

# VA 2.3 - 02 Deployment Microservice

Created by Bui, Thoan, last modified by Nguyen Thi Bao Quyen on Sep 05, 2019

- Step 0: Microservcie Prerequisites
- Step 1: Create Provision and Create Release
  - 1.1 Create Provision
    - 1.1.1 Create Provision from existing template of Delivery Kit subscription
    - 1.1.2 Create owner template for others subscription
  - 1.2 Create release
  - 1.3 Verify resource group, application insights was created on Azure
- Step 2: Published API Management on Azure
  - 2.1 Published API for Skype
    - Create 'API Management Service'
    - Create OpenAPI.json template file
    - Add API for Skype
  - 2.2 Add API for MS Chat
  - 2.3 Add API for MS Team
- Step 3: Microservice Capabilities
  - 3.1 Create New Folder
  - 3.2 Import release pipeline
  - 3.3 Enter information for all Microservice
- Step 4: Add variables
  - MS Chat
  - S4B
  - MS TEAM
  - WEBCHAT
- Step 5: Link variables to pipeline
- Step 6: Deploy
  - 6.1 Deploy
  - 6.2 Configuration for MS Team on Azure
    - Create Register Channel for MS Team
    - Update message endpoint on Setting
    - Add new channel
    - Update value for variable for BOT\_ENTRIES on DEV Tool and build again
  - 6.3 Configuration for Web Chat on Azure
    - Update Policy on MS Chat so that can support Web Chat
    - Configuration Authorization
    - Add User & Group to Webchat
    - Register Webchat to App Register
- Step 7: Testing
  - Skype for Business
  - MS Team
  - WebChat
- Step 8: Translation Microservice
  - 8.1 Configuration Service Now
  - 8.2 Using existing Translation Microservice
  - 8.3 Deploy new Translation Microservice
    - Create provision
    - Check Resource Group, Translation MicroService and App Insights was created on Portal Azure
    - Check App Insights information
    - Create API Management
    - Update VA Core and rebuild VA Core
    - Create Capacity
    - Add Variables
    - Link variables
    - Deployment
  - 8.4 Verify Translation on Skype, MS Team, WebChat....
- Step 9: Custom Build
  - Create custom build
  - Link to Capability
  - Follow the same step for Skype, MS Team, Translation, WebChat...
- Step 10: Trouble Shoot

## Step 0: Microservcie Prerequisites

1. VA 2.3 - 00 Prerequisites - Configuration Standard
2. Follow the process to request permission on Azure DevOp  
<https://confluence.csc.com/display/PD/VA+2.x+Delivery+Azure+Dev+Op+permissions+Request>
3. Build and release permissions and security roles please follow all permission before starting. Otherwise, will take much time
4. With version 2.3. We need to create account on Service Now are the same account of Skype and Portal Azure (Example: Skype account is tbui21@dxc.com → on service now must have tbui21@dxc.com. Portal Azure account is tbui21@csc.com, make sure on service now exist a account tbui21@csc.com)

## Step 1: Create Provision and Create Release

### 1.1 Create Provision

If you want to deploy Microservices on "DXC WM Production" subscription, please follow step 1.1.1

If not (you want to deploy Microservices on others subscription), you need create a owner provision, please follow step 1.1.2

#### 1.1.1 Create Provision from existing template of Delivery Kit subscription

If you want to deploy Microservices on Delivery Kit subscription with the name "**DXC WM Production**". You can use existing template to create provision

The screenshot shows the Azure DevOps interface for the 'wm-microservice' project. On the left, there's a sidebar with 'My organizations' and a 'wm-microservice' card. The main area shows a grid of projects: 'Provision' (highlighted with a red box), 'Microservices Capabilities', 'All projects' (with 'MC' icon), 'Microservices Capabilities' (under 'All projects'), and 'ms-botframework'. Below the grid, there are sections for 'What's new', 'Sprint 154 release notes', and 'Organization settings'.

If you want deploy on Dev environment, please select template "2.3 Template Provision - Dev Environment"

If you want deploy on Testing environment, please select template "2.3 Template Provision - Testing Environment"

**After select template, please move to Step 1.2 to Create release**

The screenshot shows the Azure DevOps interface for the 'wm-microservice' project, specifically the 'Pipelines' section. On the left, there's a sidebar with various pipeline icons. The main area shows a list of pipelines under 'All pipelines': 'Delivery', 'Version\_2\_3', '2.3 Provision', '2.3 Template Provision - Dev Environment' (highlighted with a red box), '2.3 Template Provision - Testing Environment', 'Customer Development Provision', 'Customer Staging Provision', 'Delivery Kit Provision', 'Demo', 'Framework', and 'Development Provision'. To the right, there's a detailed view of the '2.3 Template Provision - Dev Environment' pipeline, showing 'Releases', 'Deployments', and 'Analytics' tabs, and a list of releases including 'Release-2.3-1'.

### 1.1.2 Create owner template for others subscription

If you want to use other subscriptions (**not DXC WM Production**), you must create owner template by following below steps

Access <https://dev.azure.com/wm-microservice/>

The screenshot shows the Azure DevOps dashboard for the 'wm-microservice' project. On the left, there's a sidebar with 'My organizations', 'Announcements from the Microsoft Build Conference', '+ New organization', and 'Organization settings'. The main area displays several project cards: 'Provision' (highlighted with a red box), 'Microservices Capabilities', 'All projects', 'Microservices Capabilities', and 'ms-botframework'. Below these cards, there's a section for 'Build' announcements.

You can see the template Provision as image. However, this template is set up to Build Team subscription. You need clone and create owner provision and set correct to your subscription.

The owner template provision that I create will be used for all customers who have the same subscription.

Select template provision and Clone

The screenshot shows the 'wm-microservice / Provision / Pipelines / Releases' page. The left sidebar has 'Pipelines' selected. In the main area, it shows a list of releases under the '2.3 Provision' pipeline. A context menu is open over the '2.3 Provision' release, with the 'Clone' option highlighted by a red box.

Change the name and select TASKS

The screenshot shows the 'wm-microservice / Provision / Pipelines / Releases' page with the '2.3 Provision - TEST' pipeline selected. The left sidebar has 'Pipelines' selected. In the main area, the 'Tasks' tab is highlighted by a red box. The pipeline stages are listed: 'MS Chat' (Deployment process) and 'Agent job' (Run on agent). The 'Stage name' field contains 'MS Chat'.

Select each Microservice and change subscription

The screenshot shows the Azure DevOps Pipelines interface for a release definition named '2.3 Provision - TEST'. On the left, the 'Provision' section is selected. In the main area, there is a 'MS Chat' stage under a 'Deployment process'. The stage name is 'MS Chat'. Below the stage, a list of tasks is shown, with 'MS Chat' highlighted and surrounded by a red box.

#### Change subscription to your subscription

The screenshot shows the same pipeline configuration as above, but focusing on the 'Agent job' task. The task is labeled 'Azure Deployment:Create Or Update Reso...' and is highlighted with a red box. In the 'Azure Details' section, the 'Azure subscription' dropdown is also highlighted with a red box, showing the value 'DXC Workplace and Mobility Dev (279adfe2-476c-490c-...)'.

The same for others Microservice such as Skype, MS Team, Webchat....

This screenshot shows the configuration for the 'MS Skype For Business' stage. The 'Agent job' task is highlighted with a red box. The 'Azure Details' section shows the 'Azure subscription' dropdown again, with the value 'DXC Workplace and Mobility Dev (279adfe2-476c-490c-...)' highlighted by a red box.

After change subscription for all Microservice , select "Variables" tab and change Subscription

The screenshot shows the 'Variables' tab in the Azure DevOps Pipelines interface. A red box highlights the 'SubscriptionId' variable, which has a value of '279adfe2-476c-490c-a1cf-be00be890369'. This indicates that the pipeline is configured to use a specific subscription.

After change all subscription, need Save template provision

The screenshot shows the 'Variables' tab in the Azure DevOps Pipelines interface. A red box highlights the 'Save' button, indicating that the changes made to the variables need to be saved.

After you create owner template provision, please select it and move to step 1.2

## 1.2 Create release

The template provision is used for all customers that is the same subscription. After created owner template provision, we will create release for each customer.

The screenshot shows the 'Releases' tab in the Azure DevOps Releases interface. A red box highlights the 'Create release' button, which is used to initiate the creation of a new release for the selected provision.

9/29/2019

## VA 2.3 - 02 Deployment Microservice - MyWorkStyle AI Agent - Confluence

For default, the template will create provision for all Microservices : MS Chat, MS Skype, MS Bot, MS Webchat, MS Handover, MS Notification. If you need deploy all, please keep it.

Otherwise, we can exclude Microservice by select on "Stages for a trigger changes from automated to manual"

The screenshot shows the 'Create a new release' dialog for '2.3 Provision'. A dependency graph is displayed, with 'Shared Resource' at the top, followed by 'MS Chat', then 'MS Skype For Business', 'MS Botframework', 'MS Webchat', 'MS Handover', and 'MS Notification' at the bottom. All nodes are blue with a small circular icon. A red box highlights the 'Shared Resource' node and its connection to 'MS Chat'. Below the graph, a message says 'Stages for a trigger change from automated to manual.' A dropdown menu is open, showing options like 'Select all', 'Shared Resource', 'MS Chat', 'MS Skype For Business', 'MS Botframework', 'MS Webchat', 'MS Handover', and 'MS Notification'. The 'MS Skype For Business' and 'MS Botframework' checkboxes are checked. At the bottom are 'Create' and 'Cancel' buttons.

If you need exclude some Microservice, please follow as below

The screenshot shows the 'Create a new release' dialog for '2.3 Provision'. The 'Stages for a trigger change from automated to manual' dropdown is open, showing a list of microservices: 'Select all', 'Shared Resource', 'MS Chat', 'MS Skype For Business', 'MS Botframework', 'MS Webchat', 'MS Handover', and 'MS Notification'. The 'MS Skype For Business' and 'MS Botframework' checkboxes are checked and highlighted with a red box. At the bottom are 'Create' and 'Cancel' buttons.

After select and make decision which Microservice should be deployed, you need change Source alias and Prefix name as image

The screenshot shows the 'Create a new release' dialog for '2.3 Provision'. In the 'Artifacts' section, the 'Source alias' is set to '\_Provision' and the 'Version' is '743703f0 (Updated manifest.yml)'. In the 'Variables' section, there is one entry: 'Prefix' with value 'Delivery-Kit-Latest' and scope 'Release'. At the bottom are 'Create' and 'Cancel' buttons, with 'Create' highlighted with a red box.

Then click Create button

The screenshot shows the Azure DevOps Pipelines interface for a release named 'Release-2.3-1'. The pipeline has three stages:

- Manually triggered**: Triggered by Thoan Bui on 7/2/2019, 9:59 AM. Artifacts: Provision.
- Bluemix VA Core**: Not deployed.
- Create Bot Channel**: Not deployed.

There is different between 2.2 and 2.3. On version 2.3, the provision will create Shared Resource and we can reuse to save the cost as well as resource...

The screenshot shows the Azure DevOps Pipelines interface for the same release 'Release-2.3-1'. The pipeline now includes a 'Shared Resource' stage, which is highlighted with a red border. The stages are:

- Shared Resource**: Succeeded.
- MS Chat**: In progress.
- MS Skype For Business**: Not deployed.

On pipelines, you can see the provision what we have just created as below

The screenshot shows the Azure DevOps Pipelines interface for the '2.3 Provision' release. The release 'Release-2.3-1' is highlighted with a red border. The details are:

- Name**: Release-2.3-1
- Created**: 7/2/2019, 9:59:38 AM

In case successful. All Microservices succeed and It look like

The screenshot shows a deployment pipeline for 'Release-2.3-2'. The stages are: 'Shared Resource' (Succeeded), 'MS Chat' (Succeeded), 'MS Skype For Business' (Succeeded), 'MS Botframework' (Succeeded), and 'MS Webchat' (Succeeded). Each stage is represented by a box with a green checkmark and the status 'Succeeded'.

In case there is error when create provision. We can click on Fail Microservice and review Log

The screenshot shows a deployment pipeline for 'Release-2.3-1'. The stages are: 'Single Tenant MS Tar' (Not deployed), 'Shared Resource' (Succeeded), 'MS Chat' (Failed), and 'MS Skype For Business' (Not deployed). The 'MS Chat' stage is highlighted with a red box. A tooltip below it says 'Azure Deployment>Create Or Update Resource Group act... on 7/2/2019, 10:05 AM'.

The screenshot shows the logs for 'Deployment attempt #2' of the 'MS Chat' stage. It includes tasks like 'Initialize job', 'Download Artifacts', and 'Azure Deployment>Create Or Update Resource Group act...'. A red box highlights the error message: 'At least one resource deployment operation failed. Please list deployment operations for details. Please see https://aka.ms/arm-debug for usage details.' Below this, it says 'Details:' and '2 more errors. Click on expand view in the context menu to view complete logs.'

Click on "4 errors" character, we can see detail

```

1 [2019-07-02T03:04:54.2326644Z ##[section]Starting: Azure Deployment:Create Or Update Resource Group action on MS Chat
2 2019-07-02T03:04:54.2638154Z =====
3 2019-07-02T03:04:54.2638658Z Task : Azure resource group deployment
4 2019-07-02T03:04:54.2638717Z Description : Deploy an Azure Resource Manager (ARM) template to a resource group and manage virtual machines
5 2019-07-02T03:04:54.2638786Z Version : 2.15.2
6 2019-07-02T03:04:54.2638828Z Author : Microsoft Corporation
7 2019-07-02T03:04:54.2638832Z Help : https://docs.microsoft.com/azure/devops/pipelines/tasks/deploy/azure-resource-group-deployment?view=azure-devops
8 2019-07-02T03:04:54.2639127Z =====
9 2019-07-02T03:04:54.56.9833642Z Checking if the following resource group exists: Delivery-Kit-Version-2.3-Microservice-Chat.
10 2019-07-02T03:04:54.57.4873464Z Resource group exists: true.
11 2019-07-02T03:04:57.4946175Z Creating deployment parameters.
12 2019-07-02T03:04:57.4946175Z The detected encoding for file 'D:\a\1\a\_Provision\101-microservice-chat\template.json' is 'utf-8'.
13 2019-07-02T03:04:57.4946175Z The detected encoding for file 'D:\a\1\a\_Provision\101-microservice-chat\Development\parameters.json' is 'utf-8'.
14 2019-07-02T03:04:57.5055510Z Starting Deployment...
15 2019-07-02T03:04:57.5070950Z Deployment name is template-20190702-030457-ffce
16 2019-07-02T03:04:57.5070950Z Deployment failed. One or more resources were not deployed in your deployment. Error code: DeploymentFailed.
17 2019-07-02T03:04:57.5070950Z 20190702-030457-ffce: Error: At least one resource deployment operation failed. Please list deployment operations for detail.
18 2019-07-02T03:04:57.5070950Z 20190702-030457-ffce: #*[error]BadRequest: [
19 2019-07-02T03:04:57.5070950Z     "Code": "BadRequest",
20 2019-07-02T03:04:57.5070950Z     "Message": "The host name delivery-kit-version-2.3-microservice-chat.azurewebsites.net is invalid.",
21 2019-07-02T03:04:57.5070950Z     "Target": null,
22 2019-07-02T03:04:57.5070950Z     "Details": [
23 2019-07-02T03:04:57.5070950Z         {
24 2019-07-02T03:04:57.5070950Z             "Message": "The host name delivery-kit-version-2.3-microservice-chat.azurewebsites.net is invalid."
25 2019-07-02T03:04:57.5070950Z         },
26 2019-07-02T03:04:57.5070950Z         {
27 2019-07-02T03:04:57.5070950Z             "Code": "BadRequest"
28 2019-07-02T03:04:57.5070950Z         },
29 2019-07-02T03:04:57.5070950Z     ]
30 2019-07-02T03:04:57.5070950Z #*[error]BadRequest: [

```

### 1.3 Verify resource group, application insights was created on Azure

After create provision successful, all resource group of all Microservices was created as below and app detail was created on resource group

NAME	SUBSCRIPTION	LOCATION
Delivery-Kit-03-Shared-Resource	DXC Workplace and Mobility D...	East US
Delivery-Kit-Latest-Microservice-BotFrameWork	DXC Workplace and Mobility D...	East US
Delivery-Kit-Latest-Microservice-Chat	DXC Workplace and Mobility D...	East US
Delivery-Kit-Latest-Microservice-Handover	DXC Workplace and Mobility D...	East US
Delivery-Kit-Latest-Microservice-Notification	DXC Workplace and Mobility D...	East US
Delivery-Kit-Latest-Microservice-S4B	DXC Workplace and Mobility D...	East US
Delivery-Kit-Latest-Microservice-WebChat	DXC Workplace and Mobility D...	East US
Delivery-Kit-Latest-Shared-Resource	DXC Workplace and Mobility D...	East US
Delivery-kit-Microservice-BotFrameWork	DXC Workplace and Mobility D...	East US

Search "App Insights" , we will find shared App Insights was created

NAME	RESOURCE GROUP	LOCATION	SUBSCRIPTION
Delivery-Kit-App-Insights-Bot	A...	East US	DXC Workplace and Mobil...
Delivery-Kit-App-Insights-Chat	A...	East US	DXC Workplace and Mobil...
Delivery-Kit-App-Insights-S4B	A...	East US	DXC Workplace and Mobil...
Delivery-Kit-Latest-Shared-Application-Insight	A...	East US	DXC Workplace and Mobil...
Delivery-kit-Shared-Application-Insight	A...	East US	DXC Workplace and Mobil...
Delivery-Kit-Shared-Component-Application-Insight	A...	East US	DXC Workplace and Mobil...
Delivery-Kit-Translation-AppInsights	A...	East US	DXC Workplace and Mobil...
Demo3-Shared-Application-Insight	A...	East US	DXC Workplace and Mobil...
Demo4-Shared-Application-Insight	A...	East US	DXC Workplace and Mobil...

## Step 2: Published API Management on Azure

### 2.1 Published API for Skype

#### Create 'API Management Service'

Access Azure → search API Management → select API Management Service → select "Add" button.

This is API Management service, what include all API of MS Chat, Skype, Team, Web Chat...Each customer should have 1 API Management, and this API will include API of Skype, MS Chat...

Use Shared Resource Group and Shared App Insights what was created by Provision

**API Management service**

- Name: Delivery-Kit-Latest-API-Management
- Subscription: DXC Workplace and Mobility Dev
- Resource group: Delivery-Kit-Latest-Shared-Resource
- Location: East US
- Organization name: DXC
- Administrator email: tbui21@csc.com

Pricing tier (View full pricing details): Developer (No SLA)

**Create**   **Automation options**

**API Management service**

- Organization name: DXC
- Administrator email: tbui21@csc.com

Pricing tier (View full pricing details): Developer (No SLA)

The developer tier of API Management doesn't include SLA and shouldn't be used for production purposes. Your service may experience intermittent outages, for example during upgrades. Learn more about API Management service tiers

Enable Application Insights

Application Insights instance: Delivery-Kit-Latest-Shared-Application-Insight

**Create**   **Automation options**

The screenshot shows the Microsoft Azure portal interface. On the left, there's a sidebar with various service icons like App Services, SQL databases, and Virtual machines. The main area is titled 'API Management services' under 'DXC Workplace and Mobility Microservice'. It shows a list of 'Subscriptions' with the following details:

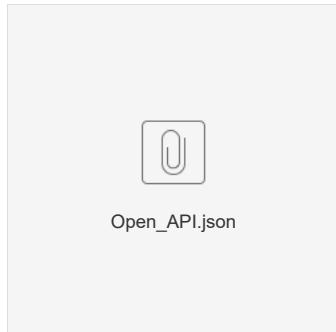
NAME	TIER	LOCATION	RESOURCE GROUP	SUBSCRIPTION
Delivery-Kit-API-Management	Or. Developer	East US	Delivery-kit-Micros...	DXC Workplace and ...
Delivery-Kit-Latest-API-Management	Ac. Developer	East US	Delivery-Kit-Latest-S...	DXC Workplace and ...
Delivery-Kit-MSChat-API	Or. Developer	East US	Delivery-kit-Micros...	DXC Workplace and ...
Delivery-Kit-S4B-API	Or. Developer	Central US	Delivery-kit-Micros...	DXC Workplace and ...
Delivery-Kit-Translation-API	Or. Developer	East US	Delivery-Kit-Micros...	DXC Workplace and ...
dxc-sandbox1-handover	Or. Developer	Central US	Delivery-kit-Micros...	DXC Workplace and ...
dxc-wm-dev	Or. Consumption	West US	WM-Shared-Compon...	DXC Workplace and ...
dxc-wm-qa	Or. Consumption	West US	WM-Shared-Compon...	DXC Workplace and ...
dxc-wm-test	Or. Consumption	West US	WM-Shared-Compon...	DXC Workplace and ...

### Create OpenAPI.json template file

Refer to below link to get latest content for Open API and create a file with name "OpenAPI.json" with content the same swagger.json file.

<https://github.dxc.com/WM-Microservice-Framework/ms-s4b/blob/master/apim/swagger.yml>

If you can not access above link, can use the attached file



### Add API for Skype

Select "Delivery-Kit-Latest-API-Management" has just created and Select "APIs" on the left menu → then select "Open APIs"

The screenshot shows the Microsoft Azure portal interface. The URL in the address bar is <https://portal.azure.com/#/resource/subscriptions/279adfe2-476c-490c-a1cf-be00be890369/resourceGroups/Delivery-Kit-Latest-Shared-Resource/providers/Microsoft.ApiManagement/service/Delivery-Kit-Latest-API-Management>. The main area is titled 'Delivery-Kit-Latest-API-Management - APIs' under 'API Management service'. The left sidebar shows various service options, and the 'APIs' option is highlighted by a red box. The main content area shows a 'Add a new API' section with several options: 'Blank API', 'OpenAPI' (which is highlighted by a red box), 'WADL', 'Echo API', 'Login API', 'API App', and 'Function'. The 'OpenAPI' option is described as 'Standard, language-agnostic interface to REST APIs'.

On Create Open API screen → select Browsers button and select OpenAPI.json file that created

Change the display name

Add API URL suffix as "api"

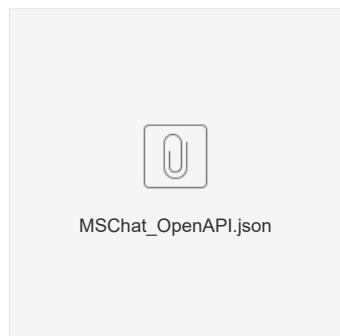
Select value for Products as "Unlimited".

Then click 'Create' button

## 2.2 Add API for MS Chat

Refer to below link and copy the content and save as "MS\_Chat\_OpenAPI.json" file

<https://github.dxc.com/WM-Microservice-Framework/ms-chat/blob/master/apim/swagger.yml>



Then follow the same step of Skype to create API for MS Chat

The screenshot shows the Microsoft Azure API Management service configuration page. The URL is <https://portal.azure.com/#/dxcwmicroservice.onmicrosoft.com/resource/subscriptions/279adfe2-476c-490c-a1cf-be00be890369/resourceGroups/Delivery...>. The left sidebar shows 'API Management' selected under 'APIs'. The main area displays the 'Delivery-Kit-Latest-API-Management - APIs' service. A modal dialog is open for 'Add API', with the 'Publisher portal' tab selected. The 'OpenAPI specification' field contains 'MSChat\_OpenAPI.json'. The 'Display name' field is set to 'delivery-kit-mschat-api'. The 'Name' field is also set to 'delivery-kit-mschat-api'. The 'API URL suffix' field is set to 'mschat'. The 'Products' field has 'Unlimited x' selected. The right side of the screen shows the XML representation of the Tfl API.

After create API for MS Chat, need update API URL and Base URL as below

The screenshot shows the Microsoft Azure API Management service interface. The left sidebar lists several APIs: 'delivery-kit-botframework', 'delivery-kit-mschat-api' (which is selected and highlighted in blue), 'delivery-kit-sfb-api', and 'Echo API'. The main content area is titled 'Delivery-Kit-Latest-API-Management - APIs' and shows the configuration for the selected API. The 'Design' tab is active. The 'Web service URL' is set to `https://delivery-kit-latest-microservice-chat.azurewebsites.net/api`. The 'URL scheme' is set to 'HTTPS'. The 'API URL suffix' is 'mschat/api'. The 'Base URL' is also `https://delivery-kit-latest-api-management.azure-api.net/mschat/api`. Under 'Tags', there is a placeholder 'e.g. Booking'. Under 'Products', it says 'Unlimited'. At the bottom, there are 'Save' and 'Discard' buttons.

Add Policies for MS Chat API by copy content on attached file and follow

The screenshot shows the Microsoft Azure API Management service interface. On the left, there's a sidebar with various options like Overview, Activity log, Access control (IAM), Tags, and API Management. Under API Management, 'APIs' is selected. In the main area, the 'Delivery-Kit-Latest-API-Management - APIs' page is shown. The 'Design' tab is active. A red box highlights the 'Inbound processing' section, which contains a 'Policies' tab with a red border. Another red box highlights the 'Outbound processing' section, which also has a 'Policies' tab with a red border.

Copy content and paste here and Save

This screenshot shows the 'Definitions' tab for the 'delivery-kit-mschat-api'. It displays the JSON configuration for the API's policies. A large red box highlights the policy code. Below the code, there are 'Save', 'Discard', and 'Reset to default' buttons, with the 'Save' button being highlighted by a red box.

```

12
13 <policies>
14   <inbound>
15     <base />
16     <cors>
17       <allowed-origins>
18         <origin></origin>
19       </allowed-origins>
20       <allowed-methods>
21         <method>GET</method>
22         <method>POST</method>
23       </allowed-methods>
24       <allowed-headers>
25         <header></header>
26       </allowed-headers>
27     <expose-headers>

```

After creating, you will see as below

This screenshot shows the 'Test' tab for the 'delivery-kit-mschat-api'. The 'SendMessage' operation is selected. The 'Request URL' field contains the URL 'https://delivery-kit-latest-api-management.azure-api.net/mschat/api/SendMessage'. The 'HTTP request' field shows the raw HTTP request: 'POST https://delivery-kit-latest-api-management.azure-api.net/mschat/api/SendMessage HTTP/1.1 Host: delivery-kit-latest-api-management.azure-api.net Ocp-Apim-Subscription-Key: ..... Ocp-Apim-Trace: true'. Both the 'Request URL' and the 'HTTP request' field are highlighted with red boxes.

## 2.3 Add API for MS Team

Refer to below link and copy the content and save as "MS\_Team\_OpenAPI.json" file

Then follow the same step of Skype to create API for MS Team

<https://github.com/WM-Microservice-Framework/ms-bot-framework/blob/master/apim/swagger.yml>



## Step 3: Microservice Capabilities

### 3.1 Create New Folder

The screenshot shows the Azure DevOps interface for the 'wm-microservice' project. On the left, there's a sidebar with organization settings and announcements. The main area has tabs for 'Projects', 'My work items', and 'My pull requests'. Under 'Projects', there are three cards: 'Provision' (blue), 'Microservices Capabilities' (red, highlighted with a red box), and 'ms-botframework' (green). A 'Filter projects' button is also visible.

The screenshot shows the Azure DevOps interface for the 'Microservices Capabilities' pipeline. The left sidebar has a 'Releases' section highlighted with a red box. The main area shows a 'Delivery' section with a table of releases. The table includes columns for 'Name', 'Release pipeline', 'Created', and 'Stage'. Several releases are listed, such as 'Build-Development-Release-1', 'UTC-Test-Release-7', and 'UTC-Test-Release-6'.

Name	Release pipeline	Created	Stage
Build-Development-Release-1	Development Micro...	2019-05-24 13:11	Green circle
UTC-Test-Release-7	UTC Test Microservice...	2019-05-23 18:28	Orange circle
UTC-Dev-Release-6	UTC Dev Microservice...	2019-05-21 23:43	Red circle
UTC-Dev-Release-5	UTC Dev Microservice...	2019-05-21 23:28	Red circle
UTC-Test-Release-6	UTC Test Microservice...	2019-05-21 13:22	Orange circle
UTC-Test-Release-5	UTC Test Microservice...	2019-05-21 11:57	Orange circle

[https://dev.azure.com/wm-microservice/Microservices%20Capabilities/\\_release?view=all&path=%5CDelivery](https://dev.azure.com/wm-microservice/Microservices%20Capabilities/_release?view=all&path=%5CDelivery)

**Delivery**

Releases

Release pipeline	Created	Stage
Build-Development-Release-1	Development Micro... 2019-05-24 13:11	[Icon]
UTC-Test-Release-7	UTC Test Microservice... 2019-05-23 18:28	[Icon]
UTC-Dev-Release-6	UTC Dev Microservice... 2019-05-21 23:43	[Icon]
UTC-Dev-Release-5	UTC Dev Microservice... 2019-05-21 23:28	[Icon]
UTC-Test-Release-6	UTC Test Microservice... 2019-05-21 13:22	[Icon]
UTC-Test-Release-5	UTC Test Microservice... 2019-05-21 11:57	[Icon]

[https://dev.azure.com/wm-microservice/Microservices%20Capabilities/\\_release?view=all&path=%5CDelivery](https://dev.azure.com/wm-microservice/Microservices%20Capabilities/_release?view=all&path=%5CDelivery)

**Create new folder**

Path: \Delivery

Name:  Delivery-Kit-2.3

Cancel OK

[https://dev.azure.com/wm-microservice/Microservices%20Capabilities/\\_release?view=all&path=%5CDelivery%5CDelivery-Kit-2.3](https://dev.azure.com/wm-microservice/Microservices%20Capabilities/_release?view=all&path=%5CDelivery%5CDelivery-Kit-2.3)

**Delivery-Kit-2.3**

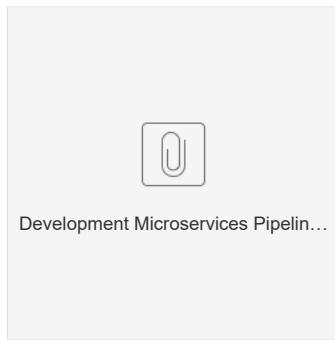
Releases

No releases found

You can create a new release manually or [setup triggers](#) to create it automatically

### 3.2 Import release pipeline

Use the template file and save on PC



## Import the template

Screenshot of the Azure DevOps Pipelines screen for the 'Delivery-Kit-2.3' project.

The left sidebar shows the following navigation:

- Overview
- Boards
- Repos
- Pipelines** (selected)
- Builds
- Releases
- Library
- Task groups
- Deployment groups
- Test Plans

The main area displays the 'Delivery-Kit-2.3' pipeline structure under the 'Delivery' folder:

- New release pipeline
- Import release pipeline** (highlighted with a red box)
- New folder
- Delivery-Kit-2.3 (highlighted with a red box)
- Delivery-Kit-Test
- OCIO
- UTC
- Framework

A modal window titled 'Import release pipeline' is open, prompting to 'Upload a release pipeline JSON file to import its contents'. A 'Browse...' button is highlighted with a red box.

Screenshot of the Azure DevOps Pipelines screen for the 'Delivery-Kit-2.3' project, showing the 'Import release pipeline' dialog.

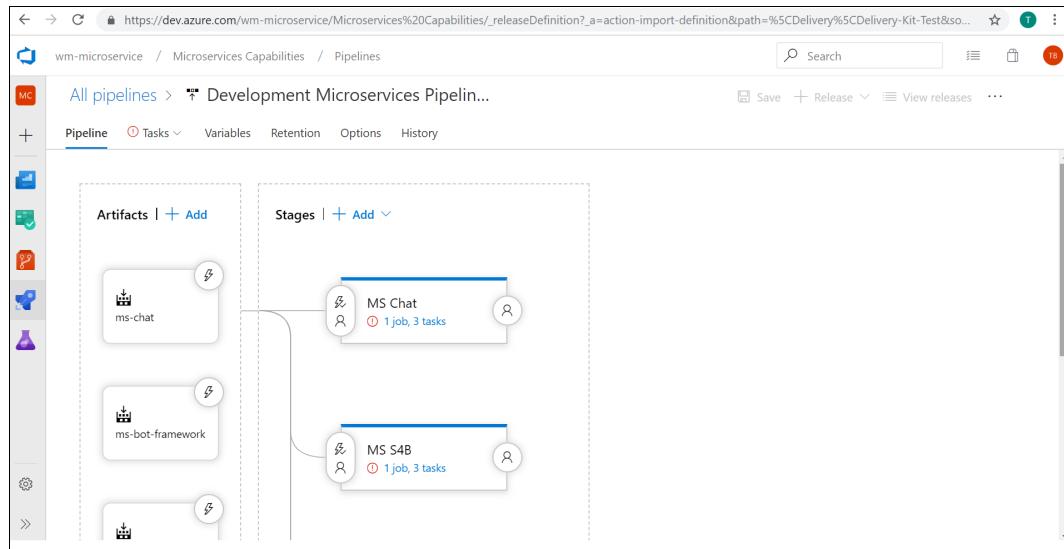
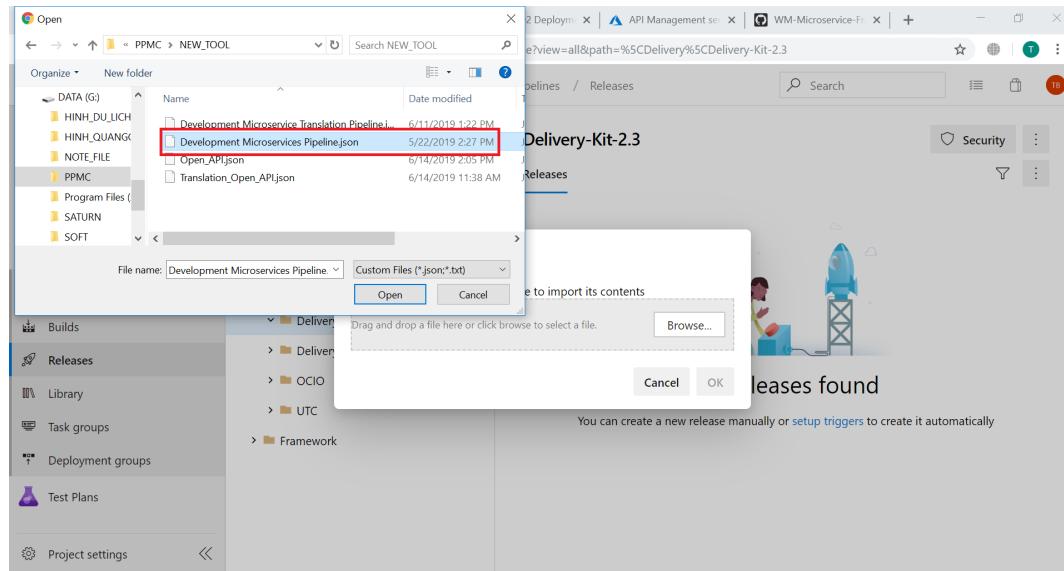
The left sidebar shows the following navigation:

- Overview
- Boards
- Repos
- Pipelines** (selected)
- Builds
- Releases
- Library
- Task groups
- Deployment groups
- Test Plans

The main area displays the 'Delivery-Kit-2.3' pipeline structure under the 'Delivery' folder:

- Build
- Delivery (highlighted with a red box)
- Delivery-Kit-2.3 (highlighted with a red box)
- Delivery-Kit-Test
- OCIO
- UTC
- Framework

A modal window titled 'Import release pipeline' is open, prompting to 'Upload a release pipeline JSON file to import its contents'. A 'Browse...' button is highlighted with a red box.



### 3.3 Enter information for all Microservice

Select "Tasks" and need input all information which red color

A screenshot of the Azure DevOps Pipelines interface, specifically the 'Tasks' tab for the 'MS Chat' stage of the 'Development Microservices Pipeline...'. The 'Tasks' tab is highlighted with a red box. On the right, the task configuration for 'MS Chat' is shown, including fields for 'Stage name' (set to 'MS Chat'), 'Parameters', 'Azure subscription' (set to 'Dev Subscription'), 'App type' (set to 'Function App for Containers (Linux)'), 'App Service name' (set to 'dxc-wm-build-chat-dev'), 'Registry or Namespace' (set to 'virtualagent.azurecr.io'), and 'Repository'.

All pipelines > Development Microservices ...

MS Chat

Agent job

Enable Continuous Monitoring

Configure Application Insights Alerts

Azure App Service Deploy

Stage name: MS Chat

Parameters: Unlink all

Azure subscription: DXC Workplace and Mobility Dev (279adfe2-476c-490c-...)

App type: Function App for Containers (Linux)

App Service name: Delivery-Kit-Latest-Microservice-Chat

Registry or Namespace: virtualagent.azurecr.io

Repository:

Select Shared Resource Group and Shared App Insight what was created from Provision.

The repository is linked to latest code

All pipelines > Development Microservices ...

MS Chat

Agent job

Enable Continuous Monitoring

Configure Application Insights Alerts

Azure App Service Deploy

Function App for Containers (Linux)

App Service name: Delivery-Kit-Latest-Microservice-Chat

Registry or Namespace: virtualagent.azurecr.io

Repository: wm-microservice-framework/ms-chat

Resource Group name for Application Insights: Delivery-Kit-Latest-Shared-Resource

Application Insights resource name: Delivery-Kit-Latest-Shared-Application-Insig

All pipelines > Development Microservices ...

MS Chat

Agent job

Run on agent

Enable Continuous Monitoring

Configure Application Insights Alerts

Azure App Service Deploy

Display name: Agent job

Agent selection:

Agent pool: Hosted Ubuntu 1604

Demands:

Name	Condition	Value

Execution plan:

This screenshot shows the Azure DevOps Pipelines interface for the 'MS Chat' deployment stage. The pipeline consists of three tasks: 'MS Chat', 'MS S4B', and 'MS Botframework'. A red box highlights the first task, 'MS Chat'. The configuration pane on the right shows the following settings:

- Stage name:** MS Chat
- Parameters:** Unlink all
- Azure subscription:** DXC Workplace and Mobility Dev (279adfe2-476c-490c-...)
- App type:** Function App for Containers (Linux)
- App Service name:** Delivery-kit-Microservice-Chat
- Registry or Namespace:** virtualagent.azurecr.io
- Repository:** (empty)

This screenshot shows the Azure DevOps Pipelines interface for the 'MS S4B' deployment stage. The pipeline consists of four tasks: 'MS S4B', 'Agent job', 'Enable Continuous Monitoring', and 'Configure Application Insights Alerts'. A red box highlights the 'Agent job' task. The configuration pane on the right shows the following settings:

- Web App for Containers (Linux)**
- App Service name:** Delivery-Kit-Latest-Microservice-S4B
- Registry or Namespace:** virtualagent.azurecr.io
- Repository:** wm-microservice-framework/ms-s4b
- Resource Group name for Application Insights:** Delivery-Kit-Latest-Shared-Resource
- Application Insights resource name:** Delivery-Kit-Latest-Shared-Application-Insig

## MS Team

This screenshot shows the Azure DevOps Pipelines interface for the 'MS Botframework' deployment stage. The pipeline consists of four tasks: 'MS Botframework', 'Agent job', 'Enable Continuous Monitoring', and 'Configure Application Insights Alerts'. A red box highlights the 'Agent job' task. The configuration pane on the right shows the following settings:

- Web App for Containers (Linux)**
- App Service name:** dxc-wm-build-bot-framework-dev
- Registry or Namespace:** virtualagent.azurecr.io
- Repository:** wm-microservice-framework/ms-bot-framework
- Resource Group name for Application Insights:** Delivery-Kit-Latest-Shared-Resource
- Application Insights resource name:** Delivery-Kit-Latest-Shared-Application-Insig

This screenshot shows the Azure DevOps Pipelines interface for the 'MS Botframework' deployment process. The pipeline consists of several tasks: 'Agent job' (Run on agent), 'Enable Continuous Monitoring' (Azure App Service manage), 'Configure Application Insights Alerts' (Azure Monitor alerts), and 'Azure App Service Deploy' (Azure App Service deploy). The 'Agent selection' section is highlighted with a red box, showing the 'Hosted Ubuntu 1604' pool selected.

## WebChat

This screenshot shows the Azure DevOps Pipelines interface for the 'MS Webchat' deployment process. The pipeline includes 'Agent job' (Run on agent), 'Enable Continuous Monitoring' (Azure App Service manage), 'Configure Application Insights Alerts' (Azure Monitor alerts), and 'Azure App Service Deploy' (Azure App Service deploy). The 'Azure App Service Deploy' task configuration is highlighted with a red box, showing details like 'Web App for Containers (Linux)', 'App Service name' (dxc-wm-build-webchat-dev), 'Registry or Namespace' (virtualagent.azurecr.io), 'Repository' (wm-microservice-framework/webchat), and 'Resource Group name for Application Insights' (Delivery-Kit-Latest-Shared-Resource).

This screenshot shows the Azure DevOps Pipelines interface for the 'MS Webchat' deployment process. The pipeline includes 'Agent job' (Run on agent), 'Enable Continuous Monitoring' (Azure App Service manage), 'Configure Application Insights Alerts' (Azure Monitor alerts), and 'Azure App Service Deploy' (Azure App Service deploy). The 'Agent selection' section is highlighted with a red box, showing the 'Hosted Ubuntu 1604' pool selected.

After all information was entered. Click "Save" button

The screenshot shows the Azure DevOps Pipelines interface for the 'MS Webchat' deployment process. On the left, there's a sidebar with options like Overview, Boards, Repos, Pipelines, Builds, Releases, Library, Task groups, Deployment groups, Test Plans, and Project settings. The main area displays the pipeline tasks: Agent job (Run on agent), Enable Continuous Monitoring (Azure App Service manage), Configure Application Insights Alerts (Azure Monitor alerts), and Azure App Service Deploy (Azure App Service deploy). In the top right, there are buttons for Save, Create release, and more. A red box highlights the 'Save' button.

This screenshot shows a 'Save' dialog box overlaid on the Azure DevOps Pipelines interface. The dialog has fields for 'Folder' (set to '\Delivery\Delivery-Kit-2.3') and 'Comment' (containing 'create pipeline cho version 2.3'). At the bottom are 'OK' and 'Cancel' buttons. The background shows the 'MS Webchat' pipeline configuration.

After Saving pipeline, we can see as below and we can change the name as below

This screenshot shows the Azure DevOps Pipelines list page. It lists pipelines under categories like All pipelines, Build, Delivery, and Analytics. One pipeline, 'Development Microservices Pipeline - version 2.3', is highlighted with a red box. The right side of the screen shows a summary with the message 'No releases found' and a 'Create a release' button.

## Step 4: Add variables

MS Chat

Azure DevOps Library page showing a list of variable groups. The '+ Variable group' button is highlighted with a red box.

Name	Date modified	Modified by	Description
Build MS Botframework Demo Dev	4/17/2019	Pham, Vinh Tuan	
Build MS Botframework Demo Prod	4/17/2019	Pham, Vinh Tuan	
Build MS Botframework Demo Test	4/17/2019	Pham, Vinh Tuan	
Build MS Botframework Development	4/10/2019	Pham, Vinh Tuan	
Build MS Botframework QA	4/17/2019	Pham, Vinh Tuan	
Build MS Botframework Sandbox	4/17/2019	Pham, Vinh Tuan	
Build MS Botframework Sandbox1	4/17/2019	Pham, Vinh Tuan	
Build MS Botframework Sandbox2	4/17/2019	Pham, Vinh Tuan	
Build MS Botframework Sandbox8	Sunday	Bang Cong Nguyen	
Build MS Botframework Test	4/17/2019	Pham, Vinh Tuan	
Build MS Botframework UAT	5/2/2019	Pham, Vinh Tuan	
Build MS Chat Demo Dev	4/17/2019	Pham, Vinh Tuan	

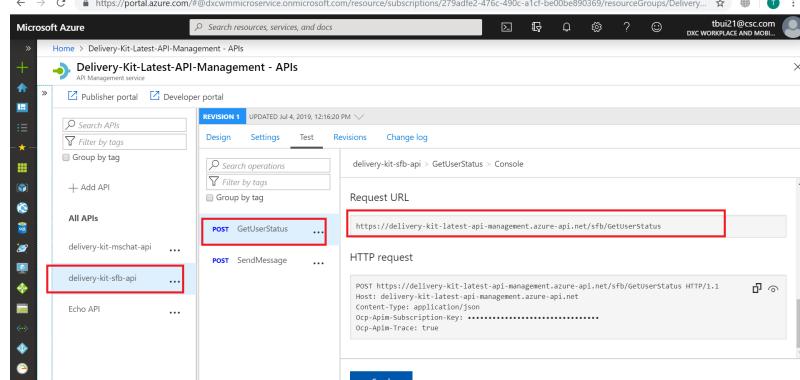
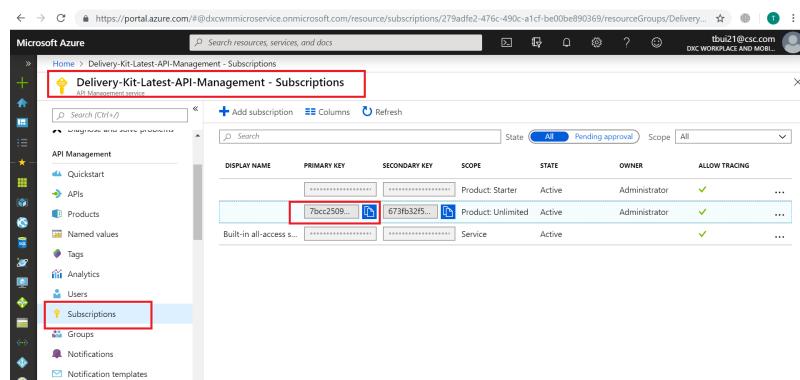
Enter group name as "Delivery Kit MS Chat Latest" and need add all variable as below. The value described on below table. You need read and follow so that know what values should enter

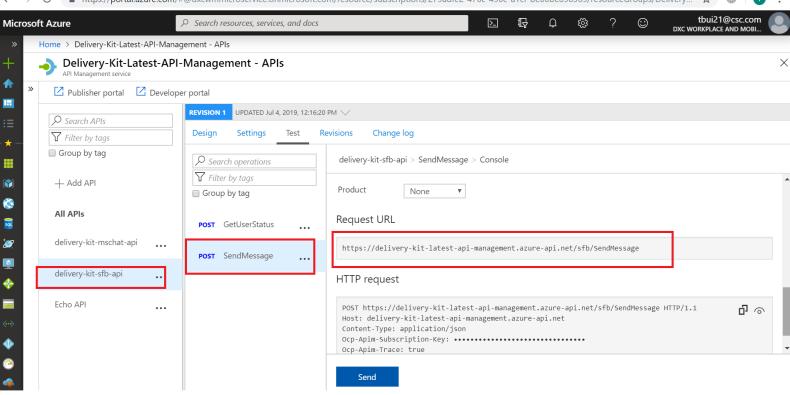
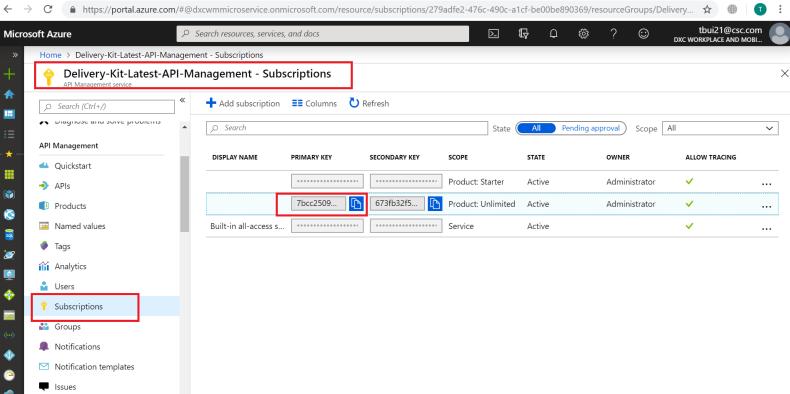
Azure DevOps Variable group page for 'Delivery Kit MS Chat Latest'. The entire content area is highlighted with a red box.

Name	Value
DOCKER-PASSWORD	cCWnqOW3szp/z/kbgoD=sGMFJoykOAKI
DOCKER-SERVER	http://virtualagent.azurecr.io
DOCKER-USERNAME	virtualagent
ENABLE_CHANNELS	skypeforbusiness, snow, snow_handover, msteams, cortana, cisco_spark...
USER_STATUS_API	{"url": "https://delivery-kit-latest-api-management.azure-api.net/api/Ge..."}
USER_STATUS_MSTEAM_API	
VA_SEND_MESSAGE_API	{"url": "https://api.us-south.apiconnect.appdomain.cloud/dxcvirtualage..."}
VA_SEND_PRO_MESSAGE_API	{"url": "https://delivery-kit-latest-api-management.azure-api.net/api/Se..."}
VA_SEND_PRO_MESSAGE_MSTEAMS_API	

Describe for value as below table

No	Parameters	Description	Require change
1	VA_SEND_MESSAGE_API	<p>We need 3 below information. These information is getting from VA Core.</p> <ul style="list-style-type: none"> <li>The URL on Bluemix of API Connect from VA Core</li> <li>x-ibm-client-id</li> <li>x-ibm-client-secret</li> </ul> <p>Please refer to below link to get 3 information from VA Core Deployment</p> <p>Client ID = ad0e0f33-16b4-402e-9316-fe29938b0b80  Client secret = aM2wV8bL2xJ7eB7iJ4tY7wH5cX1dE0hL0cB6lU5oC1bI5eW5pO  Gateway URLs = <a href="https://api.us-south.apiconnect.appdomain.cloud/dxcvirtualagent-va-dk23-dev/dk23-dev-catalog-api">https://api.us-south.apiconnect.appdomain.cloud/dxcvirtualagent-va-dk23-dev/dk23-dev-catalog-api</a>  The URL need add /api/SendMessage</p> <p>Example: {"url": "https://api.us-south.apiconnect.appdomain.cloud/dxcvirtualagent-va-dk23-dev/dk23-dev-catalog-api/api/SendMessage", "x-ibm-client-id": "ad0e0f33-16b4-402e-9316-fe29938b0b80", "x-ibm-client-secret": "aM2wV8bL2xJ7eB7iJ4tY7wH5cX1dE0hL0cB6lU5oC1bI5eW5pO"}</p>	Yes
2	DOCKER_REGISTRY_SERVER_URL	<a href="http://virtualagent.azurecr.io">http://virtualagent.azurecr.io</a>	Yes
3	DOCKER_REGISTRY_SERVER_PASSWORD	cCWnqOW3szp/z/kbgoD=sGMFJoykOAKI	Yes
4	DOCKER_REGISTRY_SERVER_USERNAME	virtualagent	Yes

No	Parameters	Description	Require change
5	USER_STATUS_API	<p>After create API Management for Skype (Step 2.1) . This parameter is URL of getUserStatus API of Skype Microservice</p> <pre>{"url":"https://delivery-kit-latest-api-management.azure-api.net/sfb/GetUserStatus", "subKey":"7bcc250922764bd8a524605ec7b40c5b"}</pre> <p>Get URL</p>  <p>Get Subkey</p> 	Yes

No	Parameters	Description	Require change
6	VA_SEND_PRO_MESSAGE_API	<p>This parameter is URL of sendMessage API of Skype Microservice (Step 2.1)</p> <pre>{"url":"https://delivery-kit-latest-api-management.azure-api.net/sfb/SendMessage", "subsKey":"7bcc250922764bd8a524605ec7b40c5b"}</pre> <p>Get URL</p>  <p>Get Subkey:</p> 	Yes
7	USER_STATUS_MSTEAM_API	This parameter is URL of getUserStatus API of MS Team Microservice (Refer Step 2.3)	Yes
8	VA_SEND_PRO_MESSAGE_MSTEAMS_API	This parameter is URL of sendMessage API of MS Team Microservice (Refer Step 2.3)	Yes
9	ENABLE_CHANNELS	<p>Default value</p> <p>skypeforbusiness, snow, snow_handover, msteams, cortana, cisco_spark, proactive_notification, webchat, s4b_timeout</p>	Yes

S4B

**Delivery Kit S4B**

**Variable group**

Delivery Kit S4B

Description

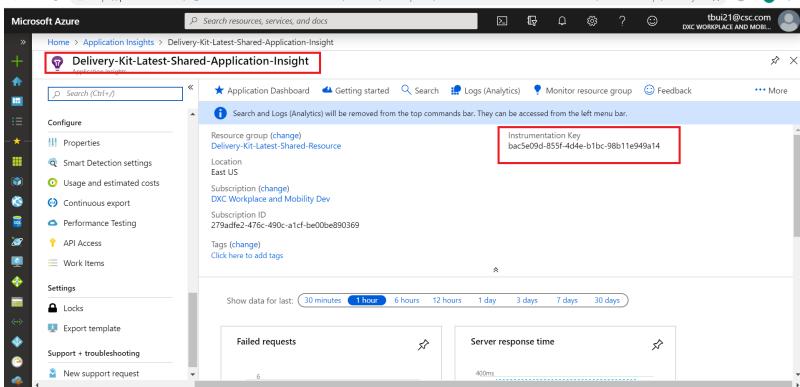
Allow access to all pipelines

Link secrets from an Azure key vault as variables

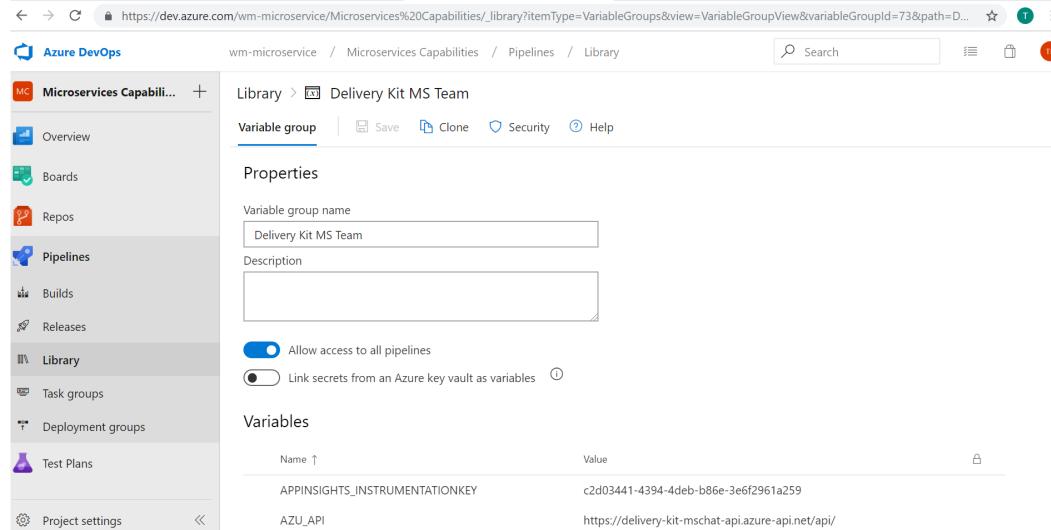
**Variables**

Name ↑	Value
APPINSIGHTS_INSTRUMENTATIONKEY	fdd19b56-217e-4a0e-9b14-f033cc9b58d4
AZU_API	https://delivery-kit-mschat-api.azure-api.net/api/
AZU_KEY	7ed052982ba6452eb917cb36d50a59ac
CODE_ACCESS_API	T9d7bjrZz2Lwzq5mkhsKuLWuMfpNn8mYfhNrnksevtRENKVkskybrAxZ...

Parameters	Description
AZU_KEY	<p><b>Subscription for MS Chat API Microservice</b></p> <p>Refer to Step 2.1</p> <p>We can get subscription</p>
AZU_API	<p><b>Request URL MS Chat API</b></p> <p>Refer to Step 2.2</p> <p>Select API of MS Chat, select Send Message → Test tab and scroll down Request URL</p> <p><b>Please remove "/SendMessage"</b></p> <p>Example:</p> <pre>https://delivery-kit-latest-api-management.azure-api.net/mschat/api</pre>
CODE_ACCESS_API	Code access to Skype for business server. Default is

SKYPE_4_BUSINESS	Only change username and password. This is username, password for Account for Bot {"identifier": "", "enable": true, "username": "Virtual-agent-delivery22@myworkstyle.info", "password": "Vuhavy123!", "domain": "myworkstyle.info"}  Refer to Prerequisites B.2 - Skype for Business Account for Bot
WEBSITE_HTTPLOGGING_RETENTION_DAYS	Default "30"
PERFROMANCE_TEST	Enable performance testing. Default is "true"
LOGGING	Logging configuration. Default is "true"
APPINSIGHTS_INSTRUMENTATIONKEY	Get app insights instruction key of Shared App Insights  Access Azure and search Application Insights, select Shared App Insights 
DOCKER_SERVER	<a href="http://virtualagent.azurecr.io">http://virtualagent.azurecr.io</a>
DOCKER_USERNAME	virtualagent
DOCKER_PASSWORD	cCWnqOW3szp/z/kbgoD=sGMFJoykOAKI

## MS TEAM



The screenshot shows the Azure DevOps interface for managing variable groups. On the left, there's a sidebar with options like Overview, Boards, Repos, Pipelines, Builds, Releases, Library (which is selected), Task groups, Deployment groups, Test Plans, and Project settings. The main area is titled 'Library > Delivery Kit MS Team'. It shows a 'Variable group' section with a 'Properties' tab. Under 'Properties', there's a 'Variable group name' field set to 'Delivery Kit MS Team' and a 'Description' field which is empty. Below that are two toggle buttons: 'Allow access to all pipelines' (which is turned on) and 'Link secrets from an Azure key vault as variables' (which is turned off). Under the 'Variables' section, there are two entries: 'APPINSIGHTS\_INSTRUMENTATIONKEY' with the value 'c2d03441-4394-4deb-b86e-3e6f2961a259' and 'AZU\_API' with the value 'https://delivery-kit-mschat-api.azure-api.net/api/'.

Parameters	Description
BOT_ENTRIES	We can set blank value first. Then after deploy, we need follow the step 02DeploymentMicroservice-6.2ConfigurationforMSTeamonAzure [{"appId": "f8838df1-6013-4bfa-904a-36fe9a5e4473", "appPassword": "h-snRromzvid-F=tjMqg4V5i1CZYKf3", "tenant": "137e5a2f-32be-433c-8333-433333333333"}]

AZU\_KEY

**Subscription for MS Chat API Microservice**

Refer to Step 2.2

After create API Management of MS Chat. We can get subscription

The screenshot shows the Azure portal interface for managing an API Management service. The left sidebar lists various Azure services like Home, All services, Favorites, Resource groups, App Services, SQL databases, etc. The main content area shows the 'Delivery-Kit-Latest-API-Management' service with its 'Subscriptions' section selected. A table displays three subscriptions: one for 'Product: Starter' and two for 'Product: Unlim.', both marked as 'Active'. The subscription row for '7bcc2...' is specifically highlighted with a red box.

AZU\_API

**Request URL MS Chat API**

Refer to Step 2.2

Select API of MS Chat, select Send Message → Test tab and scroll down Request URL

**Please remove "/SendMessage"**

Example:

`https://delivery-kit-latest-api-management.azure-api.net/mschat/api`

The screenshot shows the Azure portal interface for managing an API Management service. The left sidebar lists various Azure services like Home, All services, Favorites, Resource groups, App Services, SQL databases, etc. The main content area shows the 'Delivery-Kit-Latest-API-Management' service with its 'APIs' section selected. A table lists several APIs, including 'delivery-kit-mschat-api' which is highlighted with a red box. The 'Test' tab for this API is active, and the 'Request URL' input field is highlighted with a red box, showing the full URL 'https://delivery-kit-latest-api-management.azure-api.net/mschat/api/SendMessage'.

CODE\_ACCESS\_API

Code access to Skype for business server. Default is

`"T9d7bjrZz2Lwzq5mksKuLWuMfpNn8mYfhNrnksevtRENKVkskybrAxXzZVhDFxknCSc94XeeWAqrqfHe2YRWfKb8uMBWMUaby7Jf`

WEBSITE\_HTTPLOGGING\_RETENTION\_DAYS

LOGGING

PERFROMANCE\_TEST

DB\_INFO\_USERTEAMS

It is DB cloudant and username, password from VA Core. Please contact with people who deploy VA Core.

Each version need deploy and update this value so if you deploy another version. Please contact and get correct version

`{"database": "r1_userinfo_teams", "username": "eac243c3-a15b-4693-a434-036b6de3dddf-bluemix", "password": "c4772435c3702c229f}`

APPINSIGHTS\_INSTRUMENTATIONKEY

Get app insights instrumentatation key of Shared App Insights what was created on Provision step

The screenshot shows the Microsoft Azure Application Insights interface for the 'Delivery-Kit-Latest-Shared-Application-Insight'. It displays various monitoring metrics and settings. A red box highlights the 'Instrumentation Key' field, which is set to 'be5e09d-85f-4d4e-b1bc-98b11e949a14'. Below this, there are sections for 'Failed requests' (6) and 'Server response time' (40ms). The left sidebar includes options like 'Configure', 'Properties', 'Smart Detection settings', 'Usage and estimated costs', 'Continuous export', 'Performance Testing', 'API Access', 'Work items', 'Settings', 'Locks', 'Export template', 'Support + troubleshooting', and 'New support request'.

DOCKER_SERVER	<a href="http://virtualagent.azurecr.io">http://virtualagent.azurecr.io</a>
DOCKER_PASSWORD	cCWnqOW3szp/z/kg0D=sGMFJoykOAKI
DOCKER_USERNAME	virtualagent

DB\_MSTEAMS\_SESSION  
 It is DB cloudant and username, password from VA Core. Please contact with people who deploy VA Core.  
 Each version need deploy and update this value so if you deploy another version. Please contact and get correct version  
 {"database": "r1\_msteams\_session", "username": "eac243c3-a15b-4693-a434-036b6de3ddf-bluemix", "password": "c4772435c3702c2":  
 When we add this variable, we also need add on pipeline as below

#### Limitation need fix

The template is missing this variable. So we must add "DB\_MSTEAMS\_SESSION" variable as below

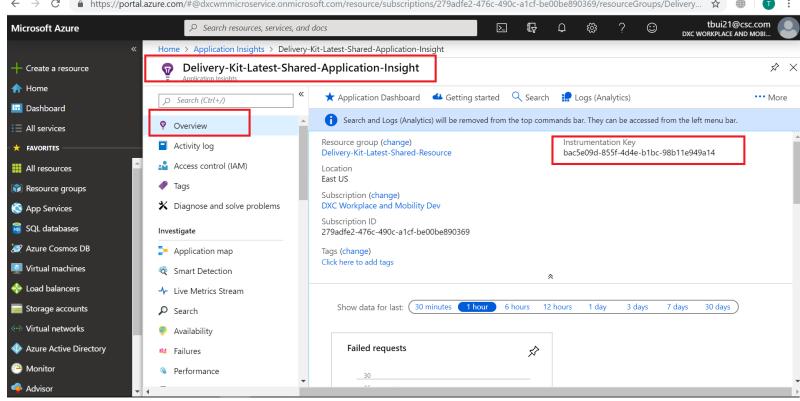
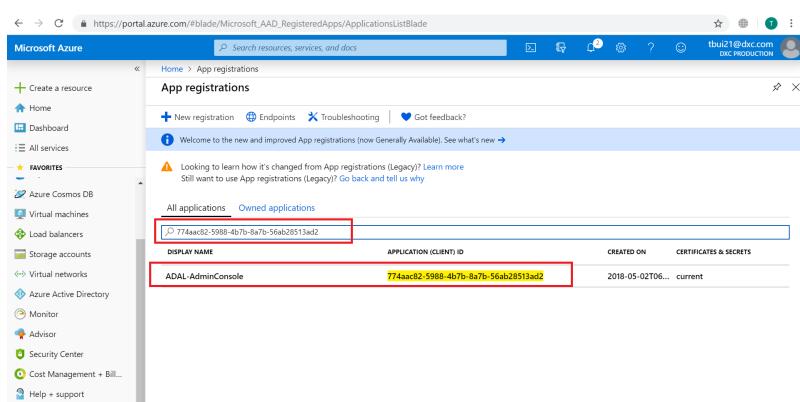
The screenshot shows the Azure DevOps Pipeline editor for the 'Development Microservices Pipeline'. The 'Tasks' tab is selected. A red box highlights the 'Azure App Service Deploy' task. The configuration pane on the right shows the 'App settings' section with several environment variables defined, including 'DOCKER\_REGISTRY\_SERVER' and 'DB\_MSTEAMS\_SESSION'. The 'Configuration settings' section is also visible.

The screenshot shows the 'App settings' dialog box from the Azure DevOps Pipeline editor. A red box highlights the 'DB\_MSTEAMS\_SESSION' entry, which is set to '\$(DB\_MSTEAMS\_SESSION)'. The 'OK' button is visible at the bottom of the dialog.

## WEBCHAT

The screenshot shows the Azure DevOps Pipelines Library interface. A variable group named "Delivery Kit WebChat" is created. It contains three variables:

- ACCESS\_KEY**: Value: GOkZRwgZ61q0XXVvxIB8TS1D6lrG7Vb9V8YwRDfy3YGAN7TM7EnxWHqdbIZfheZ
- APPINSIGHTS\_INSTRUMENTATIONKEY**: Value: 9d2ffdf2-2ef2-43db-9e58-7323860b1b7a
- AZURE\_REG\_APPID**: Value: 39d7409c-1f9e-40a2-bc0a-cdaa0926ada7

Parameters	Description	Required changes
ACCESS_KEY	<b>Default</b> <b>GOkZRwgZ61q0XXVvxIB8TS1D6lrG7Vb9V8YwRDfy3YGAN7TM7EnxWHqdbIZfheZ</b>	
APPINSIGHTS_INSTRUMENTATIONKEY	Used Shared App Insights was created by Provision and get Instrumentation Key 	Yes
AZURE_REG_APPID	Refer to Steps 6.3 Need set owner and User and Group so that user can access to webchat. Refer to Deploy step 6.3. Then search "App Registration" and select App Regis for Webchat and copy the Application ID 774aac82-5988-4b7b-8a7b-56ab28513ad2 	Yes
CHAT_MICROSERVICE	{"sendMessageAPI": "https://delivery-kit-mschat-api.azure-api.net/api/SendMessage", "Ocp-Apim-Subscription-Key": "7ed052982ba6452eb917cb36d50a59ac"}	

## API

Microsoft Azure

Home > Delivery-Kit-Latest-API-Management - APIs

Delivery-Kit-Latest-API-Management - APIs

All APIs

delivery-kit-mschat-api

delivery-kit-sfb-api

Echo API

Request URL

https://delivery-kit-latest-api-management.azure-api.net/mschat/api/sendMessage

HTTP request

POST https://delivery-kit-latest-api-management.azure-api.net/mschat/api/sendMessage HTTP/1.1  
Ocp-Azure-Subscription-Id: 4e5...  
Ocp-Azure-Tracer: True

Send

## Subscription Key

Microsoft Azure

Home > API Management services > Delivery-Kit-Latest-API-Management - Subscriptions

Delivery-Kit-Latest-API-Management - Subscriptions

Subscriptions

DISPLAY NAME	PRIMARY KEY	SECONDARY KEY	SCOPE	STATE	OWNER	ALLOW TRACING
7bcc2...	...	...	Product: Starter	Active	Administrator	...
673fb...	...	...	Product: Uni...	Active	Administrator	...
Built-in all acc...	...	...	Service	Active	...	...

DOCKER_SERVER	<a href="http://virtualagent.azurecr.io">http://virtualagent.azurecr.io</a>
DOCKER_USERNAME	virtualagent
DOCKER_PASSWORD	cCWnqOW3szp/z/kgD=sGMFJoykOAKI
DOTENVENC_KEY	P@ssword123!
TENANT	<p>Refer to Steps 6.3</p> <p>This is an organization information, we need get Tenant name</p> <p>Need switch to "DXC Production" subscription and get the tenant name</p> <p><a href="https://CSCPortal.onmicrosoft.com">CSCPortal.onmicrosoft.com</a></p>
WEBCHAT_AUTHENTICATION	<p>There is 2 values:</p> <ul style="list-style-type: none"> <li>False: if not require user log in before use Webchat. Set this value is false</li> <li>True: if require user log in before using WebChat ==&gt; Should set TRUE for user logging required</li> </ul>
AUTH_PROTOCOL	<p>On version 2.2, we only apply for Azure authentication OAuth2</p> <p>For SAML, not yet applied on Version 2.2</p> <p>Therefore, we no need declare this parameter</p>

## Step 5: Link variables to pipeline

Select pipeline --> Release --> select our release --> Edit --> Variable --> Variable group --> Link --> create Link for all Microservice

The screenshot shows the Azure DevOps interface for a project named 'wm-microservice'. In the left sidebar, 'Pipelines' is selected. On the main page, the 'Development Microservices Pipeline - version 2.3' is listed under 'All pipelines'. The 'Edit' button for this pipeline is highlighted with a red box.

The screenshot shows the 'Variables' tab within the pipeline settings. Under 'Variable groups', the 'Link variable group' button is highlighted with a red box.

The screenshot shows the 'Link variable group' dialog box. It lists several variable groups: 'Delivery Kit \$4B (11)', 'Delivery Kit WebChat (10)', 'Delivery Kit WebChat Latest (10)', 'Delivery UTC Dev MS Chat (8)', and 'Delivery UTC Dev2 MS Chat (8)'. The 'Delivery Kit WebChat (10)' group is selected and highlighted with a blue border.

The same for others

**Special, must add Docker Variables as Global variables**

The screenshot shows the 'Variables' tab in the Azure DevOps Pipelines interface. On the left is a navigation sidebar with options like Overview, Boards, Repos, Pipelines, Builds, Releases, Library, Task groups, Deployment groups, Test Plans, and Project settings. The main area displays a table of 'Pipeline variables' under 'Variable groups'. The groups listed are 'Delivery Kit MS Chat Latest (9)', 'Delivery Kit MS Team Latest (13)', 'Delivery Kit S4B Latest (11)', 'Delivery Kit WebChat (10)', and 'Docker Variables (3)'. A red box highlights the 'Delivery Kit MS Chat Latest (9)' group. At the bottom of the table are buttons for 'Link variable group' and 'Manage variable groups'.

After add all variables, click Save button

The screenshot shows a 'Save' dialog box overlaid on the Azure DevOps interface. The dialog has a 'Comment' field containing the text 'save link variable'. Below the comment field are 'OK' and 'Cancel' buttons. The background shows the same Pipelines Variables page as the previous screenshot, with the 'Delivery Kit MS Chat Latest (9)' group highlighted by a red box.

## Step 6: Deploy

### 6.1 Deploy

The screenshot shows the 'Releases' tab in the Azure DevOps interface. The left sidebar includes 'Overview', 'Boards', 'Repos', 'Pipelines' (which is selected), 'Builds', 'Releases' (selected), 'Library', 'Task groups', 'Deployment groups', 'Test Plans', and 'Project settings'. The main area displays a pipeline named 'Development Microservices Pipeline - version 2.3'. A red box highlights the 'Development Microservices Pipeline' name. To the right of the pipeline name is a 'Create release' button, which is also highlighted with a red box. Below the pipeline name, there are sections for 'Releases', 'Deployments', and 'Analytics'. A large central image features a person and a dog interacting with a rocket launching from a platform. Below the image, the text 'No releases found' is displayed, along with the instruction 'You can create a new release manually or [setup triggers](#) to create it automatically' and a 'Create a release' button.

The screenshot shows the 'Create a new release' dialog for the 'Development Microservices Pipeline - version 2.3'. The pipeline stages are listed on the right: MS Chat, MS S4B, MS Botframework, and MS Webchat. The 'Create' button is at the bottom.

The screenshot shows the 'Create a new release' dialog with the 'Artifacts' section open. It lists four artifacts: ms-bot-framework, ms-chat, ms-s4b, and webchat, all set to 'latest'. The 'Create' button is at the bottom.

8

If successful, you can see as below

The screenshot shows the 'Build-Development-Release-1' pipeline progress. The 'MS Chat' stage has succeeded, 'MS S4B' has succeeded, and 'MS Botframework' is currently in progress. The pipeline status bar indicates 'Manually triggered' by 'Thoan Bui' on 7/2/2019, 1:38 PM.

## 6.2 Configuration for MS Team on Azure

### Create Register Channel for MS Team

## Update message endpoint on Setting

Access portal Azure and select from resource group of MS team then follow below steps to update Setting & Channel

Copy the link and paste to notepad. Then come back resource group and select Chanel

NAME	TYPE	LOCATION
(E) Delivery-Kit-Latest-Bot-Channel	Bot Channels Re...	global
Delivery-Kit-Latest-Bot-Channelwjmvb	Application Ins...	East US
Delivery-Kit-Latest-Microservice-BotFrameWork	App Service	East US

Select Setting and change message endpoint with format

<url link>/ms-api/<Microsoft App ID>/messages

Need change url link and Microsoft App ID

Example: <https://delivery-kit-microservice-botframework.azurewebsites.net/ms-api/00b7b23c-bee6-4238-9b30-794fe3515fc5/messages>

The screenshot shows the Azure portal interface for managing a bot channel. The left sidebar lists various services like Home, Dashboard, All services, and Resource groups. The main content area is titled 'Delivery-Kit-Latest-Bot-Channel - Settings'. Under the 'Bot management' section, 'Channels' is selected. The 'Configuration' section is expanded, showing the 'Messaging endpoint' field with the URL 'https://delivery-kit-latest-microservice-botframework.azurewebsites.net/ms-api/f8838df1-6013-4bfa-904a-36fe9a5e4473/message'. Below this, the 'Microsoft App ID (Manage)' field also displays the same URL. Both fields are highlighted with red boxes.

In case "Microsoft App ID" don't have data. We need register. Please click on "Manage" to register

The screenshot shows the Microsoft Application Registration Portal. The title bar says 'Application Registration Portal'. The main content area is titled 'Pick an account'. It shows a list of accounts, with one account named 'Bui, Thoan' and the email 'tbui21@csc.com' listed as signed in. Below this is a 'Use another account' button. The entire window is framed by a red border.

Save setting

### Add new channel

The screenshot shows the Azure portal interface for managing channels. The left sidebar lists various services like Home, Dashboard, All services, and Resource groups. The main content area is titled 'Delivery-kit-Channel - Channels'. Under the 'Bot management' section, 'Channels' is selected. The 'Connect to channels' section shows a table with one row: 'Name' (Web Chat), 'Health' (Running), and 'Published' (--). There is an 'Edit' button next to the row. Below the table is a 'Get bot embed codes' button. At the bottom, there is a 'Add a featured channel' section with icons for various platforms, including Microsoft Teams, which is highlighted with a red box.

The screenshot shows the Microsoft Azure portal interface. The left sidebar has a 'Bot management' section with 'Channels' selected. The main content area is titled 'Configure MSTeams' and shows a 'Messaging' tab. At the bottom, there are 'Cancel', 'Save' (which is highlighted with a red box), and 'Delete Channel' buttons.

### Update value for variable for BOT\_ENTRIES on DEV Tool and build again

After register, we need get 3 information to update BOT\_ENTRIES variables. Please follow below steps to get values of AppId, appPassword and tenant

This is example:

AppId: Microsoft App ID

appPassword:

tenant:

```
[{"appId":"00b7b23c-bee6-4238-9b30-794fe3515fc5","appPassword":"oopnFNQL9@jicjTNX8315[","tenant":"137e5a2f-32be-4811-b649-15bcce76b74f"}]
```

### Get App ID

The screenshot shows the Microsoft Azure portal interface. The left sidebar has a 'Bot management' section with 'Settings' selected. The main content area is titled 'Delivery-Kit-Latest-Bot-Channel - Settings' and shows a 'Configuration' section with a 'Messaging endpoint' field containing 'https://delivery-kit-latest-microservice.botframework.azurewebsites.net/ms-api/f8838df1-6013-4bfa-904a-36fe9a5e4473/messages'. Below it, there is a 'Microsoft App ID (Manage)' field with the value 'f8838df1-6013-4bfa-904a-36fe9a5e4473' highlighted with a red box.

Search "App Registration" and Select "Certificates & secrets" to create new password. The password only appear once time

The screenshot shows the Microsoft Azure portal interface. On the left, the navigation menu includes 'Create a resource', 'Home', 'Dashboard', 'All services', 'FAVORITES' (with 'All resources' selected), 'Resource groups', 'App Services', 'SQL databases', 'Azure Cosmos DB', 'Virtual machines', 'Load balancers', 'Storage accounts', 'Virtual networks', 'Azure Active Directory', 'Monitor', and 'Advisor'. The main content area shows the 'Delivery-Kit-Latest-Bot-Channel - Certificates & secrets' page. A red box highlights the 'Certificates & secrets' link in the left sidebar. Another red box highlights the '+ New client secret' button. Below it, a table lists client secrets:

DESCRIPTION	EXPIRES	VALUE
No description	7/4/2024	jtI*****
password	12/31/2299	tkV*****

This screenshot shows the 'Add a client secret' dialog box overlaid on the Azure portal. The 'Description' field is empty. The 'Expires' section has three options: 'In 1 year', 'In 2 years', and 'Never', with 'Never' selected. A red box highlights the 'Never' option. At the bottom are 'Add' and 'Cancel' buttons, with 'Add' highlighted by a red box. To the right, the client secret table remains the same as in the previous screenshot.

The password only appear once time, so copy and paste to notepad and use

#### Get Tenant ID

The screenshot shows the Microsoft Azure portal interface. The navigation menu is identical to the first screenshot. The main content area shows the 'DXC Workplace and Mobility Microservice - Properties' page under 'Azure Active Directory'. A red box highlights the 'Properties' link in the left sidebar. In the center, the 'Directory properties' section includes fields for 'Name' (set to 'DXC Workplace and Mobility Microservice'), 'Country or region' (United Kingdom), 'Location' (EU Model Clause compliant datacenters), 'Notification language' (English), and 'Directory ID' (highlighted by a red box and containing the value '137e5a2f-32be-4811-b649-15bcce76b74f'). Other sections like 'Security' and 'Notifications settings' are also visible.

Update BOT\_ENTRIES variable as below

[{"appId":"f8838df1-6013-4bfa-904a-36fe9a5e4473","appPassword":"h-snRromzvid-F=tjMqg4V5i1CZYKf3","tenant":"137e5a2f-32be-4811-b649-15bcce76b74f"}]

The screenshot shows the Azure DevOps interface for a project named 'wm-microservice'. The left sidebar has 'Pipelines' selected. The main area shows a 'Library' view for 'Delivery Kit MS Team Latest\*'. A variable group named 'BOT\_ENTRIES' is highlighted with a red box. Other variables listed include AZU\_API, AZU\_KEY, CODE\_ACCESS\_API, DB\_MSTEAMS\_SESSION, DOCKER\_PASSWORD, DOCKER\_SERVER, DOCKER\_USERNAME, LOGGING, USER\_STATUS\_API, and WEBSITE\_HTTPLOGGING\_RETENTION\_DAYS.

Build again

## 6.3 Configuration for Web Chat on Azure

### Update Policy on MS Chat so that can support Web Chat

Select "Inbound Processing" → click "Add Policy"

The screenshot shows the Microsoft Azure API Management service for the 'Delivery-Kit-Latest-API-Management - APIs' service. In the 'Operations' tab, the 'Send Message' operation is selected. In the 'Definitions' tab, under 'Inbound processing', two policies are added: 'base' and 'cors'. The 'cors' policy is highlighted with a red box.

Select "Allow cross-origin resource sharing (CORS)"

The screenshot shows the Microsoft Azure API Management service for the same service. In the 'Operations' tab, the 'Send Message' operation is selected. In the 'Definitions' tab, the 'cors' policy under 'Allow cross-origin resource sharing (CORS)' is highlighted with a red box.

Type \* on Allowed origins and Save

Microsoft Azure

Delivery-Kit-Latest-API-Management - APIs

All APIs

- delivery-kit-mschat-api
- delivery-kit-sfb-api
- Echo API

Operations Definitions

delivery-kit-mschat-api > SendMessage > Policies

Allow cross-origin resource sharing (CORS)

Set cross-origin resource sharing (CORS) policy to allow cross-domain calls from br

Learn more about "cors" policy.

Basic | Full

Allowed origins

Save Discard

## Configuration Authorization

Create new App Register for webchat

Microsoft Azure

App registrations

+ New registration Endpoints Troubleshooting Got feedback?

	Name	Created	Status
WC	wm-microservice-Microservices Capabilities-279	5/22/2019	Current
WC	wm-microservice-Microservices Capabilities-279	5/22/2019	Current
WC	wm-microservice-Microservices Capabilities-279	5/22/2019	Current
WC	wm-microservice-Microservices Capabilities-279	5/22/2019	Current
WC	wm-microservice-Microservices Capabilities-279	5/22/2019	Current
WC	wm-microservice-Microservices Capabilities-279	5/22/2019	Current
WC	wm-microservice-Microservices Capabilities-279	5/22/2019	Current
WC	wm-microservice-Microservices Capabilities-279	5/22/2019	Current
WC	wm-microservice-Microservices Capabilities-279	5/22/2019	Current
WC	wm-microservice-MS-chat-279adfe2-476c-490c-5ef310fe-34bf-4431-8672-5303193e986a	6/24/2019	Current
DE	Delivery-Kit-Bot-App_register	6/26/2019	Current
DE	Delivery-Kit-Latest-Bot-Channel	7/4/2019	Current
DE	Delivery-Kit-Latest-MSTeam-App-Regis	7/4/2019	Current

Microsoft Azure

Home > App registrations > Register an application

Register an application

\* Name

The user-facing display name for this application (this can be changed later).

Delivery-Kit-WebChat-AppRegis

Supported account types

Who can use this application or access this API?

Accounts in this organizational directory only (DXC Workplace and Mobility Microservice)

Accounts in any organizational directory

Accounts in any organizational directory and personal Microsoft accounts (e.g. Skype, Xbox, Outlook.com)

Help me choose...

By proceeding, you agree to the Microsoft Platform Policies

Register

**Delivery-Kit-WebChat-AppRegis - Authentication**

**Advanced settings**

Logout URL (e.g. https://myapp.com/logout)

**Implicit grant**

Allows an application to request a token directly from the authorization endpoint. Recommended only if the application has a single page architecture (SPA), has no backend components, or invokes a Web API via JavaScript.

To enable the implicit grant flow, select the tokens you would like to be issued by the authorization endpoint:

Access tokens  
 ID tokens

**Default client type**

Treat application as a public client. Required for the use of the following flows where a redirect URI is not used:

Yes  No

## Update URL

**Delivery-Kit-WebChat-AppRegis - Authentication**

**Redirect URIs**

The URIs that we will accept as destinations when returning authentication responses (tokens) after successfully authenticating users. Also referred to as reply URLs.

[Learn more about adding support for web, mobile and desktop clients](#)

TYPE	REDIRECT URI
Web	https://delivery-kit-latest-microservice-webchat.azurewebsites.net
Web	e.g. https://myapp.com/auth

**Suggested Redirect URIs for public clients (mobile, desktop)**

If you are using the Microsoft Authentication Library (MSAL) or the Active Directory Authentication Library (ADAL) to build applications for desktop or mobile devices, you may select from the suggested Redirect URIs below or enter a custom redirect URI above. For more information, refer to the library documentation.

- msal39d7409c-1f9e-40a2-bc0a-cdaa0926ada7://auth (MSAL only)
- https://login.microsoftonline.com/common/oauth2/nativeclient
- https://login.live.com/oauth20\_desktop.srf (LiveSDK)

## Add User &amp; Group to Webchat

**Delivery-Kit-WebChat-AppRegis**

**Overview**

Welcome to the new and improved App registrations. Looking to learn how it's changed from App registrations (Legacy)? [Learn more](#)

Display name: Delivery-Kit-WebChat-AppRegis

Application (client) ID: 39d7409c-1f9e-40a2-bc0a-cdaa0926ada7

Directory (tenant) ID: 137e5a2f-32be-4811-b649-15bcc76b74f

Object ID: 060669bb-f881-4ea2-a5d4-e7ea6a921c45

Supported account types: My organization only

Redirect URIs: 1 web, 0 public client

Managed application in local directory: Delivery-Kit-WebChat-AppRegis

**Call APIs**

**Documentation**

- Microsoft identity platform
- Authentication scenarios
- Authentication libraries
- Code samples
- Microsoft Graph
- Glossary

Click on Delivery-Kit-WebChat-AppRegis

The screenshot shows the Microsoft Azure portal interface. On the left, there's a sidebar with various icons and a navigation tree. The main area is titled 'Delivery-Kit-WebChat-AppRegis - Users and groups'. At the top, there are buttons for 'Add user', 'Edit', 'Remove', 'Update Credentials', and 'Columns'. A note says 'The application will appear on the access panel for assigned users. Set visible to users? to no in properties to prevent this.' Below is a search bar and a table with four rows:

DISPLAY NAME	OBJECT TYPE	ROLE ASSIGNED
Bui, Thoan	User	Default Access
Kulkarni, Ravi	User	Default Access
Ly, Duong Quy	User	Default Access
Nguyen, Quyen	User	Default Access

Click Add User

This screenshot shows the 'Add Assignment' step in the Azure portal. The left sidebar has a 'Users and groups' section with 'None Selected' highlighted by a red box. The main area is titled 'Users and groups' and shows a search bar and a list of users. One user, 'Ly, Duong Quy', is selected and highlighted with a red box. Below is a 'Selected members:' list with two entries: 'Bui, Thoan' and 'Ly, Duong Quy'. At the bottom is a 'Select' button highlighted with a red box.

Click Select. Then click "Assign"

This screenshot shows the 'Users and groups' section again. The table now highlights the entire row for the selected users (Bui, Thoan and Ly, Duong Quy) with a red box. The rest of the table remains the same as in the previous screenshots.

## Register Webchat to App Register

Access portal.azure. Get the Webchat link from App Service of Webchat was created on Provision steps

<https://confluence.csc.com/display/PD/VA+2.3+-+02+Deployment+Microservice>

<https://delivery-kit-latest-microservice-webchat.azurewebsites.net>

The screenshot shows the Microsoft Azure portal interface. In the top navigation bar, the URL is https://portal.azure.com/#@dxcmicroservice.onmicrosoft.com/resource/subscriptions/279adfe2-476c-490c-a1cf-be00be890369/resourceGroups/Delivery... The main content area displays the 'Delivery-Kit-Latest-Microservice-WebChat' App Service blade. Key details shown include:

- Resource group (change):** Delivery-Kit-Latest-Microservice-WebChat
- Status:** Running
- Location:** East US
- Subscription (change):** DXC Workplace and Mobility Dev
- Subscription ID:** 279adfe2-476c-490c-a1cf-be00be890369
- Tags (change):** Click here to add tags
- URL:** <https://delivery-kit-latest-microservice-webchat.azurewebsites.net> (highlighted with a red box)
- App Service Plan:** Delivery-Kit-Latest-Shared-App-Service (B2: 1)
- Location:** No FTP/deployment user set
- FTP hostname:** ftp://waws-prod-blu-103.ftp.azurewebsites.windows.net
- FTPS hostname:** https://waws-prod-blu-103.ftp.azurewebsites.windows.net

#### Access to Directory "DXC Production"

The screenshot shows the Microsoft Azure portal interface. The user profile 'Bui, Thoan' (tbui21@csc.com) is visible in the top right corner. A 'Switch directory' button is highlighted with a red box. Other user profiles shown are 'csc' (tbui21@dvc.com) and 'Sign in with a different account'. The 'Recent resources' section lists several items, and the 'Useful links' section includes links to Technical Documentation, Azure Services, Recent Azure Updates, and Azure Blog.

The screenshot shows the Microsoft Azure portal interface. The user profile 'Bui, Thoan' (tbui21@csc.com) is visible in the top right corner. The 'Global subscription filter' dropdown is set to 'All subscriptions'. The 'Switch directory' section shows the current directory as 'dxcmicroservice.onmicrosoft.com'. The 'Favorites' list contains entries for 'DXC Production' (CSCPortal.onmicrosoft.com) and 'DXC Workplace and Mobility Microservice' (dxcmicroservice.onmicrosoft.com). The 'Recent resources' section lists several items, and the 'Useful links' section includes links to Technical Documentation, Azure Services, Recent Azure Updates, and Azure Blog.

Search "App registration" and Select "ADAL-AdminConsole"

The screenshot shows the Microsoft Azure portal interface. The left sidebar lists various services like Home, Dashboard, All services, and Favorites. The main content area is titled 'App registrations' under 'All applications'. A search bar at the top has 'ADAL-AdminConsole' typed into it. Below the search bar, there are columns for 'DISPLAY NAME', 'APPLICATION (CLIENT) ID', 'CREATED ON', and 'CERTIFICATES & SECRETS'. A single row is highlighted with a red box, showing 'ADAL-AdminConsole' as the display name and '774aac82-5988-4b7b-8a7b-56ab28513ad2' as the client ID. The 'CREATED ON' field shows '2018-05-02T06...' and 'current' under 'CERTIFICATES & SECRETS'.

Double click "ADAL-AdminConsole" select "Authentication" and Add the Webchat link and Save

<https://delivery-kit-latest-microservice-webchat.azurewebsites.net>

The screenshot shows the Microsoft Azure portal interface. The left sidebar lists various services. The main content area is titled 'ADAL-AdminConsole - Authentication' under 'Manage'. On the left, a navigation menu includes 'Overview', 'Quickstart', 'Manage', 'Branding', and 'Authentication'. The 'Authentication' option is selected and highlighted with a red box. The right pane shows a list of 'Web' type authentication configurations. One configuration has its 'Redirect URI' field set to '<https://delivery-kit-latest-microservice-webchat.azurewebsites.net>' and is marked with a green checkmark. A 'Save' button is visible at the top right of the configuration pane.

In case, you don't have permission on "DXC Production" subscription, please ask people who can do it and help you add the link

Update 2 variables on Azure DevOp again for Webchat and build Webchat again

TENANT = [CSCPortal.onmicrosoft.com](https://CSCPortal.onmicrosoft.com)

AZURE\_REG\_APPID = 774aac82-5988-4b7b-8a7b-56ab28513ad2

## Step 7: Testing

### Skype for Business

Note:

Account of Skype is dxc domain. Therefore, we need check on Service Now also need exist the same account with Skype

Example: my account of Skype [tbui21@dxc.com](mailto:tbui21@dxc.com) → make sure on Service now also has account [tbui21@dxc.com](mailto:tbui21@dxc.com)

Last message received on 5/29/2019 at 2:10 PM.

## MS Team

Name	Health	Published	Action
Microsoft Teams	Running	--	Edit
Web Chat	Running	--	Edit

## WebChat

Get the link to verify.

### Note:

Webchat is automatic to get account on Azure to chat. Therefore, we need check that on Service Now also need exist the same account with Azure

Example: my account on Azure is tbui21@csc.com → make sure on Service now also has account tbui21@csc.com

The screenshot shows the Microsoft Azure portal interface. The URL in the address bar is <https://portal.azure.com/#@dxcwmmicroservice.onmicrosoft.com/resource/subscriptions/279adfe2-476c-490c-a1cf-be00be890369/resourceGroups/Delivery-Kit-Latest-Microservice-WebChat>. The main page displays the 'Delivery-Kit-Latest-Microservice-WebChat' resource group. The 'Overview' tab is selected, indicated by a red box. On the right side, there is a detailed view of the resource settings, including the URL <https://delivery-kit-latest-microservice-webchat.azurewebsites.net>, App Service Plan 'Delivery-Kit-Latest-Shared-App-Service (B2: 1)', and other deployment details like FTP hostnames.

Access the link <https://delivery-kit-latest-microservice-webchat.azurewebsites.net>

The screenshot shows a web browser displaying the 'WebChat Channel - Welcome' page. The URL in the address bar is <https://delivery-kit-microservice-webchat.azurewebsites.net/#/welcome>. The page has a clean design with a large circular icon containing three stylized human figures. The text 'WebChat Channel - Welcome' is centered above a yellow button with the text 'Click here to login'.

Welcome to Virtual Agent. Please type your question or issues in text box below.

You: hi 11:55 am

**Virtual Agent Alfred:**  
Hi !  
I'm Alfred, DXC Technology's virtual support agent. I hope you are well today!

Please note that what you tell me is logged to various systems to improve me.  
Do not enter sensitive or restricted data (such as passwords or confidential data.)

How can I help you today?

If you need to know more about my capabilities just type: "Tell me more."

Type your question or issue here SEND

## Step 8: Translation Microservice

### 8.1 Configuration Service Now

The Translation Microservice will detect language that declared on Service Now of user and make decision using which language to response. Therefore we need double check language that declared on Service Now

Access <https://cscdemoukpv2.service-now.com>

Select "Profile" and set Language that user using

Example: Language= German

### 8.2 Using existing Translation Microservice

Because Translation Microservice is a product and all customers can use the same information for Translation Microservice.

Update VA Core with below information

```
MS_LANGUAGE_DETECTION={"url": "https://dxc-wm-qa.azure-api.net/shared-language-qa/api/detect", "subKey": "bf0c342f2b3546c0983bc7aab8c259df", "name":"Delivery-Kit-Translation"}  
MS_LANGUAGE_TRANSLATION={"url": "https://dxc-wm-qa.azure-api.net/shared-language-qa/api/translate", "subKey": "bf0c342f2b3546c0983bc7aab8c259df", "name":"Delivery-Kit-Translation"}
```

SNOW\_SERVER\_URL=<https://cscdemoukpv2.service-now.com>

### 8.3 Deploy new Translation Microservice

#### Create provision

wm-microservice

Projects My work items My pull requests

Provision

Microservices Capabilities

All projects

Microservices Capabilities

ms-botframework

The screenshot shows the Azure DevOps Pipelines interface. On the left, there's a sidebar with various project links. In the center, the 'Framework Provision Pipeline' is selected. At the top right, there's a 'Create release' button highlighted with a red box. Below it, a table lists a single release: 'Release-1' (PT), created on '2019-06-06 15:46', with status 'Shared Re...' and 'Translation'.

Select Version as picture and enter the Prefix value. Then click Create button

This screenshot shows the 'Create a new release' dialog for the 'Framework Provision Pipeline'. In the 'Artifacts' section, the 'Source alias' is '\_Provision' and the 'Version' is '743703f0 (Updated manifest.yml)'. In the 'Variables' section, the 'Prefix' is 'Translation-Version23'. At the bottom, there's a 'Create' button highlighted with a red box.

The screenshot shows the 'Framework Provision Pipeline' list again. A new release, 'Release-2' (TB), has been added and is highlighted with a red box. It was created on '2019-06-10 14:24' from branch 'master'. The table also includes 'Release-1' (PT) from '2019-06-06 15:46'.

Check Resource Group, Translation MicroService and App Insights was created on Portal Azure

Resource groups

Subscriptions: 1 of 3 selected – Don't see a subscription? Open Directory + Subscription settings

NAME	SUBSCRIPTION	LOCATION
HDemoWM-Build-Microservice-Translation	DXC Workplace and Mobility Dev	East US
translation-resource-delivery-kit	DXC Workplace and Mobility Dev	Central US
<b>Translation-Version23-Microservice-Translation</b>	<b>DXC Workplace and Mobility Dev</b>	<b>East US</b>
<b>Translation-Version23-Shared-Components</b>	<b>DXC Workplace and Mobility Dev</b>	<b>East US</b>
TranslationLogging	DXC Workplace and Mobility Dev	South Central US
TranslationLogging-Test	DXC Workplace and Mobility Dev	South Central US
vpham20-t2-Microservice-Translation	DXC Workplace and Mobility Dev	East US
vpham-t1-Microservice-Translation	DXC Workplace and Mobility Dev	East US
WM-Build-Microservice-Translation	DXC Workplace and Mobility Dev	East US

## Check App Insights information

Translation-Version23-Shared-Components

Subscription (change)  
DXC Workplace and Mobility Dev

Subscription ID  
279adfe2-476c-490c-a1cf-be00be890369

Tags (change)  
Click here to add tags

2 items  
NAME LOCATION

Translation-Version23-Shared-Component-Application-Insight	East US
Translation-Version23-Shared-Component-App-Service	East US

## Create API Management

Create API Management.

Search API Management and Add new API Management service as below

API Management services

Subscriptions: 1 of 3 selected – Don't see a subscription? Open Directory + Subscription settings

NAME	STATUS	TIER	LOCATION	RESOURCE GROUP	SUBSCRIPTION
Delivery-Kit-Latest-API-Man...	Online	Developer	East US	Delivery-Kit-Latest-Sh...	DXC Workplace and ...
dxc-sandbox1-handover	Online	Developer	Central US	Delivery-Kit-Microser...	DXC Workplace and ...
dxc-wm-dev	Online	Consumption	West US	WM-Shared-Compon...	DXC Workplace and ...
ms-skype-delivery-kit-api	Online	Developer	Central US	msfb-resource-dev	DXC Workplace and ...
sbr-chat-api	Online	Developer	Central US	sbr-resource-dev	DXC Workplace and ...
sbr-sfb-api1	Online	Developer	Central US	sbr-sfb-resource-dev	DXC Workplace and ...
translation-delivery-kit-api	Online	Developer	Central US	translation-resource-...	DXC Workplace and ...
utc-ms-chat-api	Online	Developer	Central US	utc-chat-rg-test	DXC Workplace and ...

Microsoft Azure

API Management service

- Name: translation-version23-api
- Subscription: DXC Workplace and Mobility Dev
- Resource group: Translation-Version23-Shared-Components
- Location: East US
- Organization name: dxc
- Administrator email: tbui21@csc.com

Pricing tier (View full pricing details): Developer (No SLA)

Create Automation options

Microsoft Azure

API Management service

- Organization name: dxc
- Administrator email: tbui21@csc.com

The developer tier of API Management doesn't include SLA and shouldn't be used for production purposes. Your service may experience intermittent outages, for example during upgrades. Learn more about API Management service tiers.

Enable Application Insights

Application Insights instance: Translation-Version23-Shared-Component-Application-Insight

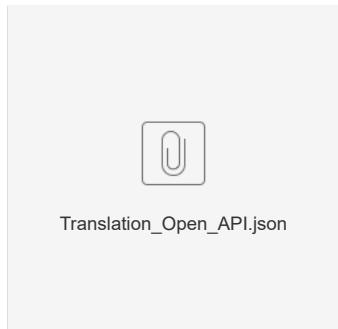
Create Automation options

## Open API.

The template is copied the content from the link as below

<https://github.dxc.com/WM-Microservice-Framework/ms-translation/blob/develop/apim/swagger.yml>

If you can not access the link. Please using attached file



Select "Select a file" and choose Translation\_Open\_API.json file.

Change the "Display name"

Click "Create"

The screenshot shows the Microsoft Azure API Management service interface. On the left, the navigation bar includes 'Home', 'Overview', 'Activity log', 'Access control (IAM)', 'Tags', 'Diagnose and solve problems', 'API Management', 'Quickstart', 'APIs', 'Products', 'Named values', 'Tags', 'Analytics', 'Users', and 'Subscriptions'. The 'APIs' option is selected. In the center, a modal window titled 'translation-version23-api - APIs' is open, showing the 'Publisher portal' tab. A red box highlights the 'Name' input field, which contains 'translation23-azurewebsites.net'. Other fields visible include 'Display name' (also 'translation23-azurewebsites.net'), 'API URL suffix' (e.g. httpbin), and 'Products' (Unlimited). The 'Create' button is at the bottom right.

## Update VA Core and rebuild VA Core

The screenshot shows the Microsoft Azure API Management service interface. The 'APIs' section on the left is selected. In the center, the 'translation-version23-api' is selected. The 'Design' tab is active, showing 'REVISION 1' (UPDATED Jul 24, 2019, 10:23:46 AM). A red box highlights the 'translate' endpoint under the 'POST' method. The 'Request URL' field shows 'https://translation-version23-api.azure-api.net/api/translate'. Below it, the 'HTTP request' section shows a POST request to the same URL with headers: Host: translation-version23-api.azure-api.net, Ocp-Apim-Subscription-Key: (redacted), and Ocp-Apim-Trace: true. A 'Send' button is at the bottom.

The screenshot shows the Microsoft Azure API Management service interface. The 'APIs' section on the left is selected. In the center, the 'translation-version23-api' is selected. The 'Design' tab is active, showing 'REVISION 1' (UPDATED Jul 24, 2019, 10:23:46 AM). A red box highlights the 'detect' endpoint under the 'POST' method. The 'Request URL' field shows 'https://translation-version23-api.azure-api.net/api/detect'. Below it, the 'HTTP request' section shows a POST request to the same URL with headers: Host: translation-version23-api.azure-api.net, Ocp-Apim-Subscription-Key: (redacted), and Ocp-Apim-Trace: true. A 'Send' button is at the bottom.

Get Subscription

<https://confluence.csc.com/display/PD/VA+2.3+-+02+Deployment+Microservice>

The screenshot shows the Microsoft Azure API Management Subscriptions page. The left sidebar includes options like Home, Activity log, Access control (IAM), Tags, Diagnose and solve problems, API Management (Quickstart, APIs, Products, Named values, Tags, Analytics, Users), and Subscriptions. The Subscriptions option is highlighted with a red box. The main content area displays a table of subscriptions:

DISPLAY NAME	PRIMARY KEY	SECONDARY KEY	SCOPE	STATE	OWNER	ALLOW TRACING
076dd3d8...	[REDACTED]	[REDACTED]	Product: Starter	Active	Administrator	<input checked="" type="checkbox"/>
62ebf612...	[REDACTED]	[REDACTED]	Product: Unlimited	Active	Administrator	<input checked="" type="checkbox"/>
Built-in all-access S...	[REDACTED]	[REDACTED]	Service	Active	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

## Get App Insight Key

The screenshot shows the Microsoft Azure Application Insights API Access page. The left sidebar includes Create a resource, Home, Dashboard, All services, Favorites (All resources, Resource groups, App Services, SQL databases, Azure Cosmos DB, Virtual machines, Load balancers, Storage accounts, Virtual networks, Azure Active Directory). The API Access option under Configuration is highlighted with a red box. The main content area shows the Application ID field, which is highlighted with a red box and contains the value: c442be04-52a4-499e-a50d-cf7b0ac01dfe.

## Create API Key

The screenshot shows the Microsoft Azure Create API key page. The left sidebar includes Create a resource, Home, Dashboard, All services, Favorites (All resources, Resource groups, App Services, SQL databases, Azure Cosmos DB, Virtual machines, Load balancers, Storage accounts, Virtual networks, Azure Active Directory). The Create API key option under Configuration is highlighted with a red box. The main content area shows a form for creating an API key:

- Description: translation api key
- Permissions (checkboxes):
  - Read telemetry
  - Write annotations
  - Authenticate SDK control channel

A "Generate key" button is at the bottom.

Create an API key to read Application Insights data.

API keys are used by applications outside the browser to access this resource. Your API keys should be managed like passwords. Keep them secret.

**Key:** oy5ei3a487b5mpvh8f8wg5fzzyrgh8v87yeygek

Make sure you copy this key now. We don't store it, and after you close this blade you won't be able to see it again.

```
MS_LANGUAGE_DETECTION={"url": "https://translation-version23-api.azure-api.net/api/detect", "subKey": "076dd3d82b7f47bd8a87413a171491d3", "name": "Delivery-Kit-Translation"}  
MS_LANGUAGE_TRANSLATION={"url": "https://translation-version23-api.azure-api.net/api/translate", "subKey": "076dd3d82b7f47bd8a87413a171491d3", "name": "Delivery-Kit-Translation"}
```

SNOW\_SERVER\_URL=https://cscdemoukpv2.service-now.com

App Insight = c442be04-52a4-499e-a50d-cf7b0ac01dfe

Access key = oy5ei3a487b5mpvh8f8wg5fzzyrgh8v87yeygek

## Create Capacity

Access dev tool and create new Folder on Framework

+ New

Name

All pipelines

- Build
- Delivery
- Framework**
- Development
- Production
- Staging
- Testing

Create new folder

Path: \Framework

Name: Delivery-Kit

OK

Release pipeline	Created	Stages
Production Test Mic...	2019-05-17 14:40	MS
Production Develop...	2019-05-17 14:39	MS
Production Test Mic...	2019-05-17 14:37	MS
Production Develop...	2019-05-17 14:35	MS
WM-Prod-Development-Release-MS-Translation-...	2019-05-17 13:40	MS
Test-Release-MS-Translation-2	2019-05-16 14:05	MS
Test-Release-MS-Translation-1	2019-05-16 13:55	MS

Download the template json file to import to new folder

Development Microservice Translat...

Search all pipelines

**Delivery-Kit**

Releases

Name + New

All Import release pipeline

Delivery

Framework

Delivery-Kit

New release pipeline

New folder

Development

Production

Staging

Testing

No releases found

You can create a new release manually or [setup triggers](#) to create it automatically

Select json file and create

All pipelines > Development Microservice Translat...

Save Create release View releases

Pipeline Tasks Variables Retention Options History

Artifacts | Add

Stages | Add

MS Translation 1 job, 3 tasks

Schedule not set

Select "Disabled" Post deployment and Gate

All pipelines > Development Microservice Translat...

Save Create release View releases

**Pipeline** Tasks Variables Retention Options History

Artifacts | Add

Stages | Add

MS Translation 1 job, 3 tasks

Post-deployment conditions

MS Translation

Post-deployment approvals

Select the users who can approve or reject deployments to this stage

Disabled

Gates

Define gates to evaluate after the deployment. [Learn more](#)

Disabled

Click on MS Translation → 3 tasks

Update all data of MS Translation, Agent job, Enable Continuous Monitoring.....Then click Save

[https://dev.azure.com/wm-microservice/Microservices%20Capabilities/\\_releaseDefinition?definitionId=35&\\_a=definition-tasks](https://dev.azure.com/wm-microservice/Microservices%20Capabilities/_releaseDefinition?definitionId=35&_a=definition-tasks)

The screenshot shows the Azure DevOps interface for a deployment pipeline named 'MS Translation'. The 'Agent job' stage is selected. A red box highlights the configuration for this stage, which includes:

- Stage name: MS Translation
- Parameters:
  - Azure subscription: DXC Workplace and Mobility Dev (279adfe2-476c-490c-...)
  - App type: Function App for Containers (Linux)
  - App Service name: Translation-Version23-Language-Translate
  - Registry or Namespace: virtualagent.azurecr.io
- Repository: wwm-microservice-framework/ms-translation

[https://dev.azure.com/wm-microservice/Microservices%20Capabilities/\\_releaseDefinition?definitionId=35&\\_a=definition-tasks](https://dev.azure.com/wm-microservice/Microservices%20Capabilities/_releaseDefinition?definitionId=35&_a=definition-tasks)

The screenshot shows the Azure DevOps interface for a deployment pipeline named 'MS Translation'. The 'Azure App Service Deploy' stage is selected. A red box highlights the configuration for this stage, which includes:

- App Service name: Translation-Version23-Language-Translate
- Registry or Namespace: virtualagent.azurecr.io
- Repository: wwm-microservice-framework/ms-translation
- Resource Group name for Application Insights: Translation-Version23-Shared-Components
- Application Insights resource name: Translation-Version23-Shared-Component-A

[https://dev.azure.com/wm-microservice/Microservices%20Capabilities/\\_releaseDefinition?path=%5Cframework%5CDelivery-Kit&source=ReleaseImport&environmentId=35&\\_a=definition-tasks](https://dev.azure.com/wm-microservice/Microservices%20Capabilities/_releaseDefinition?path=%5Cframework%5CDelivery-Kit&source=ReleaseImport&environmentId=35&_a=definition-tasks)

The screenshot shows the Azure DevOps interface for a deployment pipeline named 'MS Translation'. The 'Agent job' stage is selected. A red box highlights the configuration for this stage, which includes:

- Display name: Agent job
- Agent selection:
  - Agent pool: Hosted Ubuntu 1604

Click Save data

## Add Variables

The screenshot shows the Azure DevOps Library page. On the left, there's a sidebar with icons for MC, Pipelines, Builds, Releases, and Library. The 'Library' icon is highlighted with a red box. Below the sidebar, there are tabs for 'Variable groups' (selected), 'Secure files', and '+ Variable group'. A search bar at the top right says 'Search variable groups'. The main area lists variable groups with columns for Name, Date modified, and Modified by. One entry, 'Library', is highlighted with a red box.

Name	Date modified	Modified by
Build MS Botframework Demo Dev	4/17/2019	Pham, Vinh Tuan
Build MS Botframework Demo Prod	4/17/2019	Pham, Vinh Tuan
Build MS Botframework Demo Test	4/17/2019	Pham, Vinh Tuan
<b>Library</b>	4/17/2019	Pham, Vinh Tuan
Builds	4/17/2019	Pham, Vinh Tuan
Releases	dbox	Pham, Vinh Tuan
Task groups	dbox1	Pham, Vinh Tuan
Deployment groups	dbox2	Pham, Vinh Tuan
Build MS Botframework Test	4/17/2019	Pham, Vinh Tuan
Build MS Botframework UAT	5/2/2019	Bang Cong Nguyen
	6/4/2019	Duong Quy Ly

The screenshot shows the 'Framework-Delivery-Kit-Translation' variable group configuration page. The 'Variables' section is highlighted with a red box. It contains a table with columns for 'Name' and 'Value'. The table rows show variable definitions for IBM and Microsoft language detection services.

Name	Value
LANGUAGE_DETECTION_LANGUAGE_CODE_SY...	[{"key": "zh-CN", "values": [{"zh": "zh-Hans"}]}, {"key": "zh-TW", "values": [{"zh-...
LANGUAGE_DETECTION_SERVICE_IBM	{"detection_vendor": "IBM", "supported_language_code": "ALL", "user_n...
LANGUAGE_DETECTION_SERVICE_MS	{"detection_vendor": "Microsoft", "supported_language_code": "ALL", "s...
LANGUAGE_TRANSLATION_LANGUAGE_CODE	{"translation_vendor": "Microsoft", "synonyms": [{"key": "zh-Hans", "values": [{"zh": "zh-..."}]}]

Key	Description
Key	Description
LANGUAGE_DETECTION_SERVICE_IBM	<p>On Step 3.2 03DeploymentLanguageTranslateMicroservice-Step3:Registerlanguage translation service from vendor we can get username and password encodes of IBM Translation service password: "pNzaUwCo__t61x30fSsLs5U191njZbGK_4wB0ki_1xD" username: "apikey"</p> <pre>{"detection_vendor": "IBM", "supported_language_code": "ALL", "user_name": "apikey", "password": "pNzaUwCo__t61x30fSsLs5U191njZbGK_4wB0ki_1xD", "confidence_min": 0.01}</pre>
LANGUAGE_DETECTION_SERVICE_MS	<p>On Step 3.1 03DeploymentLanguageTranslateMicroservice-Step3:Registerlanguage translation service from vendors(IBM,Microsoft) We can get subscription_key and input it on <b>subscription_key</b></p> <pre>{"detection_vendor": "Microsoft", "supported_language_code": "ALL", "subscription_key": "7444a2817fa045b1a44178dcdb68657"}</pre>
TRANSLATION_SERVICE_MS	<p>On Step 3.1 03DeploymentLanguageTranslateMicroservice-Step3:Registerlanguage translation service from vendors(IBM,Microsoft) We can get subscription_key and input it on <b>subscription_key</b></p> <pre>{"translation_vendor": "Microsoft Neural Machine", "subscription_key": "7444a2817fa045b1a44178dcdb68657"}</pre>

Key	Description
TRANSLATION_SERVICE_IBM	<p>On Step 3.2 03DeploymentLanguageTranslateMicroservice-Step3:Registerlanguagetranslationservicefromvendors(IBM,Microsoft)</p> <p>We can get username and password encodes of IBM Translation service</p> <pre>{"translation_vendor": "IBM Neural Machine", "user_name": "apikey", "password": "pNzaUwCo__t61x30fSsLs5U"}</pre>
LANGUAGE_DETECTION_LANGUAGE_CODE_SYNONYMS	<p>The list of synonyms of language code. This is to standardize the language code from handling multiple detection vendors.</p> <p>Whenever the detection vendor returns a language code which exists in the "values" array, then the Microservice will return the "detected language".</p> <p>If not, then return the detected language returned from detection vendor.</p> <p>Can use default as below</p> <pre>[{"key": "zh-CN", "values": ["zh", "zh-Hans"]}, {"key": "zh-TW", "values": ["zh-Hant"]}]}}</pre>
LANGUAGE_TRANSLATION_LANGUAGE_CODE_SYNONYMS	<p>The list of synonyms of language code. This is to standardize the language code from handling multiple translation vendors.</p> <p>Whenever the input params passed into the Microservice have vendor name and language code exist in the "value" array, then the Microservice will consider "key" value is the final language code, and use this language code paste to translation vendor.</p> <p>If not, then use the original language code from parameters and paste it to translation vendor.</p> <p>Can use default as below</p> <pre>[{"vendor": "Microsoft", "synonyms": [{"key": "zh-Hans", "values": ["zh", "zh-CN"]}, {"key": "zh-Hant", "values": ["zh-TW"]}]}, {"vendor": "IBM", "synonyms": [{"key": "zh", "values": ["zh-Hans", "zh-CN"]}, {"key": "zh-TW", "values": ["zh-Hant"]}]}]}</pre>

## Link variables

The screenshot shows the 'Variables' tab in the Azure DevOps Pipelines interface. On the left, there's a sidebar with icons for Overview, Boards, Repos, Pipelines, Builds, Releases, Library, Task groups, Deployment groups, Test Plans, and Project settings. The main area shows a list of variable groups: 'Framework MS Translation Production Development (6)', 'Framework MS Translation Production Staging (6)', 'Framework MS Translation QA (6)', 'Framework MS Translation Test (6)', and 'Framework-Delivery-Kit-Translation (6)'. The last item is highlighted with a red box. Below the list, there's a section for 'Variable group scope' with radio buttons for 'Release' (selected) and 'Stages', and a dropdown menu labeled 'Select'. At the bottom right is a large blue 'Link' button.

Need Link Docker variables as below

This screenshot shows the same 'Variables' tab in the Azure DevOps Pipelines interface as the previous one. The sidebar and variable group list are identical. However, the 'Docker Variables (3)' group is now highlighted with a red box. Below the list, the 'Link variable group' button is also highlighted with a red box.

## Deployment

The screenshot shows the 'Releases' tab in the Azure DevOps Pipelines interface. The sidebar includes icons for Search all pipelines, New, Overview, Boards, Repos, Pipelines, Builds, Releases, Library, Task groups, Deployment groups, Test Plans, and Project settings. The main area displays the 'Development Microservice Translation Pipeline - Copy' pipeline. It features a cartoon illustration of a person launching a rocket. Below the illustration, it says 'No releases found'. A note at the bottom says 'You can create a new release manually or setup triggers to create it automatically'. At the top right of the pipeline card, there's a 'Create release' button highlighted with a red box.

The screenshot shows the 'Create a new release' dialog for the 'Development Microservice Translation Pipeline - Copy'. The 'Artifacts' section is highlighted with a red box, showing the source alias '-wm-microservice-framework\_ms...' and the selected version 'latest'.

The screenshot shows the pipeline details for 'Development-Release-MS-Translation-1'. The 'Stages' section displays the 'MS Translation' stage, which is currently in progress. The stage details show 4/4 tasks completed.

## 8.4 Verify Translation on Skype, MS Team, WebChat....

Because language response depend on Profile of Service Now. So if we define Language is "German". We must chat by German. Otherwise, system won't response.

## Step 9: Custom Build

Some customer need different version and it is specific for this customer. We need use specific code so we need custom build and link to pipeline

### Create custom build

Access Home on Dev Azure tool

The screenshot shows the Azure DevOps 'Projects' page. On the left, there's a sidebar with 'My organizations' and a selected project 'wm-microservice'. Below it are sections for 'What's new', 'Sprint 153 release notes', and 'New organization'. The main area lists projects under 'ms-botframework': 'ms-chat' (selected and highlighted with a red box), 'ms-s4b', 'ms-translation', 'ms-webchat', and 'Provision'. A search bar at the top right says 'Search' and 'Filter projects'.

Select ms-chat. Then select Pipeline → Build

The screenshot shows the Azure DevOps 'Pipelines / Builds' page for the 'ms-chat' project. The left sidebar has 'Pipelines' and 'Builds' selected (highlighted with red boxes). The main area shows two pipelines: 'delivery-kit-container' (selected and highlighted with a red box) and 'container-CL'. The 'delivery-kit-container' pipeline has a commit history entry for a merge pull request from 'WM-Microservice-Framework' to 'master' on '20190619.1'. A 'Queue' button is visible above the pipeline list.

Click on Folder icon and create new folder

The screenshot shows the same 'Pipelines / Builds' page for 'ms-chat'. In the 'Builds' section, there's a 'New build pipeline' dialog open. The 'Name' field is empty, and the 'Import a pipeline' dropdown is open, showing 'Folder' selected (highlighted with a red box). Other options like 'New build pipeline' and 'Import a pipeline' are also visible.

A screenshot of the Azure DevOps Pipelines interface. On the left sidebar, 'Pipelines' is selected. In the main area, there's a 'Delivery-Kit' folder under 'All build pipelines'. A modal dialog box titled 'New folder' is open, showing the text 'Delivery-Kit' in a input field. At the bottom of the dialog are 'Cancel' and 'Create' buttons.

Clone the container-CI and save to new folder what we have just created

A screenshot of the Azure DevOps Pipelines interface. The 'container-CI' pipeline is selected. On the right, a context menu is open with several options: 'Edit', 'Queue' (which is highlighted with a red box), '...', 'Clone' (also highlighted with a red box), 'Save as a template', 'Export', 'Status badge', and 'Delete'.

Change the name and save to new folder

A screenshot of the Azure DevOps Pipelines interface showing a cloned pipeline named 'Delivery-Kit-Test-container-CI'. The 'Name' field in the pipeline configuration is highlighted with a red box and contains the text 'Delivery-Kit-Test-container-CI'. Other visible pipeline details include 'Get sources' (from 'WM-Microservice-Framework/ms-chat' branch 'master'), 'Agent job 1' (run on agent 'Hosted Ubuntu 1604'), and a 'buildAndPush' task.

The screenshot shows the Azure DevOps Pipelines interface for a project named 'ms-chat'. On the left, there's a sidebar with various navigation options like Overview, Boards, Pipelines, Builds, Releases, Library, Task groups, Deployment groups, Test Plans, Artifacts, and Project settings. The main area displays a build pipeline named 'delivery-kit-test-CI'. The pipeline consists of two tasks: 'Get sources' (using WM-Microservice-Framework/ms-chat as the source) and 'Agent job 1' (which runs a 'buildAndPush' Docker task). At the top right of the pipeline editor, there are several buttons: 'Save & queue', 'Save', 'Save as draft', 'Discard', 'Summary', 'Queue', and more. A red box highlights the 'Save' button.

This screenshot shows a 'Save build pipeline' dialog box overlaid on the Azure DevOps interface. The dialog has fields for 'Select folder' (set to '/Delivery-Kit-Test') and 'Comment' (set to 'create build pipeline'). It includes 'Save' and 'Cancel' buttons. A red box highlights the entire dialog box.

After saving to our folder. Need edit information and link to Github where contain source code

The screenshot shows the Azure DevOps Builds interface. The left sidebar is identical to the Pipelines interface. The main area lists pipelines under the 'Builds' tab. One pipeline is visible: 'delivery-kit-test-container-CI'. To its right are 'Edit', 'Queue', and more options buttons. A red box highlights the 'Edit' button.

Select Get source --> Git hub Enterprise Server → select

The screenshot shows the Azure DevOps Pipelines interface. A pipeline named 'Delivery-Kit-Test' is selected. The 'Get sources' task, which connects to 'VADelivery/DK2-CHAT-TEST' on master, is highlighted with a red box. An 'Agent job 1' step labeled 'buildAndPush' using Docker is also visible. On the right, a list of service connections includes 'GitHub Enterprise Server', which is also highlighted with a red box. Below it, a note says 'Authorized using connection GitHub Delivery Kit'. The repository 'VADelivery/DK2-CHAT-TEST' is listed under 'Manage on GitHub Enterprise Server'.

The screenshot shows the 'Project Settings' page under 'Service connections'. The 'New service connection' button is highlighted with a red box. A dropdown menu lists several options: 'Delivery-Kit-Connection', 'DXC Workplace and Mobility Dev...', 'GitHub Enterprise', 'My Workstyle', and 'Virtual Agent Registry'. The 'GitHub Enterprise' option is also highlighted with a red box.

The screenshot shows the 'Project Settings' page under 'Service connections'. The 'New service connection' button is highlighted with a red box. A dropdown menu lists various service types: Docker Registry, Generic, GitHub, GitHub Enterprise Server, Jenkins, Jira, Kubernetes, npm, NuGet, and Other Git. The 'GitHub Enterprise Server' option is highlighted with a red box.

Enter the connection name and enter personal access token. Then click OK

The screenshot shows the 'Service connections' dialog for a GitHub Enterprise Server connection named 'Github Delivery Kit'. The 'Personal Access Token' option is selected. The 'Server URL' is set to 'https://github.dxc.com'. A 'Personal access token' field contains a redacted value. A 'Verify connection' button is visible at the bottom right.

Check connection was created

The screenshot shows the 'Service connections' section under 'Project Settings'. A new connection named 'Github Delivery Kit' is listed, highlighted with a red box. Other options like 'Delivery-Kit-Connection' and 'DXC Workplace and Mobility De...' are also shown.

Come back DEV Azure and select "Change" and select connection has just created

The screenshot shows a build pipeline named 'Delivery-Kit-Test' with a single step 'Get sources' from 'VADelivery/DK2-CHAT-TEST' master branch. The 'Repository' dropdown shows 'GitHubEnterprise' and 'Github Delivery Kit', with 'Github Delivery Kit' highlighted with a red box. A tooltip indicates the connection is authorized.

Click on "..." button

... > Delivery-Kit-Test > delivery-kit-test-container-CI

Tasks Variables Triggers Options Retention History Save & queue Discard Summary Queue ...

Pipeline Build pipeline

Get sources VADelivery/DK2-CHAT-TEST master

Agent job 1 Run on agent

buildAndPush Docker

Subversion Bitbucket Cloud Other Git

Authorized using connection: Github Delivery Kit Change

Repository \* | Manage on GitHub Enterprise Server

VADelivery/DK2-CHAT-TEST

Default branch for manual and scheduled builds \* master

Clean false

Select source code branch we need build

Select a repository

Showing all repositories Filter repositories...

- VADelivery/DK2-BOT-AUTO
- VADelivery/DK2-BOT-DEMO
- VADelivery/DK2-BOT-TEST
- VADelivery/DK2-CHAT
- VADelivery/DK2-CHAT-AUTO
- VADelivery/DK2-CHAT-DEMO
- VADelivery/DK2-CHAT-QC
- VADelivery/DK2-CHAT-TEST**
- VADelivery/DK2-SFB
- VADelivery/DK2-SFB-AUTO
- VADelivery/DK2-SFB-DEMO

Select Cancel

Click "Save" button to save information. Then select "buildAndPush" and click "New" button

... > Delivery-Kit-Test > delivery-kit-test-container-CI

Tasks Variables Triggers Options Retention History Save & queue Discard Summary Queue ...

Pipeline Build pipeline

Get sources WM-Microservice-Framework/ms-chat master

Agent job 1 Run on agent

buildAndPush Docker

Display name \* buildAndPush

Container Repository ^

Container registry | Manage

Virtual Agent Registry + New

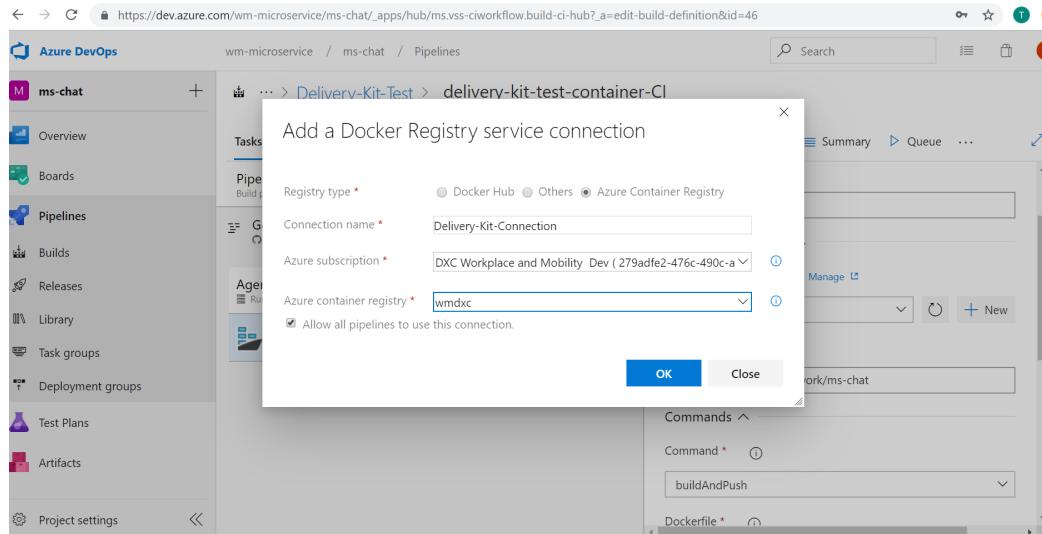
Container repository

wm-microservice-framework/ms-chat

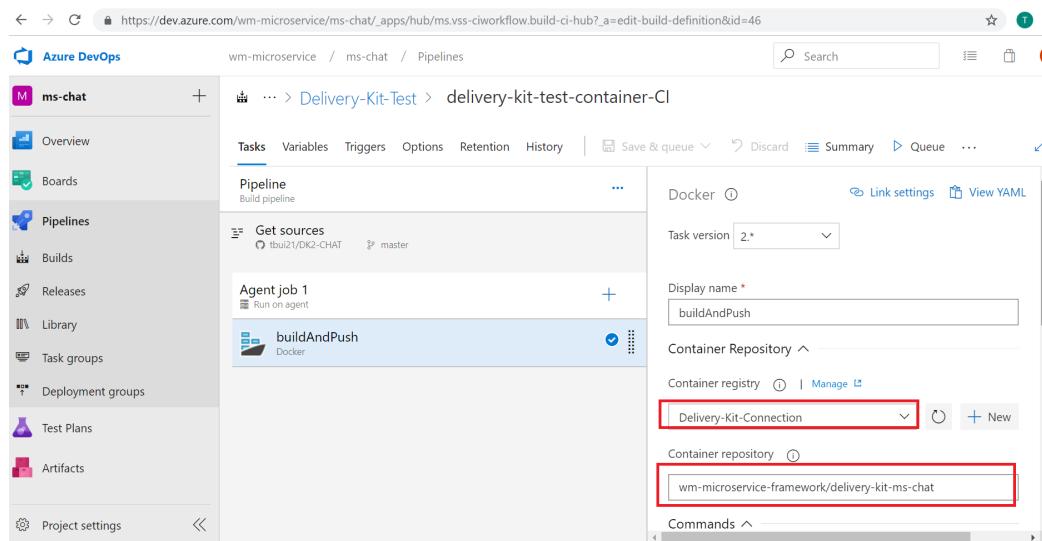
Commands ^

Command \* buildAndPush

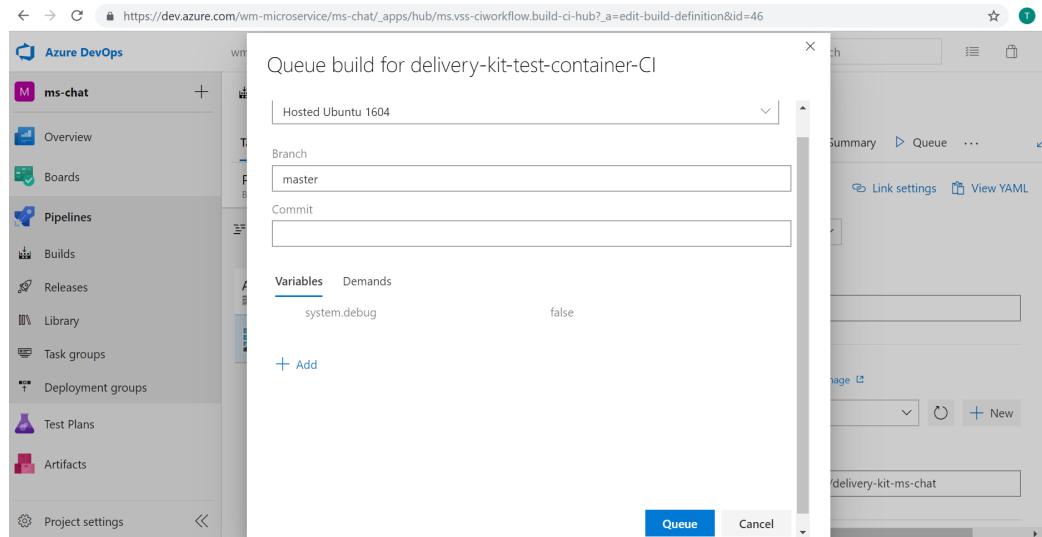
Dockerfile \*



Change the "Container repository" to your code on github such as VADelivery/DK2-CHAT-AUTO



After all information was entered. Then click "Queue" build



The screenshot shows the Azure DevOps Pipelines interface for the 'ms-chat' project. A green notification bar at the top states 'Build #20190624.1 has been queued.' Below this, the pipeline configuration is visible, featuring a 'Get sources' step and an 'Agent job 1' step containing a 'buildAndPush' task. The task is set to run on a Docker agent.

Click on the Build link to see the build process

The screenshot shows the build results for build #20190624.1. The summary indicates it was manually run by Thoan Bui. The build started at 6/24/2019, 11:15:36 AM and completed in 1m 15s. The 'Agent job 1' step details three successful tasks: 'Initialize job', 'Download secrets: vmkeyvault', and 'Checkout'. The final task, 'buildAndPush', is shown as 'Waiting for console output...'.

## Link to Capability

Open Capabilities → select Release → select pipeline we created → click Edit

The screenshot shows the Azure DevOps Releases page for the 'Microservices Capabilities' release. It lists several releases, including 'Build-Development-Release-13', 'Build-Development-Release-12', 'Build-Development-Release-11', 'Build-Development-Release-10', 'Build-Development-Release-9', and 'Build-Development-Release-8'. The 'Development Microservices Pipeline' is highlighted with a red box. The 'Edit' button for this pipeline is also highlighted with a red box.

Copy the name of MS Chat Artifact , then delete exist Artifact of MS Chat

<https://confluence.csc.com/display/PD/VA+2.3+-+02+Deployment+Microservice>

All pipelines > Development Microservices ...

Pipeline Tasks Variables Retention Options History

Artifacts | + Add Stages | + Add

ms-chat

MS Chat 1 job, 3 tasks

MS S4B 1 job, 3 tasks

Copy the name and paste to Notepad

All pipelines > Development Microservices ...

Pipeline Tasks Variables Retention Options History

Virtual Agent Container

Resource Group \* ⓘ VirtualAgent

Azure Container Registry \* ⓘ virtualagent

Repository \* ⓘ wwm-microservice-framework/ms-chat

Default version \* ⓘ Latest

Source alias \* ⓘ ms-chat

Delete current MS Chat Artifact

All pipelines > Development Microservices ...

Pipeline Tasks Variables Retention Options History

Artifact

AzureContainerRepository - ms-chat

Delete

Service connection | Manage ⓘ

Virtual Agent Container

Resource Group \* ⓘ VirtualAgent

Azure Container Registry \* ⓘ virtualagent

Repository \* ⓘ wwm-microservice-framework/ms-chat

After delete, now we create new Artifact for MS Chat

The screenshot shows the Azure DevOps Pipelines interface for a project named 'wm-microservice'. The left sidebar is visible with various options like Overview, Boards, Repos, Pipelines, Builds, Releases, Library, Task groups, Deployment groups, Test Plans, and Project settings. The main area displays a pipeline named 'Development Microservices ...' with two stages: 'MS Chat' (1 job, 3 tasks) and 'MS S4B' (1 job, 3 tasks). On the left, under 'Artifacts', there is a list with 'ms-chat' and 'ms-bot-framework', and a red box highlights the '+ Add' button.

On Add Artifact screen, select Azure Container and enter information

This screenshot shows the 'Add an artifact' screen within the Azure DevOps Pipelines interface. The left sidebar shows the same project structure as the previous screenshot. The main area has tabs for Pipeline, Tasks, Variables, Retention, Options, and Artifacts. Under 'Artifacts', there is a list with 'ms-s4b', 'webchat', and 'Add an artifact'. On the right, there are several artifact types: Build, Azure Repos ..., GitHub, TFVC, Azure Artifacts, GitHub Release ..., Docker Hub, and Jenkins. A red box highlights the 'Azure Container...' option. Below these options, there are dropdowns for Service connection (set to 'DXC Workplace and Mobility Dev'), Resource Group (set to 'WM-Shared-Components'), and Azure Container Registry (set to 'wmdxc').

This screenshot shows the 'Add an artifact' screen with more detailed configuration. The 'Service connection' dropdown is set to 'DXC Workplace and Mobility Dev'. The 'Resource Group' dropdown is set to 'WM-Shared-Components'. The 'Azure Container Registry' dropdown is set to 'wmdxc'. The 'Repository' dropdown contains the value 'wm-microservice-framework/delivery-kit-ms-chat'. The 'Default version' dropdown is set to 'Latest'. The 'Source alias' dropdown contains the value '\_wm-microservice-framework\_delivery-kit-ms-chat'. A red box highlights the entire configuration area.

We must delete "ms-chat" artifact what already exist. Then change new Artifact have just created the same name with old artifact

The screenshot shows the Azure DevOps Pipelines interface for a 'Development Microservices' pipeline. On the left, the 'Pipelines' section is selected. In the main area, there's a step labeled 'Artifact' which includes a 'Service connection' (DXC Workplace and Mobility Dev), a 'Resource Group' (WM-Shared-Components), an 'Azure Container Registry' (wmdbc), and a 'Repository' (wm-microservice-framework/delivery-kit-ms-chat). The 'Repository' field is highlighted with a red box.

Change the name to "ms-chat" and click "Save"

The screenshot shows the same Azure DevOps Pipelines interface after saving the changes. The 'Repository' field now contains 'wm-microservice-framework/delivery-kit-ms-chat'. Below it, a 'Default version' dropdown is set to 'Latest' and a 'Source alias' field is present, both of which are highlighted with red boxes.

After Save data, need click "Create release" to rebuild all microservice again.

Follow the same step for Skype, MS Team, Translation, WebChat...

## Step 10: Trouble Shoot

Permission is error and you can see as below

No labels

### 4 Comments

Joel Louie Miranda  
Under...

#### 3.3 Enter information for all Microservice

Select "Tasks" and need input all information which red color

For the "Application Service Name", kindly ensure you change it using the one that was deployed under "Provisions".



Bohol Jr., Arsenio D.

There are steps on how to configure **MS Team – 6.2 Configuration for MS Team on Azure**

and **Web Chat – 6.3 Configuration for Web Chat on Azure**

But I can't seem to find the steps on how to configure **S4B** after deployment



Joel Louie Miranda

Under

- **6.2 Configuration for MS Team on Azure**

Based on the last setup we did for MS Teams, there seems to be missing a step for creating a channel and bot registration service. I tried to create, but experiencing access denied. Already coordinating with Vinh on the access issue.

Once this works, will proceed on testing directly with MS Teams.



Joel Louie Miranda

Hi,

Here are some recollections from today.

- Another step that's probably missing is the ITSM/Service Now account creation. If your account is not yet created on Service Now, the bot will not answer you back.
- Also encountered issue on the correct Cloudant credentials, all that is needed is a re-verification of the credentials.
- There was a new AZU\_API and AZU\_KEY that was provided by Vinh, it might be related to someone using it.

Once those have been redeployed on the ms-botframework, we tried on teams and I can now talk to Alfred.

