ['debug', 'release']

runs-on: ${ matrix.agent }
name: Integration in ${matrix.agent}

steps:
- uses: actions/checkout@v2
- uses: actions/setup-dotnet@v1
  with:
    dotnet-version: '2.2.108' # SDK Version to use
- run: dotnet build "Ejemplo NetCore.sln" -c ${matrix.configuration}
- run: dotnet test "Ejemplo NetCore.sln" -c ${matrix.configuration}
```


Servicios

```
name: Build
runs-on: ubuntu-latest
services:
  sql-server:
    image: microsoft/mssql-server-linux
    env:
      ACCEPT_EULA: Y
      MSSQL_SA_PASSWORD: Password12!
    ports:
      - 5433:1433
steps:
  - uses: actions/checkout@v1
  - name: Setup .NET Core SDK
    uses: actions/setup-dotnet@v1.1.0
```

Agentes

1. Agentes hospedados por Github
2. Agentes autohospedados

¿Y qué nos falta?

1. Posibilidad de ejecutar triggers manuales
2. Posibilidad de crear templates
3. Posibilidad de crear integraciones multistage

plain concepts 

iGRACIAS!

www.plainconcepts.com

[@plainconcepts](https://twitter.com/plainconcepts)



MADRID

Paseo de la Castellana 163, 10º
28046 Madrid. España
T. (+34) 91 5346 836



BARCELONA

Carrer Compte d'Urgell 240 4º 1A
08036 Barcelona. España
T. (+34) 93 7978 566



BILBAO

Calle Ledesma 10-bis 3º
48001 Bilbao. España
T. (+34) 94 6073 371



SEVILLA

Avenida de la innovación s/n
Edificio Renta Sevilla, 3º A
41020 Sevilla. España



DUBAI

Dubai Internet City. Building 1
73030 Dubai. EAU
T. (+971) 4 551 6653



LONDON

Impact Hub Kings Cross
24B York Way, N1 9AB
London. UK



SEATTLE

1511, Third Ave
Seattle WA 98101. USA
T. (+1) 206 708 1285