Module 3: Hosting Web Applications on the Azure Platform

Lab: Creating an ASP.NET Web App Using an Azure Web App

Scenario

You have an events administration application that is currently used by a static set of users. The application must be upgraded to handle all the users in your organization in the future. You need a hosting option that provides the least amount of friction so that you can immediately deploy the web application for immediate use. You also need the hosting option to be flexible enough so that it allows you to configure and scale the web application, thereby ensuring that it can handle an increase in the number of administrative users. For these reasons, you have chosen to deploy the application to Web Apps. Web Apps will also give you the flexibility to integrate your application with Azure Active Directory in the future so that all of your organization's users can access the application. In this lab, you will create a Web App, deploy your existing application, and then configure the Web App after deployment.

Objectives

After you complete this lab, you will be able to:

- Create a Web App.
- Create a linked resource for a Web App.
- Publish an ASP.NET web application to a Web App.
- Modify the configuration of the Web App in the Portal.

Lab Setup

Estimated Time: 60 minutes

Before starting this lab, you must complete the lab in Module 2. For the lab in this module, you will use the available host machine. Also, you must complete the following steps:

1. On the host computer, click **Start**, type **Remote**, and then click **Remote Desktop Connection**.
2. In Remote Desktop Connection, provide the name of your virtual machine in the **Computer** box by using the following format:
   - **[Your VM IP Address]:[*Your VM RDP Port*]**

> **Note:** The name and port for your virtual machine might be saved in the Computer drop-down list. If this is the case, use this value instead of typing it in manually. If you are unsure about your virtual machine's RDP port, use either of the Azure portals to find your virtual machine's endpoints. The endpoint with the name **Remote Desktop** is the correct port for RDP. This port is randomized to protect your virtual machine from unauthorized access.

3. In Remote Desktop Connection, click **Connect**. Wait until the RDP client accesses the virtual machine.

4. If necessary, sign in by using the following credentials:
   - User name: **Student**
   - Password: **AzurePa$$w0rd**
5. Verify that you received the credentials to sign in to the Azure portal from your training provider. You will use these credentials and the Azure account throughout the labs in this course.

Exercise 1: Creating an Azure Web App

Scenario

In this exercise, you will:

- Create a Web App instance in the Portal.
- Go to the URL for the new Web App.

The main tasks for this exercise are as follows:

1. Create a Web App instance using the Azure Portal.
2. Go to the newly created Web App's placeholder page.

Task 1: Create a Web App instance by using the Portal

1. Sign in to the Azure Portal (https://portal.azure.com).
2. Create a new **Web App + SQL** instance by using a unique name and the following details:

> **Note:** Be sure to note down your SQL administrator user name and password. If you lose the information, it will be very difficult and time consuming to recover that information.

- Url: **Pick a unique name for the website**
- Resource Group: **20532**
- App Service Plan Name: **SamplePlan**
- Pricing Tier: **Free**
- Database Tier: **Free**
- Region: **Select the region that is closest to your location**
- Database Name: **EventsContextDB**
- Server Admin Login/Password: **Supply your own values**

Task 2: Go to the newly created Web App's placeholder page

1. View the blade for your Web App.
2. Go to your new Web App.

Results: After completing this exercise, you will have used the Portal to create a Web App instance.

Exercise 2: Deploying an ASP.NET Web Application to an Azure Web App

Scenario

In this exercise, you will:

- Download the publish profile for an existing Web App.
- Publish a web application by using the publish wizard in Visual Studio 2017.

The main tasks for this exercise are as follows:

1. Open an existing ASP.NET web application project with Visual Studio 2017.
2. Download the publish profile for the Azure Web App.
3. Publish the ASP.NET web application to the Azure Web App.
4. Verify that the web application is successfully published.

Task 1: Open an existing ASP.NET web application project with Visual Studio 2017

1. Open the **Contoso.Events** solution from the following location:

- File location: **Allfiles (F):\Mod03\Labfiles\Starter\Contoso.Events**

2. Debug the web application to verify that it works.

Task 2: Download the publish profile for the Azure Web App

1. Sign in to the Azure Portal (https://portal.azure.com).
2. Go to your previously created Web App’s blade.
3. Download the publish profile for your Web App to the following location:

- Save location: **Allfiles (F):\Mod03\Labfiles**

Task 3: Publish the ASP.NET web application to the Azure Web App

1. Publish the ASP.NET web application by using the publish profile.

Task 4: Verify that the web application is successfully published

1. Verify that the Web App is working correctly in Azure.

Results: After completing this exercise, you will have used a publish profile to publish web applications directly to a Web App.

Exercise 3: Configuring an Azure Web App

Scenario

In this exercise, you will:

- Use the **ConfigurationManager** class to read custom app settings from a web configuration file.
- Modify custom app settings by using the Portal.

The main tasks for this exercise are as follows:

1. Implement logic to read a configuration setting from app settings.
2. Publish web application to the Azure Web App.
3. Modify the app settings in the Portal.
4. Verify that the app settings are successfully updated.

Task 1: Implement logic to read a configuration setting from app settings

1. View the **EventsListViewModel** class in the **Contoso.Events.ViewModels** project.
2. Update the **EventsListViewModel** class to use an app setting that sets the **EventCount** property in **EventsListViewModel** constructor:

- App Setting Key: **EventsListViewModel.EventCount**

3. Add an app setting to the root of the **Web.config** for the **Contoso.Events.Management** project:

- App Setting Key: **EventsListViewModel.EventCount**
- App Setting Value: **5**

4. Verify that the local web application is now using the app setting.

Task 2: Publish web application to the Azure Web App

1. Publish the ASP.NET web application by using the most recent publish profile.
2. Verify that the cloud-based web application is now using the app setting.

Task 3: Modify the app settings in the Portal

1. Sign in to the Azure Portal (*https://portal.azure.com*).
2. Go to your previously created Web App's **Application Settings** blade.
3. Add an **App Settings** entry to the Web App configuration to override the app setting in the Web.config file.

- App Setting Key: **EventsListViewModel.EventCount**
- App Setting Value: **2**

Task 4: Verify that the app settings are successfully updated

1. Verify that the web application in the cloud is now using the updated App Settings entry.

Results: After completing this exercise, you will have configured custom app settings by using a Web.config file and the Azure Portal.

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.

Version: 1.0