

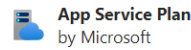
4.1 Modificar nuestro pipeline para incluir el deploy en QA y PROD de Imagenes Docker en Servicio Azure App Services con Soporte para Contenedores

[Home](#) > [Sample01TP5](#) > [Marketplace](#) >

Create App Service Plan ...

Basics   Tags   Review + create

Summary



Details

|                  |                       |
|------------------|-----------------------|
| Subscription     | Azure subscription 1  |
| Resource Group   | Sample01TP5           |
| Name             | MiAppPlanIF           |
| Operating System | Linux                 |
| Region           | Brazil South          |
| SKU              | Free                  |
| ACU              | Shared infrastructure |
| Memory           | 1 GB memory           |

[Home](#) >

Microsoft.Web-ASP-Portal-124e290c-a58d | Overview ...

Overview

Inputs

Outputs

Template

Your deployment is complete

Deployment name : Microsoft.Web-ASP-Portal-124e290c-a58d   Start time : 10/16/2024, 6:23:22 PM  
Subscription : Azure subscription 1   Correlation ID : 418cd94e-3074-40e0-8c80-9ef0c9854423  
Resource group : Sample01TP5

> Deployment details

∨ Next steps

[Go to resource](#)

Give feedback

Tell us about your experience with deployment

```
52 - name: WebAppApiNameContainersQA
53   value: 'igfr-crud-api-qa'
54 - name: AppServicePlanLinux
55   value: 'MiAppPlanIF'
```

✓ #20241016.18 • Update azure-pipelines.yml for Azure Pipelines  
sample04 Angular Employee

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

Summary Environments Associated pipelines Code Coverage

Triggered by Ignacio Franciosi

Repository and version  
sample04-Angular-Employee  
main 2a78e41b

Time started and elapsed  
Today at 7:27 p. m.  
15m 10s

Warnings 2

No data was written into the file /home/vsts/work/\_temp/task\_outputs/build\_1729118010052.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\_1729117919161.txt  
Construir y Subir Imágenes Docker a ACR • Construir y Subir Imágenes Docker de front a ACR • Construir Imagen Docker para Front

Stages Jobs

Build and Test API a... 2 jobs completed 3m 12s 4 artifacts

Construir y Subir Im... 2 jobs completed 2m 44s

Desplegar Imágenes ... 1 job completed 2m 15s

Desplegar PROD 1 job completed 4m 3s

Desplegar en Azure ... 1 job completed 1m 57s

Refresh

Filter by name

Status: All

| Name ↑ ↓         | Type ↑ ↓ | Kind ↑ ↓            | Resource Group ↑ ↓ | Status ↑ ↓ |
|------------------|----------|---------------------|--------------------|------------|
| igfr-crud-api-qa | App      | app.linux.container | Sample01TP5        | Running    |





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

```
- name: WebAppFrontNameContainersQA
  value: 'igfr-front-qa'
```

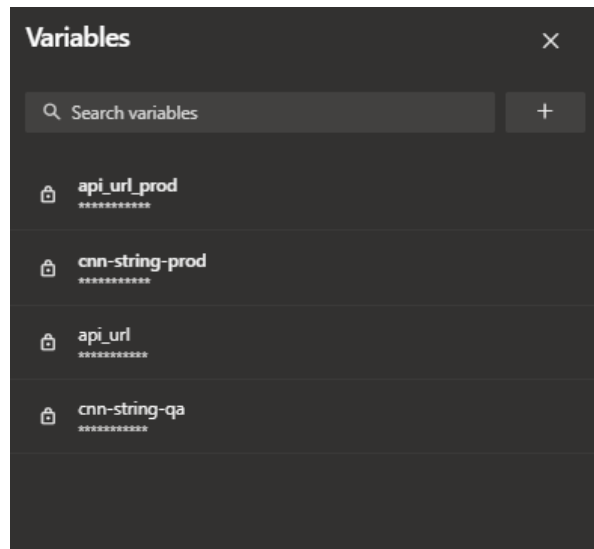
```
#App Service FRONT
Settings:
- task: AzureCLI@2
  displayName: 'Verificar y crear el recurso Azure App Service para FRONT si no existe'
  inputs:
    azureSubscription: '$(ConnectedServiceName)'
    scriptType: 'bash'
    scriptLocation: 'inlineScript'
    inlineScript: |
      # Verificar si el App Service para la FRONT ya existe
      if ! az webapp list --query "[?name=='$(WebAppFrontNameContainersQA)'] && resourceGroup=='$(ResourceGroupName)'] | length(@)" -o tsv | grep -q '^1$'; then
        echo "El App Service para FRONT no existe. Creando..."
        # Crear el App Service sin especificar la imagen del contenedor
        az webapp create --resource-group $(ResourceGroupName) --plan $(AppServicePlanLinux) --name $(WebAppFrontNameContainersQA) --deployment-container-image-name "nginx"
      else
        echo "El App Service para FRONT QA ya existe. Actualizando la imagen..."
      fi

      # Configurar el App Service para usar Azure Container Registry (ACR)
      az webapp config container set --name $(WebAppFrontNameContainersQA) --resource-group $(ResourceGroupName) \
        --container-image-name $(acrLoginServer)/$(frontImageName):$(frontImageTag) \
        --container-registry-url https://$(acrLoginServer) \
        --container-registry-user $(acrName) \
        --container-registry-password $(az acr credential show --name $(acrName) --query "passwords[0].value" -o tsv)

      # Establecer variables de entorno
      az webapp config appsettings set --name $(WebAppFrontNameContainersQA) --resource-group $(ResourceGroupName) \
        --settings ConnectionStrings_DefaultConnection="$(conn-string-qa)"
```

| Name ↑ ▾   | Type ↑ ▾ | Kind ↑ ▾            | Resource Group ↑ ▾   | Status ↑ ▾ |
|--|----------|---------------------|--|------------|
|  igfr-crud-api-qa | App      | app,linux,container |  Sample01TP5 | Running    |
|  igfr-front-qa    | App      | app,linux,container |  sample01tp5 | Running    |

**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.**



**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.**

```
- name: WebAppApiNameContainersProd
  value: igfr-container-back-prod
- name: WebAppFrontNameContainersProd
  value: igfr-container-front-prod
```

```
##### STAGE: PROD

- stage: DeployImagesToAppServiceProd
  displayName: 'Desplazar Imágenes de API y Front en Azure App Service (Prod)'
  dependsOn:
    - DeployImagesToAppServiceQA

  condition: succeeded()

  jobs:
    - job: DeployImagesToAppServiceProd
      displayName: 'Desplazar Imágenes de API y Front en Azure App Service (Prod)'
      pool:
        vmImage: 'ubuntu-latest'
      steps:
        - script: 'az deployment group create --resource-group $(ResourceGroupName) --template-file $(TemplatePath)'
          # DEPLOY DOCKER API IMAGE TO AZURE APP SERVICE (PROD)
          # =====
        - task: AzureCLI@2
          displayName: 'Verificar y crear el recurso Azure App Service para API (Prod) si no existe'
          inputs:
            azureSubscription: '$(ConnectedServiceName)'
            scriptType: 'bash'
            scriptLocation: 'inlineScript'
            inlineScript: |
              # Verificar si el App Service para la API ya existe
              if ! az webapp list --query "[?name=='$(WebAppApiNameContainersProd)' && resourceGroup=='$(ResourceGroupName)']" | length 0 -o tsv | grep -q "15"; then
                echo "El App Service para API Prod no existe. Creando..."
                # Crear el App Service sin especificar la imagen del contenedor
                az webapp create --resource-group $(ResourceGroupName) --plan $(AppServicePlanLinux) --name $(WebAppApiNameContainersProd) --deployment-container-image-name "nginx" # Especifica una imagen temporal para permitir la creación
              else
                echo "El App Service para API Prod ya existe. Actualizando la imagen..."
              fi

          # Configurar el App Service para usar Azure Container Registry (ACR)
          az webapp config container set --name $(WebAppApiNameContainersProd) --resource-group $(ResourceGroupName) \
            --container-image-name $(acrloginserver)/$(backImageName):$(backImageTag) \
            --container-registry-url https://$(acrloginserver) \
            --container-registry-user $(acrName) \
            --container-registry-password $(az acr credential show --name $(acrName) --query "passwords[0].value" -o tsv)

          # Establecer variables de entorno
          az webapp config appsettings set --name $(WebAppApiNameContainersProd) --resource-group $(ResourceGroupName) \
            --settings ConnectionStrings__DefaultConnection=$(conn-string-prod)
```

```

544
545 # Establecer variables de entorno
546 az webapp config appsettings set --name $(WebAppApiNameContainersProd) --resource-group $(ResourceGroupName) \
547 --settings ConnectionStrings_DefaultConnection="$(cnn-string-prod)"
548
549 #App-Service-PROD-FRONT
550 Settings
551 - task: AzureCLI@2
552   displayName: 'Verificar y crear el recurso Azure App Service para FRONT si no existe'
553   inputs:
554     azureSubscription: '$(ConnectedServiceName)'
555     scriptType: 'bash'
556     scriptLocation: 'inlineScript'
557     inlineScript: |
558       # Verificar si el App Service para la FRONT ya existe
559       if ! az webapp list --query "[?name=='$(WebAppFrontNameContainersProd)' && resourceGroup=='$(ResourceGroupName)']" | length(@) -o tsv | grep -q '^1$'; then
560         echo "El App Service para FRONT no existe. Creando..."
561         # Crear el App Service sin especificar la imagen del contenedor
562         az webapp create --resource-group $(ResourceGroupName) --plan $(AppServicePlanLinux) --name $(WebAppFrontNameContainersProd) --deployment-container-image-name "nginx"
563       else
564         echo "El App Service para FRONT Prod ya existe. Actualizando la imagen..."
565       fi
566
567 # Configurar el App-Service para usar Azure Container Registry (ACR)
568 az webapp config container set --name $(WebAppFrontNameContainersProd) --resource-group $(ResourceGroupName) \
569 --container-image-name $(acrLoginServer)/$(frontImageName):$(frontImageTag) \
570 --container-registry-url https://$(acrLoginServer) \
571 --container-registry-user $(acrName) \
572 --container-registry-password $(az acr credential show --name $(acrName) --query "passwords[0].value" -o tsv)
573
574 # Establecer variables de entorno
575 az webapp config appsettings set --name $(WebAppFrontNameContainersProd) --resource-group $(ResourceGroupName) \
576 --settings ConnectionStrings_DefaultConnection="$(cnn-string-prod)"
577

```

## 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

**#202410172 • Update azure-pipelines.yml for Azure Pipelines**  
sample04 Angular Employee

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

**Summary** Tests Environments Associated pipelines Code Coverage

Triggered by **Ignacio Franciosi**

Repository and version: sample04-Angular-Employee, # main, 1c7189da  
Time started and elapsed: Today at 9:25 p. m., 22m 38s

**Warnings**

No data was written into the file /home/vsts/work/\_temp/task\_outputs/build\_1729125166077.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\_1729125222057.txt  
Construir y Subir Imágenes Docker a ACR • Construir y Subir Imágenes Docker a ACR • Construir imagen Docker para Front

**Stages** Jobs

| Build and Test API a...  | Construir y Subir Im...   | Desplegar Imágenes ...                              | Desplegar Imágenes ...                      |
|--|---------------------------|---|---|
| 2 jobs completed<br>5m 17s<br>100% tests passed<br>4 artifacts | 2 jobs completed<br>2m 6s | 1 job completed<br>2m 23s                           | 1 job completed<br>4m 28s                   |
|  |                           | Desplegar en Azure ...<br>1 job completed<br>2m 12s | Desplegar PROD<br>1 job completed<br>3m 45s |