# Module 5: Implementing Azure App Service

# Lab: Implementing web apps

### Scenario

The A. Datum Corporation's public-facing web app currently runs on an IIS web server at the company's chosen ISP. A. Datum wants to migrate this web app into Azure. You must test the Web Apps functionality by setting up a test A. Datum web app. An internal team provides you with a test web app to deploy. You must ensure that they can continue to stage changes to the test web app before deploying those changes to the public-facing site. A. Datum is a global company, so you also want to test Azure Traffic Manager, and show your organization's decision makers how it distributes traffic to instances close to users of the web app.

### Objectives

After completing this lab, you will be able to:

* Create a new web app.
* Deploy a web app.
* Manage web apps.
* Implement Traffic Manager to load-balance web apps.

### Lab Setup

Estimated Time: 60 minutes

Virtual machine: **20533E-MIA-CL1**

User name: **Student**

Password: **Pa55w.rd**

## Exercise 1: Creating web apps

### Scenario

You must set up a test web app in Azure. As the first step in the setup process, you want to create a new web app. Later in this lab, you will deploy this web app to the test web app.

The main tasks for this exercise are as follows:

1. Create a web app
2. Add a deployment slot
3. Configure deployment credentials

#### Task 1: Create a web app

1. Ensure that you are signed in to MIA-CL1 as **Student** with the password **Pa55w.rd**.
2. Open Microsoft Edge, browse to the Azure portal, and then sign in using the Microsoft account that is the Service Administrator of your subscription.
3. To create a new web app, use the following information:

* App name: any unique valid name
* Resource Group: **20533E0501-LabRG**
* Web Hosting Plan Name: **20533E0501LabPlan**
* Location: an Azure region close to the lab location, in which you can create app service plans.
* Pricing tier: **S1 Standard**
* Application Insights: leave at its default value

#### Task 2: Add a deployment slot

1. In the Azure portal, add a new deployment slot to the web app that you created in the first task, using the following information:

* Name: **Staging**
* Configuration Source: accept the default setting

1. Open Windows PowerShell window and authenticate to your Azure subscription by signing in using the Microsoft account that is the Service Administrator of your subscription.
2. If you have multiple subscriptions, select the target one by running the Azure PowerShell **Set-AzureRmContext** cmdlet.
3. Use the Azure PowerShell **Get-AzureRmWebApp** and **Get-AzureRmWebAppSlot** cmdlets to identify the web app and staging slot that you created.
4. Keep the Azure PowerShell window open.

#### Task 3: Configure deployment credentials

1. In the Azure portal, on the web app blade, set the following deployment credentials for the web app that you created in the first task:

* FTP/Deployment User Name: a unique name
* Password: **Pa55w.rd**

**Result**: After completing this exercise, you should have created a new web app in the Azure portal, and configured the new web app with deployment slots and deployment credentials.

## Exercise 2: Deploying a web app

### Scenario

Now that you created a web app in Azure, and added a deployment slot for the web app, you can publish the internally developed web app that the A. Datum web-development team supplied. In this exercise, you will use a publishing profile in Visual Studio to connect to the new web app and deploy the web content.

The main tasks for this exercise are as follows:

1. Obtain a publishing profile
2. Deploy a web app

#### Task 1: Obtain a publishing profile

1. From the Azure portal, download the publish profile for the Web app you created in Exercise 1.
2. Open the web-application project stored in **F:\LabFiles\Lab05\Starter\AdatumWebsite\AdatumWebsite.sln** in **Visual Studio**.
3. Start debugging the web application, examine the web page automatically displayed on a new Microsoft Edge tab and then, close that tab.

**Note:** When you start the web application in Visual Studio, the web app runs in IIS Express on your local workstation.

#### Task 2: Deploy a web app

1. In Visual Studio, start the Publish Wizard for the **AdatumWebsite** project, and then import the **.PublishSettings** file that you downloaded in task 1 of this exercise.
2. Publish the new website to Azure.

**Note:** When the operation is complete, Microsoft Edge opens and displays the new web app hosted in Azure.

1. Verify that A. Datum's web app opens in Microsoft Edge and then verify the web app's current address.
2. Close Microsoft Edge.
3. Leave Visual Studio open.

**Result**: After completing this exercise, you should have deployed a web app hosted in Azure.

## Exercise 3: Managing web apps

### Scenario

The web-deployment team created an updated style sheet for the A. Datum's test web app. You have to demonstrate how you can deploy these changes to a staging slot, and then test them, before you deploy to the production A. Datum web app. In this exercise, you will upload the new web app to the staging slot that you created in Exercise 1, and you then will swap the new version of the web app into the production slot.

The main tasks for this exercise are as follows:

1. Deploy a web app for staging
2. Swap deployment slots
3. Roll back a deployment

#### Task 1: Deploy a web app for staging

1. In the Azure portal, download a publishing profile for the Staging slot for your web app.
2. Open the project in **F:\LabFiles\Lab05\Starter\NewAdatumWebsite\AdatumWebsite.sln** in **Visual Studio**.
3. Start the web app publishing process and import the staging publishing profile that you downloaded in the first step of this task.
4. Publish the new web app to the Staging slot.
5. Close Microsoft Edge.
6. Leave Visual Studio open.

#### Task 2: Swap deployment slots

1. In Microsoft Edge, in the Azure portal, navigate to the web app that you created in Exercise 1.
2. From the Azure portal, use the **URL** link for your web app to open it in another Microsoft Edge tab.
3. Notice that the color scheme has not changed, because the Web app with the new color scheme is still in the staging slot. Close the Microsoft Edge tab displaying the A. Datum web app.
4. From the web app blade in the Azure portal, swap the staging and production web-app slots.
5. When the swap completes, use the **URL** link again to browse to the web app and notice that the color scheme has changed.
6. Close the Microsoft Edge tab that displays the A. Datum's web app.

#### Task 3: Roll back a deployment

1. In the Azure portal, swap the staging and production slots again.

**Note:** By swapping the slots a second time, you simulate a deployment rollback.

1. When the swap is complete, browse to the web app. Notice that the color scheme has reverted to the original one.
2. Close the Microsoft Edge tab displaying the A. Datum web app.

**Result**: After completing this exercise, you should have an updated web app in the staging slot and have tested the slot swap functionality.

## Exercise 4: Implementing Traffic Manager

### Scenario

Because A. Datum has customers around the globe, you must ensure that the A. Datum web apps perform well when serving requests from multiple locations around the world. You must evaluate Traffic Manager to verify that web content is served from a location that is close to customers. To accomplish this, you will set up a deployment of Traffic Manager serving content of a test web app from two different Azure regions.

The main tasks for this exercise are as follows:

1. Deploy a web app to another region
2. Create a Traffic Manager profile
3. Add endpoints, and configure Traffic Manager
4. Test Traffic Manager
5. Remove the lab environment

#### Task 1: Deploy a web app to another region

1. In Azure PowerShell, identify the settings of your test web app by using the **Get-AzureRmWebApp** cmdlet. Note the name of the web app and its location.
2. Choose an Azure region where you can provision an Azure web apps and which is different from the location of the original web app, preferably on a different continent. This will become the ***SecondLocation***.
3. Use the **New-AzureRmResourceGroup** cmdlet to create a new resource group named **20533E0502-LabRG** located in the ***SecondLocation***.
4. Use the **New-AzureRmAppServicePlan** cmdlet to create a new App Service plan named **20533E0502LabPlan** with the Standard pricing tier in the resource group **20533E0502-LabRG** and the ***SecondLocation***.
5. Use the **New-AzureRmWebApp** cmdlet to create a new web app. Use the following information for the web app:

* Resource group: **20533E0502-LabRG**
* Name: a unique name (use the **Test-AzureRmDnsAvailability** cmdlet to identify it)
* Service plan: **20533E0502LabPlan**
* Location: ***SecondLocation***

1. In the Azure portal, download a publishing profile for the web app you just created.
2. Open the project in **F:\LabFiles\Lab05\Starter\AdatumWebsite\AdatumWebsite.sln** in **Visual Studio**.
3. Start the Publish Web Wizard, and then import the publish settings file that you just downloaded.
4. Publish the web app, and then close Microsoft Edge and Visual Studio.

#### Task 2: Create a Traffic Manager profile

1. In the Azure portal, create a new Traffic Manager profile by using the following information:

* Name: a unique domain name
* Routing Method: **Performance**
* Resource Group: **20533E0503-LabRG**
* Resource group location: ***the Azure region in which you provisioned the first web app***

#### Task 3: Add endpoints, and configure Traffic Manager

1. From the Traffic Manager profile blade in the Azure portal, add the web apps that you created in Exercise 1 and Exercise 4 as the Traffic Manager profile endpoints.
2. From the Traffic Manager profile blade, modify the profile configuration by setting the DNS TTL value to 30 seconds.

#### Task 4: Test Traffic Manager

1. From the Azure portal, use the DNS name of the Traffic Manager profile to browse to the web app instance corresponding to the closest endpoint.
2. Use the **nslookup** command to resolve the DNS name of the Traffic Manager profile.

**Note:** Review the DNS records listed in the output of the command to identify the web app instance returned from the Traffic Manager profile

1. In the Azure portal, disable the Traffic Manager endpoint representing the web app instance you identified in the previous step.
2. Use the **nslookup** command again to resolve the DNS NAME for your Traffic Manager profile. The results should differ from those in step 2.

**Note:** You might have to wait in order for the endpoint state change to take effect. Wait about 1 minute and re-run the **nslookup** command.

#### Task 5: Remove the lab environment

1. On MIA-CL1, close all open windows without saving any files.
2. Start **Windows PowerShell** as Administrator and, from the **Administrator: Windows PowerShell** window, run **Remove-20533EEnvironment**.
3. When prompted, sign in by using the Microsoft account that is the Service Administrator of your Azure subscription.
4. If you have multiple Azure subscriptions, select the one you want the script to target.
5. If prompted, specify the current lab number.
6. When prompted for confirmation, type **y**.
7. Start Microsoft Edge, browse to the Azure portal, and sign in by using the Microsoft account that is the Service Administrator of your Azure subscription.
8. In the Azure portal, reset the dashboard to the default state.
9. Close all open windows.

**Result**: After completing this exercise, you should have implemented two Azure web apps and a Traffic Manager profile configured to distribute requests between them.

**Question** In Exercise 2, you deployed the A. Datum production web app to Azure. In Exercise 3, you deployed a new version of the site to a staging slot. How can you tell, within Microsoft Edge, which is the production site and which is the staging site?

**Question** At the end of Exercise 4, you used an FQDN within the trafficmanager.net domain to access your web app. How can you use your own registered domain name to access this web app?

©2016 Microsoft Corporation. All rights reserved.

The text in this document is available under the [Creative Commons Attribution 3.0 License](https://creativecommons.org/licenses/by/3.0/legalcode), additional terms may apply. All other content contained in this document (including, without limitation, trademarks, logos, images, etc.) are **not** included within the Creative Commons license grant. This document does not provide you with any legal rights to any intellectual property in any Microsoft product. You may copy and use this document for your internal, reference purposes.

This document is provided "as-is." Information and views expressed in this document, including URL and other Internet Web site references, may change without notice. You bear the risk of using it. Some examples are for illustration only and are fictitious. No real association is intended or inferred. Microsoft makes no warranties, express or implied, with respect to the information provided here.