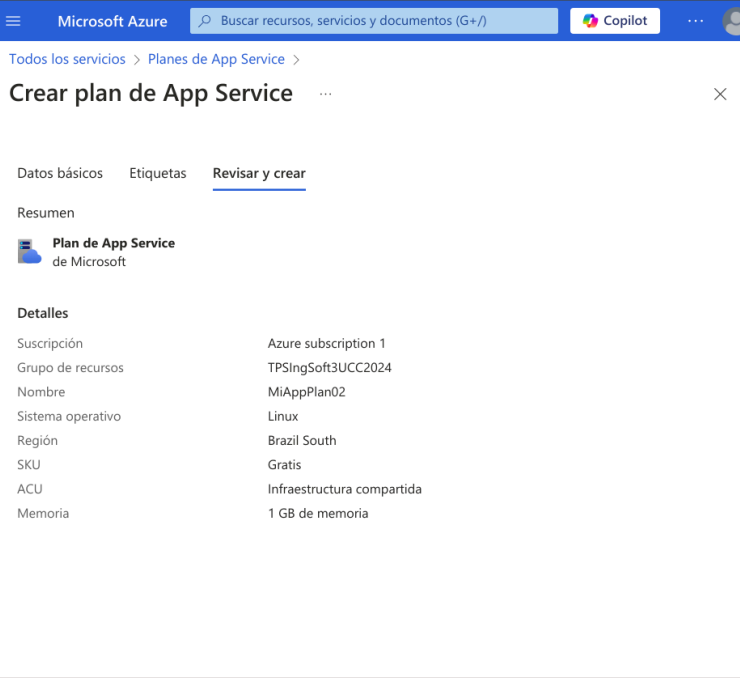


## Trabajo Práctico 9 - Implementación de Contenedores en Azure Parte 2

### 4.1 Modificar nuestro pipeline para incluir el deploy en QA y PROD de Imágenes Docker en Servicio Azure App Services con Soporte para Contenedores

#### 4.1.1 - Agregar a nuestro pipeline una nueva etapa que dependa de nuestra etapa de Construcción y Pruebas y de la etapa de Construcción de Imágenes Docker y subida a ACR realizada en el TP08

Agregar tareas para crear un recurso Azure Container Instances que levante un contenedor con nuestra imagen de back utilizando un AppServicePlan en Linux



Microsoft Azure | Buscar recursos, servicios y documentos (G+/J) | Copilot

Todos los servicios > Planes de App Service >

### Crear plan de App Service

Datos básicos | Etiquetas | **Revisar y crear**

Resumen

**Plan de App Service**  
de Microsoft

**Detalles**

Suscripción	Azure subscription 1
Grupo de recursos	TPSingSoft3UCC2024
Nombre	MiAppPlan02
Sistema operativo	Linux
Región	Brazil South
SKU	Gratis
ACU	Infraestructura compartida
Memoria	1 GB de memoria

**Crear** | < Anterior | Siguiente > | [Descargar una plantilla para la automatización](#)

Microsoft Azure

Buscar recursos, servicios y documentos (G+)

Copilot

Todos los servicios > Microsoft.Web-ASP-Portal-b5c4736d-b70b | Información general >

MiAppPlan02

Plan de Linux

✧ ☆ ...

Buscar

Eliminar

Envíenos sus comentarios

Introducción

Registro de actividad

Control de acceso (IAM)

Etiquetas

Diagnosticar y solucionar problemas

Eventos (versión preliminar)

Configuración

Supervisión

Automation

Ayuda

Las métricas del plan App Service no se aplican a las SKU gratuitas y compartidas

Información esencial

Vista JSON

Grupo de recursos (mover)

Estado

Ubicación

Suscripción (mover)

Id. de suscripción

Plan de precios

Recuento de instancias

Aplicaciones y espacios

Sistema operativo

Con redundancia de zona

Etiquetas (editar)

TPSingSoft3UCC2024

Listo

Brazil South

Azure subscription 1

5341073c-194f-47d6-9c70...

F1

1

0 / 0

Linux

Deshabilitado

Agregar etiquetas

Porcentaje de CPU

Porcentaje de memoria

```
45
46 WebAppApiNameContainersQA: 'container-crud-api-qa'
47 AppServicePlanLinux: MiAppPlan02
48
```

1803161 / tp7-angular / Pipelines

tp7-angular

main tp7-angular / azure-pipelines.yml

```
466 ## STAGE DEPLOY TO AZURE APP SERVICE QA
467 #-----
468 . stage: DeployImagesToAppServiceQA
469   displayName: 'Desplegar Imágenes en Azure App Service (QA)'
470   dependsOn:
471     - BuildAndTest
472     - DockerBuildAndPush
473   condition: succeeded()
474   jobs:
475     - job: DeployImagesToAppServiceQA
476       displayName: 'Desplegar Imágenes de API y Front en Azure App Service (QA)'
477       pool:
478         vmImage: 'ubuntu-latest'
479       steps:
480         # DEPLOY DOCKER API IMAGE TO AZURE APP SERVICE (QA)
481         #-----
482         Settings
483         - task: AzureCLI@2
484           displayName: 'Verificar y crear el recurso Azure App Service para API (QA) si no existe'
485           inputs:
486             azureSubscription: '$(ConnectedServiceName)'
487             scriptType: 'bash'
488             scriptLocation: 'inlineScript'
489             inlineScript: |
490               # Verificar si el App Service para la API ya existe
491               if ! az webapp list -q "name=$(WebAppApiNameContainersQA)" && resourceGroup=$(ResourceGroupName) | length(@) -o tsv | grep -q '^1$'; then
492                 echo "El App Service para API QA no existe. Creando..."
493                 # Crear el App Service sin especificar la imagen del contenedor
494                 az webapp create --resource-group $(ResourceGroupName) --plan $(AppServicePlanLinux) --name $(WebAppApiNameContainersQA) --deployment-container-image-name "nginx" #
495               else
496                 echo "El App Service para API QA ya existe. Actualizando la imagen..."
497               fi
498
499           # Configurar el App Service para usar Azure Container Registry (ACR)
500           az webapp config container set --name $(WebAppApiNameContainersQA) --resource-group $(ResourceGroupName) \
501             --container-image-name $(acrLoginServer)/$(backImageName):$(backImageTag) \
502             --container-registry-url https://$(acrLoginServer) \
503             --container-registry-user $(acrName) \
504             --container-registry-password $(az acr credential show --name $(acrName) --query "passwords[0].value" -o tsv)
505           # Establecer variables de entorno
506           az webapp config appsettings set --name $(WebAppApiNameContainersQA) --resource-group $(ResourceGroupName) \
507             --settings ConnectionStrings_DefaultConnection="$(cnn-string-qa)" \
508
```

Azure DevOps

/ 20241017.2

Search

1

✓

#20241017.2 • Update azure-pipelines.yml fo...

Run new

tp7-angular

ⓘ

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

View retention leases

Summary

Environments

Associated pipelines

Code Coverage

Triggered by

1 1803161

View change

Repository and version

tp7-angular

main

c1f48e84

Time started and elapsed

Today at 14:59

18m 7s

Related

0 work items

4 published

Tests and coverage

Get started

MiAppPlan02 | Aplicaciones

Plan de Linux

Actualizar

Filtrar por nombre

Estado: Todo

Nombre	Tipo	Variante	Grupo de recursos	Estado
container-crud-api-qa	Aplicación	app.linux.container	tpsingsoft3ucc2024	En ejecución

container-crud-api-qa

Aplicación web

Examinar

Detener

Intercambiar

Reiniciar

Eliminar

Actualizar

Descargar perfil de publicación

Restablecer perfil de publicación

Essentials

Vista JS

Grupo de recursos

tpsingsoft3ucc2024

Dominio predeterminado

container-crud-api-qa.azurewebsites.net

Estado

En ejecución

Plan del servicio de aplic...

MiAppPlan02 (F1: 1)

Ubicación

Brazil South

Sistema operativo

Linux

Suscripción

Azure subscription 1

Comprobación de estado

Sin configurar

container-crud-api-qa - Mic

container-crud-api-qa.azure

container-crud-api-qa.azurewebsites.net/api/Employee/getAll

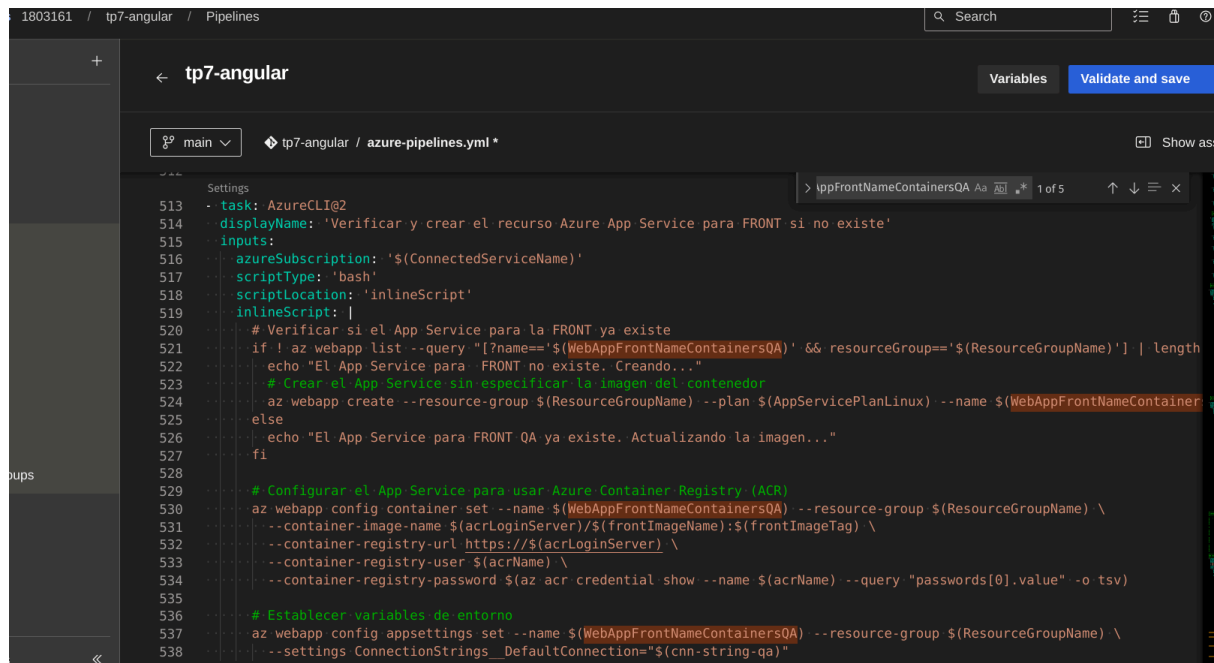
Pretty print

[{"id":1,"name":"Juan Perez","createdAt":"2024-09-05T00:00:00"}, {"id":2,"name":"Carla Ruiz","createdAt":"2024-09-05T22:22:47.440572"}, {"id":3,"name":"Carlos Gomez","createdAt":"2024-09-06T09:16:50.070943"}, {"id":4,"name":"Joaquin Zarate","createdAt":"2024-09-06T09:19:37.898716"}, {"id":5,"name":"Luis Rodriguez","createdAt":"2024-09-06T09:23:31.324434"}]

4.2 Desafíos:

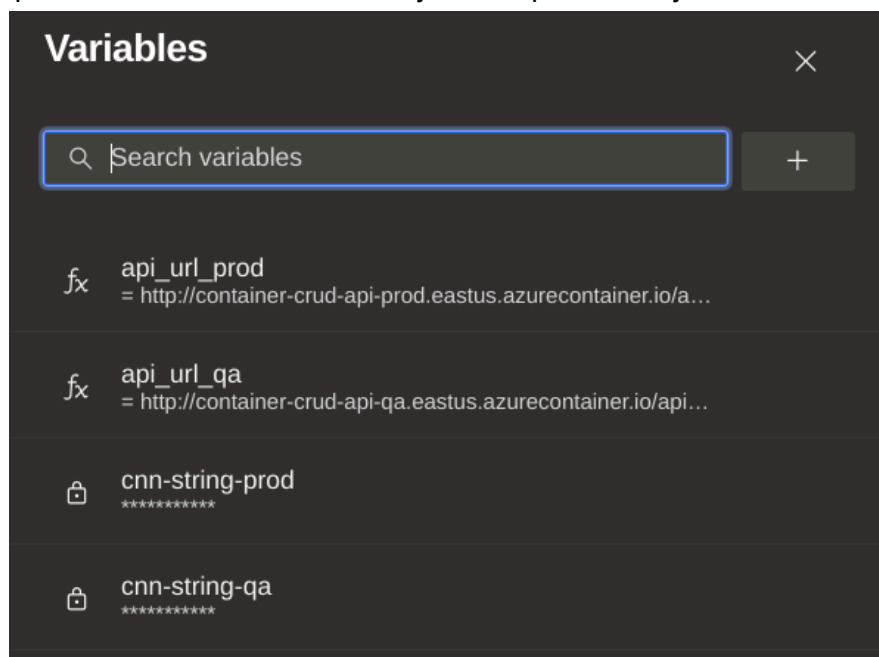
4.2.1 Agregar tareas para generar Front en Azure App Service con Soporte para Contenedores

```
WebAppFrontNameContainersQA: 'container-crud-front-qa'
```



```
1803161 / tp7-angular / Pipelines
tp7-angular
Variables Validate and save
main tp7-angular / azure-pipelines.yml
Settings
513 - task: AzureCLI@2
514   displayName: 'Verificar y crear el recurso Azure App Service para FRONT si no existe'
515   inputs:
516     azureSubscription: '$(ConnectedServiceName)'
517     scriptType: 'bash'
518     scriptLocation: 'inlineScript'
519     inlineScript: |
520       # Verificar si el App Service para la FRONT ya existe
521       if ! az webapp list --query "[?name=='$(WebAppFrontNameContainersQA)'] && resourceGroup=='$(ResourceGroupName)'] | length
522       echo "El App Service para FRONT no existe. Creando..."
523       # Crear el App Service sin especificar la imagen del contenedor
524       az webapp create --resource-group $(ResourceGroupName) --plan $(AppServicePlanLinux) --name $(WebAppFrontNameContainer
525     else
526       echo "El App Service para FRONT QA ya existe. Actualizando la imagen..."
527     fi
528
529   # Configurar el App Service para usar Azure Container Registry (ACR)
530   az webapp config container set --name $(WebAppFrontNameContainersQA) --resource-group $(ResourceGroupName) \
531     --container-image-name $(acrLoginServer)/$(frontImageName):$(frontImageTag) \
532     --container-registry-url https://$(acrLoginServer) \
533     --container-registry-user $(acrName) \
534     --container-registry-password $(az acr credential show --name $(acrName) --query "passwords[0].value" -o tsv)
535
536   # Establecer variables de entorno
537   az webapp config appsettings set --name $(WebAppFrontNameContainersQA) --resource-group $(ResourceGroupName) \
538     --settings ConnectionStrings_DefaultConnection="$(cnn-string-qa)"
```

4.2.2 Agregar variables necesarias para el funcionamiento de la nueva etapa considerando que debe haber 2 entornos QA y PROD para Back y Front.



### Variables

Search variables

- fx api\_url\_prod  
= http://container-crud-api-prod.eastus.azurecontainer.io/a...
- fx api\_url\_qa  
= http://container-crud-api-qa.eastus.azurecontainer.io/api...
- 🔒 cnn-string-prod  
\*\*\*\*\*
- 🔒 cnn-string-qa  
\*\*\*\*\*

4.2.3 Agregar tareas para correr pruebas de integración en el entorno de QA de Back y Front creado en Azure App Services con Soporte para Contenedores.

4.2.4 Agregar etapa que dependa de la etapa de Deploy en QA que genere un entorno de PROD.

```
WebAppApiNameContainersProd: 'container-crud-api-prod'
WebAppFrontNameContainersProd: 'container-crud-front-prod'
```

1803161 / tp7-angular / Pipelines

tp7-angular

main

tp7-angular / azure-pipelines.yml \*

```
542 - stages: DeployImagesToAppServiceProd
543   displayName: 'Desplegar Imagenes en Azure App Service (Prod)'
544   dependsOn:
545     - DeployImagesToAppServiceQA
546   condition: succeeded()
547   jobs:
548     - job: DeployImagesToAppServiceProd
549       displayName: 'Desplegar Imagenes de API y Front en Azure App Service (Prod)'
550       pool:
551         vmImage: 'ubuntu-latest'
552       steps:
553         - task: AzureCLI@2
554           displayName: 'Verificar y crear el recurso Azure App Service para API (Prod) si no existe'
555           inputs:
556             azureSubscription: '${ConnectedServiceName}'
557             scriptType: 'bash'
558             scriptLocation: 'inlineScript'
559             inlineScript: |
560               # Verificar si el App Service para la API ya existe
561               if ! az webapp list --query "[?name=='$(WebAppApiNameContainersProd)' && resourceGroup=='$(ResourceGroupName)']" | length(0) -o tsv | grep -q '^$'; then
562                 echo "El App Service para API Prod no existe. Creando..."
563                 # Crear el App Service sin especificar la imagen del contenedor
564                 az webapp create --resource-group $(ResourceGroupName) --plan $(AppServicePlanLinux) --name $(WebAppApiNameContainersProd) --deployment-container-image-name "nginx" -- # # #
565               else
566                 echo "El App Service para API Prod ya existe. Actualizando la imagen..."
567               fi
568
569               # Configurar el App Service para usar Azure Container Registry (ACR)
570               az webapp config container set --name $(WebAppApiNameContainersProd) --resource-group $(ResourceGroupName) \
571                 --container-image-name $(acrLoginServer)/$(backImageName):$(backImageTag) \
572                 --container-registry-url https://$(acrLoginServer) \
573                 --container-registry-user $(acrName) \
574                 --container-registry-password $(az acr credential show --name $(acrName) --query "passwords[0].value" -o tsv)
575
576               # Establecer variables de entorno
577               az webapp config appsettings set --name $(WebAppApiNameContainersProd) --resource-group $(ResourceGroupName) \
578                 --settings ConnectionStrings__DefaultConnection="$(conn-string-prod)"
579
580
```

Settings

1803161 / tp7-angular / Pipelines

tp7-angular

main

tp7-angular / azure-pipelines.yml \*

```
581 - task: AzureCLI@2
582   displayName: 'Verificar y crear el recurso Azure App Service para FRONT si no existe'
583   inputs:
584     azureSubscription: '${ConnectedServiceName}'
585     scriptType: 'bash'
586     scriptLocation: 'inlineScript'
587     inlineScript: |
588       # Verificar si el App Service para la FRONT ya existe
589       if ! az webapp list --query "[?name=='$(WebAppFrontNameContainersProd)' && resourceGroup=='$(ResourceGroupName)']" | length(0) -o tsv | grep -q '^$'; then
590         echo "El App Service para FRONT no existe. Creando..."
591         # Crear el App Service sin especificar la imagen del contenedor
592         az webapp create --resource-group $(ResourceGroupName) --plan $(AppServicePlanLinux) --name $(WebAppFrontNameContainersProd) --deployment-container-image-name "ng
593       else
594         echo "El App Service para FRONT Prod ya existe. Actualizando la imagen..."
595       fi
596
597       # Configurar el App Service para usar Azure Container Registry (ACR)
598       az webapp config container set --name $(WebAppFrontNameContainersProd) --resource-group $(ResourceGroupName) \
599         --container-image-name $(acrLoginServer)/$(frontImageName):$(frontImageTag) \
600         --container-registry-url https://$(acrLoginServer) \
601         --container-registry-user $(acrName) \
602         --container-registry-password $(az acr credential show --name $(acrName) --query "passwords[0].value" -o tsv)
603
604       # Establecer variables de entorno
605       az webapp config appsettings set --name $(WebAppFrontNameContainersProd) --resource-group $(ResourceGroupName) \
606         --settings ConnectionStrings__DefaultConnection="$(conn-string-prod)"
607
```

Settings

Ejecutamos el pipeline:

dev.azure.com/1803161/tp7-angular/\_build/results?buildId=103&view=results

1803161 / tp7-angular / Pipelines / tp7-angular / 20241017.3

#20241017.3 • Update azure-pipelines.yml for Azure Pipelines

Run new

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

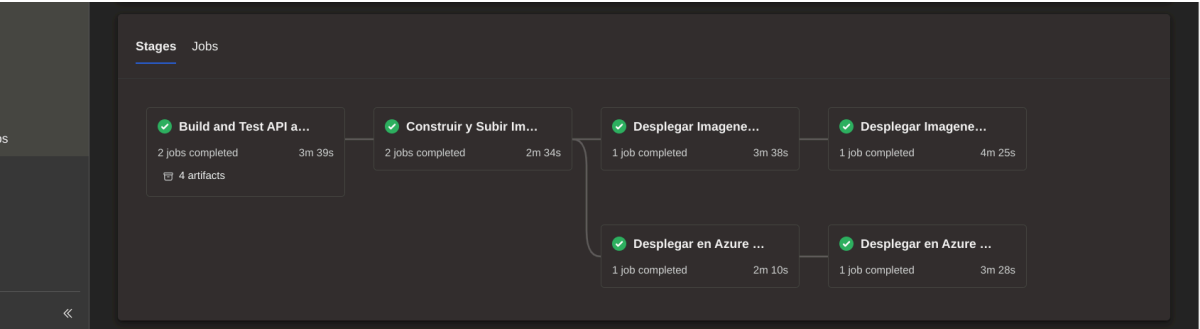
View retention leases

Summary

Triggered by 1803161

View change

Repository and version	Time started and elapsed	Related	Tests and coverage
tp7-angular	Today at 17:46	0 work items	Get started
main 520c8782	21m 59s	4 published	



Microsoft Azure

Buscar recursos, servicios y documentos (G+)

Copilot

1803161@ucc.edu.ar DIRECTORIO PREDETERMINADO

Todos los servicios > MiAppPlan02

MiAppPlan02 | Aplicaciones

Plan de Linux

Buscar

Actualizar

Filtrar por nombre Estado: Todo

Nombre	Tipo	Variante	Grupo de recursos	Estado
container-crud-api-prod	Aplicación	app.linux.container	tpsingsoft3ucc2024	En ejecución
container-crud-api-qa	Aplicación	app.linux.container	TPSingSoft3UCC2024	En ejecución
container-crud-front-prod	Aplicación	app.linux.container	tpsingsoft3ucc2024	En ejecución
container-crud-front-qa	Aplicación	app.linux.container	TPSingSoft3UCC2024	En ejecución

container-crud-api-qa.azurewebsites.net/api/Employee/getAll

```
[[{"id":1,"name":"Juan Perez","createdAt":"2024-09-05T00:00:00"}, {"id":2,"name":"Carla Ruiz","createdAt":"2024-09-05T22:22:47.440572"}, {"id":3,"name":"Carlos Gomez","createdAt":"2024-09-06T09:16:50.070943"}, {"id":4,"name":"Joaquin Zarate","createdAt":"2024-09-06T09:19:37.898716"}, {"id":5,"name":"Luis Rodriguez","createdAt":"2024-09-06T09:23:31.324434"}]]
```

Not secure container-crud-front-qa.azurewebsites.net

# EmployeeCrudAngular

Employees:

Id	Name	Created Date		
1	Juan Perez	05/09/2024 00:00:00		
2	Carla Ruiz	05/09/2024 22:22:47		
3	Carlos Gomez	06/09/2024 09:16:50		
4	Joaquin Zarate	06/09/2024 09:19:37		
5	Luis Rodriguez	06/09/2024 09:23:31		

Not secure container-crud-api-prod.azurewebsites.net/api/Employee/getAll

```
[[{"id":1,"name":"Juan Perez","createdAt":"2024-09-05T00:00:00"}, {"id":2,"name":"Carla Ruiz","createdAt":"2024-09-05T22:22:47.440572"}, {"id":3,"name":"Carlos Gomez","createdAt":"2024-09-06T09:16:50.070943"}, {"id":4,"name":"Joaquin Zarate","createdAt":"2024-09-06T09:19:37.898716"}, {"id":5,"name":"Luis Rodriguez","createdAt":"2024-09-06T09:23:31.324434"}]]
```

#### 4.2.5 Entregar un pipeline que incluya:

- A) Etapa Construcción y Pruebas Unitarias y Code Coverage Back y Front
- B) Construcción de Imágenes Docker y subida a ACR
- C) Deploy Back y Front en QA con pruebas de integración para Azure Web Apps
- D) Deploy Back y Front en QA con pruebas de integración para ACI
- E) Deploy Back y Front en QA con pruebas de integración para Azure Web Apps con Soporte para contenedores
- F) Aprobación manual de QA para los puntos C,D,E
- G) Deploy Back y Front en PROD para Azure Web Apps
- H) Deploy Back y Front en PROD para ACI
- I) Deploy Back y Front en PROD para Azure Web Apps con Soporte para contenedores

**Pipeline llamado “tp7-angular” en proyecto: “tp7-angular”**