

Bot Web App

Author: Tyler Lu

Contents

[1. LUIS App 2](#_Toc498020264)

[1.1. Import LUIS App 2](#_Toc498020265)

[1.2. Set application as public 3](#_Toc498020266)

[1.3. Train and Publish the App 3](#_Toc498020267)

[1.4. Creating Subscription Keys on Azure 4](#_Toc498020268)

[2. Bot 6](#_Toc498020269)

[2.1. Create a Bot 6](#_Toc498020270)

[2.2. Customize and Configure the Bot 7](#_Toc498020271)

[2.3. Add Microsoft Teams Channel 11](#_Toc498020272)

[3. Bot Web App 14](#_Toc498020273)

[3.1. Create App registrations in AAD 14](#_Toc498020274)

[3.2. Create a Web App on Azure 15](#_Toc498020275)

[4. Follow-up steps 16](#_Toc498020276)

[4.1. Add Reply URL to the App Registration 16](#_Toc498020277)

[4.2. Admin consent 16](#_Toc498020278)

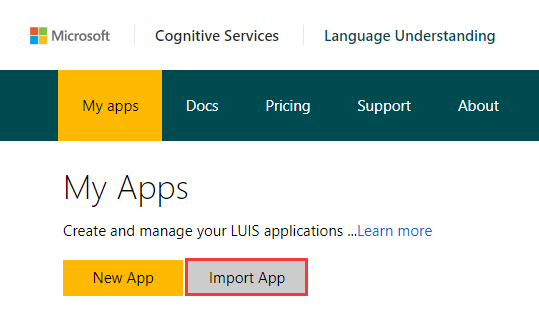
[4.3. Update Bot Messaging Endpoint 17](#_Toc498020279)

[5. Verify the Bot 18](#_Toc498020280)

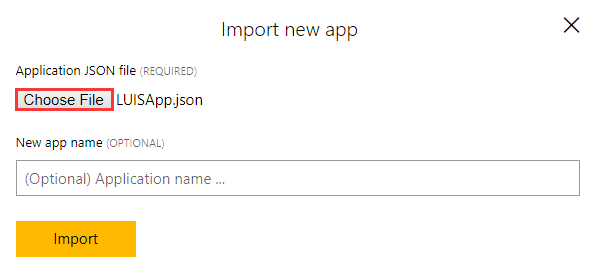
# LUIS App

## Import LUIS App

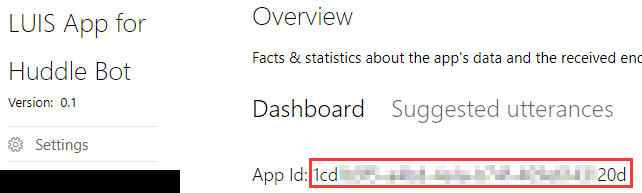
1. Open <https://www.luis.ai/>, and sign in.
2. Finish or skip the welcome page. Then go to the applications page:



1. Click **Import App**



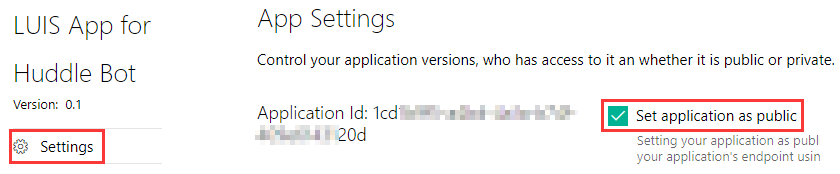
* + Click **Choose File**, and *select /Files/* *LUISApp.json*.
  + Click **Import**.



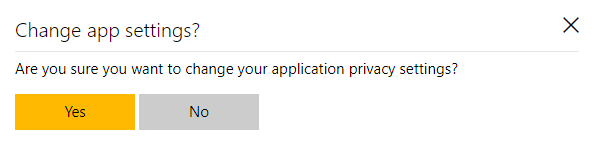
1. Copy aside the **App Id**. It will be used as the value of **LuisAppId** App setting.

## Set application as public

1. Click **Settings**.



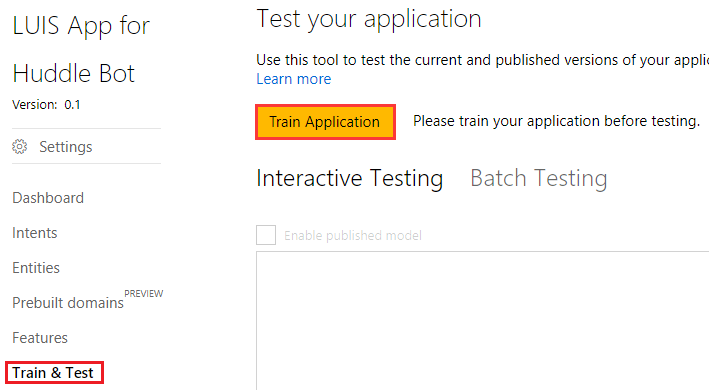
1. Check **Set application as public**.



1. Click **Yes**.

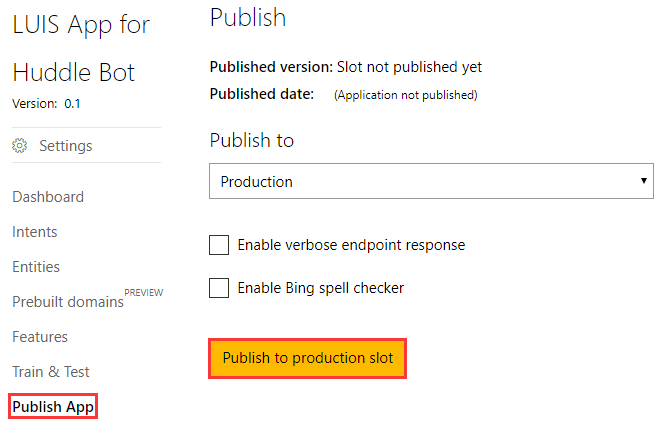
## Train and Publish the App

1. Click **Train & Test** at the left.



Click **Train Application**.

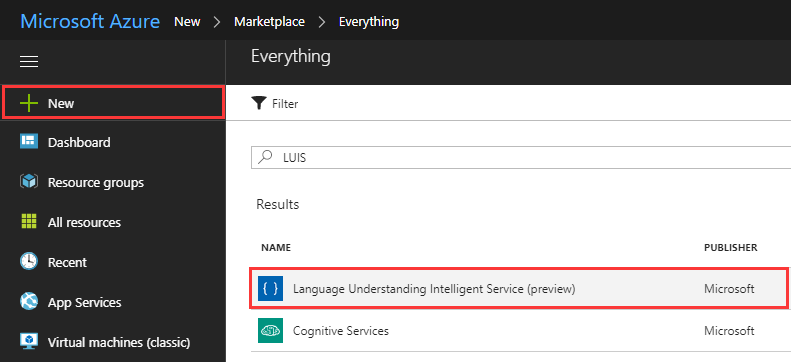
1. Click **Publish App** at the left.



Click **Publish to production slot**.

## Creating Subscription Keys on Azure

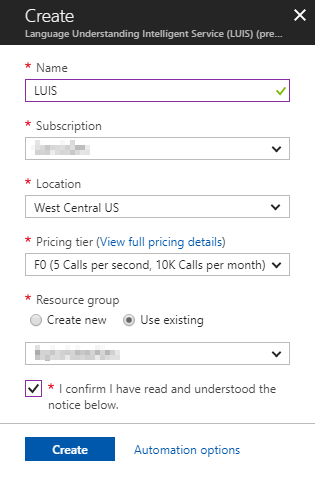
1. Sign in to Azure Portal: <https://portal.azure.com>
2. Click **+ New**, then search “LUIS”



Click **Language Understanding Intelligent Service (preview)**.

Follow the **create experience** to create a LUIS subscription account.

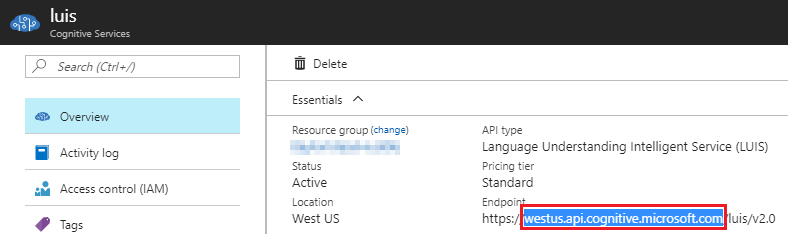
1. Fill the form.



* Input the **Name**: LUIS
* Choose the **Subscription** and **Location**
* Choose a proper **Pricing tier**
* Choosethe **Resource group.**
* Check **I confirm…**

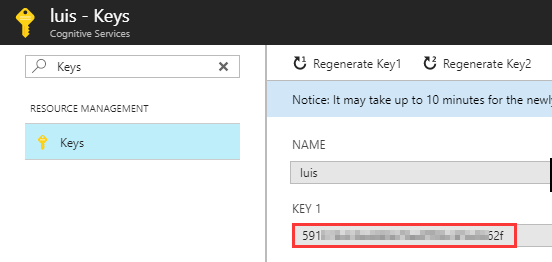
Click **Create**.

1. Go to the created resource



Copy aside the domain name from **Endpoint**. It will be used as the value of **LuisAPIDomain** App setting.

1. Click **Keys**.

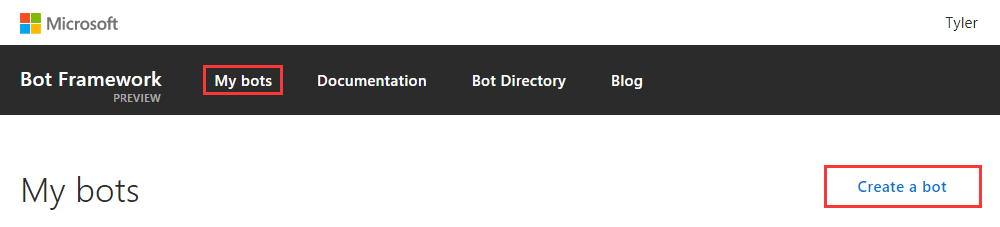


Copy aside one of the key. It will be used as the value of **LuisAPIKey** App setting.

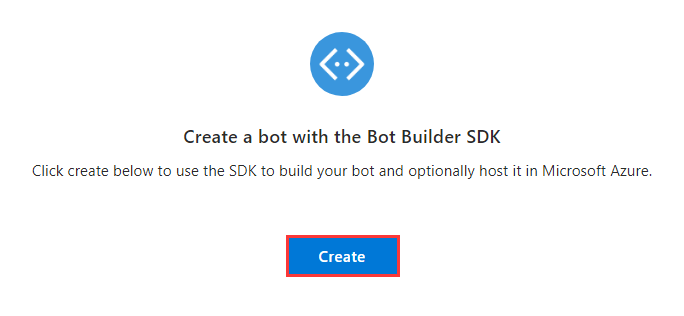
# Bot

## Create a Bot

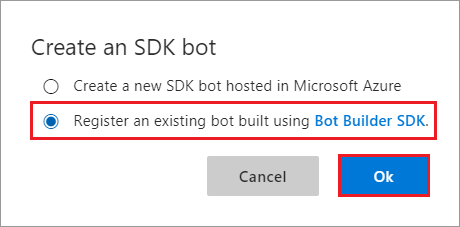
1. Open <https://dev.botframework.com> in browser, then sign in with a work account.



1. Click **My bots**, then click **Create a bot**.



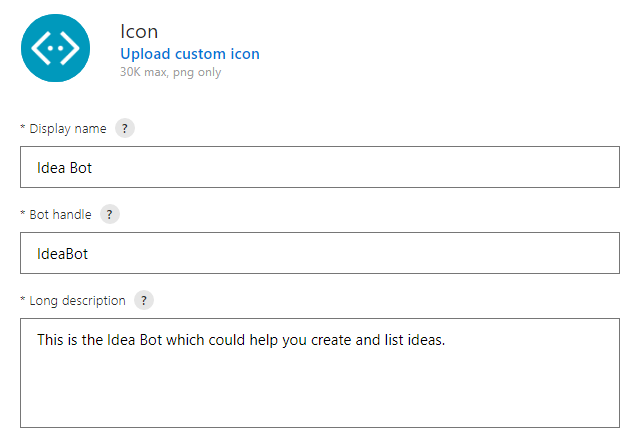
1. Click **Create**



1. Select **Register an existing bot built using Bot Builder SDK**. Then click **Ok**.

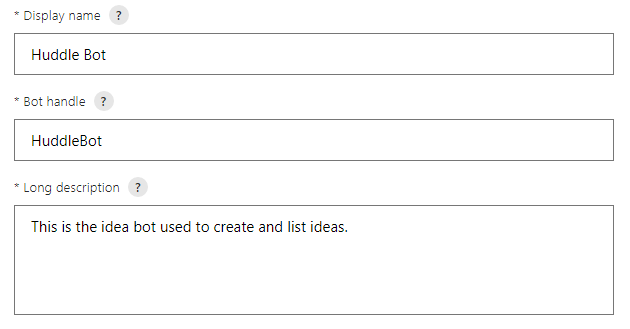
## Customize and Configure the Bot

1. Upload icon:



Click **Upload custom icon**, then select /Files/*HuddleBotIcon.png*.

1. Input the fields:

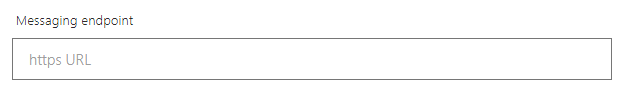


* + **Display name**: Huddle Bot
  + **Bot handle**: HuddleBot
  + **Long description**: This is the idea bot used to create and list ideas.

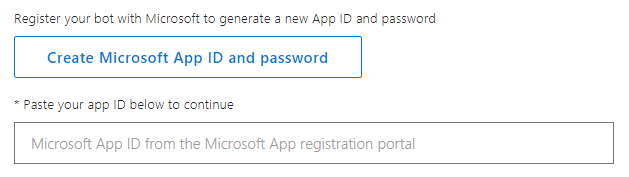
Note:

* + Bot handle should be unique, please add some suffix to avoid confliction.
  + Bot handle will be used as the value of **BotId** App setting. Please copy it aside.

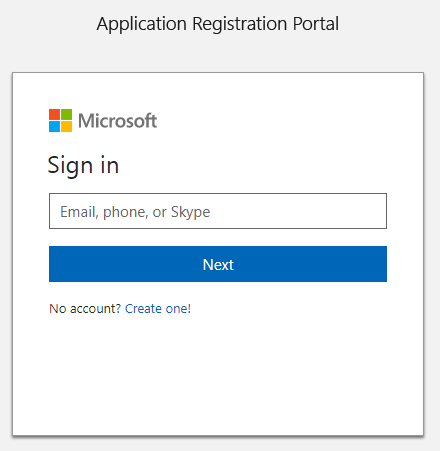
1. Leave Message endpoint empty



1. Configure Microsoft App

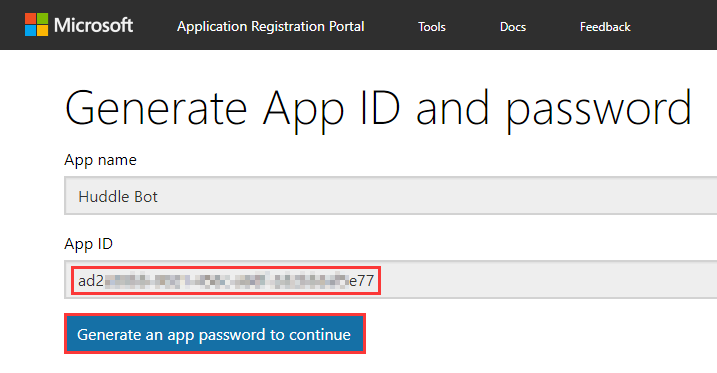


Click **Create Microsoft App ID and password**.



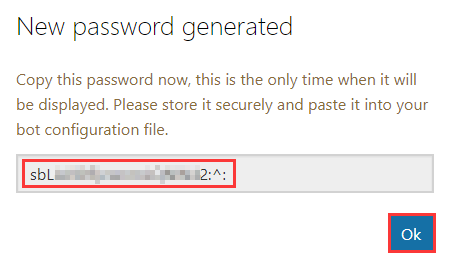
Login in with the work account.

An app will be created automatically:



Copy aside the App ID. It will be used as the value of **MicrosoftAppId** App setting. It will also be used in the Teams App **manifest.json** file as id and botId.

Then click **Generate an app password to continue**.



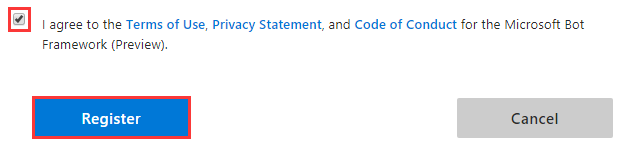
Copy aside the password, It will be used as the value of **MicrosoftAppPassword** App setting.

Then click **OK**.

Click Finish and go back to Bot Framework.



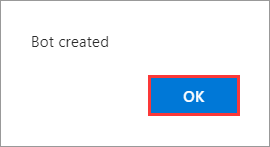
1. Skip **Analytics** and **Admin** sections.
2. Go to the end of the page:



Check the checkbox, and click **Register**.

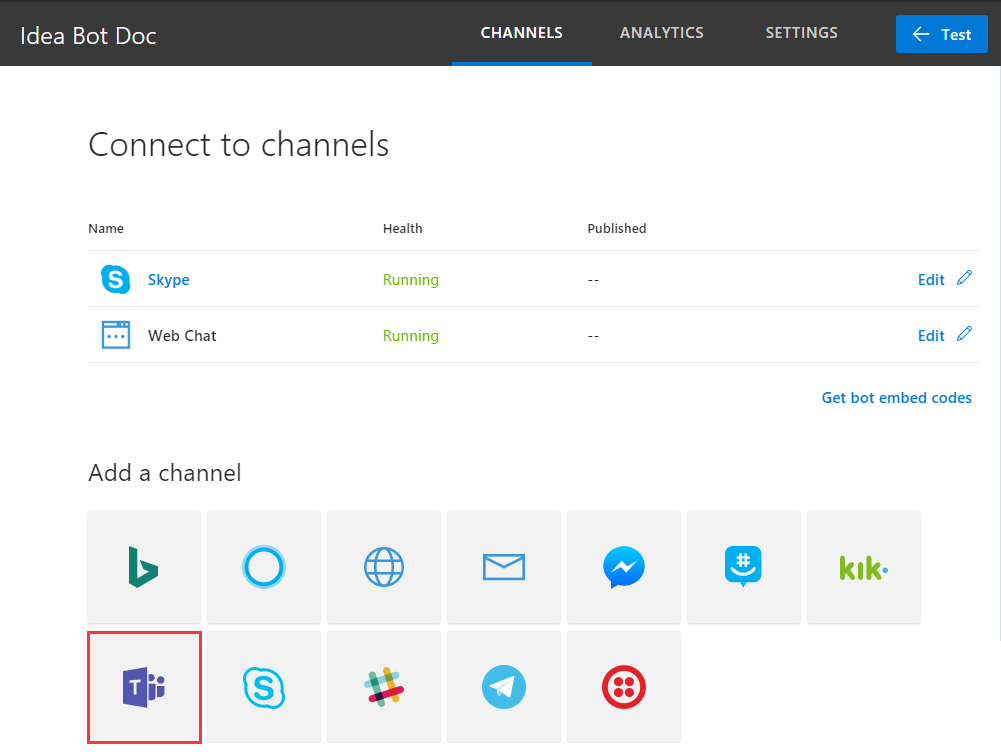
Note: If you got error of the Bot handle: Id is already in use. Please use another one.

1. Click **OK**.

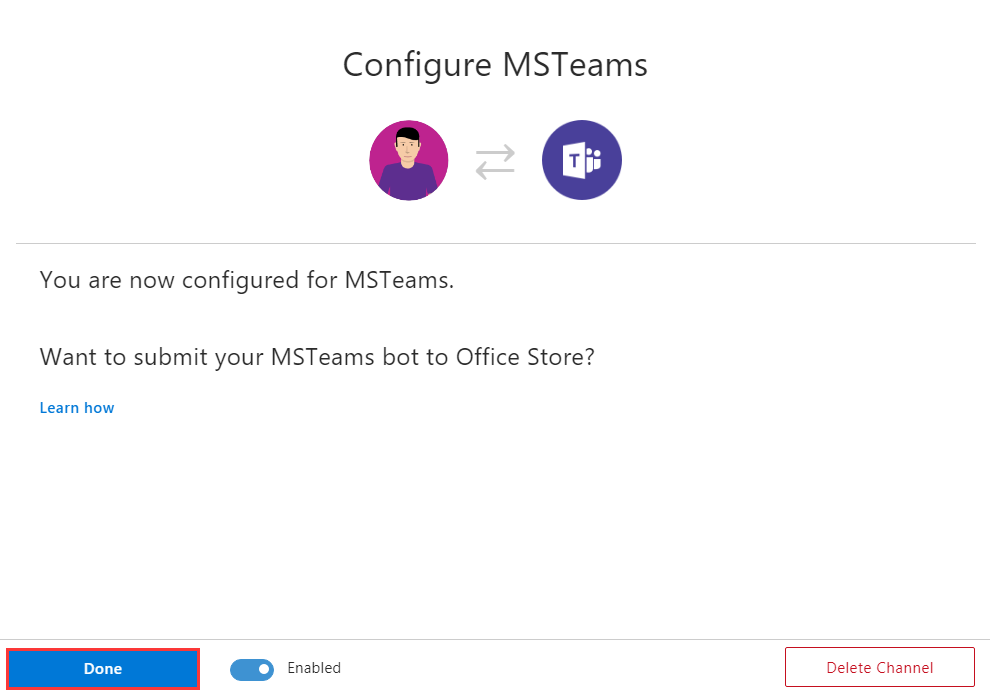


## Add Microsoft Teams Channel

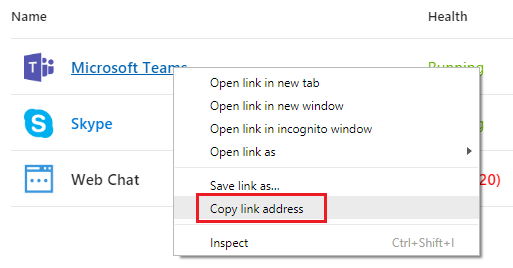
1. Click the **Microsoft Teams Icon** under **Add a channel** section.



1. Click **Done**.



1. Right click the new added **Microsoft Teams** channel.

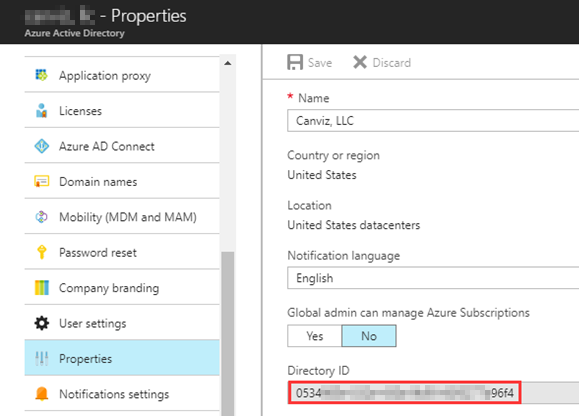


Click **Copy link address**, and paste the URL to someplace. It will be used to add the Bot to Microsoft Teams later.

# Bot Web App

## Create App registrations in AAD

1. Navigate to the AAD you want register the app to.



Copy aside the Directory Id. It will be used as the value of **ida:TenantId** App setting.

1. Create an App Registration in AAD with the following settings:

* **Name**: Huddle Bot Web App
* **Application Type**: Web app/API
* **Sign-on URL**: https://<Your Domain>/bot-web-app
* **Required Permissions**:

|  |  |
| --- | --- |
| API | Permissions |
| Office 365 SharePoint Online  (Microsoft.SharePoint) | Application Permissions:   * Read and write items and lists in all site collections |
| Microsoft Graph | Delegated Permissions   * Read and write all groups * Read all users’ full profiles |

1. After the App Registration created:

* Copy aside the Application ID. It will be used as the value of **ida:ClientID** App setting.
* Create a new key, and copy aside its value. The value will be used as the value of **ida:ClientSecret** App setting.

1. Follow chapter 3 in *2. App-only certificate.docx* to add keyCredential of the certificate to the manifest of the app registration.

## Create a Web App on Azure

1. Create a Web in Azure with the following App settings:

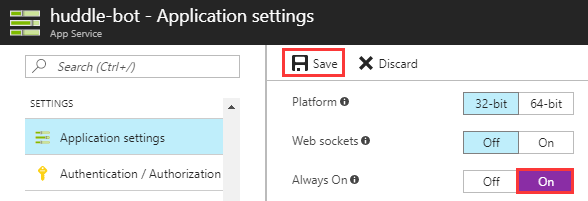
|  |  |  |
| --- | --- | --- |
|  | Key | Value |
| LUIS App | LuisAppId | Use the corresponding value you got from previous chapters. |
| LuisAPIKey |
| LuisAPIDomain |
| Bot | BotId |
| MicrosoftAppId |
| MicrosoftAppPassword |
| AAD App registration | ida:TenantId |
| ida:ClientId |
| ida:ClientSecret |
| Certification | ida:ClientCertThumbprint | Thumbprint of the certificate |
| WEBSITE\_LOAD\_CERTIFICATES | Thumbprint of the certificate |
| SharePoint | BaseSPSiteUrl | The URL of the SharePoint site containing Issue and Metric lists.  For example:  https://xxx.sharepoint.com/sites/huddle |

After created, copy aside the URL of the website, and change the schema to https. For example:

https://huddle-bot.azurewebsites.net

This secure URL will be used in next chapter.

1. Turn on the **Always On** option in **Application Settings**.



1. Upload certificate to Web App

If this Web App uses the same App Service Plan as the Metric Web App. The Huddle App-only Cert is there already.

Otherwise, please follow steps in chapter 4 in 2. App-only certificate.docx to upload it.

1. Build and publish *Huddle.BotWebApp* project to this Web App.

# Follow-up steps

## Add Reply URL to the App Registration

Go to the App registration you created in AAD.

Add a new reply URL: secure URL + /

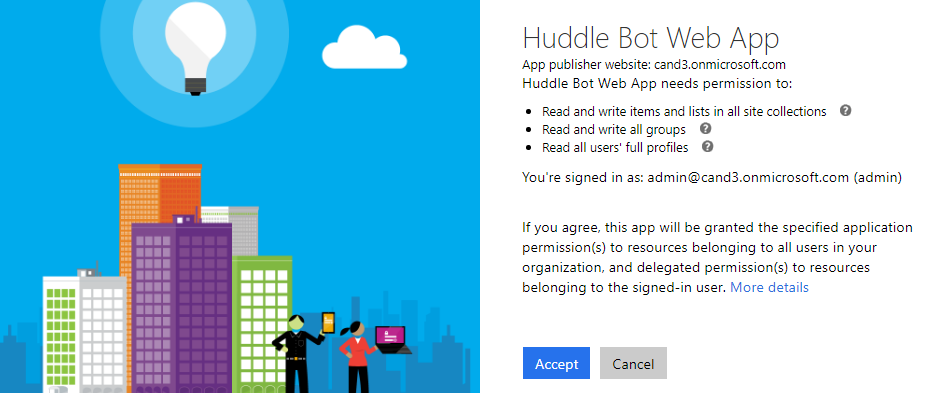
For example: https://huddle-bot.azurewebsites.net/

## Admin consent

Build the admin consent URL: secure URL + /admin/consent

For example: https://huddle-bot.azurewebsites.net/admin/consent

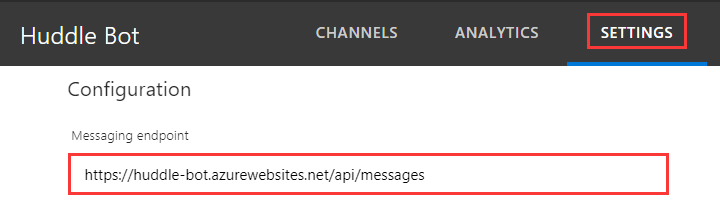
Open the admin consent URL in browser, then sign in with an admin account:



Click **Accept**.

## Update Bot Messaging Endpoint

Go to the Bot you created, then click **SETTINGS**.

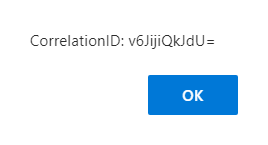


Fill the **Message endpoint** with the secure URL + /api/messages

For example: https://huddle-bot.azurewebsites.net/api/messages

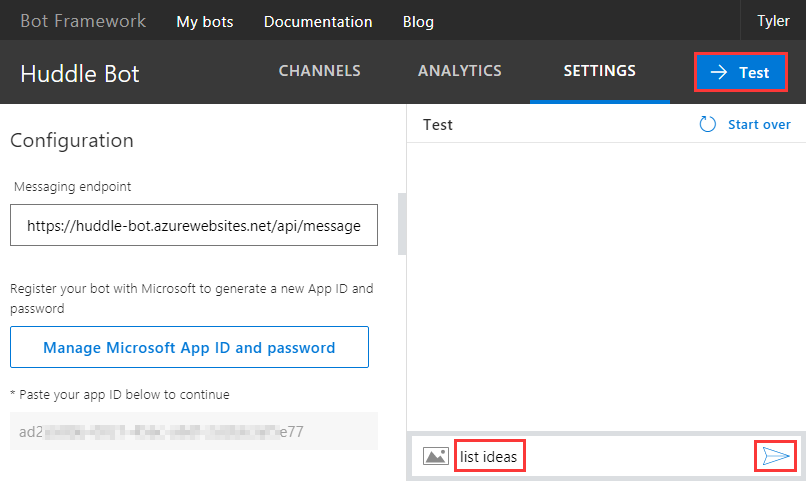
Click **Save Changes** at the bottom.

Note: if you got error like below, please refresh the page and try it several times.

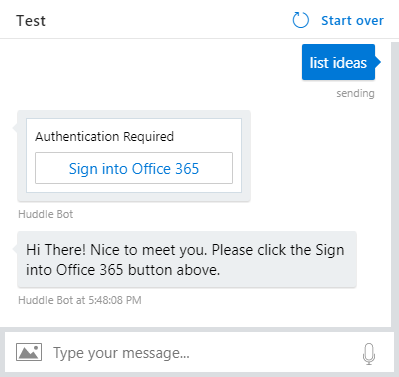


# Verify the Bot

1. Navigate to the bot you registered.



1. Click -**> Test**, input “list ideas”, then send.
2. If you get message like below, the Bot is deployed successfully.



Note: If the message could not be send, please click **retry** for a few times**.**

