

**Semana 4**

DevOps (ISY2201)

**Guía paso a paso para trabajar imágenes de contenedores en ACR y despliegue de la imagen en Docker en Máquina Virtual a través de Azure DevOps.**

| **Profesor:** Ignacio Pastenet | **Autor:** Ignacio Pastenet |
| --- | --- |
| **Asignatura:** DevOps | **Carrera:** Ingeniería en Desarrollo de Software |
| **Versión:** 1.0 | **Fecha:** 19-08-2024 |

Contenido

[Descripción de la actividad 3](#_Toc176050570)

[Instrucciones específicas 3](#_Toc176050571)

[Requisitos técnicos 4](#_Toc176050572)

[Evidencias y ejecución paso a paso 5](#_Toc176050573)

[Evidencias Paso 1: Validación de máquina virtual creada y grupo de recurso en Azure Portal 5](#_Toc176050574)

[Evidencias Paso 2: Creación de imagen Docker y subir a Azure Container Registry (ACR) 6](#_Toc176050575)

[Evidencias Paso 3: Preparar ambiente y ejecución del Pipeline para despliegue de imagen Docker en máquina virtual. 13](#_Toc176050576)

[Actividad Finalizada. 20](#_Toc176050577)

# Descripción de la actividad

Siguiendo con el aprendizaje anterior de creación de máquina virtual y su entorno de red virtual de forma automatizada a través de Azure DevOps, ocuparemos la misma máquina virtual, por lo que la actividad contempla la configuración de contenedores en una máquina virtual para lograr la segmentación de aplicaciones, lo que resultará en la independencia de códigos y tiempos de ejecución. Además, deberás utilizar repositorios de contenedores con el propósito de centralizar imágenes, optimizando así el manejo de recursos. Esta guía incluye el paso a paso que te ayudara a crear una imagen de contenedor y subir a tu ACR para posteriormente configurar y ejecutar en Docker la imagen en una máquina virtual Linux.

## Instrucciones específicas

Deberán realizar lo siguiente:

**\*PASO1: Validación previa**

1. Validar que se este ejecutando la máquina virtual en Azure Portal
2. Si no esta en ejecución se debe desplegar nuevamente a través del pipeline de despliegue de infraestructura realizado en la experiencia n°1.
3. Confirmar la IP pública de la máquina virtual.
4. Nota: Si esta en ejecución el apache deben bajarlo con (sudo systemctl stop apache2), esto para que no choque el puerto del nginx en puerto 80, o si bien cambiar el en la imagen y ejecución del Docker.
5. Contar con un grupo de recursos creado, ideal mismo donde reside la maquina virtual.

**\*PASO 2: Creación de imagen Docker y subir a Azure Container Registry (ACR)**

1. Creación de Registro de contenedor (ACR).
2. Creación de imagen Docker (Dockerfile) “ej. Imagen nginx” y sus archivos de configuración default.conf y index.html.
3. Creación de código pipeline en yaml para subir imagen Dockerfile a ACR.
4. Subir archivos a GitHub: Dockerfile, default.conf, index.html en una estructura de directorio.
5. Subir archivo código pipeline a GitHub para subir imagen Dockerfile hacia ACR.
6. Crear Service Connections Docker Registry de tipo Azure Container Registry.
7. Creación y Ejecución de Azure Pipeline con código YAML desde repositorio GitHub para subir imagen Dockerfile.
8. Evidencias de la ejecución del pipeline en Azure Portal sección Registros de contenedor y validar que este la imagen subida.

**\*PASO 3: Preparar ambiente y ejecución del Pipeline para despliegue de imagen Docker en maquina virtual.**

1. Crear Service Connections con Azure Resource Manager y el grupo de recurso.
2. Crear un entorno en Azure pipeline para registrar la máquina virtual y dar permisos al repositorio pipeline.
3. Creación de código pipeline en yaml para instalar Docker, azure cli y hacer pull y run de imagen en ACR.
4. Subir archivo código pipeline a GitHub para posterior ejecución en Azure Pipeline.
5. Creación y Ejecución de Azure Pipeline con código YAML desde repositorio GitHub para instalar y ejecutar Docker con la imagen nginx y página de bienvenida.
6. Revisión de la ejecución del job del pipeline.
7. Ejecutar IP publica de la máquina virtual en navegador web.
8. Evidencias de la ejecución del pipeline desde máquina virtual como ejecución de Docker ps.

## Requisitos técnicos

Contar con las siguientes plataformas y herramientas:

1. Cuenta duoc registrada en Azure Portal.
2. Cuenta duoc registrada en Azure DevOps
3. Cuenta duoc o personal en GitHub (Repositorio de código) ó usar Azure Repos
4. Contar con la herramienta Putty para acceso a la máquina Linux.
5. Contar con la instalación de la herramienta Azure Cli en la laptop para validaciones extras.
6. Contar con la instalación de Docker desktop en la laptop para validaciones extras.

**Nota:** La región seleccionada debe ser EAST US 2 para todos los casos de despliegue de servicios en Azure Portal.

## Evidencias y ejecución paso a paso

**Nota:** La región que se debe seleccionar es la **EAST US 2**, para todo el aprovisionamiento de recursos cloud en **Azure Portal**.

## Evidencias Paso 1: Validación de máquina virtual creada y grupo de recurso en Azure Portal

* Login en azure portal
* URL: <http://aka.ms/devtoolsforteaching>
* Ingresar con su cuenta DUOC

Texto, Aplicación

Descripción generada automáticamente con confianza media

* Validación grupo de recurso creado

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

* Validación máquina virtual creada en grupo de recurso asignado.

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

* Validación IP Pública

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

* Validar ejecución de apache2 y bajada del servicio.

Texto

Descripción generada automáticamente

## Evidencias Paso 2: Creación de imagen Docker y subir a Azure Container Registry (ACR)

* Creación de Registro de contenedor (ACR).

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

* Detalles de la creación del registro del contenedor.

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

* Habilitar claves de acceso necesarias para la configuración por pipeline en código de despliegue.

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

* Creación de imagen Docker (Dockerfile) “ej. Imagen nginx” y sus archivos de configuración default.conf y index.html.
* Archivo Dockerfile

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente



* Archivo default.conf

Texto

Descripción generada automáticamente

* Archivo index.html

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente



* Creación de código pipeline en yaml para subir imagen Dockerfile a ACR.

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

* Crear repositorio y subir archivos a GitHub: Dockerfile, default.conf, index.html en una estructura de directorio.

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

* Subir archivo código pipeline a GitHub para subir imagen Dockerfile hacia ACR.

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

* Crear Service Connections Docker Registry de tipo Azure Container Registry.

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

* Creación y Ejecución de Azure Pipeline con código YAML desde repositorio GitHub para subir imagen Dockerfile a ACR.

Interfaz de usuario gráfica, Aplicación, Word, Teams

Descripción generada automáticamente

Interfaz de usuario gráfica, Aplicación, Teams

Descripción generada automáticamente

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

Texto

Descripción generada automáticamente

* Evidencias de la ejecución del pipeline en Azure Portal sección Registros de contenedor y validar que este la imagen subida.

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

## Evidencias Paso 3: Preparar ambiente y ejecución del Pipeline para despliegue de imagen Docker en máquina virtual.

* Crear Service Connections con Azure Resource Manager y el grupo de recurso.

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

* Crear un entorno en Azure pipeline para registrar la máquina virtual y dar permisos al repositorio pipeline.

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Interfaz de usuario gráfica

Descripción generada automáticamente

Interfaz de usuario gráfica, Aplicación, Teams

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

* Creación de código pipeline en yaml para instalar Docker, azure cli y hacer pull y run de imagen en ACR.

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

* Subir archivo código pipeline a GitHub para posterior ejecución en Azure Pipeline.

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

* Creación y Ejecución de Azure Pipeline con código YAML desde repositorio GitHub para instalar y ejecutar Docker con la imagen nginx y página de bienvenida.

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

Interfaz de usuario gráfica, Aplicación, Teams

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto

Descripción generada automáticamente

* Revisión de la ejecución del job del pipeline.

Texto

Descripción generada automáticamente

Texto

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

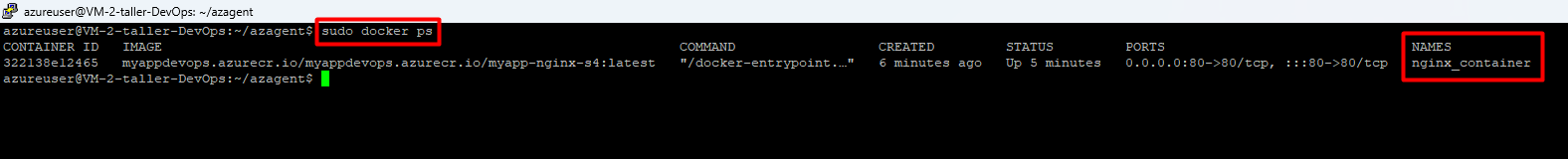
Descripción generada automáticamente

* Ejecutar IP publica de la máquina virtual en navegador web.

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

* Evidencias de la ejecución del pipeline desde máquina virtual como ejecución de Docker ps.



# Actividad Finalizada.

Con este paso a paso se evidencia la correcta ejecución de la creación de una imagen Docker y subirla al repositorio de imágenes en Azure Container Registry (ACR) de manera automatizada con Azure DevOps Pipelines.

También se evidencia paso a paso el despliegue de la instalación de Docker y la descarga de la imagen en ACR para luego ejecutar el Docker con la imagen, donde posteriormente se evidencia la ejecución del contenedor Docker y la evidencia de la página web del contenedor en el navegador web con la ip publica de la máquina virtual y su puerto asociado. Todo esto se creó de forma automatizada mediante código YAML a través de Azure DevOps Pipelines.

Con esto el alumno logra diferenciar la forma de automatizar el despliegue de contenedores e imágenes en Azure DevOps.



Reservados todos los derechos Fundación Instituto Profesional Duoc UC. No se permite copiar, reproducir, reeditar, descargar, publicar, emitir, difundir, de forma total o parcial la presente obra, ni su incorporación a un sistema informático, ni su transmisión en cualquier forma o por cualquier medio (electrónico, mecánico, fotocopia, grabación u otros) sin autorización previa y por escrito de Fundación Instituto Profesional Duoc UC La infracción de dichos derechos puede constituir un delito contra la propiedad intelectual.