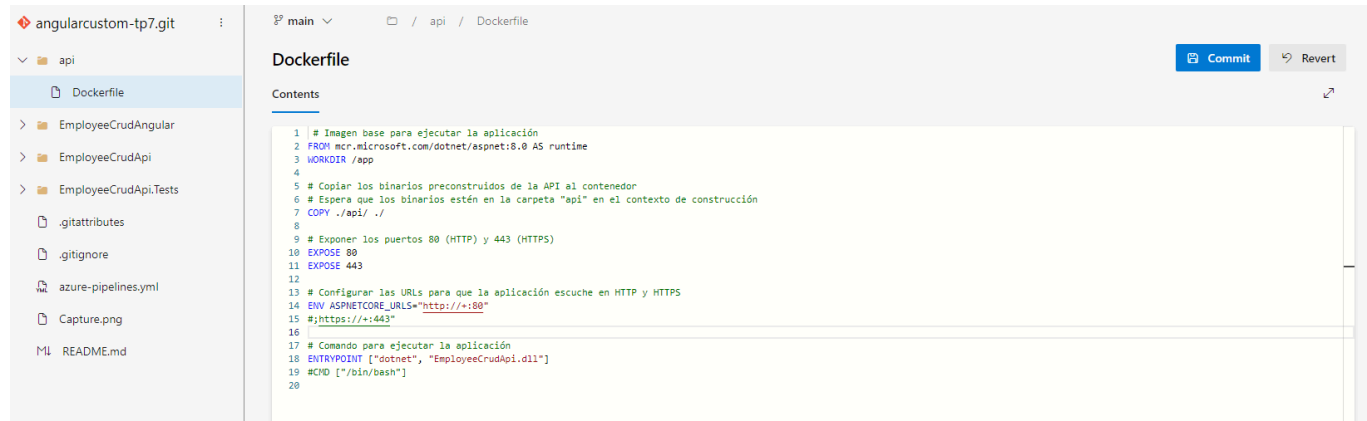


4.1 Modificar nuestro pipeline para construir imágenes Docker de back y front y subirlas a ACR

4.1.1 Crear archivos DockerFile para nuestros proyectos de Back y Front

Back:



The screenshot shows the Dockerfile for the 'api' service. The file is located at 'api / Dockerfile'. The contents are as follows:

```
1 # Imagen base para ejecutar la aplicación
2 FROM mcr.microsoft.com/dotnet/aspnet:8.0 AS runtime
3 WORKDIR /app
4
5 # Copiar los binarios precompilados de la API al contenedor
6 # Espera que los binarios estén en la carpeta "api" en el contexto de construcción
7 COPY ./api/ ./
8
9 # Exponer los puertos 80 (HTTP) y 443 (HTTPS)
10 EXPOSE 80
11 EXPOSE 443
12
13 # Configurar las URLs para que la aplicación escuche en HTTP y HTTPS
14 ENV ASPNETCORE_URLS="http://*:80"
15 #https://*:443"
16
17 # Comando para ejecutar la aplicación
18 ENTRYPOINT ["dotnet", "EmployeeCrudApi.dll"]
19 #CMD ["./bin/bash"]
20
```

Front:

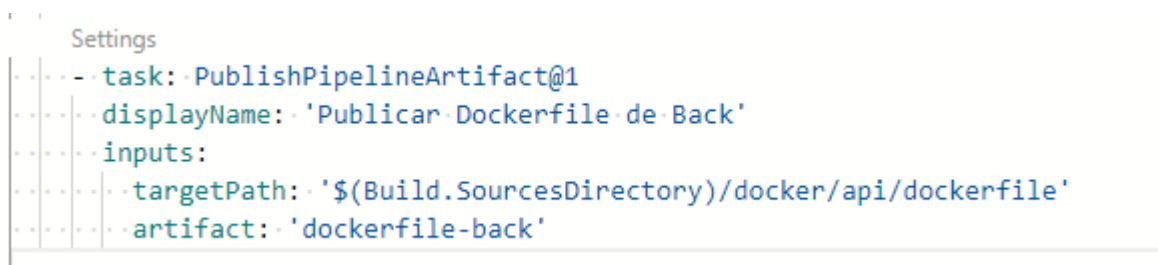


The screenshot shows the Dockerfile for the 'front' service. The file is located at 'front / Dockerfile'. The contents are as follows:

```
1 # Utilizar la imagen base de nginx
2 FROM nginx:alpine
3
4 # Establecer el directorio de trabajo en el contenedor
5 WORKDIR /usr/share/nginx/html
6
7 # Eliminar los archivos existentes de la imagen base de nginx
8 RUN rm -rf /*
9
10 # Copiar los archivos compilados de Angular al directorio de nginx
11 COPY ./ .
12
13 # Exponer el puerto 80 para servir la aplicación Angular
14 EXPOSE 80
15 CMD sh -c "echo 'window[\"env\"] = { apiUrl: \"${API_URL}\" };\" > /usr/share/nginx/html/assets/env.js && nginx -g 'daemon off;'"
16
```

4.1.3 Modificar nuestro pipeline en la etapa de Build y Test

Back:



The screenshot shows the 'Settings' section of the pipeline configuration. The configuration is as follows:

```
task: PublishPipelineArtifact@1
displayName: 'Publicar Dockerfile de Back'
inputs:
  targetPath: '$(Build.SourcesDirectory)/docker/api/dockerfile'
  artifact: 'dockerfile-back'
```

Front:

```
    Settings
  - task: PublishPipelineArtifact@1
    displayName: 'Publicar Dockerfile de Back'
    inputs:
      targetPath: '$(Build.SourcesDirectory)/docker/front/dockerfile'
      artifact: 'dockerfile-front'
```

4.1.5 Agregar a nuestro pipeline variables

```
variables:
- ConnectedServiceName: 'ServiceConnectionARM' #Por ejemplo 'ServiceConnectionARM'
- acrLoginServer: 'ingsoft3ucc.azurecr.io' #Por ejemplo 'ascontainerregistry.azurecr.io'
- backImageName: 'employee-crud-api' #Por ejemplo 'employee-crud-api'
```

4.1.6 Agregar a nuestro pipeline una nueva etapa que dependa de nuestra etapa de Build y Test

```

126 | | | | #-----
127 | | | | # BUILD DOCKER BACK IMAGE Y PUSH A AZURE CONTAINER REGISTRY
128 | | | | #-----
129 | | | |
130 | | | | Settings
131 | | | | - task: DownloadPipelineArtifact@2
132 | | | | displayName: 'Descargar Artefactos de Back'
133 | | | | inputs:
134 | | | | buildType: 'current'
135 | | | | artifactName: 'backend'
136 | | | | targetPath: '$(Pipeline.Workspace)/drop-back'
137 | | | |
138 | | | | Settings
139 | | | | - task: DownloadPipelineArtifact@2
140 | | | | displayName: 'Descargar Dockerfile de Back'
141 | | | | inputs:
142 | | | | buildType: 'current'
143 | | | | artifactName: 'dockerfile-back'
144 | | | | targetPath: '$(Pipeline.Workspace)/dockerfile-back'
145 | | | |
146 | | | | Settings
147 | | | | - task: AzureCLI@2
148 | | | | displayName: 'Iniciar Sesión en Azure Container Registry (ACR)'
149 | | | | inputs:
150 | | | | azureSubscription: '$(ConnectedServiceName)'
151 | | | | scriptType: bash
152 | | | | scriptLocation: inlineScript
153 | | | | inlineScript: |
154 | | | | az acr login --name $(acrLoginServer)
155 | | | |
156 | | | | Settings
157 | | | | - task: Docker@2
158 | | | | displayName: 'Construir Imagen Docker para Back'
159 | | | | inputs:
160 | | | | command: build
161 | | | | repository: '$(acrLoginServer)/$(backImageName)'
162 | | | | dockerfile: '$(Pipeline.Workspace)/dockerfile-back/dockerfile'
163 | | | | buildContext: '$(Pipeline.Workspace)/drop-back'
164 | | | | tags: 'latest'
165 | | | |
166 | | | | Settings
167 | | | | - task: Docker@2
168 | | | | displayName: 'Subir Imagen Docker de Back a ACR'
169 | | | | inputs:
170 | | | | command: push
171 | | | | repository: '$(acrLoginServer)/$(backImageName)'
172 | | | | tags: 'latest'

```

4.1.7 - Ejecutar el pipeline y en Azure Portal acceder a la opción Repositorios de nuestro recurso Azure Container Registry. Verificar que exista una imagen con el nombre especificado en la variable backImageName asignada en nuestro pipeline

#20241022.11 • PRUEBA CONSTRUCCION IMAGENES Y SUBIDA A ACR

angularcustom-tp7.git

Run new

This run is being retained as one of 3 recent runs by main (Branch).

View retention leases

SummaryAssociated pipelinesCode Coverage

Triggered by Santiago Tomas Muscellini

View 18 changes

Repository and version

angularcustom-tp7.git

main 016f78de

Time started and elapsed

Today at 1:08 PM

6m 2s

Related

0 work items

4 published

Tests and coverage

Get started

Warnings 2

No data was written into the file /home/vsts/work/_temp/task_outputs/build_1729613634062.txt

Construir y Subir Imágenes Docker a ACR • Construir y Subir Imágenes Docker a ACR • Construir Imagen Docker para Back

No data was written into the file /home/vsts/work/_temp/task_outputs/build_1729613660201.txt

Construir y Subir Imágenes Docker a ACR • Construir y Subir Imágenes Docker a ACR • Construir Imagen Docker para Front

StagesJobs

Build Stage

2 jobs completed4m 12s

4 artifacts

Construir y Subir Imá...

1 job completed1m 33s

Activate Windows

Go to Settings to activate Windows.

IngSoft3UCC | Repositories

Container registry

Search

Refresh

Manage Deleted Repositories

Overview

Activity log

Access control (IAM)

Tags

Quick start

Events

Settings

Services

Repositories

Webhooks

Geo-replications

Tasks

Connected registries (Preview)

Cache

New to ACR, Artifact streaming helps pull images faster from AKS clusters. The 'Artifact streaming status' column shows which repositories are using this feature. [Learn more](#)

Search to filter repositories ...

Repositories	Cache Rule
drop-back	
drop-front	

4.1.8 - Agregar tareas para generar imagen Docker de Front (DESAFIO)

```

201 | ..... | #-----
202 | ..... | # BUILD DOCKER FRONT IMAGE Y PUSH A AZURE CONTAINER REGISTRY
203 | ..... | #-----
204 | ..... |
205 | ..... | Settings
206 | ..... | -- task: DownloadPipelineArtifact@2
207 | ..... | -- displayName: 'Descargar Artefactos de Front'
208 | ..... | -- inputs:
209 | ..... | -- buildType: 'current'
210 | ..... | -- artifactName: 'drop-front'
211 | ..... | -- targetPath: '$(Pipeline.Workspace)/drop-front'
212 | ..... |
213 | ..... | Settings
214 | ..... | -- task: DownloadPipelineArtifact@2
215 | ..... | -- displayName: 'Descargar Dockerfile de Front'
216 | ..... | -- inputs:
217 | ..... | -- buildType: 'current'
218 | ..... | -- artifactName: 'dockerfile-front'
219 | ..... | -- targetPath: '$(Pipeline.Workspace)/dockerfile-front'
220 | ..... |
221 | ..... | Settings
222 | ..... | -- task: Docker@2
223 | ..... | -- displayName: 'Construir Imagen Docker para Front'
224 | ..... | -- inputs:
225 | ..... | -- command: build
226 | ..... | -- repository: $(acrLoginServer)/$(frontImageName)
227 | ..... | -- dockerfile: $(Pipeline.Workspace)/dockerfile-front/Dockerfile
228 | ..... | -- buildContext: $(Pipeline.Workspace)/drop-front
229 | ..... | -- tags: 'latest'
230 | ..... |
231 | ..... | Settings
232 | ..... | -- task: Docker@2
233 | ..... | -- displayName: 'Subir Imagen Docker de Front a ACR'
234 | ..... | -- inputs:
235 | ..... | -- command: push
236 | ..... | -- repository: $(acrLoginServer)/$(frontImageName)
237 | ..... | -- tags: 'latest'

```

4.1.9 - Agregar a nuestro pipeline una nueva etapa que dependa de nuestra etapa de Construcción de Imagenes Docker y subida a ACR

- Agregar variables a nuestro pipeline:

```
variables:
- nodeVersion: '20.x'
- configuration: 'Release'
- buildOutput: '$(Build.ArtifactStagingDirectory)'
- projectPath: 'EmployeeCrudAngular'
- solution: '**/*.sln'
- ConnectedServiceName: 'Azure_subscription_1-(a5bc7875-738f-4ed0-8e89-c9ef66bca3a7)' #Por ejemplo 'ServiceConnectionARM'
- acrLoginServer: 'ingsoft3ucc.azurecr.io' #Por ejemplo 'ascontainerregistry.azurecr.io'
- acrName: 'IngSoft3UCC'
- backImageName: 'drop-back' #Por ejemplo 'employee-crud-api'
- frontImageName: 'drop-front'
- ResourceGroupName: 'TPsIngSoft3' #Por ejemplo 'TPS_INGSOFT3_UCC'
- backContainerInstanceNameQA: 'as-crud-api-qa-sm'
- frontContainerInstanceNameQA: 'as-crud-front-qa-sm'
- backImageTag: 'latest'
- frontImageTag: 'latest'
- container-cpu-api-qa: 1 #CPU's de nuestro container de QA
- container-memory-api-qa: 1.5 #RAM de nuestro container de QA
- cnn_string_qaa: '$(cnn_string_qa)'
```

- Agregar variable secreta cnn-string-qa desde la GUI de ADO que apunte a nuestra BD de SQL Server de QA como se indica en el instructivo 5.3

Variables



Search variables



cnn_string_qa

- Agregar tareas para crear un recurso Azure Container Instances que levante un contenedor con nuestra imagen de back

```
..
- stage: DeployToACIQA
  displayName: 'Desplegar en Azure Container Instances (ACI) QA'
  dependsOn: DockerBuildAndPush
  jobs:
  - job: deploy_to_aci_qa
    displayName: 'Desplegar en Azure Container Instances (ACI) QA'
    pool:
      vmImage: 'ubuntu-latest'
    steps:
      # -----
      # DEPLOY DOCKER BACK IMAGE A AZURE CONTAINER INSTANCES QA
      # -----
      Settings
      - task: AzureCLI@2
        displayName: 'Desplegar Imagen Docker de Back en ACI QA'
        inputs:
          azureSubscription: '$(ConnectedServiceName)'
          scriptType: bash
          scriptLocation: inlineScript
          inlineScript: |
            echo "Resource Group: $(ResourceGroupName)"
            echo "Container Instance Name: $(backContainerInstanceNameQA)"
            echo "ACR Login Server: $(acrLoginServer)"
            echo "Image Name: $(backImageName)"
            echo "Image Tag: $(backImageTag)"
            echo "Connection String: $(cnn-string-qa)"

            az container delete --resource-group $(ResourceGroupName) --name $(backContainerInstanceNameQA) --yes

            az container create --resource-group $(ResourceGroupName) \
              --name $(backContainerInstanceNameQA) \
              --image $(acrLoginServer)/$(backImageName):$(backImageTag) \
              --registry-login-server $(acrLoginServer) \
              --registry-username $(acrName) \
              --registry-password $(az acr credential show --name $(acrName) --query "passwords[0].value" -o tsv) \
              --registry-password $(az acr credential show --name $(acrName) --query "passwords[0].value" -o tsv) \
              --dns-name-label $(backContainerInstanceNameQA) \
              --ports 80 \
              --environment-variables ConnectionStrings__DefaultConnection="$(cnn-string-qa)" \
              --restart-policy Always \
              --cpu $(container-cpu-api-qa) \
              --memory $(container-memory-api-qa)
```

- **4.1.10 - Ejecutar el pipeline y en Azure Portal acceder al recurso de Azure Container Instances creado. Copiar la url del contenedor y navegarlo desde browser. Verificar que traiga datos.**

#20241022.15 • prueba nueva deploy en acr back

angularcustom-tp7.git

Run new

This run is being retained as one of 3 recent runs by main (Branch).

View retention leases

SummaryAssociated pipelinesCode Coverage

Triggered by Santiago Tomas Muscellini

View 4 changes

Repository and version

angularcustom-tp7.git

mainc2664d41

Time started and elapsed

Today at 2:32 PM

3m 7s

Related

0 work items

4 published

Tests and coverage

Get started

Warnings 2

No data was written into the file /home/vsts/work/_temp/task_outputs/build_1729618650037.txt

Construir y Subir Imágenes Docker a ACR • Construir y Subir Imágenes Docker a ACR • Construir Imagen Docker para Back

No data was written into the file /home/vsts/work/_temp/task_outputs/build_1729618667040.txt

Construir y Subir Imágenes Docker a ACR • Construir y Subir Imágenes Docker a ACR • Construir Imagen Docker para Front

StagesJobs

Build Stage

2 jobs completed4m 11s

4 artifacts

Construir y Subir Imá...

1 job completed1m 10s

Desplegar en Azure C...

1 job completed3m 4s

Activate Windows
Go to Settings to activate Windows.

as-crud-api-qa-sm

Container instances

Search

StartRestartStopDeleteRefreshGive feedback

Overview

Activity log

Access control (IAM)

Tags

Settings

Monitoring

Automation

Help

Essentials

Resource group (move) : TPxingSoft3

Status : Running

Location : West Europe

Subscription (move) : Azure subscription 1

Subscription ID : a5bc7875-738f-4ed0-8e89-c9ef66bca3a7

Tags (edit) : Add tags

SKU : Standard

OS type : Linux

IP address (Public) : 20.61.242.17

FQDN : as-crud-api-qa-sm.westeurope.azurecontainer.io

Container count : 1

CPU

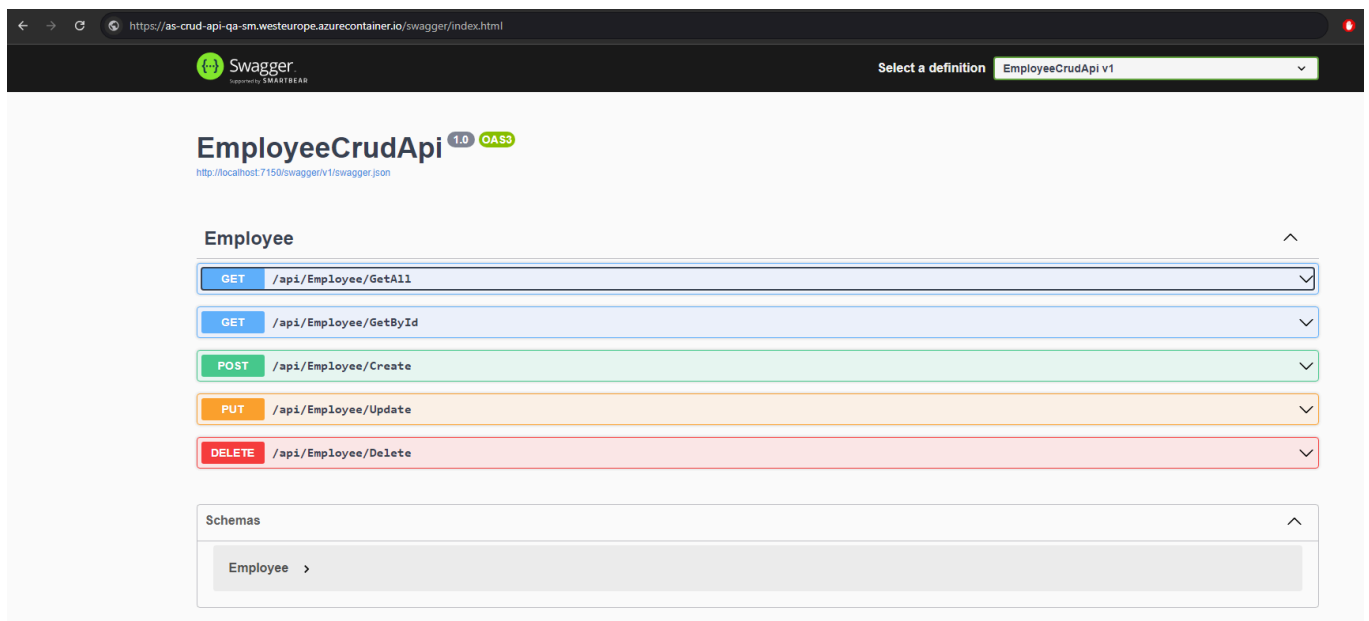
CPU Usage (Avg), as-crud-api-qa-sm | 1.52

Memory

Memory Usage (Avg), as-crud-api-qa-sm | 30.7MB

Network bytes received

Network Bytes Received Per Second (Avg), as-c... | 35.86B



4.1.11 - Agregar tareas para generar un recurso Azure Container Instances que levante un contenedor con nuestra imagen de front (DESAFIO)


```

295 # DEPLOY DOCKER FRONT IMAGE TO ACI QA
296 Settings
297 ---task: AzureCLI@2
298 ---displayName: 'Desplegar Imagen Docker de Front en ACI QA'
299 ---inputs:
300 ---azureSubscription: '$(ConnectedServiceName)'
301 ---scriptType: bash
302 ---scriptLocation: inlineScript
303 ---inlineScript: |
304 echo "Resource Group: $(ResourceGroupName)"
305 echo "Container Instance Name: front-container-qa"
306 echo "ACR Login Server: $(acrLoginServer)"
307 echo "Image Name: $(frontImageName)"
308 echo "Image Tag: $(frontImageTag)"
309
310 # Delete the existing front container if present
311 az container delete --resource-group $(ResourceGroupName) --name front-container-qa --yes || true
312
313 # Create the front-end container
314 az container create --resource-group $(ResourceGroupName) \
315 --location westeurope \
316 --name $(frontContainerInstanceNameQA) \
317 --image $(acrLoginServer)/$(frontImageName):$(frontImageTag) \
318 --registry-login-server $(acrLoginServer) \
319 --registry-username $(acrName) \
320 --registry-password $(ACR_PASSWORD) \
321 --dns-name-label $(frontContainerInstanceNameQA) \
322 --ports 80 \
323 --environment-variables API_URL=$(container-url-api-qa) \
324 --restart-policy Always \
325 --cpu 1 \
326 --memory 1.5

```

Azure DevOps mussalovesmuzska / sample02 / Pipelines / angularcustom-tp7.git / 20241022.19

Search

sample02

Overview

Boards

Repos

Pipelines

Pipelines

Environments

Releases

Library

Task groups

Deployment groups

Test Plans

Artifacts

Jobs in run #20241022.19

angularcustom-tp7.git

Build Stage

> Build y Analisis del Back-End 1m 59s

> Build y Analisis del Front-End 1m 23s

Construir y Subir Imágenes Docker a ACR

> Construir y Subir Imágenes Doc... 1m 16s

Desplegar en Azure Container Instances (ACI) QA

Desplegar en Azure Container L... 1m 43s

Initialize job 2s

Checkout angularcustom-tp7.git... 2s

Fetch ACR credentials 9s

Desplegar Imagen Docker de... 1m 22s

Desplegar Imagen Docker de Fron... 5s

Post-job: Checkout angularcustom... 1s

Finalize Job <1s

Finalize build

Report build status <1s

Desplegar Imagen Docker de Front en ACI QA

View raw log

1 Starting: Desplegar Imagen Docker de Front en ACI QA

2 =====

3 Task : Azure CLI

4 Description : Run Azure CLI commands against an Azure subscription in a PowerShell Core/Shell script when running on Linux agent or PowerShell/PowerShell Core/Batch script when running on Windows agent

5 Version : 2.247.1

6 Author : Microsoft Corporation

7 Help : https://docs.microsoft.com/azure/devops/pipelines/tasks/deploy/azure-cli

8 =====

9 /usr/bin/az --version

10 azure-cli 2.65.0

11

12 core 2.65.0

13 telemetry 1.1.0

14

15 Extensions:

16 azure-devops 1.0.1

17

18 Dependencies:

19 msal 1.31.0

20 azure-ngint-resource 23.1.1

21

22 Python location '/opt/az/bin/python3'

23 Extensions directory '/opt/az/azcliextensions'

24

25 Python (Linux) 3.11.8 (main, Sep 25 2024, 11:33:44) [GCC 11.4.0]

26 Legal docs and information: aka.ms/AzureCLILegal

27

28

29

30 Your CLI is up-to-date.

31 Setting AZURE_CONFIG_DIR env variable to: /home/vsts/work/_temp/.azclitask

32 Setting active cloud set to: AzureCloud

33 /usr/bin/az cloud set -n AzureCloud

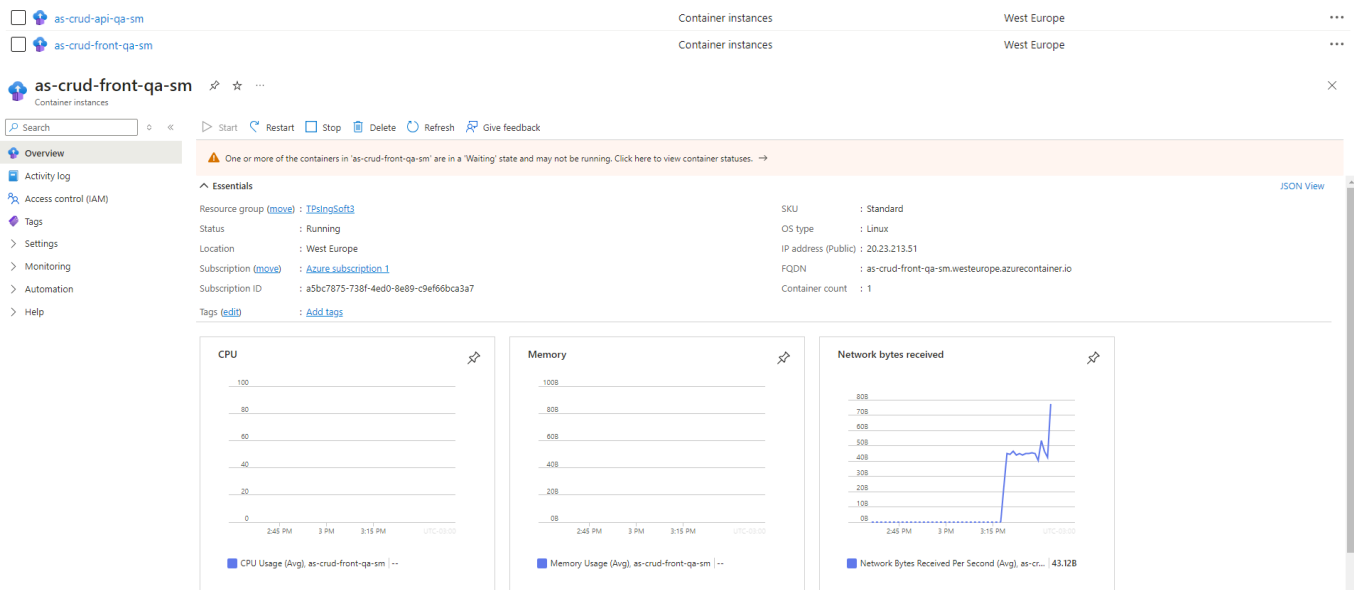
34 /usr/bin/az login --service-principal -u *** --password=*** --tenant ed5d3524-c224-4514-b065-d2369518433c --allow-no-subscriptions

35 {

36 {

37 "cloudName": "AzureCloud",

Activate Windows
Go to Settings to activate Windows.



- 4.1.12 - Agregar tareas para correr pruebas de integración en el entorno de QA de Back y Front creado en ACI.

```
327 | | | | | # Ejecutar pruebas unitarias del backend dentro del contenedor de ACI QA
    | | | | | Settings
328 | | | | | - task: AzureCLI@2
    | | | | | displayName: 'Ejecutar Pruebas del Back-End en ACI QA'
329 | | | | | inputs:
330 | | | | | - azureSubscription: '${ConnectedServiceName}'
331 | | | | | - scriptType: bash
332 | | | | | - scriptLocation: inlineScript
333 | | | | | - inlineScript: |
334 | | | | | # Ejecutar pruebas usando el archivo .dll de pruebas
335 | | | | | docker exec $(backContainerInstanceNameQA) dotnet test /app/EmployeeCrudApi.Tests.dll --logger trx --results-directory /app/tests
336 | | | | |
337 | | | | |
338 | | | | | # Publicar resultados de pruebas del backend
    | | | | | Settings
339 | | | | | - task: PublishTestResults@2
    | | | | | displayName: 'Publicar Resultados de Pruebas de Back'
340 | | | | | inputs:
341 | | | | | - testResultsFormat: 'VSTest'
342 | | | | | - testResultsFiles: '**/*.trx'
343 | | | | | - failTaskOnFailedTests: true
344 | | | | |
345 | | | | |
346 | | | | | # Ejecutar pruebas E2E del Frontend en el contenedor de ACI QA
    | | | | | Settings
347 | | | | | - task: AzureCLI@2
    | | | | | displayName: 'Ejecutar Pruebas del Front-End en ACI QA'
348 | | | | | inputs:
349 | | | | | - azureSubscription: '${ConnectedServiceName}'
350 | | | | | - scriptType: bash
351 | | | | | - scriptLocation: inlineScript
352 | | | | | - inlineScript: |
353 | | | | | # Ejecutar pruebas E2E usando Cypress
354 | | | | | docker exec $(frontContainerInstanceNameQA) npx cypress run
355 | | | | |
356 | | | | |
```

```

346 | . . . . . # Ejecutar pruebas E2E del Frontend en el contenedor de ACI QA
      | Settings
347 | . . . . . - task: AzureCLI@2
348 | . . . . . displayName: 'Ejecutar Pruebas del Front-End en ACI QA'
349 | . . . . . inputs:
350 | . . . . . azureSubscription: '$(ConnectedServiceName)'
351 | . . . . . scriptType: bash
352 | . . . . . scriptLocation: inlineScript
353 | . . . . . inlineScript: |
354 | . . . . . # Ejecutar pruebas E2E usando Cypress
355 | . . . . . docker exec $(frontContainerInstanceNameQA) npx cypress run
356 |
357 | . . . . . # Publicar resultados de pruebas del frontend
      | Settings
358 | . . . . . - task: PublishTestResults@2
359 | . . . . . displayName: 'Publicar Resultados de Pruebas de Front'
360 | . . . . . inputs:
361 | . . . . . testResultsFormat: 'JUnit'
362 | . . . . . testResultsFiles: '**/cypress/results/*.xml'
363 | . . . . . failTaskOnFailedTests: true

```

#20241022.19 • Update azure-pipelines.yml for Azure Pipelines
 Run new

This run is being retained as one of 3 recent runs by main (Branch).
 View retention leases

SummaryAssociated pipelinesCode Coverage

Triggered by Santiago Tomas Muscellini
 View 4 changes

Repository and version	Time started and elapsed	Related	Tests and coverage
angularcustom-tp7.git main ↕ 2871327e	Today at 3:20 PM 8m 7s	0 work items 4 published	Get started

Warnings 2

No data was written into the file /home/vsts/work/_temp/task_outputs/build_1729621567047.txt
 Construir y Subir Imágenes Docker a ACR • Construir y Subir Imágenes Docker a ACR • Construir Imagen Docker para Back

No data was written into the file /home/vsts/work/_temp/task_outputs/build_1729621597054.txt
 Construir y Subir Imágenes Docker a ACR • Construir y Subir Imágenes Docker a ACR • Construir Imagen Docker para Front

StagesJobs

Build Stage
 2 jobs completed 4m 16s
 4 artifacts

Construir y Subir Imá...
 1 job completed 1m 21s

Desplegar en Azure C...
 1 job completed 1m 48s

Activate Windows
 Go to Settings to activate Windows.