

- 4.1 Verificar acceso a Pipelines concedido

The screenshot displays the 'Jobs in run #20240827.1' for 'sample02'. On the left, a table lists the tasks and their durations. On the right, a console window shows the job's execution details.

Task	Duration
Job	1m 14s
Initialize job	13s
Checkout sample02@m...	6s
NuGetToolInstaller	<1s
NuGetCommand	22s
VSBuild	28s
VSTest	1s
Post-job: Checkout sa...	<1s
Finalize Job	<1s
Report build status	<1s

**Job Details:**

- 1 Pool: [Azure Pipelines](#)
- 2 Image: windows-latest
- 3 Agent: Hosted Agent
- 4 Started: Just now
- 5 Duration: 1m 14s
- 6
- 7 ▶ Job preparation parameters
- 42 Job live console data:
- 43 Finishing: Job

- 4.2 Agregar en pipeline YAML una tarea de Publish.

```

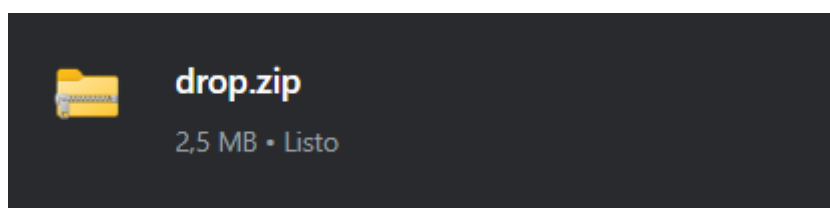
Settings
- task: DotNetCoreCLI@2
  inputs:
    command: publish
    publishWebProjects: True
    arguments: '--configuration $(BuildConfiguration) --output $(Build.ArtifactStagingDirectory) --zipAfterPublish: true
Settings
- task: PublishBuildArtifacts@1
  inputs:
    PathtoPublish: '$(Build.ArtifactStagingDirectory)'
    ArtifactName: 'drop'
    publishLocation: 'Container'

```

- 4.3 Explicar por qué es necesario contar con una tarea de Publish en un pipeline que corre en un agente de Microsoft en la nube.

Porque permiten que los archivos generados en la ejecución del pipeline, como ejecutables, se almacenen y se encuentren disponibles. Estos después se pueden usar para ejecutarlos o en etapas posteriores, como test y deploy.

- 4.4 Descargar el resultado del pipeline y correr localmente el software compilado.



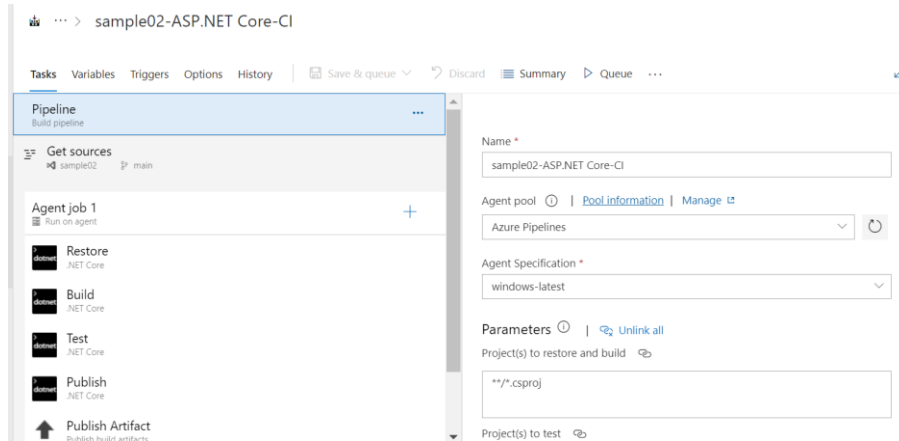
- 4.5 Habilitar el editor clásico de pipelines. Explicar las diferencias claves entre este tipo de editor y el editor YAML.



#### Disable creation of classic build pipelines

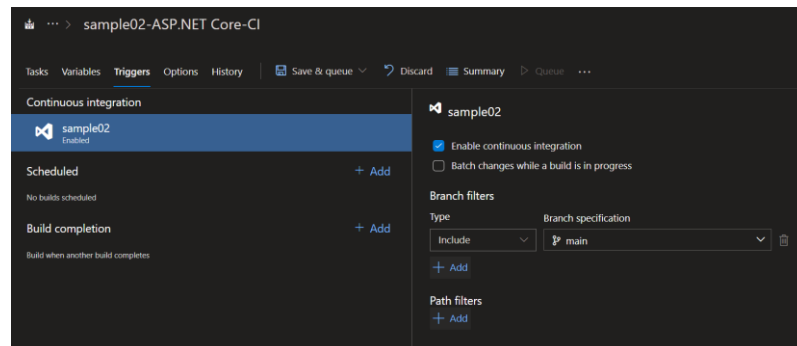
No classic build pipelines can be created / imported. Existing ones will continue to work.

- 4.6 Crear un nuevo pipeline con el editor clásico. Descargar el resultado del pipeline y correr localmente el software compilado.

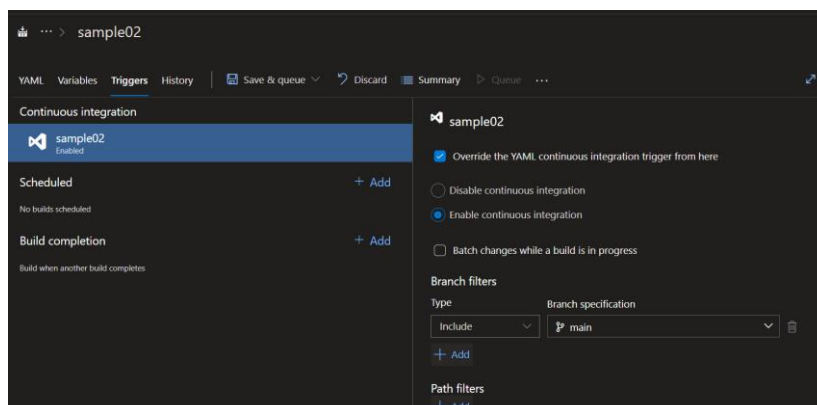


- 4.7 Configurar CI en ambos pipelines (YAML y Classic Editor). Mostrar resultados de la ejecución automática de ambos pipelines al hacer un commit en la rama main.

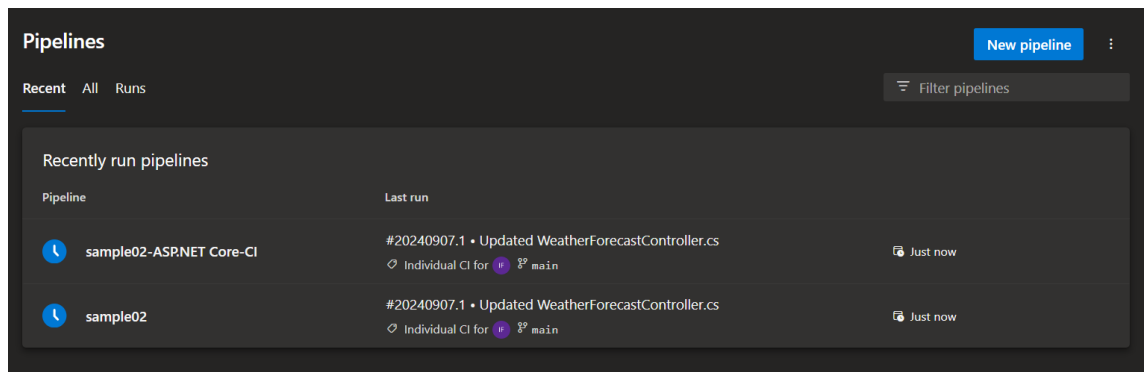
## Classic Editor



## YAML



Al hacer un cambio y commitarlo, se reinician automáticamente los pipelines



- 4.8 Explicar la diferencia entre un agente MS y un agente Self-Hosted. Qué ventajas y desventajas hay entre ambos? Cuándo es conveniente y/o necesario usar un Self-Hosted Agent?

Agente MS-Hosted (Microsoft-hosted agent): Este tipo de agente es proporcionado y gestionado por Microsoft. Está listo para ser utilizado con solo configurarlo en Azure DevOps, sin necesidad de instalación o mantenimiento por parte del usuario.

Ventajas: Microsoft provee y actualiza estos agentes. Puedes empezar a usarlos de inmediato, lo que los hace ideales para proyectos pequeños o pruebas rápidas. Dispone de varias imágenes de sistemas operativos preconfiguradas con herramientas comunes (Windows, Ubuntu, MacOS).

Desventajas: Si hay muchos usuarios usando agentes MS-hosted en el mismo momento, puede haber tiempos de espera para que el pipeline se ejecute. Genera costos.

--

Agente Self-Hosted: Los agentes Self-Hosted son configurados y mantenidos por el usuario en su propia infraestructura, ya sea en servidores físicos, máquinas virtuales o contenedores Docker.

Ventajas: Puedes configurar el entorno a tu medida. Esto incluye instalar cualquier software o herramienta adicional que necesites para tu pipeline. No hay tiempos de espera. Sin límites en minutos. Si tienes hardware potente y optimizado, puedes mejorar los tiempos de ejecución.

Desventajas: Eres responsable de configurar, actualizar, y mantener el agente, incluyendo la instalación de las herramientas necesarias. Costos de infraestructura como en servidores, almacenamiento, etc.

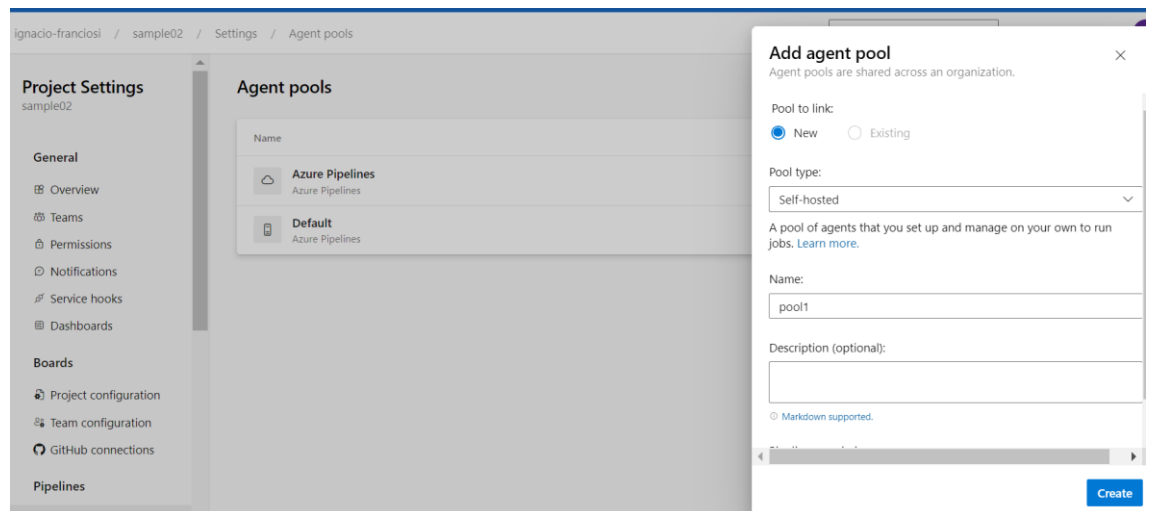
### ¿Cuándo es conveniente usar un Self-Hosted Agent?

Si tu proyecto necesita software específico o versiones que no están disponibles en los agentes MS-hosted.

Cuando necesitas un entorno muy específico que los agentes de Microsoft no pueden ofrecer.

Privacidad o seguridad, es decir, cuando tus pipelines contienen información sensible o no deseas que se ejecute en servidores externos, por razones de compliance o políticas de seguridad.

- 4.8 Crear un Pool de Agentes y un Agente Self-Hosted



Token: crjd3ryxqodknneals623ow52pj3d67tiu4wautmxafq7l7rrttq

- 4.9 Instalar y correr un agente en nuestra máquina local.

Get the agent

×

Windows

macOS

Linux

x64

x86

[System prerequisites](#)

Configure your account

Configure your account by following the steps outlined here.

Download the agent

Download



Create the agent

```
PS C:\> mkdir agent ; cd agent
PS C:\agent> Add-Type -AssemblyName System.IO.Compression.FileSystem ;
[System.IO.Compression.ZipFile]::ExtractToDirectory("$HOME\Downloads\vsts
agent-win-x64-3.243.1.zip", "$PWD")
```

Configure the agent [Detailed instructions](#)

```
PS C:\agent> .\config.cmd
```

Optionally run the agent interactively

If you didn't run as a service above:

```
PS C:\agent> .\run.cmd
```

That's it!

```
Administrador: Símbolo del sistema
Microsoft Windows [Versión 10.0.22631.4112]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Windows\System32>cd C:\agents\vsts-agent-win-x64-3.243.1

C:\agents\vsts-agent-win-x64-3.243.1>.\config.cmd

          agent v3.243.1          (commit 3bb22cd)


>> Connect:

Enter server URL > https://dev.azure.com/ignacio-franciosi/
Enter authentication type (press enter for PAT) >
Enter personal access token > *****
Connecting to server ...


>> Register Agent:

Enter agent pool (press enter for default) >
Enter agent name (press enter for DESKTOP-VB9BSN0) > AgentSelfHosted
Scanning for tool capabilities.
Connecting to the server.
Successfully added the agent
Testing agent connection.
Enter work folder (press enter for _work) >
2024-09-08 01:15:34Z: Settings Saved.
Enter run agent as service? (Y/N) (press enter for N) > Y
Enter enable SERVICE_SID_TYPE_UNRESTRICTED for agent service (Y/N) (press enter for N) > N
Enter User account to use for the service (press enter for NT AUTHORITY\Servicio de red) >
Granting file permissions to 'NT AUTHORITY\Servicio de red'.
Service vstsagent.ignacio-franciosi.Default.AgentSelfHosted successfully installed
Service vstsagent.ignacio-franciosi.Default.AgentSelfHosted successfully set recovery option
Service vstsagent.ignacio-franciosi.Default.AgentSelfHosted successfully set to delayed auto start
Service vstsagent.ignacio-franciosi.Default.AgentSelfHosted successfully configured
Enter whether to prevent service starting immediately after configuration is finished? (Y/N) (press enter for N) > Y

C:\agents\vsts-agent-win-x64-3.243.1>
```

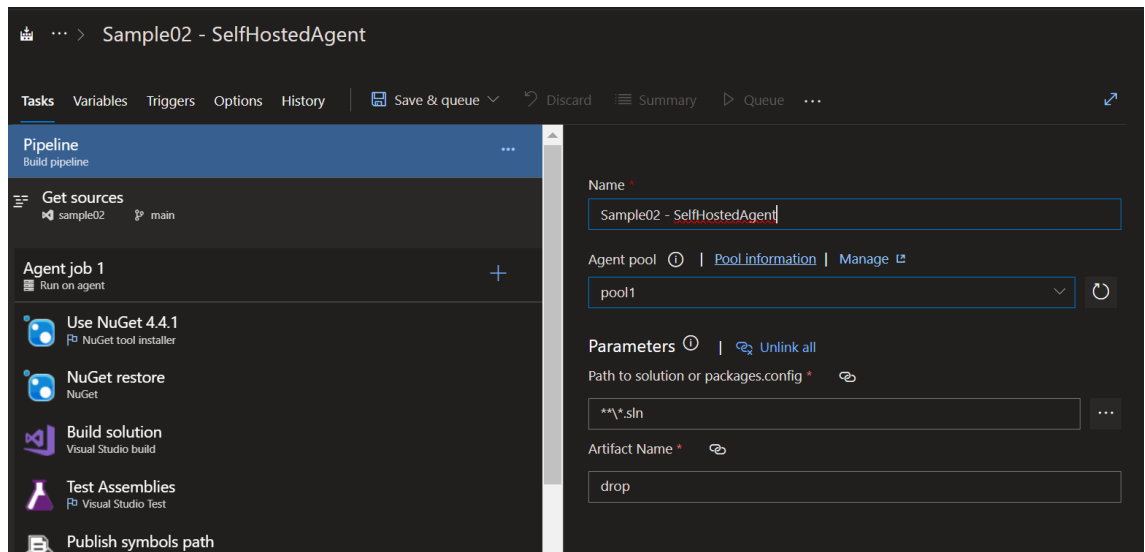
Jobs	Agents	Details	Security	Approvals and checks	Analytics
Name	Last run	Current status	Agent version	Enabled	
AgentSelfHosted ● Offline		Idle	3.243.1	 On	

```
C:\agents\vsts-agent-win-x64-3.243.1>.\run.cmd
Scanning for tool capabilities.
Connecting to the server.
2024-09-08 01:21:54Z: Listening for Jobs
```

Name	Last run	Current status	Agent version	Enabled	
AgentSelfHosted ● Online		Idle	3.243.1	 On	

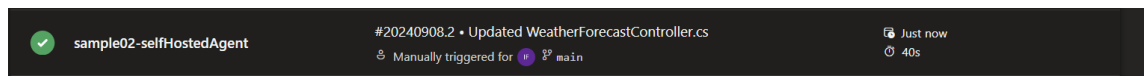
- 4.10 Crear un pipeline que use el agente Self-Hosted alojado en nuestra máquina local.

EN EL PIPELINE CLASSIC VOY A LA CONFIG Y EN VEZ DE AZURE LE PONGO EL NOMBRE DEL POOL



- 4.11 Buscar el resultado del pipeline y correr localmente el software compilado.

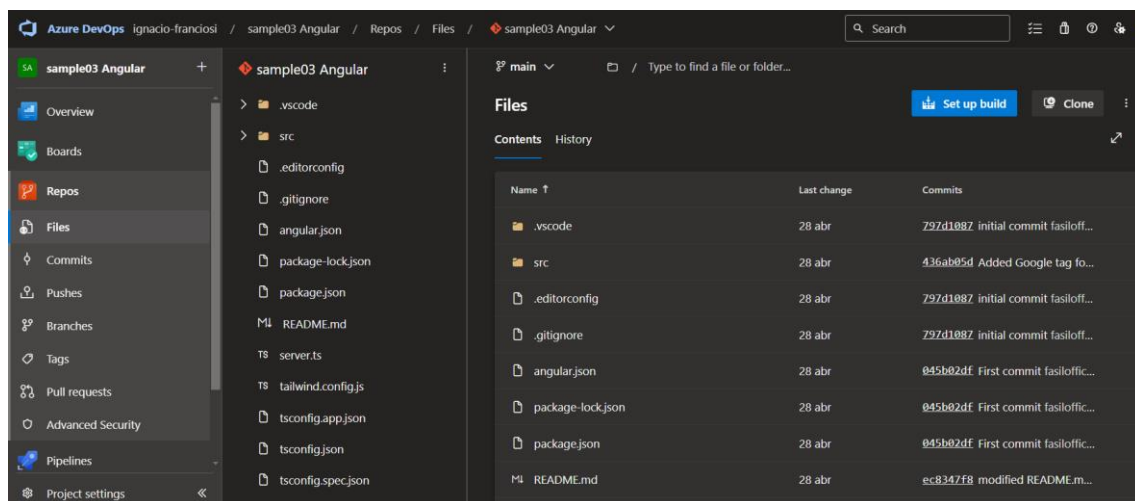
```
2024-09-08 02:05:12Z: Running job: Agent job
2024-09-08 02:05:50Z: Job Agent job completed with result: Succeeded
```



6

- 4.12 Crear un nuevo proyecto en ADO clonado desde un repo que contenga una aplicación en Angular como por ejemplo <https://github.com/ingsoft3ucc/angular-demo-project.git>

Se clonó el repositorio en un nuevo proyecto “sample03 Angular”



- 4.13 Configurar un pipeline de build para un proyecto de tipo Angular como el clonado.

Uso la opción de YAML con “Node.js con Angular”

The screenshot displays the Azure DevOps interface for configuring a pipeline. At the top, the breadcrumb navigation shows 'sample03 Angular'. Below this, a dropdown menu is set to 'main' and the file 'sample03 Angular / azure-pipelines.yml' is selected. The main area shows the YAML configuration for the pipeline, which includes a trigger for 'main' and a pool of 'ubuntu-latest'. The steps section is configured with a task 'NodeTool@0' and a script to install Node.js and build the project. Below the main configuration, a 'Settings' section shows the task 'PublishBuildArtifacts@1' with inputs for 'PathToPublish' and 'ArtifactName'. At the bottom, a 'Recently run pipelines' table shows the pipeline 'sample03 Angular' with a status of 'Succeeded' and a last run time of 'Just now'.

```
1 # Node.js with Angular
2 # Build a Node.js project that uses Angular.
3 # Add steps that analyze code, save build artifacts, deploy, and more:
4 # https://docs.microsoft.com/azure/devops/pipelines/languages/javascript
5
6 trigger:
7   - main
8
9 pool:
10  vmImage: ubuntu-latest
11
12 steps:
13   - task: NodeTool@0
14     inputs:
15       versionSpec: '18.x'
16       displayName: 'Install Node.js'
17
18   - script: node --version
19     displayName: 'Check Node.js version'
20
21   - script: |
22     npm install -g @angular/cli
23     npm install
24     ng build --configuration=production
25     displayName: 'npm install and build'
26
27
28 Settings
29 - task: PublishBuildArtifacts@1
30   inputs:
31     PathToPublish: 'dist'
32     ArtifactName: 'dist'
```

Pipeline	Last run
sample03 Angular	#20240908.3 • Update azure-pipelines.yml for Azure Pipelines Individual CI for main Just now 1m 9s

- 4.14 Habilitar CI para el pipeline.

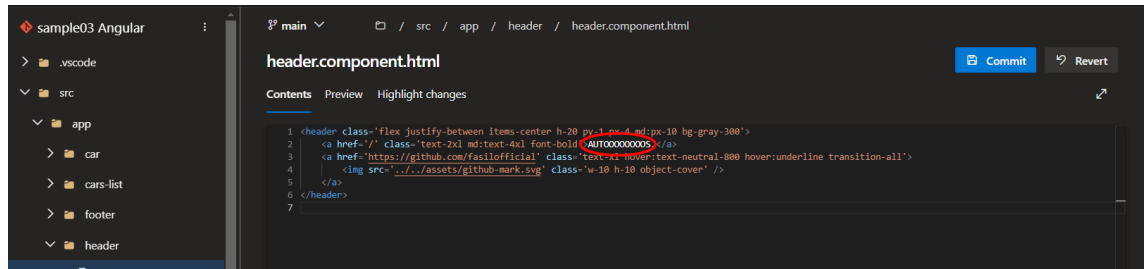
The screenshot displays the Azure DevOps interface for configuring a pipeline. At the top, the breadcrumb navigation shows 'sample03 Angular'. Below this, a dropdown menu is set to 'main' and the file 'sample03 Angular / azure-pipelines.yml' is selected. The main area shows the YAML configuration for the pipeline, which includes a trigger for 'main' and a pool of 'ubuntu-latest'. The steps section is configured with a task 'NodeTool@0' and a script to install Node.js and build the project. Below the main configuration, a 'Settings' section shows the task 'PublishBuildArtifacts@1' with inputs for 'PathToPublish' and 'ArtifactName'. At the bottom, a 'Recently run pipelines' table shows the pipeline 'sample03 Angular' with a status of 'Succeeded' and a last run time of 'Just now'.

```
1 # Node.js with Angular
2 # Build a Node.js project that uses Angular.
3 # Add steps that analyze code, save build artifacts, deploy, and more:
4 # https://docs.microsoft.com/azure/devops/pipelines/languages/javascript
5
6 trigger:
7   - main
8
9 pool:
10  vmImage: ubuntu-latest
11
12 steps:
13   - task: NodeTool@0
14     inputs:
15       versionSpec: '18.x'
16       displayName: 'Install Node.js'
17
18   - script: node --version
19     displayName: 'Check Node.js version'
20
21   - script: |
22     npm install -g @angular/cli
23     npm install
24     ng build --configuration=production
25     displayName: 'npm install and build'
26
27
28 Settings
29 - task: PublishBuildArtifacts@1
30   inputs:
31     PathToPublish: 'dist'
32     ArtifactName: 'dist'
```

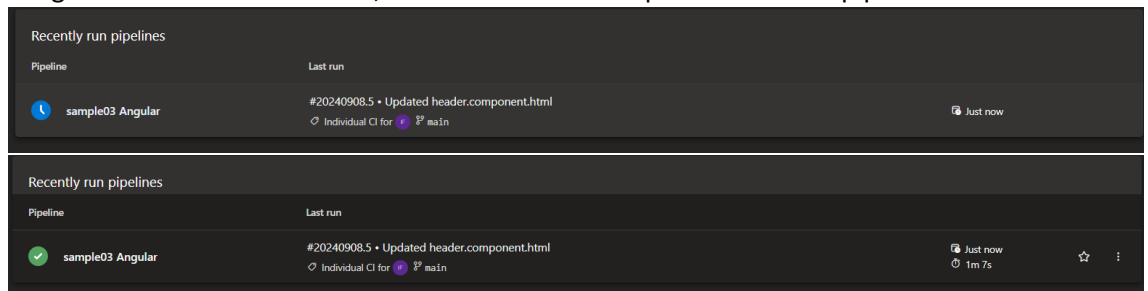
Pipeline	Last run
sample03 Angular	#20240908.3 • Update azure-pipelines.yml for Azure Pipelines Individual CI for main Just now 1m 9s



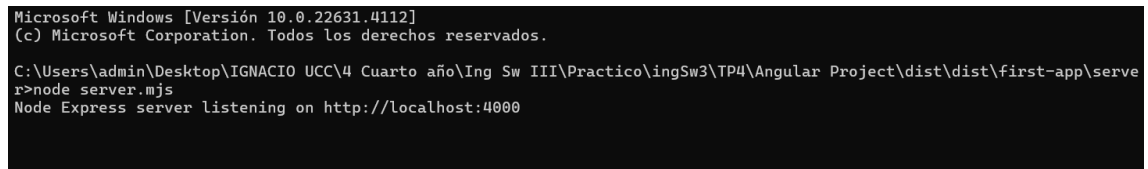
- 4.15 Hacer un cambio a un archivo del proyecto (algún cambio en el HTML que se renderiza por ejemplo) y verificar que se ejecute automáticamente el pipeline.



Luego de commitear el cambio, automáticamente empieza a correr el pipeline

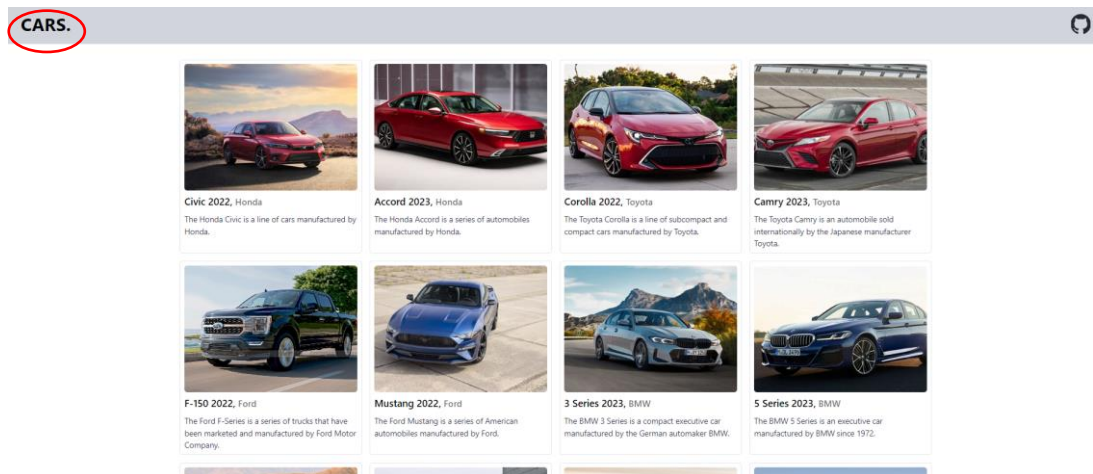


- 4.16 Descargar el resultado del pipeline y correr en un servidor web local el sitio construido.




- 4.17 Mostrar el antes y el después del cambio.

ANTES




DESPUES


AUTOooooooooos.




**Civic 2022, Honda**  
The Honda Civic is a line of cars manufactured by Honda.



**Accord 2023, Honda**  
The Honda Accord is a series of automobiles manufactured by Honda.



**Corolla 2022, Toyota**  
The Toyota Corolla is a line of subcompact and compact cars manufactured by Toyota.



**Camry 2023, Toyota**  
The Toyota Camry is an automobile sold internationally by the Japanese manufacturer Toyota.

