

Práctica de Azure

Iker García Calviño <iker.gcalvino@udc.es>

Despliegue de contenedor mediante Docker Compose en Azure

El objetivo de esta práctica es desplegar, a través de Docker Compose, un contenedor que contenga dos servicios con las imágenes de WordPress y MySQL. La comunicación entre los servicios se realiza de forma interna dentro del contenedor desplegado, exponiendo únicamente el puerto 80 del servicio WordPress al exterior.

Recursos necesarios en Azure

Azure App Service es el servicio utilizado para hospedar aplicaciones web, API REST y back-ends para dispositivos móviles. Puede ejecutar y escalar aplicaciones en entornos basados en Windows y Linux. Para lanzar un docker-compose en Azure App Service, es necesario desplegarlo sobre un Azure Web App.

Instrucciones

1. Generar un **Azure App Service** en Linux.
2. Una vez generado el **Web App**, es necesario configurar sus **App Settings** para indicar la ruta y las credenciales (usuario/contraseña) del **Azure Container Registry**. Esto permite que el servicio busque las imágenes en el registro de contenedores de Azure y no en DockerHub (por defecto).

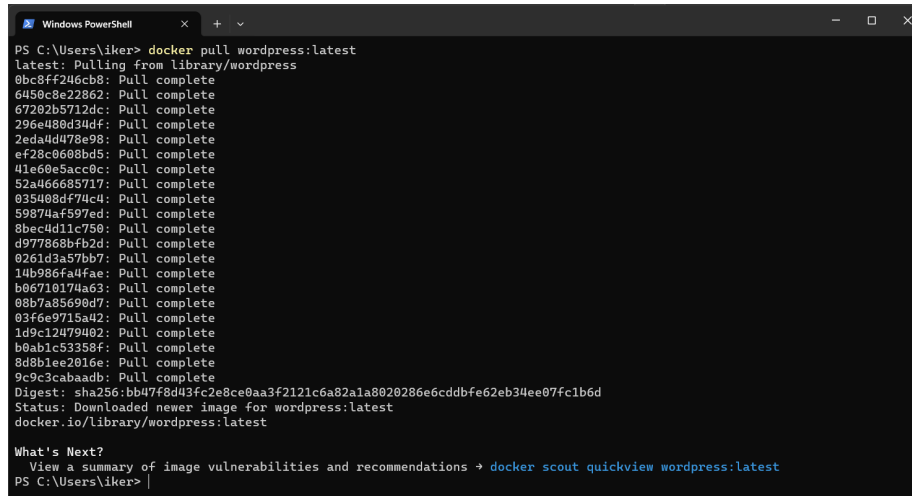
Documentación

Descripción del despliegue

Se utilizaron las imágenes de WordPress y MySQL, descargadas desde Docker-Hub, para crear un contenedor mediante Docker Compose en Azure. A continuación, se detallan los pasos realizados:

1. Descarga de imágenes desde DockerHub:

`docker pull wordpress:latest`

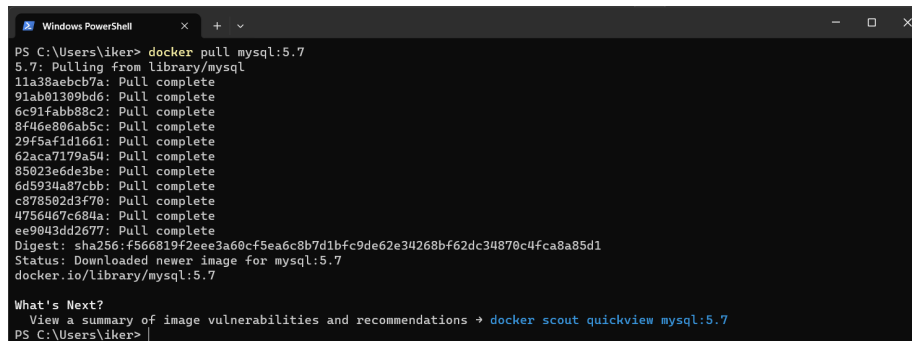


```
PS C:\Users\iker> docker pull wordpress:latest
latest: Pulling from library/wordpress
0bc8ff246cb8: Pull complete
6450c8e22862: Pull complete
67202b5712dc: Pull complete
296e480d34df: Pull complete
2eda4d478e98: Pull complete
ef28c0608bd5: Pull complete
41e60e5ace0e: Pull complete
52a466685717: Pull complete
035408df74cd: Pull complete
99874af597ed: Pull complete
8bec4d11c750: Pull complete
d977868bfb2d: Pull complete
0261d3a57bb7: Pull complete
14b986fa4fae: Pull complete
b06710174a63: Pull complete
08b7a85690d7: Pull complete
03f6e9715a42: Pull complete
1d9c12479402: Pull complete
b0ab1c53358f: Pull complete
8d8b1ee2016e: Pull complete
9c9c3cabandb: Pull complete
Digest: sha256:bb47f8d43fc2e8ce0aa3f2121c6a82a1a8020286e6cddbf62eb34ee07fc1b6d
Status: Downloaded newer image for wordpress:latest
docker.io/library/wordpress:latest

What's Next?
View a summary of image vulnerabilities and recommendations → docker scout quickview wordpress:latest
PS C:\Users\iker>
```

Figure 1: Descarga de la imagen de WordPress desde DockerHub

`docker pull mysql:5.7`



```
PS C:\Users\iker> docker pull mysql:5.7
5.7: Pulling from library/mysql
11a38aebcb7a: Pull complete
91ab01309bd6: Pull complete
6c91fab888c2: Pull complete
8f46e806ab5c: Pull complete
29f5af1d1661: Pull complete
62aca7179a5d: Pull complete
85023e6de3be: Pull complete
6d5934a87cbb: Pull complete
c878502d3f70: Pull complete
4756467c684a: Pull complete
ee9043dd2677: Pull complete
Digest: sha256:f566819f2eee3a60cf5ea6c8b7d1bfc9de62e34268bf62dc34870c4fca8a85d1
Status: Downloaded newer image for mysql:5.7
docker.io/library/mysql:5.7

What's Next?
View a summary of image vulnerabilities and recommendations → docker scout quickview mysql:5.7
PS C:\Users\iker>
```

Figure 2: Descarga de la imagen de MySQL desde DockerHub

2. Docker Compose:

Se creó un archivo `docker-compose.yml` para definir la configuración del servicio. Este archivo especifica los servicios, las imágenes, las variables de entorno y los volúmenes necesarios.

```
version: '3'

volumes:
  db:
  wordpress:

services:
  db:
    image: mysql:5.7
    restart: always
    environment:
      - MYSQL_ROOT_PASSWORD=root
      - MYSQL_DATABASE=wordpress
      - MYSQL_USER=test
      - MYSQL_PASSWORD=test
    ports:
      - "3306:3306"
    volumes:
      - db:/var/lib/mysql

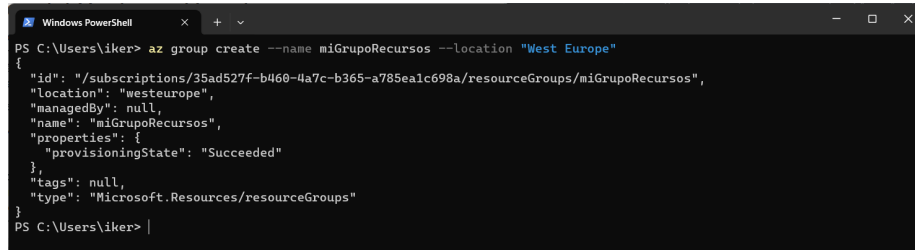
  wordpress:
    depends_on:
      - db
    image: wordpress:latest
    restart: always
    environment:
      - WORDPRESS_DB_HOST=db:3306
      - WORDPRESS_DB_USER=test
      - WORDPRESS_DB_PASSWORD=test
      - WORDPRESS_DB_NAME=wordpress
    ports:
      - "8080:80"
    volumes:
      - wordpress:/var/www/html
```

En este caso, se especifica que se van a desplegar 2 servicios: una base de datos (mysql:5.7) y una aplicación web (wordpress:latest). Este fichero también especifica que la aplicación web depende de la base de datos, por lo que se desplegará después de esta para evitar errores, y el uso de los puertos “3306:3306” y “8080:80”, respectivamente.

3. Configuración en Azure:

- Creación de un grupo de recursos:

```
az group create
--name miGrupoRecursos
--location "West Europe"
```



```
Windows PowerShell
PS C:\Users\iker> az group create --name miGrupoRecursos --location "West Europe"
{"id": "/subscriptions/35ad527f-b460-4a7c-b365-a785ealc698a/resourceGroups/miGrupoRecursos",
"location": "westeurope",
"managedBy": null,
"name": "miGrupoRecursos",
"properties": {
  "provisioningState": "Succeeded"
},
"tags": null,
"type": "Microsoft.Resources/resourceGroups"
}
PS C:\Users\iker> |
```

Figure 3: Creación de un grupo de recursos en Azure

- Creación de un registro de contenedores en Azure:

```
az acr create
--name ikergcalvinoregistry
--resource-group miGrupoRecursos
--sku Basic
```



```
PS C:\Users\iker> az acr create --name ikergcalvinoregistry --resource-group miGrupoRecursos --sku Basic
Resource provider 'Microsoft.ContainerRegistry' used by this operation is not registered. We are registering for you.
Registration succeeded.
{
  "adminUserEnabled": false,
  "anonymousPullEnabled": false,
  "creationDate": "2023-11-17T23:17:22.173159+00:00",
  "dataEndpointEnabled": false,
  "dataEndpointHostNames": [],
  "encryption": {
    "keyVaultProperties": null,
    "status": "disabled"
  },
  "id": "/subscriptions/35ad527f-b460-4a7c-b365-a785ealc698a/resourceGroups/miGrupoRecursos/providers/Microsoft.ContainerRegistry/registries/ikergcalvinoregistry",
  "identity": null,
  "location": "nesteurope",
  "loginServer": "ikergcalvinoregistry.azurecr.io",
  "name": "ikergcalvinoregistry",
  "networkRuleBypassOptions": "AzureServices",
  "networkRuleSet": null,
  "policies": {
    "azureAdAuthenticationAsArmPolicy": {
      "status": "enabled"
    },
    "exportPolicy": {
      "status": "enabled"
    },
    "quarantinePolicy": {
      "status": "disabled"
    },
    "retentionPolicy": {
      "days": 7,
      "lastUpdatedTime": "2023-11-17T23:17:28.819129+00:00",
      "status": "disabled"
    },
    "softDeletePolicy": {
      "lastUpdatedTime": "2023-11-17T23:17:28.819162+00:00",
      "retentionDays": 7,
      "status": "disabled"
    },
    "trustPolicy": {
      "status": "disabled",
      "type": "Notary"
    }
  },
  "privateEndpointConnections": [],
  "provisioningState": "Succeeded",
  "publicNetworkAccess": "Enabled",
  "resourceGroup": "miGrupoRecursos",
  "sku": {
    "name": "Basic",
    "tier": "Basic"
  },
  "status": null,
  "systemData": {
    "createdAt": "2023-11-17T23:17:22.173159+00:00",
    "createdBy": "iker.gcalvino@udc.es",
    "createdByType": "User",
    "lastModifiedAt": "2023-11-17T23:17:22.173159+00:00",
    "lastModifiedBy": "iker.gcalvino@udc.es",
    "lastModifiedByType": "User"
  },
  "tags": {},
  "type": "Microsoft.ContainerRegistry/registries",
  "zoneRedundancy": "Disabled"
}
PS C:\Users\iker>
```

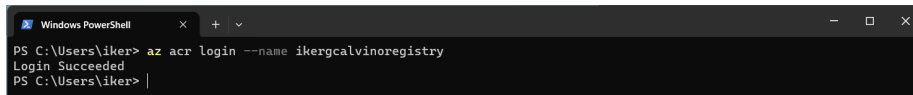
Figure 4: Creación de un registro de contenedores en Azure

- Carga de imágenes en el sistema de registro de contenedores.

Para que la aplicación web pueda utilizar las imágenes de docker, deberemos subirlas al registro de contenedores creado previamente.

- Inicio de sesión en el registro de contenedores:

```
az acr login
--name ikergcalvinoregistry
```

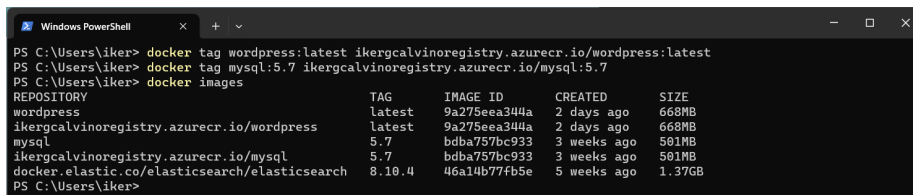


```
Windows PowerShell
PS C:\Users\iker> az acr login --name ikergcalvinoregistry
Login Succeeded
PS C:\Users\iker> |
```

Figure 5: Inicio de sesión en el registro de contenedores de Azure

- Tag y push de las imágenes al registro de contenedores de Azure:

```
docker tag wordpress:latest ikergcalvinoregistry.azurecr.io/wordpress:latest
docker tag mysql:5.7 ikergcalvinoregistry.azurecr.io/mysql:5.7
```



```
Windows PowerShell
PS C:\Users\iker> docker tag wordpress:latest ikergcalvinoregistry.azurecr.io/wordpress:latest
PS C:\Users\iker> docker tag mysql:5.7 ikergcalvinoregistry.azurecr.io/mysql:5.7
PS C:\Users\iker> docker images
REPOSITORY                                TAG                IMAGE ID           CREATED            SIZE
wordpress                                 latest             9a275eea3d4a       2 days ago        668MB
ikergcalvinoregistry.azurecr.io/wordpress latest             9a275eea3d4a       2 days ago        668MB
mysql                                      5.7                bdba757bc933       3 weeks ago       581MB
ikergcalvinoregistry.azurecr.io/mysql     5.7                bdba757bc933       3 weeks ago       581MB
docker.elastic.co/elasticsearch/elasticsearch 8.10.4            46a14b77fb5e       5 weeks ago       1.37GB
PS C:\Users\iker>
```

Figure 6: Etiquetado y envío de la imagen de WordPress al registro de contenedores de Azure

– Habilitación de la autenticación de administrador:

```
az acr update
-n ikergcalvinoregistry
--admin-enabled true
```

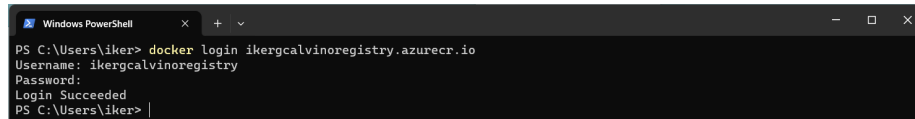


```
PS C:\Users\iker> az acr update -n ikergcalvinoregistry --admin-enabled true
{
  "adminUserEnabled": true,
  "anonymousPullEnabled": false,
  "creationDate": "2023-11-17T23:17:22.173159+00:00",
  "dataEndpointEnabled": false,
  "dataEndpointHostNames": [],
  "encryption": {
    "keyVaultProperties": null,
    "status": "disabled"
  },
  "id": "/subscriptions/35ad527f-b460-4a7c-b365-a785ealc698a/resourceGroups/miGrupoRecursos/providers/Microsoft.ContainerRegistry/registries/ikergcalvinoregistry",
  "identity": null,
  "location": "westeurope",
  "loginServer": "ikergcalvinoregistry.azurecr.io",
  "name": "ikergcalvinoregistry",
  "networkRuleBypassOptions": "AzureServices",
  "networkRuleSet": null,
  "policies": {
    "azureAdAuthenticationAsArmPolicy": {
      "status": "enabled"
    },
    "exportPolicy": {
      "status": "enabled"
    },
    "quarantinePolicy": {
      "status": "disabled"
    },
    "retentionPolicy": {
      "days": 7,
      "lastUpdatedTime": "2023-11-17T23:17:28.819129+00:00",
      "status": "disabled"
    },
    "softDeletePolicy": {
      "lastUpdatedTime": "2023-11-17T23:17:28.819162+00:00",
      "retentionDays": 7,
      "status": "disabled"
    },
    "trustPolicy": {
      "status": "disabled",
      "type": "Notary"
    }
  },
  "privateEndpointConnections": [],
  "provisioningState": "Succeeded",
  "publicNetworkAccess": "Enabled",
  "resourceGroup": "miGrupoRecursos",
  "sku": {
    "name": "Basic",
    "tier": "Basic"
  },
  "status": null,
  "systemData": {
    "createdAt": "2023-11-17T23:17:22.173159+00:00",
    "createdBy": "iker.gcalvino@udc.es",
    "createdByType": "User",
    "lastModifiedAt": "2023-11-17T23:22:13.438443+00:00",
    "lastModifiedBy": "iker.gcalvino@udc.es",
    "lastModifiedByType": "User"
  },
  "tags": {},
  "type": "Microsoft.ContainerRegistry/registries",
  "zoneRedundancy": "Disabled"
}
```

Figure 7: Habilitación de la autenticación de administrador en el registro de contenedores de Azure

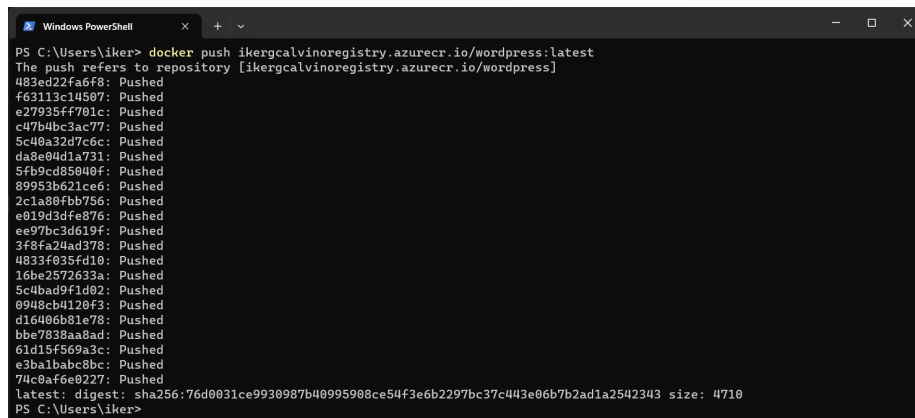
- Inicio de sesión en el registro de contenedores de Azure y push de imágenes:

```
docker login ikergcalvinoregistry.azurecr.io
docker push ikergcalvinoregistry.azurecr.io/wordpress:latest
docker push ikergcalvinoregistry.azurecr.io/mysql:5.7
```



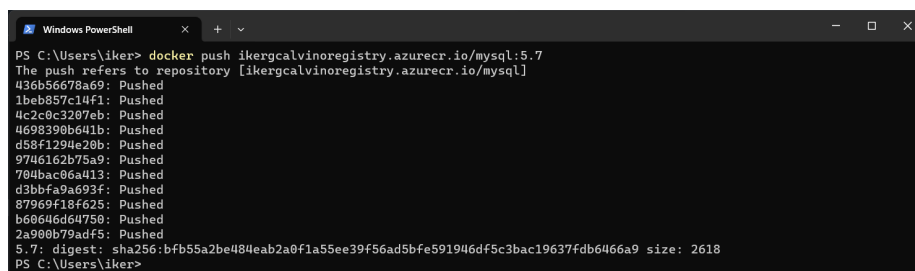
```
Windows PowerShell
PS C:\Users\iker> docker login ikergcalvinoregistry.azurecr.io
Username: ikergcalvinoregistry
Password:
Login Succeeded
PS C:\Users\iker> |
```

Figure 8: Inicio de sesión



```
Windows PowerShell
PS C:\Users\iker> docker push ikergcalvinoregistry.azurecr.io/wordpress:latest
The push refers to repository [ikergcalvinoregistry.azurecr.io/wordpress]
483ed22fa6f8: Pushed
f63113c14507: Pushed
e27935ff701c: Pushed
c47b4bc3ac77: Pushed
5c40a32d7c6c: Pushed
da8e04d1a731: Pushed
5fb9cd85040f: Pushed
89953b621ce6: Pushed
2c1a80fbb756: Pushed
e019d3dfc876: Pushed
ee97bc3d619f: Pushed
3f8fa24ad378: Pushed
4833f035fd10: Pushed
16be2572633a: Pushed
5c4bad9fd02: Pushed
0948cb4120f3: Pushed
d16406b81e78: Pushed
bbe783aa8ad: Pushed
61d15f569a3c: Pushed
e3ba1babc8bc: Pushed
74c0af6e0227: Pushed
latest: digest: sha256:76d0031ce9930987b40995908ce54f3e6b2297bc37c443e06b7b2ad1a2542343 size: 4710
PS C:\Users\iker>
```

Figure 9: Envío de la imagen de WordPress al registro de contenedores de Azure



```
Windows PowerShell
PS C:\Users\iker> docker push ikergcalvinoregistry.azurecr.io/mysql:5.7
The push refers to repository [ikergcalvinoregistry.azurecr.io/mysql]
436b56678a69: Pushed
1beb857c14f1: Pushed
4c2c0c3207eb: Pushed
4698390b641b: Pushed
d58f1294e20b: Pushed
9746162b75a9: Pushed
704bac06a413: Pushed
d3bbfa9a693f: Pushed
87969f18f625: Pushed
b66646d6d750: Pushed
2a900b79adf5: Pushed
5.7: digest: sha256:bfb55a2be484eab2a0f1a55ee39f56ad5bfe591946df5c3bac19637fdb6466a9 size: 2618
PS C:\Users\iker>
```

Figure 10: Envío de la imagen de MySQL al registro de contenedores de Azure

Podemos comprobar que se han subido correctamente desde la web de Azure.

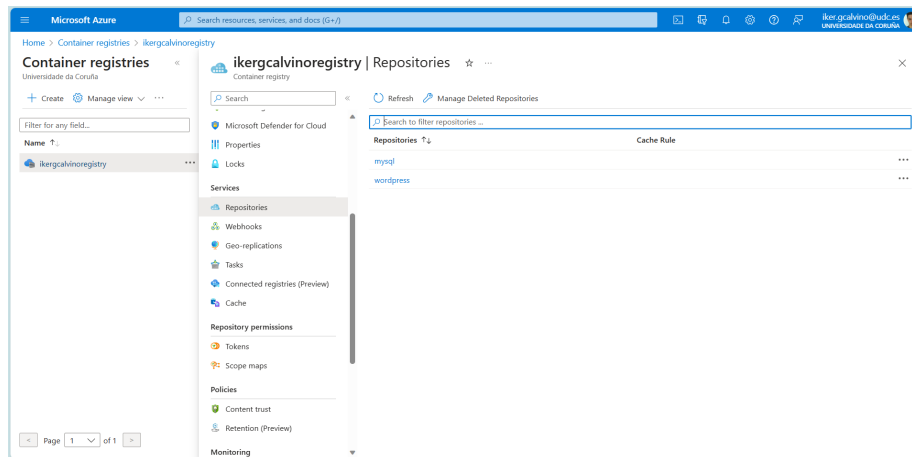
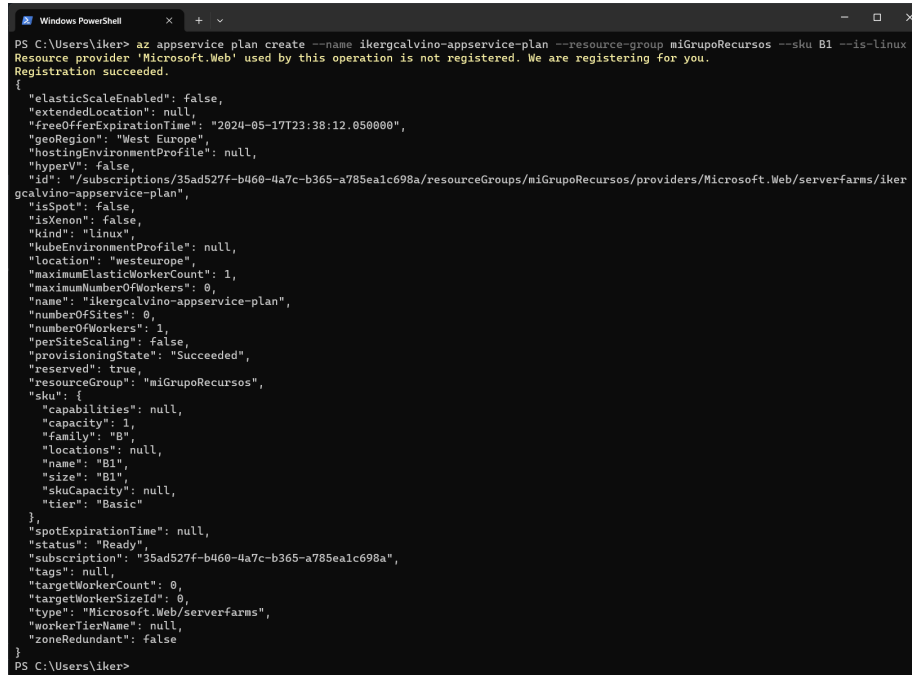


Figure 11: Verificación de imágenes en el portal de Azure

- Creación de un plan de servicio de aplicaciones:

```
az appservice plan create
--name ikergcalvino-appservice-plan
--resource-group miGrupoRecursos
--sku B1
--is-linux
```



```
PS C:\Users\iker> az appservice plan create --name ikergcalvino-appservice-plan --resource-group miGrupoRecursos --sku B1 --is-linux
Resource provider 'Microsoft.Web' used by this operation is not registered. We are registering for you.
Registration succeeded.
{
  "elasticScaleEnabled": false,
  "extendedLocation": null,
  "freeOfferExpirationTime": "2024-05-17T23:38:12.050000",
  "geoRegion": "West Europe",
  "hostingEnvironmentProfile": null,
  "hyperv": false,
  "id": "/subscriptions/35ad527f-b460-4a7c-b365-a785ealc698a/resourceGroups/miGrupoRecursos/providers/Microsoft.Web/serverfarms/iker
gcalvino-appservice-plan",
  "isSpot": false,
  "isXenon": false,
  "kind": "linux",
  "kubeEnvironmentProfile": null,
  "location": "westeurope",
  "maximumElasticWorkerCount": 1,
  "maximumNumberOfWorkers": 0,
  "name": "ikergcalvino-appservice-plan",
  "numberOfSites": 0,
  "numberOfWorkers": 1,
  "perSiteScaling": false,
  "provisioningState": "Succeeded",
  "reserved": true,
  "resourceGroup": "miGrupoRecursos",
  "sku": {
    "capabilities": null,
    "capacity": 1,
    "family": "B",
    "locations": null,
    "name": "B1",
    "size": "B1",
    "skuCapacity": null,
    "tier": "Basic"
  },
  "spotExpirationTime": null,
  "status": "Ready",
  "subscription": "35ad527f-b460-4a7c-b365-a785ealc698a",
  "tags": null,
  "targetWorkerCount": 0,
  "targetWorkerSizeId": 0,
  "type": "Microsoft.Web/serverfarms",
  "workerTierName": null,
  "zoneRedundant": false
}
PS C:\Users\iker>
```

Figure 12: Creación de un plan de servicio de aplicaciones en Azure

- Creación de un servicio de aplicaciones web:

```
az webapp create
--name IkerGarciaWebApp
--plan ikergcalvino-appservice-plan
--resource-group miGrupoRecursos
--multicontainer-config-file docker-compose.yml
--multicontainer-config-type COMPOSE
```

```
PS C:\repositorios\Prácticas\Prácticas de Azure> az webapp create --name IkerGarciaWebApp --plan ikergercalvino-appservice-plan --reso
source-group miGrupoRecursos --multicontainer-config-file docker-compose.yml --multicontainer-config-type COMPOSE
{
  "availabilityState": "Normal",
  "clientAffinityEnabled": true,
  "clientCertEnabled": false,
  "clientCertExclusionPaths": null,
  "clientCertMode": "Required",
  "cloningInfo": null,
  "containerSize": 0,
  "customDomainVerificationId": "3018898F965C7C606677813D7977126333F880809C1D1D1E13FE8B985CED91D",
  "dailyMemoryTimeQuota": 0,
  "defaultHostName": "ikergercalvino.azurewebsites.net",
  "enabled": true,
  "enabledHostNames": [
    "ikergercalvino.azurewebsites.net",
    "ikergercalvino.scm.azurewebsites.net"
  ],
  "extendedLocation": null,
  "ftpPublishingUrl": "https://www-prod-an2-769.ftp.azurewebsites.windows.net/site/wwwroot",
  "hostNameSslStates": [
    {
      "certificateResourceId": null,
      "hostType": "Standard",
      "ipBasedSslResult": null,
      "ipBasedSslState": "NotConfigured",
      "name": "ikergercalvino.azurewebsites.net",
      "sslState": "Disabled",
      "thumbprint": null,
      "toUpdate": null,
      "toUpdateIpBasedSsl": null,
      "virtualIPv6": null,
      "virtualIp": null
    },
    {
      "certificateResourceId": null,
      "hostType": "Repository",
      "ipBasedSslResult": null,
      "ipBasedSslState": "NotConfigured",
      "name": "ikergercalvino.scm.azurewebsites.net",
      "sslState": "Disabled",
      "thumbprint": null,
      "toUpdate": null,
      "toUpdateIpBasedSsl": null,
      "virtualIPv6": null,
      "virtualIp": null
    }
  ],
  "hostNames": [
    "ikergercalvino.azurewebsites.net"
  ],
  "hostNamesDisabled": false,
  "hostingEnvironmentProfile": null,
  "httpsOnly": false,
  "httpV": false,
  "id": "/subscriptions/35ad527f-b460-4a7c-b365-a785ealc698a/resourceGroups/miGrupoRecursos/providers/Microsoft.Web/sites/IkerGarciaWebApp",
  "identity": null,
  "inProgressOperationId": null,
  "isDefaultContainer": null,
  "isXenon": false,
  "keyVaultReferenceIdentity": "SystemAssigned",
  "kind": "app,linux,container",
  "lastModifiedTimeUtc": "2023-11-17T23:58:20.596666",
  "location": "West Europe",
  "maxNumberOfWorkers": null,
  "name": "IkerGarciaWebApp",
  "outboundIPAddresses": "20.101.188.174,20.101.189.236,20.101.190.64,20.101.190.255,20.101.191.10,20.101.191.37,20.86.209.248,20.86.210.26,20.86.211.60,20.86.211.198,20.86.212.16,20.86.212.165,20.105.232.49",
  "possibleOutboundIPAddresses": "20.101.188.174,20.101.189.236,20.101.190.64,20.101.190.255,20.101.191.10,20.101.191.37,20.86.209.248,20.86.210.26,20.86.211.60,20.86.211.198,20.86.212.16,20.86.212.165,20.86.212.286,20.86.213.183,20.86.215.112,20.86.215.141,20.101.184.31,20.101.185.92,20.101.186.189,20.101.187.75,20.101.187.78,20.101.187.82,20.101.188.20,20.101.188.22,20.101.188.174,20.101.189.236,20.101.190.64,20.101.190.255,20.101.191.10,20.101.191.37,20.101.191.66,20.101.191.74,20.101.191.82,20.101.191.118,20.101.191.178,20.22.40.160,20.105.232.49",
  "publicNetworkAccess": null,
  "redundancyMode": "None",
  "repositorySiteName": "IkerGarciaWebApp",
  "reserved": true,
  "resourceGroup": "miGrupoRecursos",
  "scmSiteAlsoStopped": false,
  "serverFarmId": "/subscriptions/35ad527f-b460-4a7c-b365-a785ealc698a/resourceGroups/miGrupoRecursos/providers/Microsoft.Web/serverfarms/IkerGarciaWebApp",
  "siteConfig": {
    "acrUseManagedIdentityCreds": false,
    "acrUserManagedIdentityId": null,
    "alwaysOn": false,
    "antivirusScanEnabled": null,
    "apiDefinition": null,
    "apiManagementConfig": null,
    "appCommandLine": null,
    "appSettings": null,
    "autoHealEnabled": null,
    "autoHealRules": null,
    "autoSwapSlotName": null,
    "azureMonitorLogCategories": null,
    "azureStorageAccounts": null,
    "connectionStrings": null,
    "cors": null,
    "customAppPoolIdentityAdminState": null,
    "customAppPoolIdentityTenantState": null,
    "defaultDocuments": null,
    "detailedErrorLoggingEnabled": null,
    "documentRoot": null,
    "elasticWebAppScaleLimit": 0,
    "experiments": null,
    "fileChangeAuditEnabled": null,
    "ftpState": null,
    "functionAppScaleLimit": null,
    "functionsRuntimeScaleMonitoringEnabled": null,
    "handlerMappings": null,
    "healthCheckPath": null,
    "http20Enabled": false,
    "http20ProxyFlag": null,
    "httpLoggingEnabled": null,
    "ipSecurityRestrictions": [
      {
        "action": "Allow",
        "description": "Allow all access",
        "headers": null,
        "ipAddress": "Any",
        "name": "Allow all",
        "priority": 2147483647,
        "subnetMask": null
      }
    ]
  }
}
```

Figure 13: Creación de un servicio de aplicaciones web en Azure (Parte 1)

```
Windows PowerShell
"subnetTrafficTag": null,
"tag": null,
"vnetSubnetResourceId": null,
"vnetTrafficTag": null
}
],
"ipSecurityRestrictionsDefaultAction": null,
"javaContainer": null,
"javaContainerVersion": null,
"javaVersion": null,
"keyVaultReferenceIdentity": null,
"limits": null,
"linuxFxVersion": "",
"loadBalancing": null,
"localMySQLEnabled": null,
"logsDirectorySizeLimit": null,
"machineKey": null,
"managedPipelineMode": null,
"managedServiceIdentityId": null,
"metadata": null,
"minTlsCipherSuite": null,
"minTlsVersion": null,
"minimumElasticInstanceCount": 0,
"netFrameworkVersion": null,
"nodeVersion": null,
"numberOfWorkers": 1,
"phpVersion": null,
"powershellVersion": null,
"preWarmedInstanceCount": null,
"publicNetworkAccess": null,
"publishingPassword": null,
"publishingUsername": null,
"push": null,
"pythonVersion": null,
"remoteDebuggingEnabled": null,
"remoteDebuggingVersion": null,
"requestTracingEnabled": null,
"requestTracingExpirationTime": null,
"routingRules": null,
"runtimeADUser": null,
"runtimeADUserPassword": null,
"scmIpSecurityRestrictions": [
{
"action": "Allow",
"description": "Allow all access",
"headers": null,
"ipAddress": "Any",
"name": "Allow all",
"priority": 2147483647,
"subnetMask": null,
"subnetTrafficTag": null,
"tag": null,
"vnetSubnetResourceId": null,
"vnetTrafficTag": null
}
],
"scmIpSecurityRestrictionsDefaultAction": null,
"scmIpSecurityRestrictionsUseMain": null,
"scmMinTlsVersion": null,
"scmType": null,
"sitePort": null,
"sitePrivateLinkHostEnabled": null,
"storageType": null,
"supportedTlsCipherSuites": null,
"tracingOptions": null,
"use32BitWorkerProcess": null,
"virtualApplications": null,
"vnetName": null,
"vnetPrivatePortCount": null,
"vnetRouteAllEnabled": null,
"webSocketsEnabled": null,
"websiteTimeZone": null,
"winAuthAdminState": null,
"winAuthTenantState": null,
"windowsConfiguredStacks": null,
"windowsFxVersion": null,
"xManagedServiceIdentityId": null
},
"slotSwapStatus": null,
"state": "Running",
"storageAccountRequired": false,
"suspendedUntil": null,
"tags": null,
"targetSwapSlot": null,
"trafficManagerHostNames": null,
"type": "Microsoft.Web/sites",
"usageState": "Normal",
"virtualNetworkSubnetId": null,
"vnetContentShareEnabled": false,
"vnetImagePullEnabled": false,
"vnetRouteAllEnabled": false
}
PS C:\repositorios\Prácticas\Prácticas de Azure>
```

Figure 14: Creación de un servicio de aplicaciones web en Azure (Parte 2)

- A continuación, configuramos la aplicación web para que acceda al registro de contenedores creado anteriormente.

```
az webapp config container set
--name IkerGarciaWebApp
--resource-group miGrupoRecursos
--docker-registry-server-url ikergcalvinoregistry.azurecr.io
--docker-registry-server-user ikergcalvinoregistry
--docker-registry-server-password nbXfPUWkGsKLyPLyHcBdo80WJUS6TA+4eJfZrs82n1+AC
```

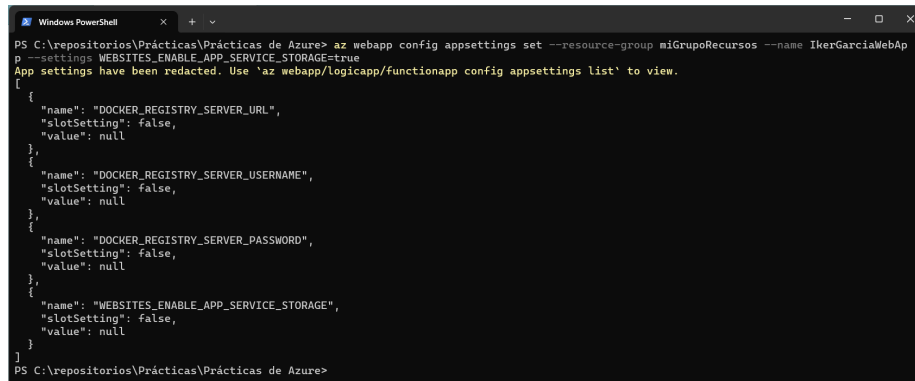
[illegible]

Figure 15: Configuración del registro de contenedores en el servicio de aplicaciones web de Azure

– Configuración adicional:

Según la documentación oficial de Azure, se afirma que la persistencia de almacenamiento de datos está habilitada de forma predeterminada para las aplicaciones web en entornos Linux. Sin embargo, en una discusión de un hilo de Stack Overflow, se menciona lo contrario. Por ello, para evitar posibles problemas, ejecutaremos el siguiente comando:

```
az webapp config appsettings set
--resource-group miGrupoRecursos
--name IkerGarciaWebApp
--settings WEBSITES_ENABLE_APP_SERVICE_STORAGE=true
```



```
Windows PowerShell
PS C:\repositorios\Prácticas\Prácticas de Azure> az webapp config appsettings set --resource-group miGrupoRecursos --name IkerGarciaWebApp --settings WEBSITES_ENABLE_APP_SERVICE_STORAGE=true
App settings have been redacted. Use 'az webapp/logicapp/functionapp config appsettings list' to view.
[
  {
    "name": "DOCKER_REGISTRY_SERVER_URL",
    "slotSetting": false,
    "value": null
  },
  {
    "name": "DOCKER_REGISTRY_SERVER_USERNAME",
    "slotSetting": false,
    "value": null
  },
  {
    "name": "DOCKER_REGISTRY_SERVER_PASSWORD",
    "slotSetting": false,
    "value": null
  },
  {
    "name": "WEBSITES_ENABLE_APP_SERVICE_STORAGE",
    "slotSetting": false,
    "value": null
  }
]
PS C:\repositorios\Prácticas\Prácticas de Azure>
```

Figure 16: Configuración adicional en el servicio de aplicaciones web de Azure

4. Acceso a la aplicación en Azure:

Al acceder por primera vez, es posible que necesites instalar y configurar WordPress. Para realizar esto, sigue estos pasos:

1. Accede a la siguiente dirección en tu navegador:

`https://ikergarciawebapp.azurewebsites.net/wp-admin/install.php`

2. Selecciona el idioma de tu preferencia (en este caso, se ha escogido la opción de instalar WordPress en español).
3. Completa la configuración requerida según tus preferencias. Puedes referirte a la documentación proporcionada en la práctica para obtener más detalles sobre la configuración realizada.

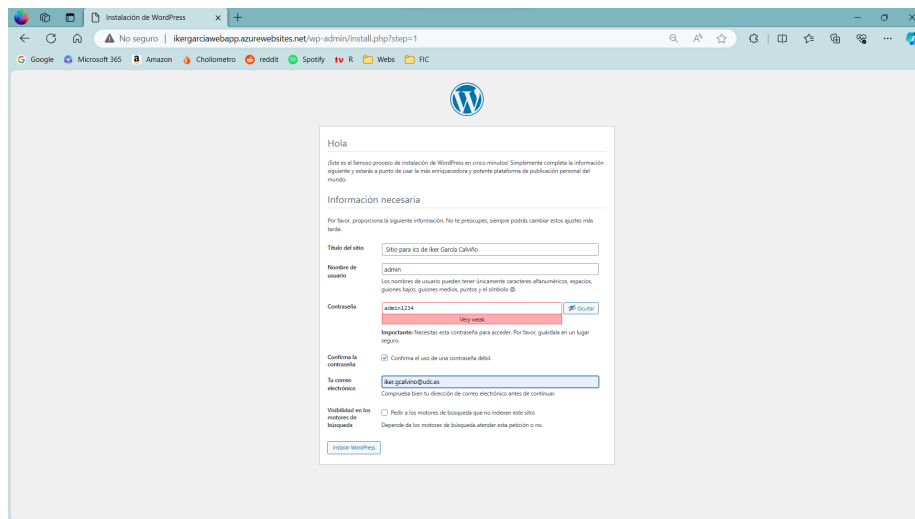


Figure 17: Acceso inicial a la instalación de WordPress en la aplicación web de Azure

4. Una vez completada la instalación, podrás observar la configuración en la interfaz de administración de WordPress.

Una vez completados los pasos anteriores, la aplicación estará disponible en la URL proporcionada por el servicio de aplicaciones web de Azure: IkerGarciaWebApp.

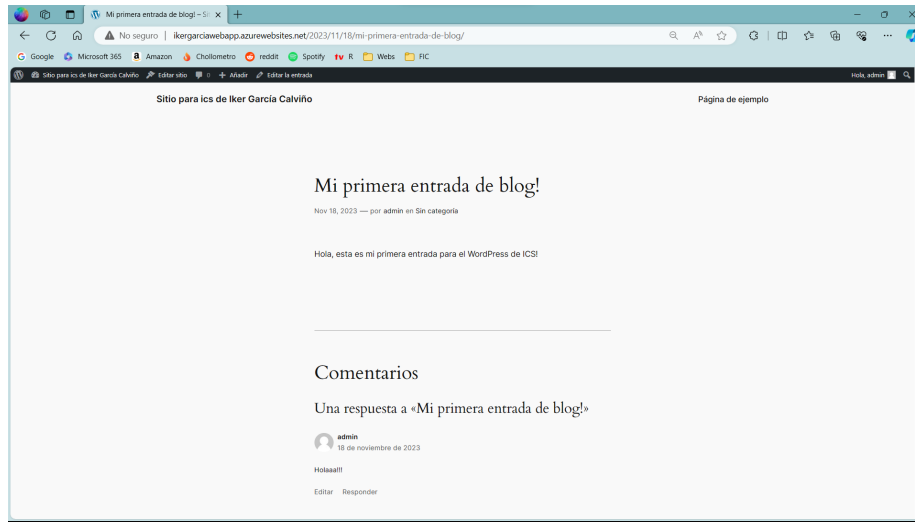


Figure 18: Sitio para ics de Iker García Calviño

Como vemos en la imagen, ya podemos tanto navegar por la web como crear, ver y comentar en los distintos posts que se encuentran en la web.