# Microsoft Dynamics CRM SDK

## Persistent Queue Listener Readme

## This folder contains sample code demonstrating how to write a listener application that reads messages from a Windows Azure persistent queue. These messages are sent to the queue from Microsoft Dynamics CRM 2013.

## Instructions

## Persistent Queue Listener Project

## Configure Windows Azure and Microsoft Dynamics CRM Integration

1. Navigate to <http://www.microsoft.com/windowsazure/> and follow the steps to sign up for a Windows Azure account. Setting up an account requires a credit card but there may be trial accounts available which at the time of this writing do not charge your credit card. Please read any agreements entered into carefully as the details may have changed.
2. Configure Windows Azure ACS for Microsoft Dynamics CRM integration by following the instructions in the SDK topic [Configure Windows Azure Integration with Microsoft Dynamics CRM](http://msdn.microsoft.com/en-us/library/gg309340.aspx).

## Download Windows Azure Service Bus Issuer Certificate from the Microsoft Dynamics CRM Server

1. On a machine with the Windows Azure SDK installed, launch the Microsoft Dynamics CRM web application.
2. Navigate to **Settings**, and then click **Customizations**.
3. Select **Developer Resources**.
4. Click **Download Certificate** under **Windows Azure Bus Issuer Certificate**. If there is no certificate, one must be [configured](http://msdn.microsoft.com/en-us/library/gg309340.aspx#bkmk_configurecrm) before proceeding.
5. Note the issuer name of the certificate mentioned above the **Download Certificate** link for later use while configuring Windows Azure using the Plug-in Registration tool.
6. Import the downloaded certificate into the local certificate store by following these steps.
   1. Run the Certificate snap-in of the MMC (Microsoft Management Console). Open the **Start Menu**, select **Run**, enter *certmgr.msc,* and press <Enter>.
   2. Expand the **Certificates** tree node.
   3. Right click **Trusted Root Certification Authorities**, select **All Tasks**, and then select **Import**.
   4. Follow the **Certificate Import Wizard** dialog steps to import the certificate.
7. Export the certificate you just imported using base-64 encoding by following these next steps.
   1. Under the **Trusted Root Certification Authorities** node, select the **Certificates** node.
   2. Right-click the certificate in the list that you just imported.
   3. In the context menu, select **All Tasks**, and then select **Export**.
   4. Follow the **Certificate Export Wizard** dialog to export a base-64 encoded file with a .cer file name extension to a machine with the Windows Azure SDK installed. This certificate file is for later use while configuring Windows Azure using the Plug-in Registration tool.

## Configuring Microsoft Dynamics CRM to post to Windows Azure on creation of a letter.

## The following procedure is similar to the walkthrough in the SDK documentation, except that some settings and values are specific to this sample. For more information, see [Walkthrough: Register an Azure-Aware Plug-in with Plug-in Registration Tool](http://msdn.microsoft.com/en-us/library/gg328524.aspx).

1. Using the Plug-in Registration tool located in the Bin folder of the SDK download, connect to the Microsoft Dynamics CRM organization which will post to the Windows Azure endpoint.
2. Select the **Register** button, and then from the dropdown list select **Register New Service Endpoint**.
3. In the **Name** field enter a meaningful identifier such as ‘PersistentQueueListener’.
4. In the **Description** field enter ‘Persistent Queue sample’.
5. In the **Solution Namespace** field enter the Windows Azure service namespace.
6. In the **Path** field enter the path of the persistent queue. This is defined in the sample code as ‘MyQueue’.
7. In the **Contract** dropdown select **PersistentQueue**.
8. Ensure the **Claim** dropdown is set to **None**.
9. If you have never configured the service endpoint for the queue, complete these next steps. Otherwise, continue with step10.

One time setup to configure Windows Azure through the Plug-in Registration tool for each new queue:

* 1. Select the **Save & Configure ACS** button.
  2. On the **ACS Configuration** screen, enter the Management Key, Certificate File, and Issuer Name.
  3. Select the **Configure ACS** button and then select the **Yes** button when prompted to confirm. The screen should display configuration setup completion without any error.
  4. Select the **Close** button to close the window.
  5. On the **Service Endpoint Details** screen, select the **Save & Verify Authentication** button to verify the configuration. A window should open and display **Verify Authentication: Success**.
  6. Select the **Close** button to close the window.

1. Select the **Save** button to save the endpoint configuration.
2. Under the **Registered Plugins & Custom Workflow Activities** list, right click the **(ServiceEndpoint) PersistentQueueListener** (the identifier from step #3) node.
3. Select **Register New Step**.
4. In the **Message** field enter *Create*.
5. In the **Primary Entity** field enter *letter*.
6. Select **Register New Step**.
7. After the Plug-in Registration tool completes the step registration successfully, close the tool.

## How to Install, Build, and Run the Sample

1. In Windows Explorer, open the **WindowsAzureVB.sln** solution.
2. Right-click the **PersistentQueueListener** project.
3. Select the **Set as StartUp Project** menu option.
4. Compile and run the project by pressing F5.
5. At the prompt, enter appropriate information (as shown in the example screenshot below) to start the listener.
6. After the console application displays: **Press [Enter] to retrieve a message from the queue (type quit to exit)**, create and save a new letter activity in Microsoft Dynamics CRM.
7. Press <Enter> in the console window to view the message posted from Microsoft Dynamics CRM. There may be a delay of seconds up to a few minutes before the Microsoft Dynamics CRM asynchronous service posts the message to the Windows Azure Service Bus, so be patient.

**Expected Results**

You should see:  


## Troubleshooting

## Build error(s)

## Ensure the proper references are added and the project is targeting the .NET Framework 4.0 and not the .NET Framework 4 Client Profile.

## Run-time errors

## If project hangs while the host is opening check if Fiddler or some other proxy is preventing the sample from working.

## If everything seems to run fine but upon creating a new letter in Microsoft Dynamics CRM nothing shows up in the console window, follow these steps.

## Open the Microsoft Dynamics CRM web application.

## Navigate to Settings, then to System Jobs.

## Look for a job regarding the created letter with ‘Waiting’ for the Status Reason.

## Open that job and view details to get troubleshooting information.

## If you are experiencing errors when running the sample, check the following.

## You have followed all instructions in this document correctly.

## The server, organization, and logon credentials you entered are correct.

## Your account on the Microsoft Dynamics CRM server has the required privileges to perform the sample’s intended operation.

## Important Notes

* The sample files should not be used in a production environment without prior testing. You should deploy an application that uses this sample code to a test environment and examine it for interaction or interference with other parts of the system.
* Before you deploy applications that use this sample code to a production environment, make sure that you consider the existing customizations that you may have implemented in Microsoft Dynamics CRM.

## Copyright

Microsoft may have patents, patent applications, trademarks, copyrights, or other intellectual property rights covering subject matter in this document. Except as expressly provided in any written license agreement from Microsoft, the furnishing of this document does not give you any license to these patents, trademarks, copyrights, or other intellectual property.

© 2013 Microsoft Corporation. All rights reserved.

Microsoft, Microsoft Dynamics, Active Directory, Outlook, Visual Studio, Windows, Windows Server, and Windows Vista are trademarks of the Microsoft group of companies.

All other trademarks are property of their respective owners.