Azure Fundamentals:   
Module 3, Lesson 12  
Mobile Apps with Microsoft Azure App Service

## Overview

Create and configure an Azure Mobile App, upload server-side API code for database access, then build a mobile app to run and sync data to Azure.

## Objectives

In this hands-on lab you will learn how to:

* Create an Azure Mobile App
* Configure the Azure Mobile App
* Download C# server-side and Xamarin.Forms solutions
* Compile and publish the server-side C# API code to your Azure server
* Compile and run the Xamarin.Forms app
* Enter and sync data from the device to Azure

## Prerequisites

The following are required to complete this hands-on lab:

* Azure account (or free trial account), Visual Studio with Xamarin installed

## Exercises

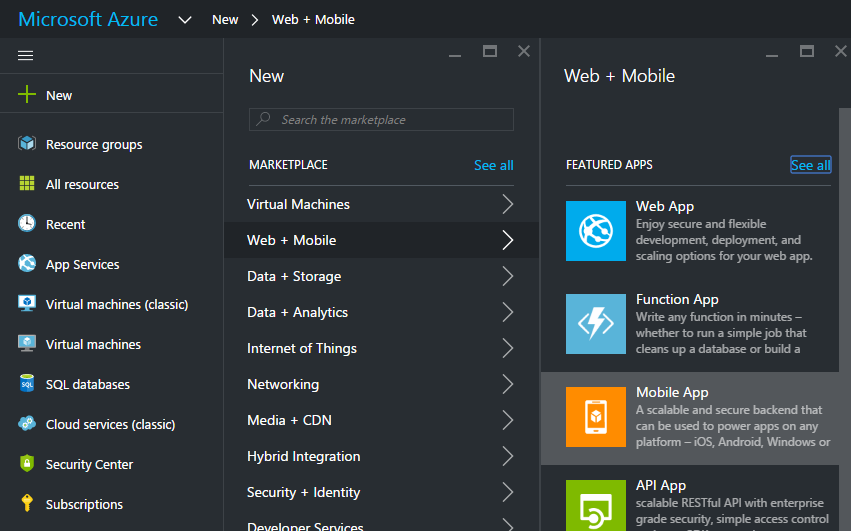
This hands-on lab includes the following exercises:

* Exercise 1: Create and configure an Azure Mobile App
* Exercise 2: Deploy the server-side solution
* Exercise 3: Run the mobile app and sync with Azure

## Exercise 1: Create and configure an Azure Mobile App

Create a .NET backend using the Azure portal.

1. Log into the [Azure Portal](https://portal.azure.com/).
2. In the top left of the window, click the **+NEW** button > **Web + Mobile** > **Mobile App**, then provide a name for your Mobile App backend.

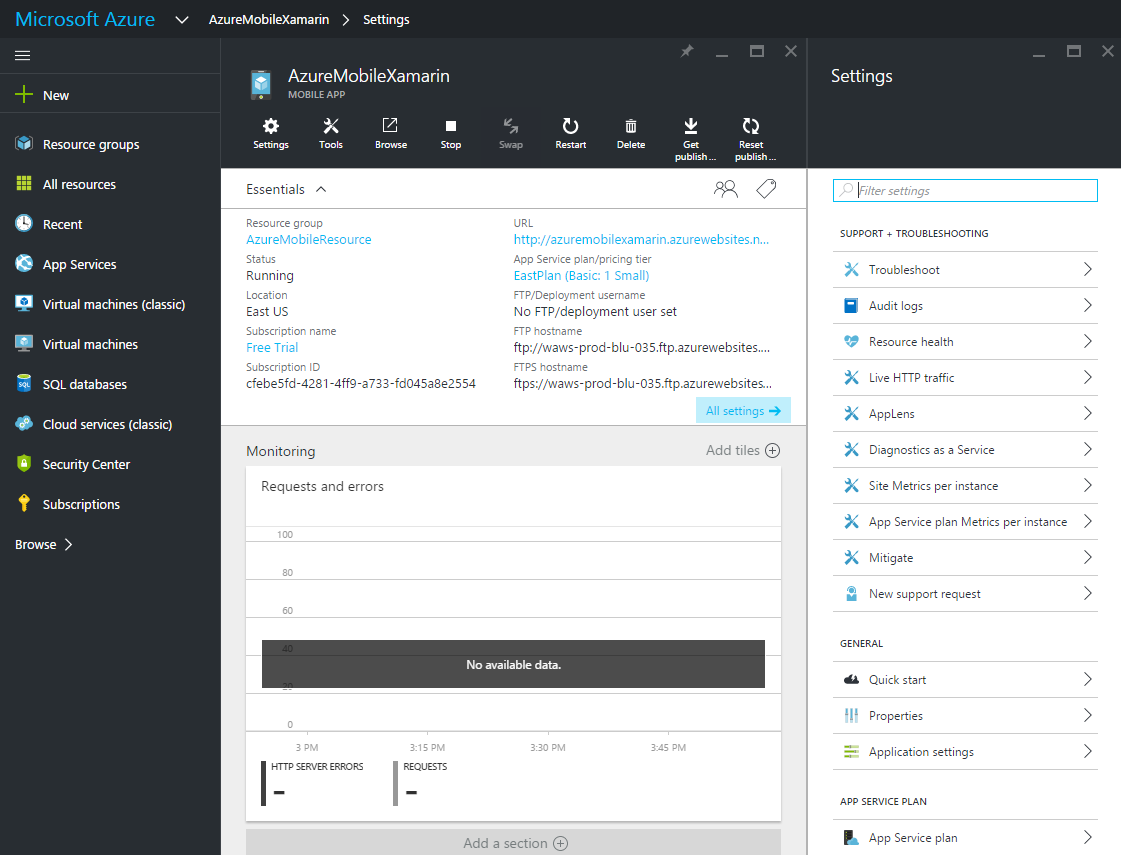


1. In the **Resource Group** box, select an existing resource group. If you have no resource groups, enter the same name as your app.

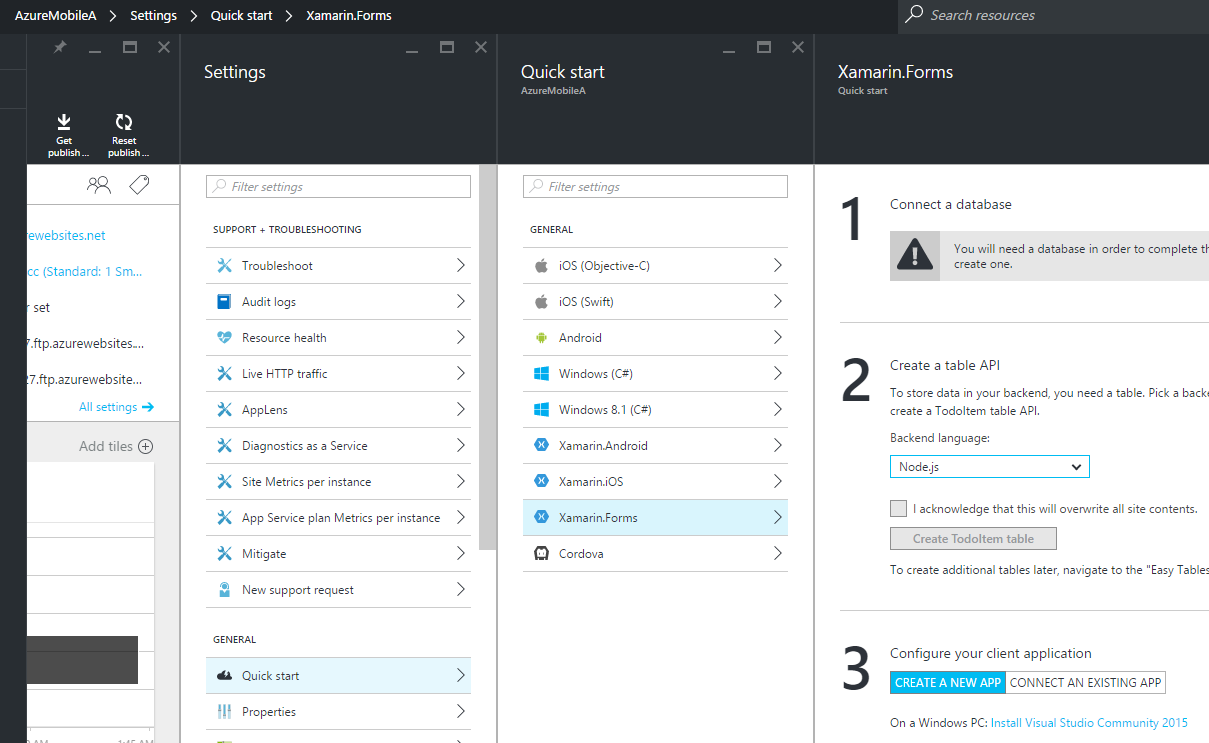
At this point, the default App Service plan is selected, which is in the Free tier. The App Service plan settings determine the location, features, cost and compute resources associated with your app. You can either select another App Service plan or create a new one.

1. Use the default App Service plan, select a different plan or create a new plan, then click **Create**.

This creates the Mobile App backend. Later you will deploy your server project to this backend. Provisioning a Mobile App backend can take several minutes; the **Settings** blade for the Mobile App backend is displayed when complete.



1. Define a connection to a data store. In the **Settings** blade for the new Mobile App backend, click **Quick start** > **Xamarin.Forms**(or whatever your platform is) > **1 Connect a database**.



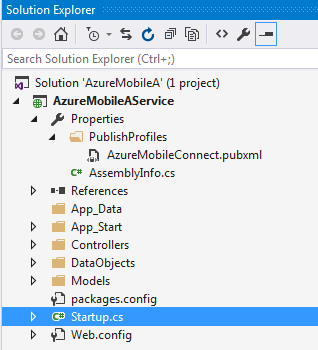
1. In the **Add data connection** blade, click **SQL Database** > **Create a new database**, type the database **Name**, choose a pricing tier (**F - Free**), then click **Server**.

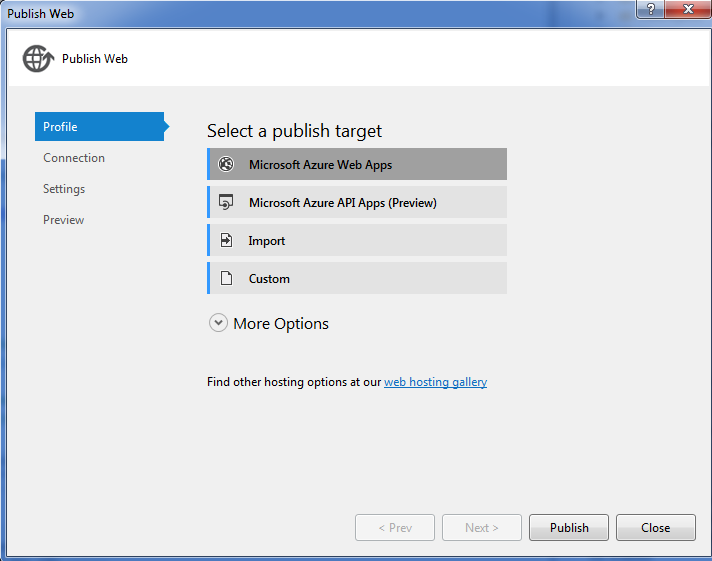
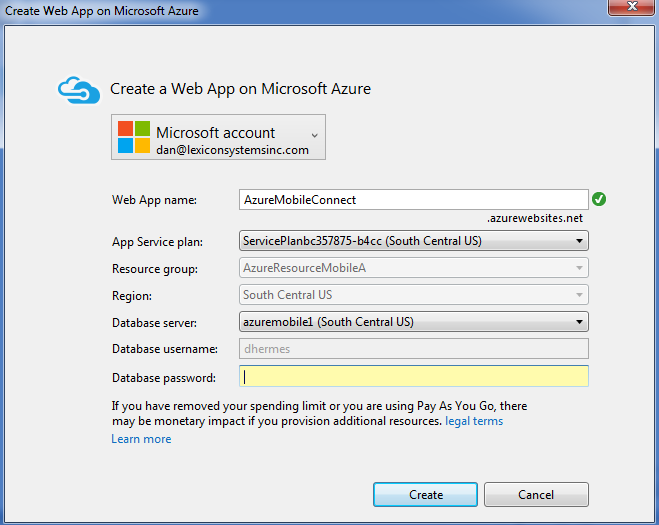
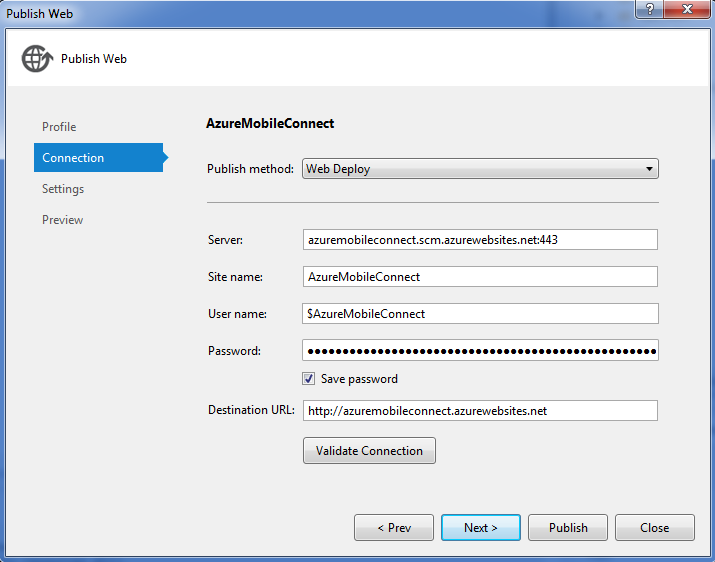
Note: You can reuse this new database and administer it as you would any other SQL Database instance. If you already have a database in the same location as the new mobile app backend, you can instead choose **Use an existing database** and then select that database. The use of a database in a different location is not recommended because of additional bandwidth costs and higher latencies. Other data storage options are available.

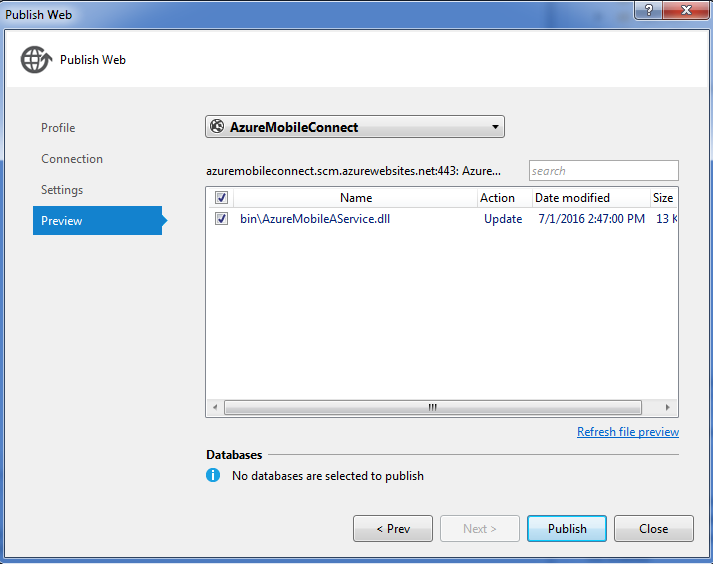
1. In the **New server** blade, type a unique server name in the **Server name** field, provide a secure **Server admin login** and **Password**, make sure that **Allow azure services to access server** is checked, then click **OK** twice. This creates the new database and server.
2. Back in the **Add data connection** blade, click **Connection string**, check that the login and password values are present for your database, then click **OK** twice.   
     
   Creation of the database can take a few minutes. Use the **Notifications** area to monitor the progress of the deployment. You cannot continue until the database has been deployed successfully.

## Exercise 2: Deploy the server-side solution

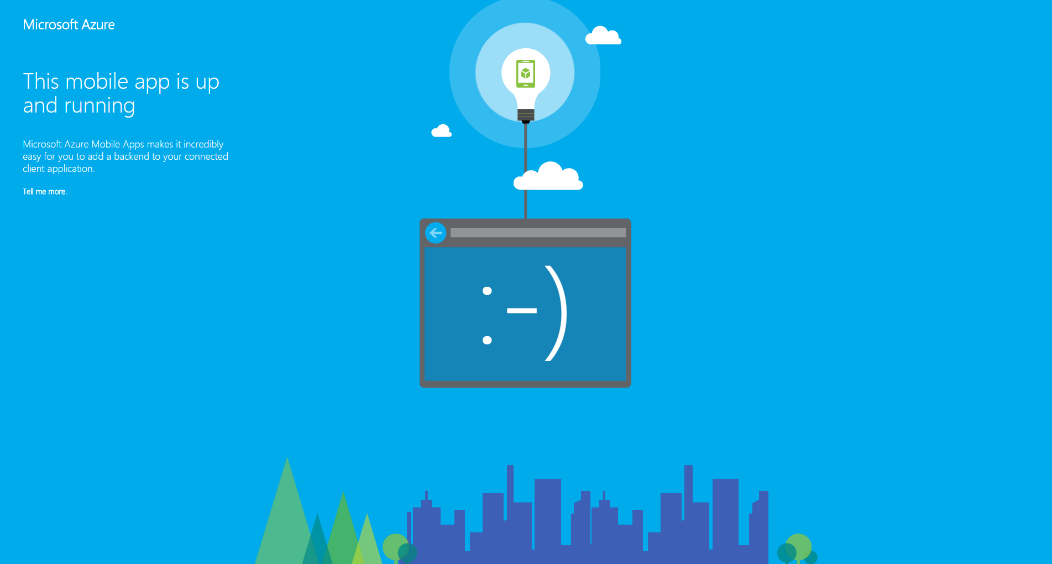
1. In the **Quick Start > Xamarin.Forms**(or whatever your platform is) blade, under **Create a table API**, select **C#** as your **Backend language**.
2. Click **Download**, extract the compressed project files to your local computer, open the solution in Visual Studio. Note the startup.cs file, and the Models, DataObjects, and Controllers folders.



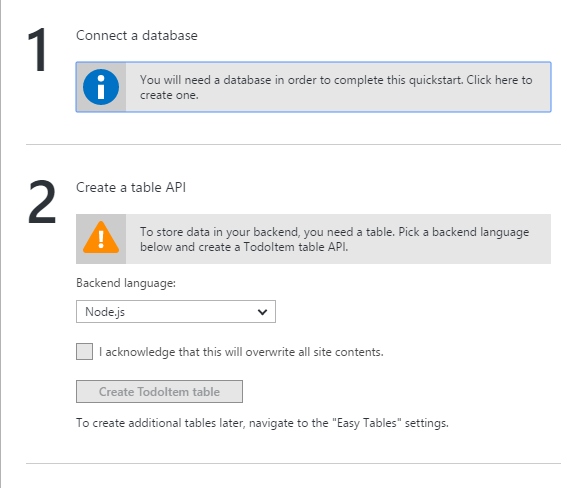
1. Build the project to restore the NuGet packages, then deploy the project to Azure. Here’s how:
2. Right-click the project and select **Publish**.
3. In the Publish Web wizard, select **Microsoft Web Apps** and click **Publish**. Azure Web Apps contain Mobile App functionality.  
     
   
4. Configure the Web App by providing a name, service plan, and database. Use the login from your Azure SQL Database defined earlier.  
     
   
5. Create an Azure site name, user name, and password.  
     
   
6. Preview the DLLs and files to publish to the server and click Publish.



1. You Mobile App backend is now ready to use with your client app.



1. Finally, add the ToDoList table to your SQL Database. Back in the Azure portal, in the **Quick Start > Xamarin.Forms**(or whatever your platform is) blade, under **Create a table API**, select **Node.js** as your **Backend language**.

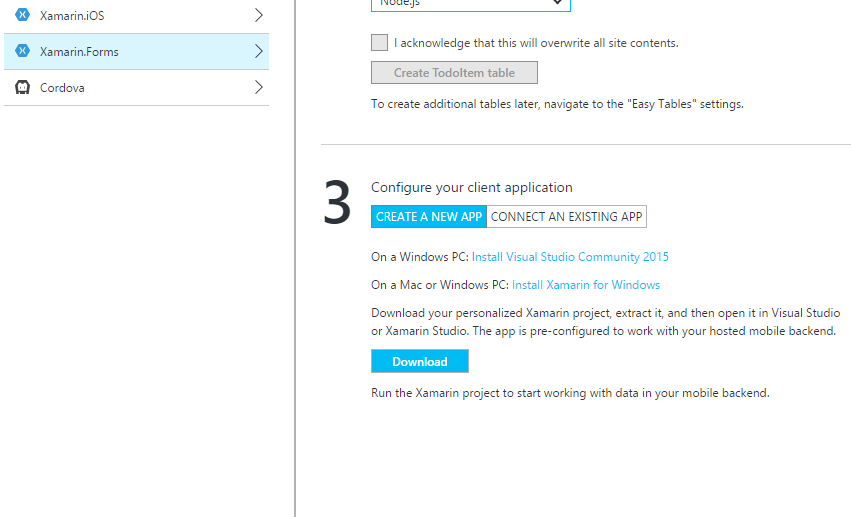


1. Check the box “I acknowledge that this will overwrite all site contents”. This will enable the button “Create TodoItem table”. Click the button and wait until the table is created.

## Exercise 3: Run the mobile app and sync with Azure

Once you have configured your Mobile App backend, you can either create a new client app or modify an existing app to connect to Azure. In this section, you download a Xamarin.Forms app that is customized to connect to your Mobile App backend.

1. Back in the **Quick Start** blade for your Mobile App backend, click **Create a new app** > **Download**, then extract the compressed project files to your local computer.



1. With the Xamarin.Forms platform of your choice (such as Android) app as the startup project, press the F5 key to rebuild the project and start the mobile app.
2. In the app, type meaningful text, such as Complete the lab, in the **Insert a TodoItem** text box, and then click **the plus sign to add.**
3. Your item is added to the list (and your Azure SQL Database) and the list updated.

## 

## Summary

In this hands-on lab, you learned how to:

* Create an Azure Mobile App
* Configure the Azure Mobile App
* Download C# server-side and Xamarin.Forms solutions
* Compile and publish the server-side C# API code to your Azure server
* Compile and run the Xamarin.Forms app
* Enter and sync data from the device to Azure

For more information about [lab topic(s)] please check out the following resources:

* [Create a Xamarin.Forms app Using Azure](https://azure.microsoft.com/en-us/documentation/articles/app-service-mobile-xamarin-forms-get-started/)
* [Creating a Web App and a Mobile App Using Azure](https://github.com/Azure-Readiness/DevCamp/tree/master/Presentation/App-Service)
* [How to Publish the Server Project](https://azure.microsoft.com/en-us/documentation/articles/app-service-mobile-dotnet-backend-how-to-use-server-sdk/#publish-server-project)