**Steps to create and deploy a Machine Learning Web service using Machine Learning Studio**

* Create your first experiment
* Deploy Machine Learning Web Service

Once you deploy an Azure Machine Learning Studio predictive model as a Web service, you can use a REST API to send it data and get predictions. You can send the data in real-time or in batch mode.

Azure Machine Learning Studio has two types of services:

* Request-Response Service (RRS) – A low latency, highly scalable service that provides an interface to the stateless models created and deployed from the Machine Learning Studio.
* Batch Execution Service (BES) – An asynchronous service that scores a batch for data records.

**Azure ML Web app templates:**

The web app templates available in the Azure Marketplace can build a custom web app that knows your web service's input data and expected results. All you need to do is give the web app access to your web service and data, and the template does the rest.

Two templates are available:

* [Azure ML Request-Response Service Web App Template](https://azure.microsoft.com/marketplace/partners/microsoft/azuremlaspnettemplateforrrs/)
* [Azure ML Batch Execution Service Web App Template](https://azure.microsoft.com/marketplace/partners/microsoft/azuremlbeswebapptemplate/)

Each template creates a sample ASP.NET application, using the API URI and Key for your web service, and deploys it as a web site to Azure. The Request-Response Service (RRS) template creates a web app that allows you to send a single row of data to the web service to get a single result. The Batch Execution Service (BES) template creates a web app that allows you to send many rows of data to get multiple results.

No coding is required to use these templates. You just supply the API URI and Key and the template builds the application for you.

**How to use Request Response Service Template:**

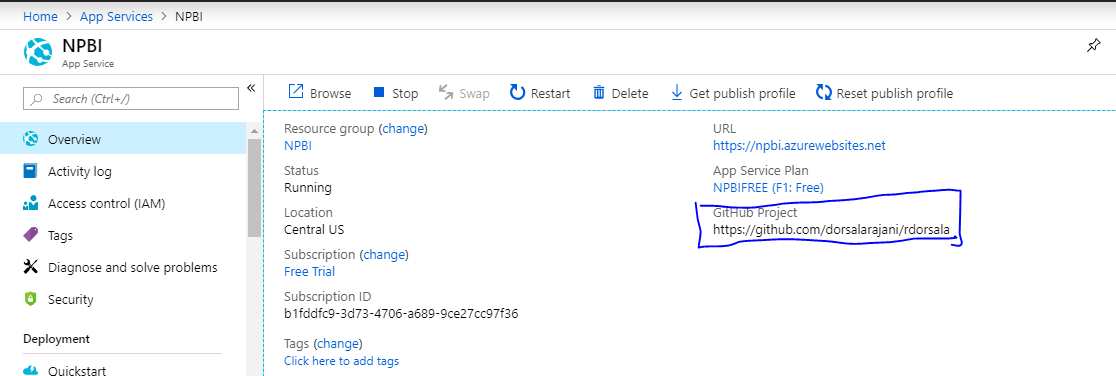
Below zip file contains the template source code for AzureMLRSS.



1. In Machine Learning Studio, open the **Web Services** tab and then open the web service you want to access. Copy the key listed under **API key** and save it.
2. Open the **REQUEST/RESPONSE** API Help Page. At the top of the help page, under **Request**, copy the **Request URI** value and save it. This value will look like this:
3. Go to the [Azure portal](https://portal.azure.com/), **Login**, click **New**, Search for and select **Azure ML Request-Response Service Web App**, then click **Create**.

* Give your web app a unique name. The URL of the web app will be this name followed by .azurewebsites.net. For example, http://NPBI.azurewebsites.net.
* Select the Azure subscription and services under which your web service is running.
* Click **Create**.

Once you successfully create the App Service, open the App overview there you will see the highlighted link. It’s a deployment center .



We have to configure the deployment center manually in azure portal. Please follow the steps in below URL to configure deployment center.

<https://docs.microsoft.com/en-us/azure/app-service/deploy-continuous-deployment>

In my practice, I have selected deployment center as “From GitHub”. Before doing this, I have Signup for GitHub account and there created a public repository and uploaded AzureMLRRSTemplate folder in that repository.

While setting up deployment center, I have selected From GitHub , there I have given repository RRS folder URL and Branch name.

1. When Azure has finished deploying the web app, click the **URL** on the web app settings page in Azure, or enter the URL in a web browser. For example, http://NPBI.azurewebsites.net.
2. When the web app first runs it will ask you for the **API Post URL** and **API Key**. Enter the values you saved earlier:
   * **Request URI** from the API Help Page for **API Post URL**
   * **API Key** from the web service dashboard for the **API Key**.

Click **Submit**.

1. The web app displays its **Web App Configuration** page with the current web service settings. Here you can make changes to the settings used by the web app.
2. When you're done, click **Save changes**, and then click **Go to Home Page**.

From the home page you can enter values to send to your web service, click **Submit**, and the result will be returned.

**Note: We have to explore various deployment center configuration techniques before creating App Service.**