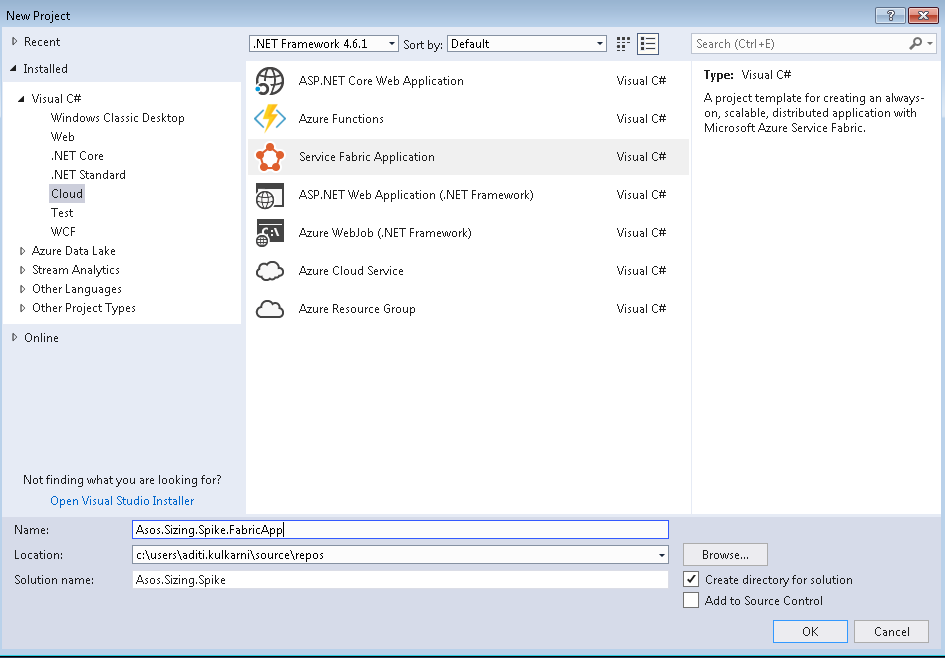
# Setup system

1. Install service fabric SDK from following path

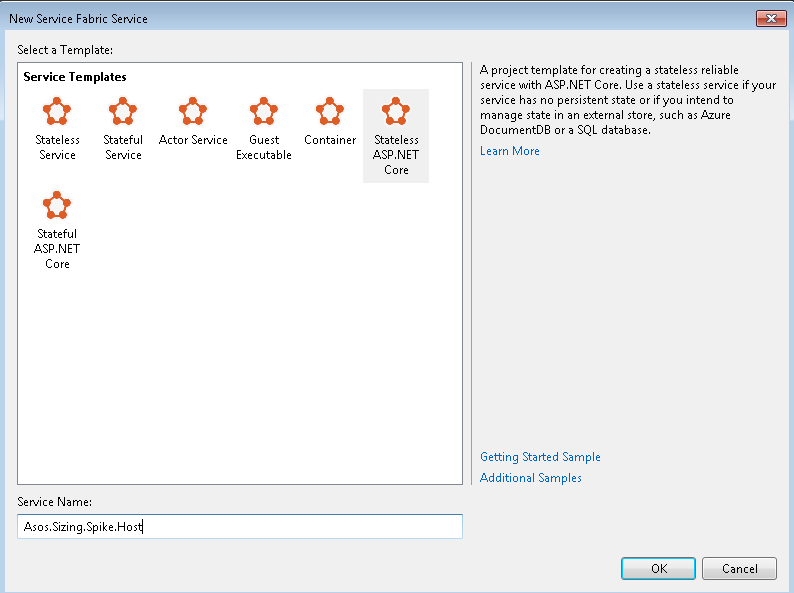
<https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-get-started>

# Steps to create service fabric application

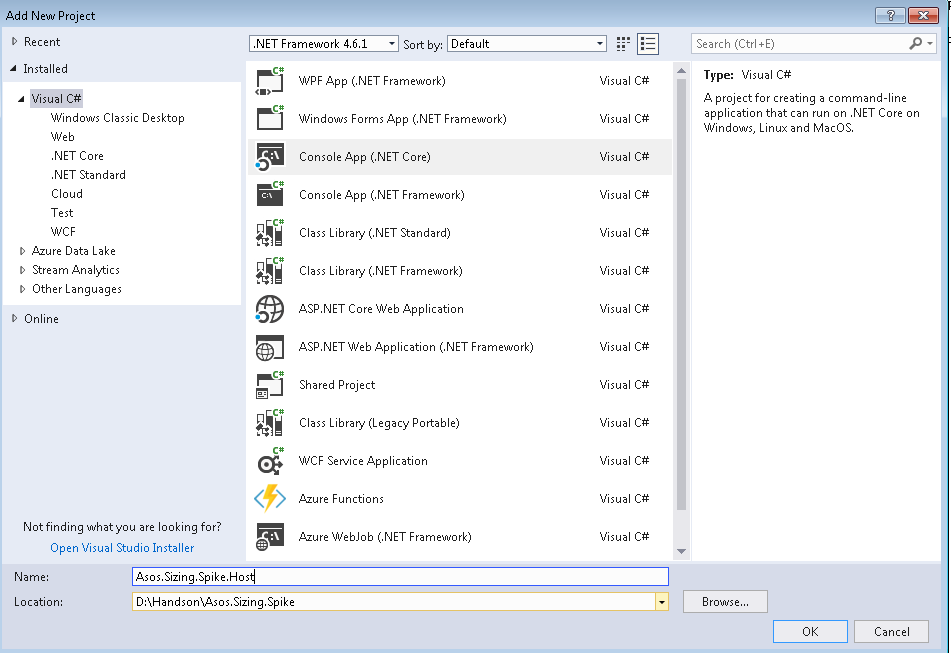
1. Add new project, select Visual C# - > Cloud -> Service Fabric Application



1. Select service template



1. Add console app project in this solution.

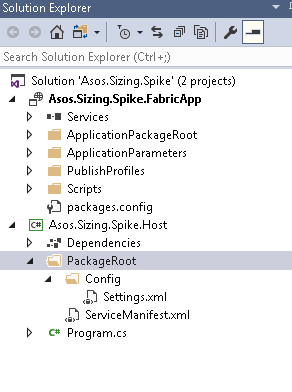


1. Create a new folder in console app, named it as ‘PackageRoot’. Add following files and folders in it.

i. XML file having name ‘ServiceManifes.xml’.

ii. Config folder

iii. XML file inside PackageRoot -> Config -> Settings.xml



1. Add following sections in ‘ServiceManifest.xml’.

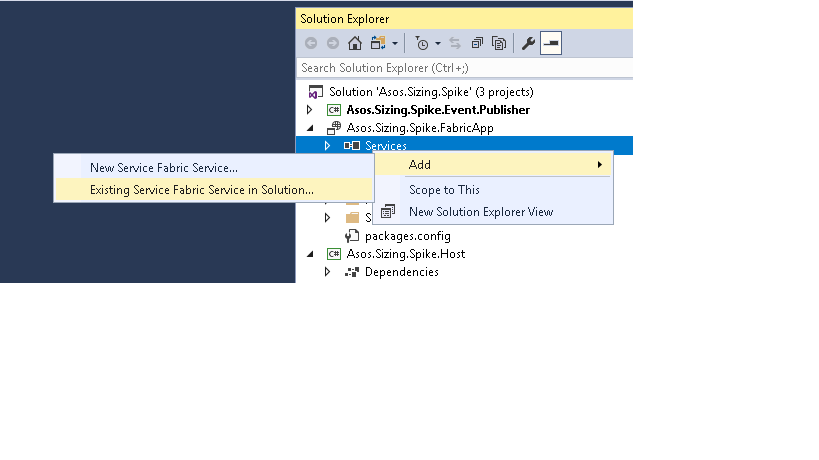
|  |
| --- |
| <?xml version="1.0" encoding="utf-8"?>  <ServiceManifest Name="SizingSpike.HostPkg"  Version="1.0.0"  xmlns="http://schemas.microsoft.com/2011/01/fabric"  xmlns:xsd="http://www.w3.org/2001/XMLSchema"  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">  <ServiceTypes>  <!-- This is the name of your ServiceType.  This name must match the string used in RegisterServiceType call in Program.cs. -->  <StatelessServiceType ServiceTypeName="SizingSpike" UseImplicitHost="true" />  </ServiceTypes>  <!-- Code package is your service executable. -->  <CodePackage Name="Code" Version="1.0.0">  <EntryPoint>  <ExeHost>  <Program>Asos.Sizing.Spike.Host.exe</Program>  <WorkingFolder>CodePackage</WorkingFolder>  </ExeHost>  </EntryPoint>  </CodePackage>  <!-- Config package is the contents of the Config directoy under PackageRoot that contains an  independently-updateable and versioned set of custom configuration settings for your service. -->  <ConfigPackage Name="Config" Version="1.0.0" />  <Resources>  <Endpoints>  <!-- This endpoint is used by the communication listener to obtain the port on which to  listen. Please note that if your service is partitioned, this port is shared with  replicas of different partitions that are placed in your code. -->  <Endpoint Protocol="http" Name="ServiceEndpoint" Type="Input" Port="8688" />  </Endpoints>  </Resources>  </ServiceManifest> |

Update highlighted values with appropriate values as per your project names.

1. Edit PropertyGroup section from consoleApp.csproj file with following settings

|  |
| --- |
| <PropertyGroup>  <RuntimeIdentifier>win-x64</RuntimeIdentifier>  <TargetFramework>netcoreapp2.0</TargetFramework>  <AssemblyName>Asos.Sizing.Spike.Host</AssemblyName>  <RootNamespace>Asos.Sizing.Spike.Host</RootNamespace>  </PropertyGroup> |

1. Right click on services present in ServiceFabricApplication and remove existing service from it, which we have added through template in step 2.



1. Add ConsoleApp in ServiceFabricApplication. Right click on ServiceFabricApplication -> Services. Select ‘Existing Service Fabric Service in Solution’ option.
2. Build solution.
3. Go to ApplicationPackageRoot -> ApplicationManifest.xml. And make sure that highlighted values are matched with values from ‘ServiceManifest.xml’ present in ConsoleApp, refer step 5.

|  |
| --- |
| <?xml version="1.0" encoding="utf-8"?>  <ApplicationManifest xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" ApplicationTypeName="Asos.Sizing.Spike.FabricAppType" ApplicationTypeVersion="1.0.0" xmlns="http://schemas.microsoft.com/2011/01/fabric">  <Parameters>  <Parameter Name="Asos.Sizing.Spike.Host\_InstanceCount" DefaultValue="-1" />  </Parameters>  <!-- Import the ServiceManifest from the ServicePackage. The ServiceManifestName and ServiceManifestVersion  should match the Name and Version attributes of the ServiceManifest element defined in the  ServiceManifest.xml file. -->  <ServiceManifestImport>  <ServiceManifestRef ServiceManifestName="SizingSpike.HostPkg" ServiceManifestVersion="1.0.0" />  <ConfigOverrides />  </ServiceManifestImport>  <DefaultServices>  <!-- The section below creates instances of service types, when an instance of this  application type is created. You can also create one or more instances of service type using the  ServiceFabric PowerShell module.    The attribute ServiceTypeName below must match the name defined in the imported ServiceManifest.xml file. -->  <Service Name="SizingSpike">  <StatelessService ServiceTypeName="SizingSpike" InstanceCount="[Asos.Sizing.Spike.Host\_InstanceCount]">  <SingletonPartition />  </StatelessService>  </Service>  </DefaultServices>  </ApplicationManifest> |

# Create Azure Service Bus Cluster

1. Go to <https://portal.azure.com/#create/Microsoft.ServiceFabricCluster>
2. In basic blade provide basic details of your cluster

Refer following link to create service fabric cluster on azure portal-

<https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-cluster-creation-via-portal>

Refer following link for service fabric naming conventions

<https://asoscom.atlassian.net/wiki/spaces/PE/pages/23268605/Service+Fabric+Naming+Standards>

1. In the Basics blade you need to provide the basic details for your cluster.

Enter the name of your cluster.

Enter a user name and password for Remote Desktop for the VMs.

Make sure to select the Subscription that you want your cluster to be deployed to, especially if you have multiple subscriptions.

Create a new resource group. It is best to give it the same name as the cluster, since it helps in finding them later, especially when you are trying to make changes to your deployment or delete your cluster.

Note

Although you can decide to use an existing resource group, it is a good practice to create a new resource group. This makes it easy to delete clusters that you do not need.

Select the region in which you want to create the cluster. You must use the same region that your Key Vault is in.

1. Cluster configuration blade

# Manual deployment

1. Publish
2. Publish setting