Lab 09b - Implement Azure Container Instances

Contoso wants to find a new platform for its virtualized workloads. You identified a number of container images that can be leveraged to accomplish this objective. Since you want to minimize container management, you plan to evaluate the use of Azure Container Instances for deployment of Docker images.

In this lab, you will:

* Task 1: Deploy a Docker image by using the Azure Container Instance
* Task 2: Review the functionality of the Azure Container Instance

A picture containing table

Description automatically generated

Exercise 1

Task 1: Deploy a Docker image by using the Azure Container Instance

In this task, we will create a new container instance for the web application.

We sign in the Azure portal, locate and open Container instances and click Create. We create a new container with the following settings:

| Setting | Value |
| --- | --- |
| Subscription | the name of the Azure subscription you are using in this lab |
| Resource group | the name of a new resource group **az104-09b-rg1** |
| Container name | **az104-9b-c1** |
| Region | the name of a region where you can provision Azure container instances |
| Image Source | **Quickstart images** |
| Image | **mcr.microsoft.com/azuredocs/aci-helloworld:latest (Linux)** |

And on the networking tab we set any valid, globally unique DNS host name in my case **vlatko283**

Graphical user interface, application, table

Description automatically generated

After reviewing the other settings and the validation passed we can now Create the instance.

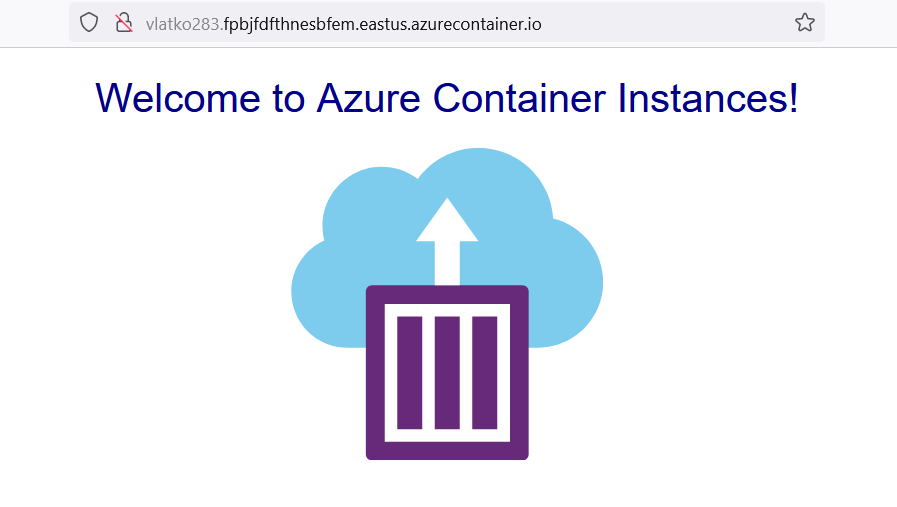
Task 2: Review the functionality of the Azure Container Instance

After the deployment is complete, we click the Go to resource link.

Graphical user interface, text, application

Description automatically generated

We can now verify that the container Status is Running. We copy the value of FQDN and paste the URL in a new browser tab to verify that the Welcome to Azure Container Instance page is displayed.



We close the tab and navigate to the Settings section of the container instance, we click on Containers and then Logs. We verify that we see the log representing the HTTP GET request generated by displaying the application in the browser.

