



BOT Work Shop

Eric Hullander

January 26th, 2017

Bellevue, WA



Microsoft Partner
Gold Data Analytics
Gold Cloud Platform

Microsoft Partner of the Year
2015 Finalist
Big Data and Analytics



Tools You Will Need

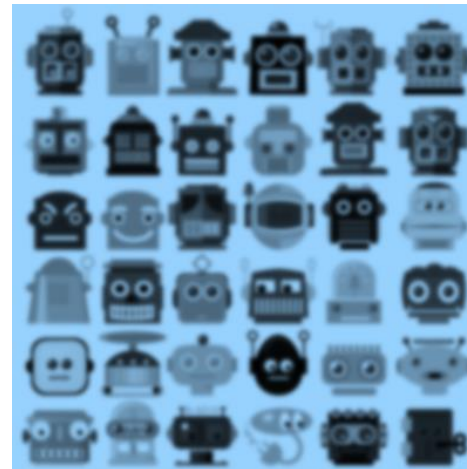
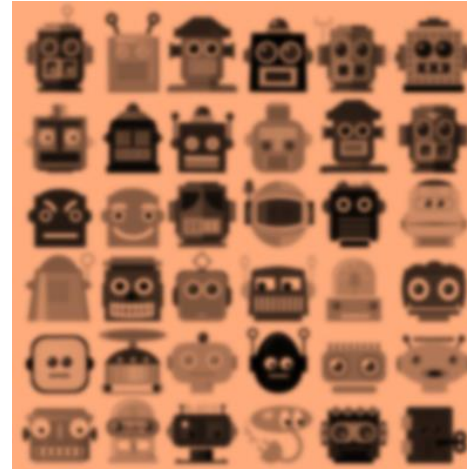
<https://docs.botframework.com/en-us/>

Required:

- Microsoft Account
- [Visual Studio](#) 2015 or higher
- Microsoft.Bot.Builder ("Manage NuGet Packages" in VS)
- [Bot emulator](#)
- Web Browser
- [Skype](#), etc.

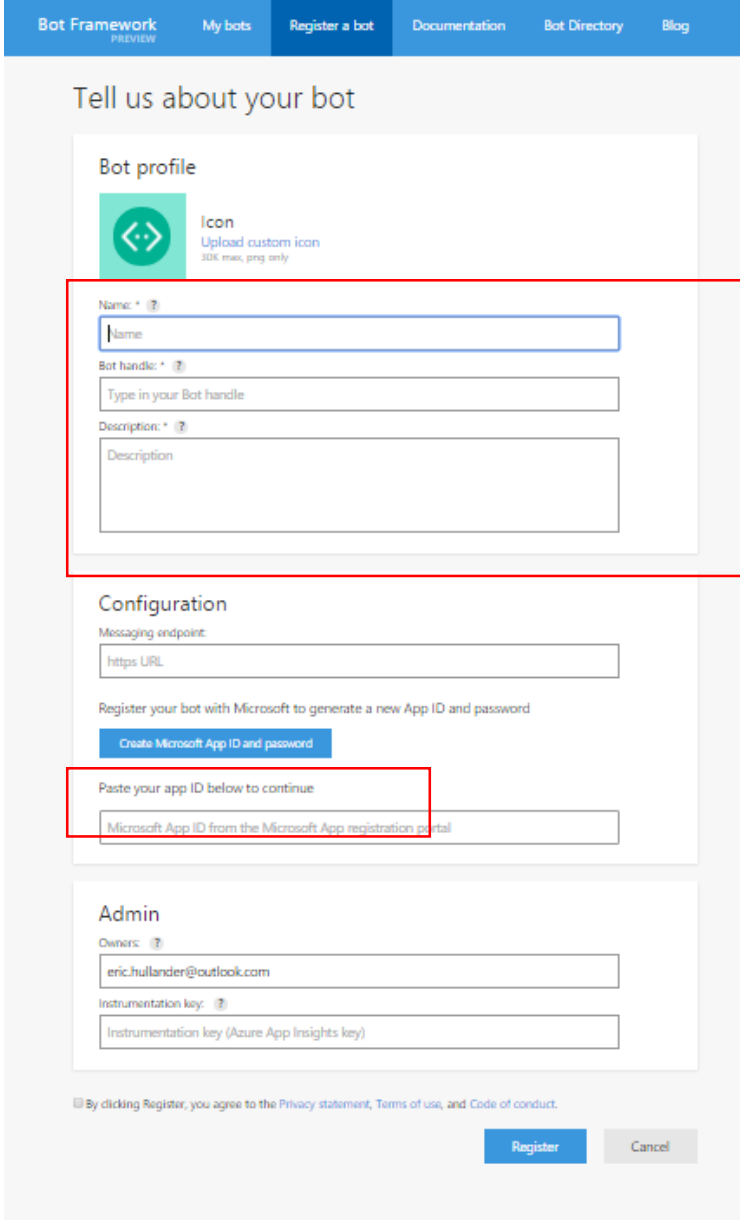
Optional:

- SQL Management Studio
- Excel
- Notepad
- [PowerBI](#)



Bot Framework

- [Sign into Azure](#)
- <https://dev.botframework.com/>
- Click Register a Bot
- Fill in
 - Name
 - Bot Handle
- Click Create Microsoft App ID and password
- Will leave “Messaging Endpoint” Blank



The screenshot shows the 'Register a bot' page on the Bot Framework Developer Portal. The page has a blue header with navigation links: 'Bot Framework', 'My bots', 'Register a bot', 'Documentation', 'Bot Directory', and 'Blog'. The main heading is 'Tell us about your bot'. Below this is the 'Bot profile' section, which includes an 'Icon' upload area and a red-bordered box containing the 'Name', 'Bot handle', and 'Description' input fields. The 'Configuration' section follows, with a 'Messaging endpoint' field and a 'Create Microsoft App ID and password' button. Below this is another red-bordered box containing a 'Paste your app ID below to continue' label and a text input field for the 'Microsoft App ID'. The 'Admin' section at the bottom includes 'Owners' and 'Instrumentation key' fields. At the very bottom, there is a disclaimer and two buttons: 'Register' and 'Cancel'.

Bot Framework
PREVIEW

My bots Register a bot Documentation Bot Directory Blog

Tell us about your bot

Bot profile

Icon
Upload custom icon
30K max, png only

Name: * ?
Name

Bot handle: * ?
Type in your Bot handle

Description: * ?
Description

Configuration

Messaging endpoint:
https URL

Register your bot with Microsoft to generate a new App ID and password

Create Microsoft App ID and password

Paste your app ID below to continue

Microsoft App ID from the Microsoft App registration portal

Admin

Owners: ?
eric.hullander@outlook.com

Instrumentation key: ?
Instrumentation key (Azure App Insights key)

By clicking Register, you agree to the Privacy statement, Terms of use, and Code of conduct.

Register Cancel

Bot Framework

- Copy the App ID to notepad for future use

Generate App ID and password

App name

EricHullandertest

App ID

c6851ae8-631c-4911-ae11-44a236edc3ad

Generate an app password to continue

☐ English (United States)

Bot Framework AppID and Password

- Copy the App ID to notepad for future use
- Click “Generate an app password...”
- Copy the Password to notepad for future use
- Click “OK”
- “Click Finish and go to Bot Framework”

App name: EricHullandertest

App ID: c6851ae8-631c-4911-ae11-44a236edc3ad

App pw: L4wqs7mXeAnz7OG

Generate App ID and password

App name
EricHullandertest

App ID
c6851ae8-631c-4911-ae11-44a236edc3ad

[Generate an app password to continue](#)

☐ English (United States)

New password generated

Copy this password now, this is the only time when it will be displayed. Please store it securely and paste it into your bot configuration file.

L4wqs7mXeAnz7OGCXWxFTm

[Ok](#)

Generate App ID and password

App name
EricHullandertest

App ID
c6851ae8-631c-4911-ae11-44a236edc3ad

Password [Learn More](#)
L4w*****

[Finish and go back to Bot Framework](#)

☐ English (United States)

Bot Framework Registration

- Add the description if you haven't
- Leave endpoint blank for now
- Click if you agree to the T&Cs
- Click "Register"
- Click "OK" on "Bot created" menu
- You have just taken your first step into a much larger world...

Option: You may add an app icon now or later if you like...

The screenshot displays the Bot Framework Registration interface. At the top is a navigation bar with links: Bot Framework, My bots, Register a bot, Documentation, Bot Directory, and Blog. The main heading is "Tell us about your bot".

The "Bot profile" section includes an icon upload area (with a default icon and text "Icon Upload custom icon 30K max, png only"), and input fields for "Name" (containing "EricHullandertest"), "Bot handle" (containing "erichullandertest"), and "Description" (containing "my test bot"). A red asterisk indicates the description is a required field.

The "Configuration" section has a "Messaging endpoint" field (containing "https://"), a button to "Manage Microsoft App ID and password", and a "Paste your app ID below to continue" field (containing "c68").

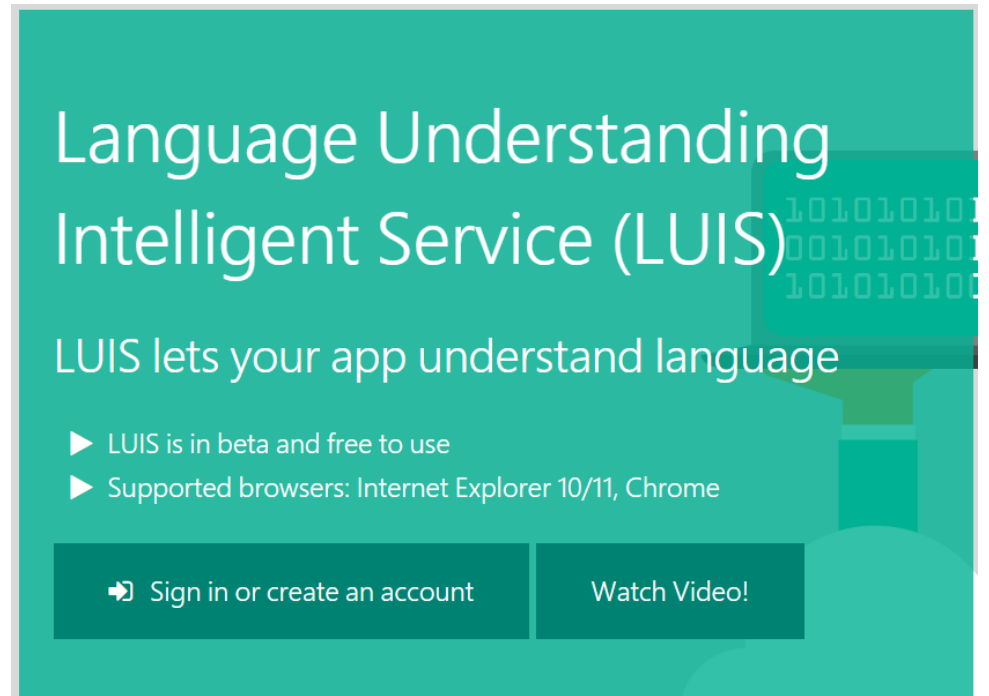
The "Admin" section includes an "Owners" field (containing "eric.hullander@outlook.com"), an "Instrumentation key" field, and a note that the key is an Azure App Insights key.

A "Bot created" confirmation dialog box is overlaid on the right, with an "OK" button.

At the bottom, a checkbox states: "By clicking Register, you agree to the Privacy statement, Terms of use, and Code of conduct." Below this are "Register" and "Cancel" buttons.

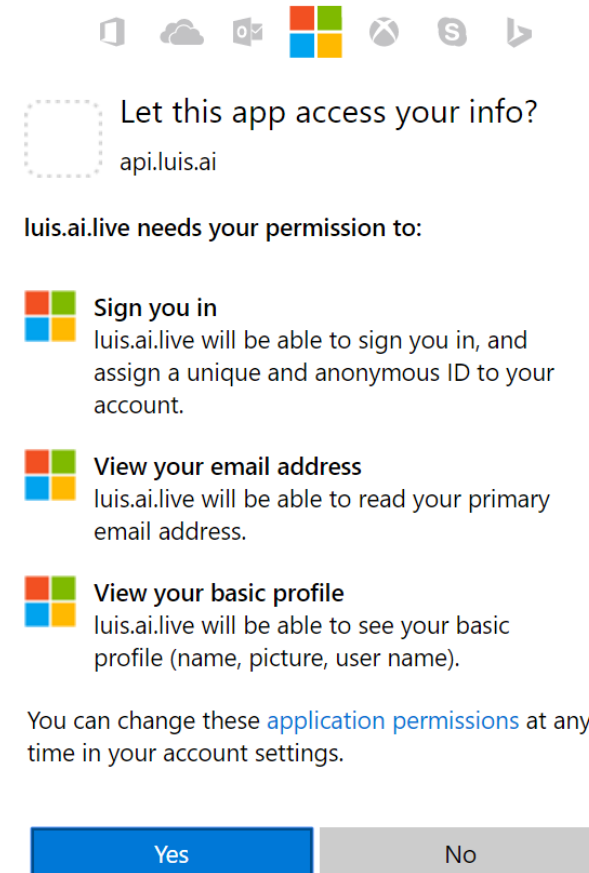
LUIS

- [Sign into Azure](#)
- <https://www.luis.ai/>
- Sign in with your Microsoft account



LUIS

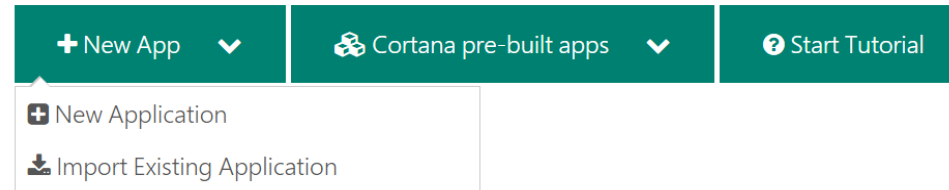
- [Sign into Azure](#)
- <https://www.luis.ai/>
- Sign in with your Microsoft account
- “Yes”



LUIS

Building a LUIS Model

- New App
- Choose “Import...”



A LUIS Application (model) takes examples of many phrases (utterances) that mean the same thing and maps them to a single intent.

A LUIS Application can have up to 40 unique intents

Given a new utterance, LUIS predicts which of the 40 intents is the best match

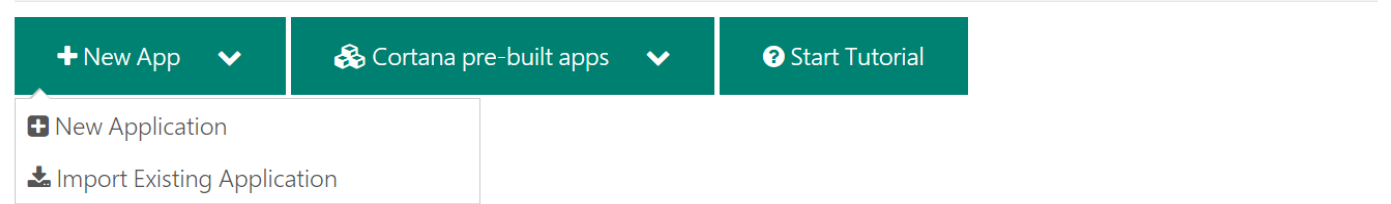
Let's get started

Build [a new language understanding application...](#)

LUIS

Building a LUIS Model

- New App
- Choose “Import...”



Further, entities (nouns, verbs, adjectives) can be inferred from within an utterance

Example:

I want to buy a **house**.

I want to buy a **car**.

Might map to the intent “purchase”

The entity could be “house” or “car”

My Applications

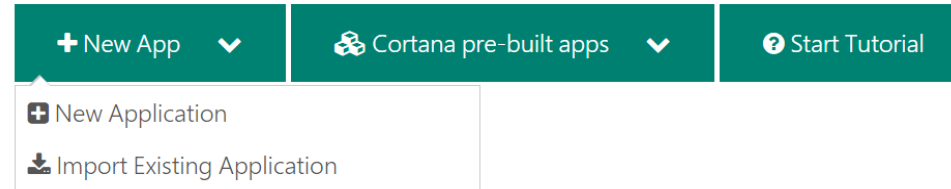
Let's get started

Build [a new language understanding application...](#)

LUIS

Building a LUIS Model

- New App
- Choose “Import...”



To work best, LUIS needs many examples to learn from

Fortunately, if you've already trained LUIS before, and your application will be using the same set of intents and entities you can recycle an existing model and build on it

Let's get started

Build [a new language understanding application...](#)

LUIS

Building a LUIS Model

- Once a model has been trained, we can export it as a JSON file
- Click “Choose File” and Navigate to the JSON file provided
- Click “Import”



HotelExampleBot.json

Import an existing application

Choose an application JSON file to import your application. Please note that importing an application may take several minutes. You will automatically be directed to your new app when it is successfully imported.

Choose File and enter (optional) imported app name:

HotelExampleBot.json

LUIS JSON File

- Within the JSON file you will find lists of intents and later, text and entities that map to that intent
- You may edit this JSON directly and upload it, or use the interface



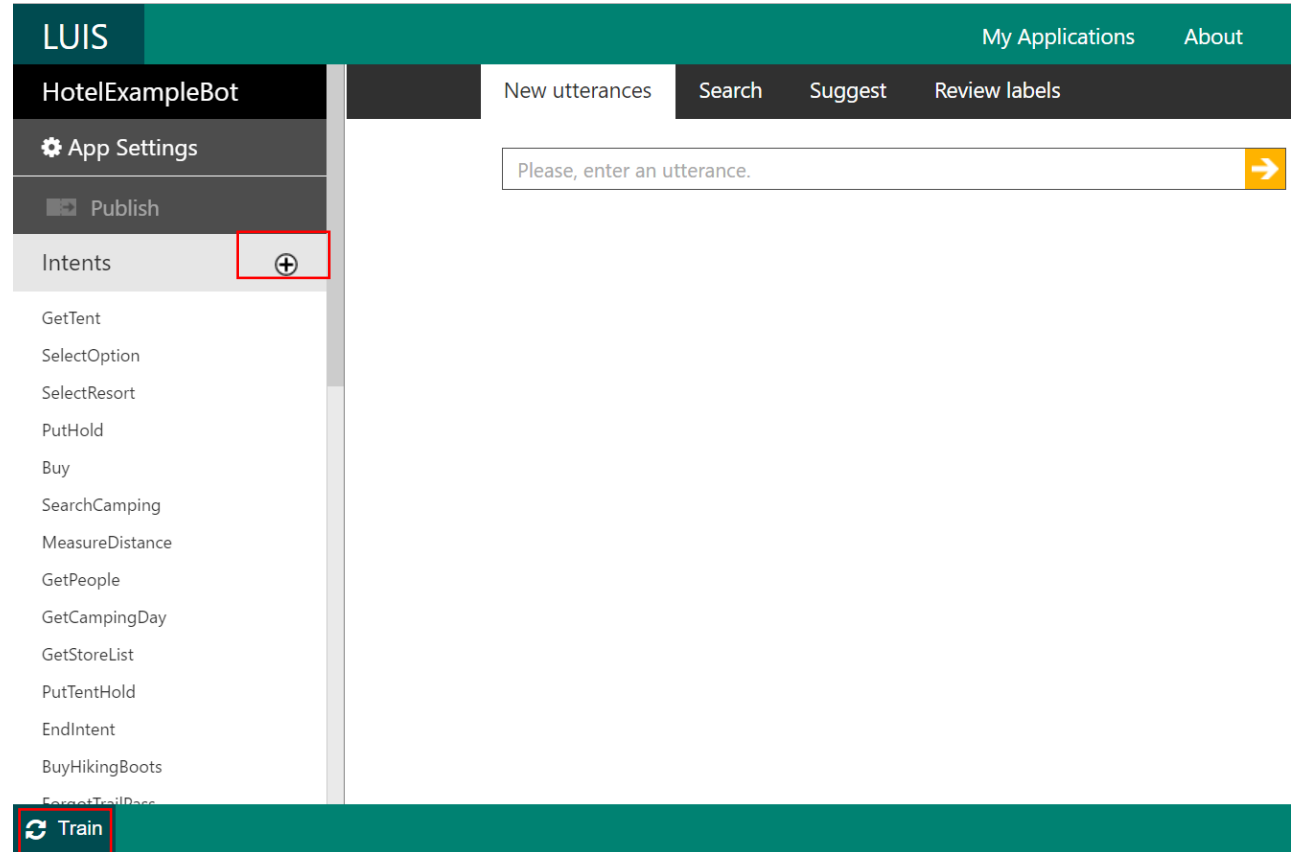
HotelExampleBot.json

```
{
  "luisc_schema_version": "1.3.0",
  "name": "HotelExampleBot",
  "desc": "",
  "culture": "en-us",
  "intents": [
    {
      "name": "Buy"
    },
    {
      "name": "BuyHikingBoots"
    },
    {
      "name": "EndIntent"
    },
    {
      "name": "FindHikes"
    },
    {
      "name": "ForgotTrailPass"
    }
  ],
  {
    "text": "camp sites",
    "intent": "SearchCamping",
    "entities": []
  },
  {
    "text": "camping",
    "intent": "SearchCamping",
    "entities": []
  },
  {
    "text": "any hikes",
    "intent": "FindHikes",
    "entities": []
  },
  {
    "text": "do they have nice hikes?",
    "intent": "FindHikes",
    "entities": []
  }
}
```

LUIS

Editing

- Once Loaded we see this screen
- The list on the left are the “intents” this model understands
- If we click (+), we can add new intents to the model
- Even though we have imported the file, we still need to train the algorithm to recognize utterances
- Click “Train”



LUIS

Publishing

- Training can take some time. Click “Publish
- Click “Publish web Service”

The screenshot displays the LUIS web interface for an application named 'HotelExampleBot'. The left sidebar contains a menu with 'App Settings' and 'Publish' highlighted by a red rectangle. The main area shows a search bar with the placeholder text 'Please, enter an utterance.' and a yellow arrow button. Below the search bar, a list of intents is visible: GetTent, SelectOption, SelectResort, PutHold, Buy, SearchCamping, MeasureDistance, and GetPeople. A modal window titled 'HTTP service' is open, showing the status 'Status: service not published' and a green 'Publish web service' button highlighted by a red rectangle. The modal also includes a note: 'Note: To enable bot integration, enable action fulfillment in one of your intents.' and a checkbox labeled 'Enable Action Binding using Microsoft Bot Framework'.

LUIS

HotelExampleBot

App Settings

Publish

Intents

GetTent

SelectOption

SelectResort

PutHold

Buy

SearchCamping

MeasureDistance

GetPeople

New utterances

Search

Suggest

Review labels

Please, enter an utterance.

HTTP service

Publish Current Application to URL for access via HTTP

Status: service not published

Publish web service

Note: To enable bot integration, enable action fulfillment in one of your intents.

☐ Enable Action Binding using Microsoft Bot Framework

LUIS

Publishing

- Training can take some time. Click “Publish
- Click “Publish web Service”

The screenshot displays the LUIS web interface for an application named 'HotelExampleBot'. The left sidebar contains a menu with 'App Settings' and 'Publish' highlighted by a red rectangle. The main area shows a search bar with the placeholder text 'Please, enter an utterance.' and a yellow arrow button. Below the search bar, a list of intents is visible: GetTent, SelectOption, SelectResort, PutHold, Buy, SearchCamping, MeasureDistance, and GetPeople. A modal window titled 'HTTP service' is open, showing the status 'Status: service not published' and a green 'Publish web service' button highlighted by a red rectangle. The modal also includes a note: 'Note: To enable bot integration, enable action fulfillment in one of your intents.' and a checkbox labeled 'Enable Action Binding using Microsoft Bot Framework'.

LUIS

Testing

- You can type an utterance into “Query” to see how the model scores it. Given an utterance it will give each possible intent in the model a score
- Example
- Query: “find a trail”

```
{ "query": "find a trail", "topScoringIntent": { "intent": "GetTrailPass", "score": 0.9974881 },  
  "intents": [ { "intent": "GetTrailPass", "score": 0.9974881 },  
    { "intent": "None", "score": 0.5163145 },  
    { "intent": "ForgotTrailPass", "score": 0.0379541256 },  
    { "intent": "Buy", "score": 0.0100398948 },  
    { "intent": "PutTentHold", "score": 0.007901444 },  
    { "intent": "Help", "score": 0.00303268922 },  
    { "intent": "SelectResort", "score": 0.00238058716 },
```

HTTP service

Publish Current Application to URL for access via HTTP

Status: Published on 1/24/2017, 2:24:59 PM

[Update published application](#)

Query:

Optional Parameters

☒ Verbose

☐ Enable Bing Spell Check For This Query [?](#)

URL: [https://westus.api.cognitive.microsoft.com/luis/v2.0/apps/02849bf2-da96-4840-825c-73d340943a4a?subscription-key=\[REDACTED\]&use=true](https://westus.api.cognitive.microsoft.com/luis/v2.0/apps/02849bf2-da96-4840-825c-73d340943a4a?subscription-key=[REDACTED]&use=true)

Note: To enable bot integration, enable action fulfillment in one of your intents.

☐ Enable Action Binding using Microsoft Bot Framework

[Download web service usage logs](#)

[Download logs](#)

LUIS

AppID and Key

- Click App Settings
- Copy the App Id to your notepad
- In the top banner, click on your account name

The screenshot displays the LUIS portal interface. At the top, there's a navigation bar with 'LUIS', 'My Applications', and 'About'. Below this, a secondary bar shows 'HotelExampleBot', 'New utterances', 'Search', 'Suggest', and 'Review labels'. A search bar with the placeholder 'Please, enter an utterance.' and a submit button is also present. On the left, a sidebar lists various intents like 'GetTent', 'SelectOption', etc. The 'App Settings' modal is open, showing 'Basic Settings' and 'Auth Settings' tabs. The 'General Info' section contains the 'App Id' (02849bf2-da96-4840-825c-73d340943a4a) and 'App Name' (HotelExampleBot). The 'Endpoint Key Assigned to app:' section includes a text input for a new key, an 'Add New Key' button, and a dropdown menu for selecting an existing key. The 'Bing Spell Check' section has a checkbox to 'Enable Bing Spell Check' and a link to subscribe for a key.

LUIS

AppID and Key

- In the top banner, click on your account name
- Click Subscription Keys
- Copy the API Key to your notepad

App name: EricHullandertest

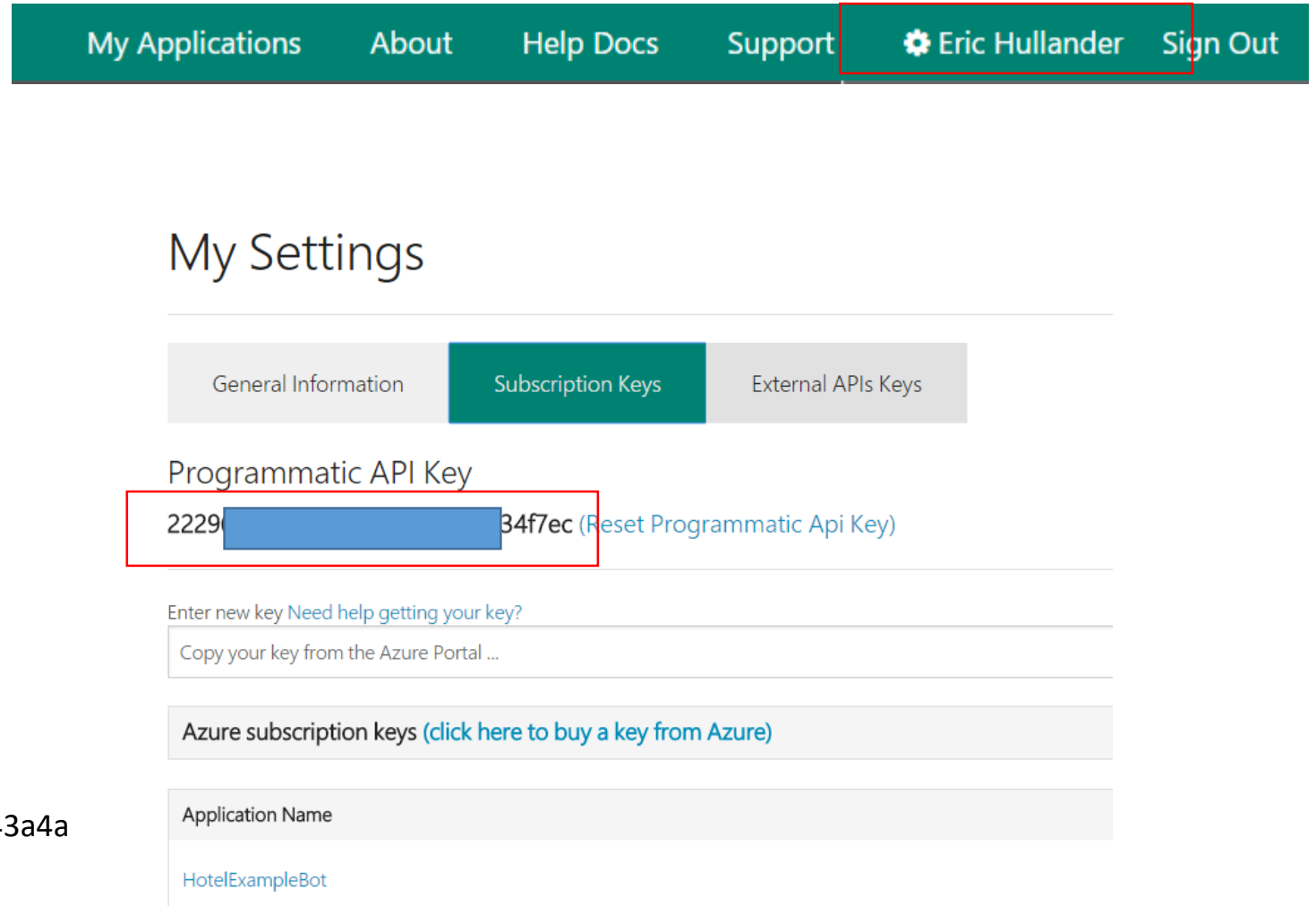
App ID: c6851ae8-631c-4911-ae11-44a236edc3ad

App pw: L4wqs7mXeAnz7OGcXWxFTm

LUIS App Name: HotelExampleBot

LUID ADD ID: 02849bf2-da96-4840-825c-73d340943a4a

LUID Key: 22296c1f55054d48b514027a0534f7ec



The screenshot shows the 'My Settings' page in the Azure portal. The top navigation bar includes 'My Applications', 'About', 'Help Docs', 'Support', and a user profile for 'Eric Hullander' with a 'Sign Out' link. The 'Subscription Keys' tab is selected. Under 'Programmatic API Key', the key '2229[REDACTED]34f7ec' is displayed, with a red box highlighting it and a '(Reset Programmatic Api Key)' link. Below this is a section for 'Enter new key' with a link 'Need help getting your key?' and a text input field containing 'Copy your key from the Azure Portal ...'. Further down, there is a section for 'Azure subscription keys (click here to buy a key from Azure)' and a section for 'Application Name' with the value 'HotelExampleBot'.

My Applications About Help Docs Support **Eric Hullander** Sign Out

My Settings

General Information **Subscription Keys** External APIs Keys

Programmatic API Key

2229[REDACTED]34f7ec ([Reset Programmatic Api Key](#))

Enter new key [Need help getting your key?](#)

Copy your key from the Azure Portal ...

Azure subscription keys ([click here to buy a key from Azure](#))

Application Name

HotelExampleBot

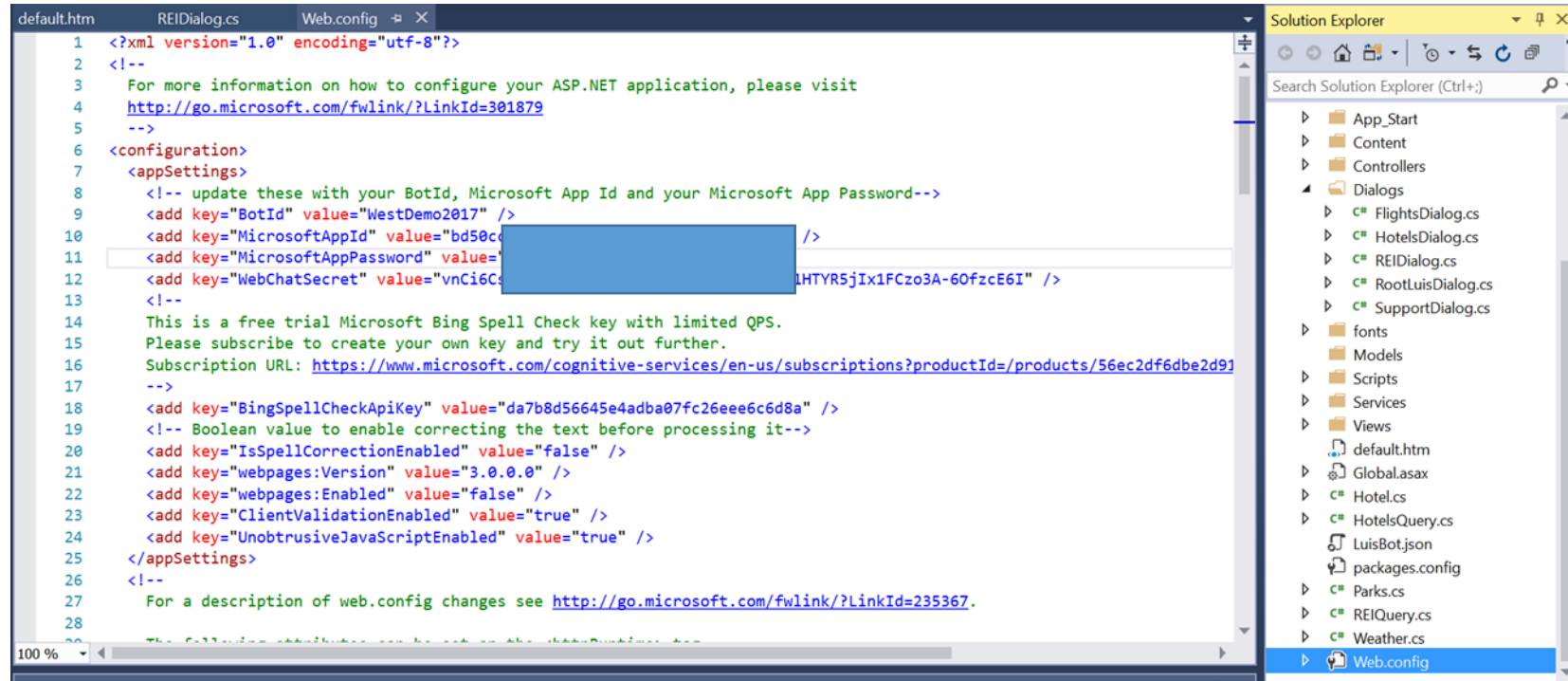
Visual Studio

- Unzip and then navigate to the .sln file in the the zip file provided
- Open in Visual Studio
- This can be used as a template for future bot building and will provide a starting point for your exploration
- Some configuration is required

Name	Date modified	Type	Size
obj	1/24/2017 12:12 AM	File folder	
packages	11/28/2016 11:54 AM	File folder	
Properties	11/28/2016 11:55 AM	File folder	
Scripts	11/28/2016 11:55 AM	File folder	
Service References	11/28/2016 11:55 AM	File folder	
Services	11/28/2016 11:55 AM	File folder	
Views	11/28/2016 11:55 AM	File folder	
default.htm	1/24/2017 12:20 AM	Chrome HTML Docu...	2 KB
Global.asax	11/28/2016 11:54 AM	ASP.NET Server Appli...	1 KB
Global.asax.cs	11/28/2016 11:54 AM	Visual C# Source File	1 KB
Hotel.cs	11/28/2016 11:55 AM	Visual C# Source File	1 KB
HotelsQuery.cs	11/28/2016 11:54 AM	Visual C# Source File	1 KB
LuisBot.csproj	1/24/2017 12:12 AM	Visual C# Project file	16 KB
LuisBot.csproj.user	1/24/2017 12:12 AM	Per-User Project Opti...	2 KB
LuisBot.csproj.vspssc	11/28/2016 11:55 AM	Source Control Proje...	1 KB
LuisBot.json	11/28/2016 11:54 AM	JSON File	5 KB
LuisBot.sln	11/28/2016 11:54 AM	Visual Studio Solution	2 KB
packages.config	11/28/2016 11:54 AM	XML Configuration File	3 KB
Parks.cs	11/28/2016 11:54 AM	Visual C# Source File	1 KB
README.md	11/28/2016 11:54 AM	MD File	9 KB
REIQuery.cs	11/28/2016 11:54 AM	Visual C# Source File	1 KB
Weather.cs	11/28/2016 11:54 AM	Visual C# Source File	3 KB
Web.config	1/23/2017 11:41 PM	XML Configuration File	6 KB
Web.Debug.config	11/28/2016 11:54 AM	XML Configuration File	2 KB
Web.Release.config	11/28/2016 11:54 AM	XML Configuration File	2 KB

Visual Studio

- Unzip and then navigate to the .sln file in the the zip file provided
- Open in Visual Studio
- Open Web.config
- Fill in from your notepad
 - BotId
 - MicrosoftAppId
 - MicrosoftAppPassword

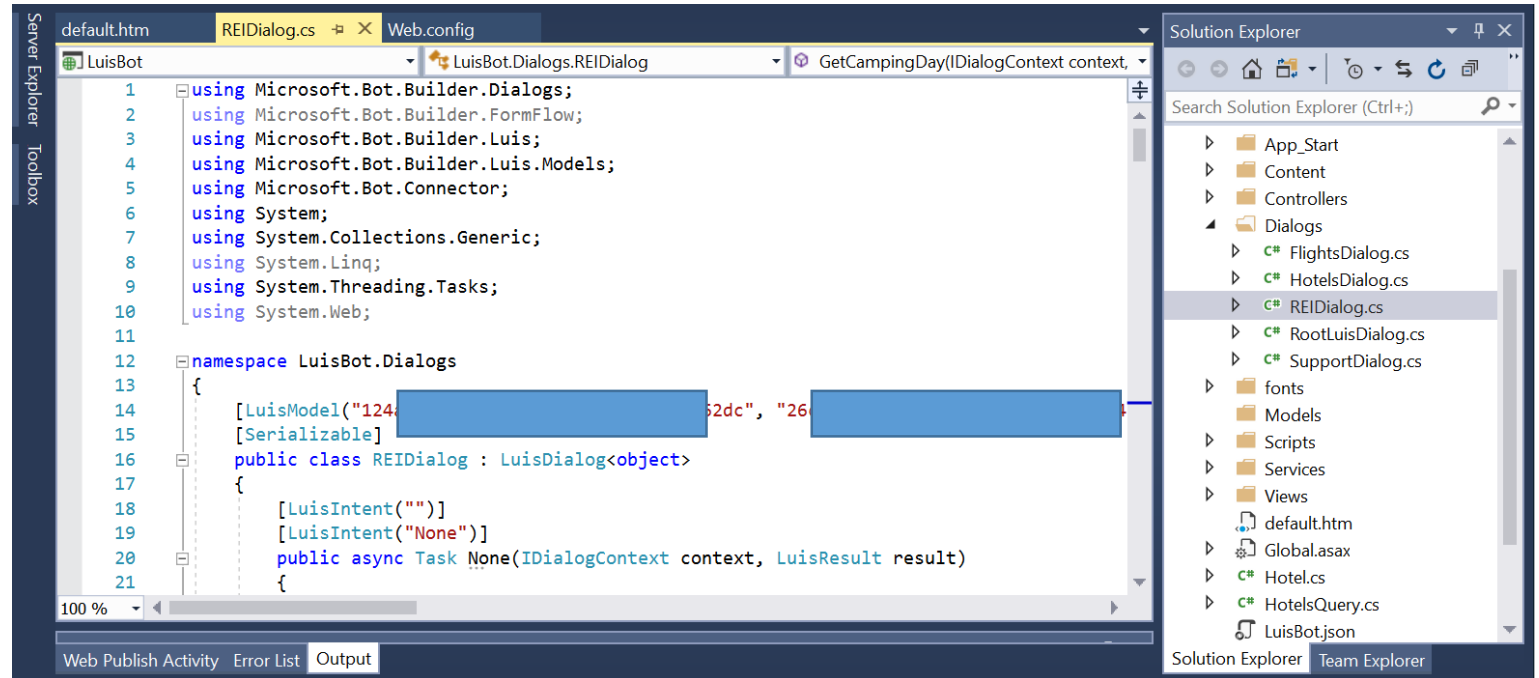


The screenshot shows the Visual Studio IDE with the `Web.config` file open in the editor. The file contains XML configuration for an ASP.NET application, including app settings for BotId, MicrosoftAppId, MicrosoftAppPassword, and WebChatSecret. The Solution Explorer on the right shows the project structure, including folders like App_Start, Content, Controllers, Dialogs, fonts, Models, Scripts, Services, Views, and files like default.htm, Global.asax, Hotel.cs, HotelsQuery.cs, LuisBot.json, packages.config, Parks.cs, REIQuery.cs, and Weather.cs. The `Web.config` file is selected in the Solution Explorer.

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <!--
3 For more information on how to configure your ASP.NET application, please visit
4 http://go.microsoft.com/fwlink/?LinkId=301879
5 -->
6 <configuration>
7   <appSettings>
8     <!-- update these with your BotId, Microsoft App Id and your Microsoft App Password-->
9     <add key="BotId" value="WestDemo2017" />
10    <add key="MicrosoftAppId" value="bd50c" />
11    <add key="MicrosoftAppPassword" value=" " />
12    <add key="WebChatSecret" value="vnCi6C" />
13  </appSettings>
14  <!-- This is a free trial Microsoft Bing Spell Check key with limited QPS.
15  Please subscribe to create your own key and try it out further.
16  Subscription URL: https://www.microsoft.com/cognitive-services/en-us/subscriptions?productId=/products/56ec2df6dbe2d91
17  -->
18  <add key="BingSpellCheckApiKey" value="da7b8d56645e4adba07fc26eee6c6d8a" />
19  <!-- Boolean value to enable correcting the text before processing it-->
20  <add key="IsSpellCorrectionEnabled" value="false" />
21  <add key="webpages:Version" value="3.0.0.0" />
22  <add key="webpages:Enabled" value="false" />
23  <add key="ClientValidationEnabled" value="true" />
24  <add key="UnobtrusiveJavaScriptEnabled" value="true" />
25 </appSettings>
26 <!--
27 For a description of web.config changes see http://go.microsoft.com/fwlink/?LinkId=235367.
28 -->
```

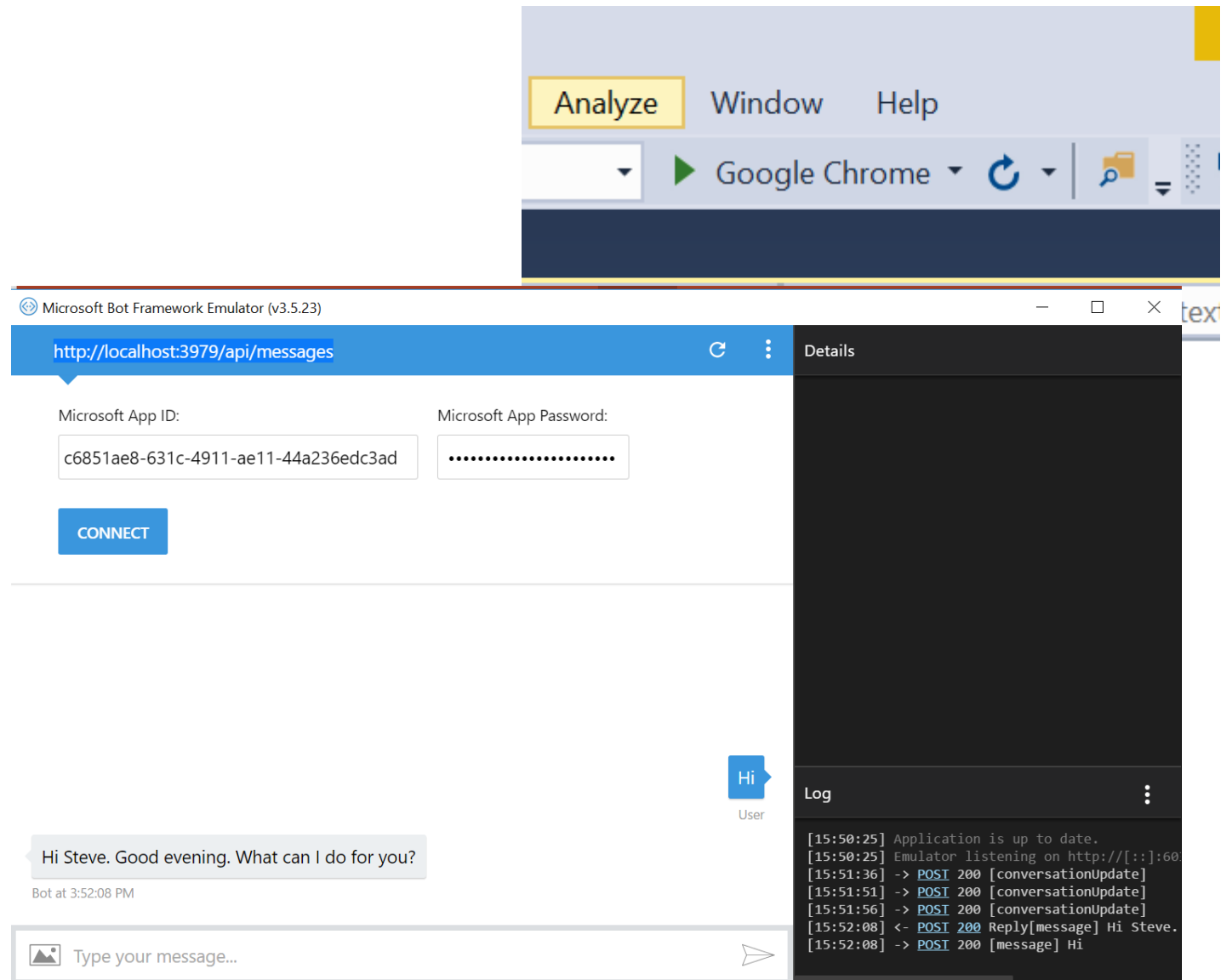
Visual Studio

- Open REIDialog.cs
- Enter the App ID and Password



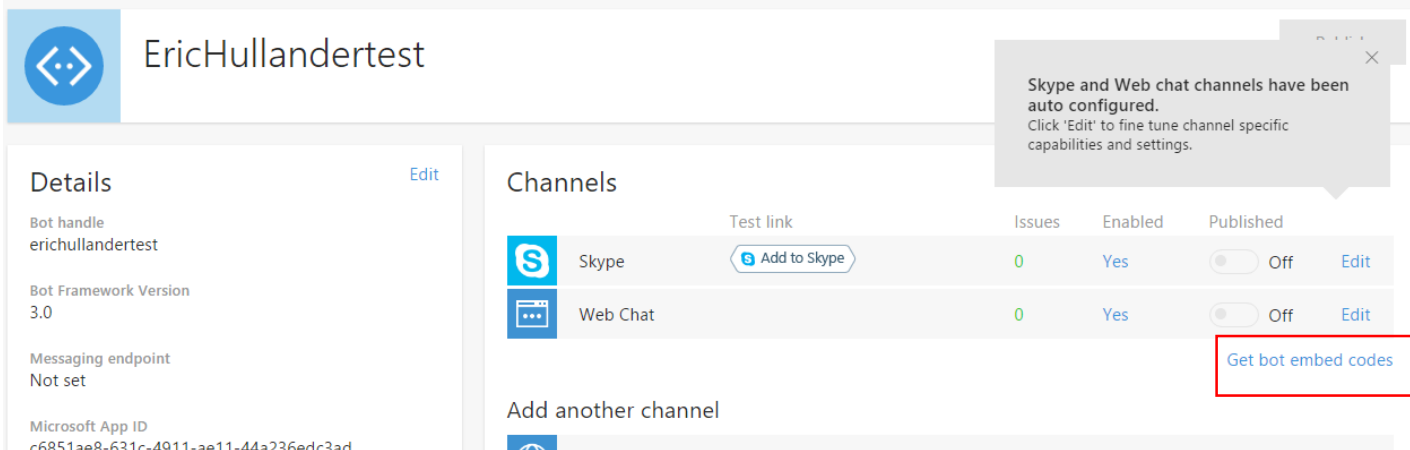
Bot Emulation

- Before we publish we will test the bot locally using the bot emulator
- Click the Play Button to have the bot run on your local host
- Launch the Bot emulator
- Input the address of your local host
- Input your AppID and PW
- Let the emulator connect
- Say “Hi”



Bot Publication Skype

- Return to <https://dev.botframework.com/bots?id=erichullandertest>
- Click the light blue Skype icon
- Click “Get bot embed codes”
- Copy the embed code to your notepad





The screenshot shows the Bot Framework Developer Portal interface for a bot named 'EricHullandertest'. The 'Details' tab is active, showing the bot handle 'erichullandertest', Bot Framework Version '3.0', and a 'Get bot embed codes' button highlighted with a red box. The 'Channels' tab shows two channels: 'Skype' and 'Web Chat', both with 'Add to Skype' buttons and 'Get bot embed codes' links. A notification at the top right states: 'Skype and Web chat channels have been auto configured. Click 'Edit' to fine tune channel specific capabilities and settings.'

Channel	Test link	Issues	Enabled	Published	Action
Skype	Add to Skype	0	Yes	Off	Edit
Web Chat		0	Yes	Off	Edit


4

Bot embed code

Choose a channel to get the embed code for it. Copy and paste the code to your website.



Embed code for Skype



```
<a href='https://join.skype.com/bot/c6851ae8-[redacted]6edc3ad'> <img src='https://dev.botframework.com/Client/Images/Add-To-Skype-Buttons.png' /> </a>
```


Bot Publication Web

- Click the dark blue web icon
- “Click here to open the web...”



The screenshot shows the Microsoft Bot Framework Developer Portal interface for a bot named 'EricHullandertest'. The 'Channels' section lists 'Skype' and 'Web Chat'. The 'Web Chat' channel is highlighted with a red box, and a 'Get bot embed codes' link is visible. A modal window titled 'Bot embed code' is open, displaying the Skype and Web Chat icons. The modal text states: 'Choose a channel to get the embed code for it. Copy and paste the code to your website.' Below the icons, it says 'Embed code for Web Chat has moved' and provides a link: 'Click here to open the Web Chat configuration page >'. A notification at the top right states: 'Skype and Web chat channels have been auto configured. Click 'Edit' to fine tune channel specific capabilities and settings.'

Channel	Test link	Issues	Enabled	Published	Action
Skype	Add to Skype	0	Yes	Off	Edit
Web Chat		0	Yes	Off	Edit

Bot Publication Web

- Click “Add New Site”
- Give your site a name
 - “MySite”
- Click Done

Configure Web Chat



+ Add new site

How would you name your site?

Site name is for your reference and you can change it anytime.

Cancel Done

Bot Publication Web

- Show
- Copy Secret to notepad
- Copy Embed Code to notepad
- Replace "YOUR_SECRET_HERE" in embed code with your secret

+ Add new site

EricHullanderTest

EricHullanderTest Disable

Secret keys

XXXXXXXXXXXXXXXXXXXXXXXXXXXX

Show | Regenerate

XXXXXXXXXXXXXXXXXXXXXXXXXXXX

Show | Regenerate

Embed code

```
<iframe src='https://webchat.botframework.com/embed/erichullander?test?s=YOUR_SECRET_HERE'></iframe>
```

Copy

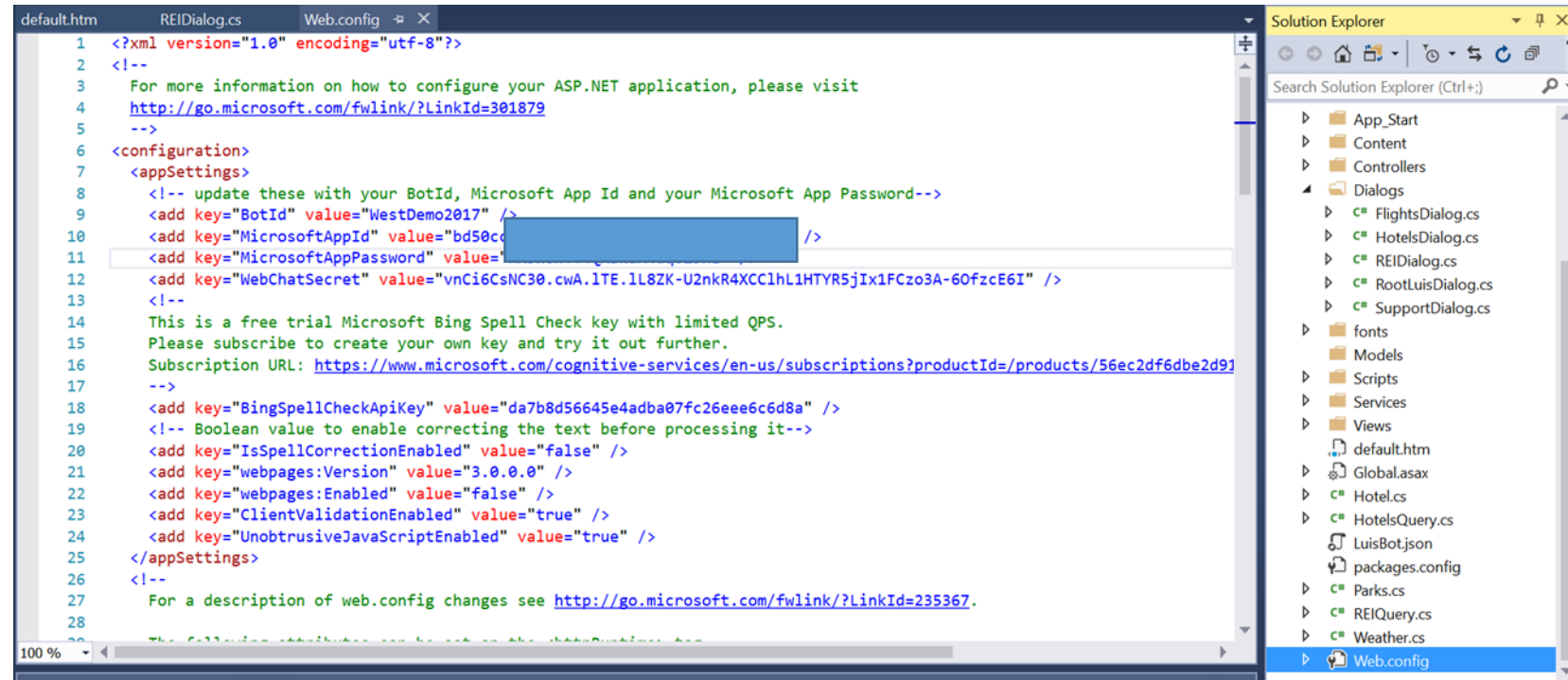
Preview

Enabling Web Chat Preview on this site will give users of your bot a preview version of the Bot Framework Web Chat control. The preview version includes new, beta features that have not yet been officially released.

☒ Enable Preview

Visual Studio

- The web secret goes here too

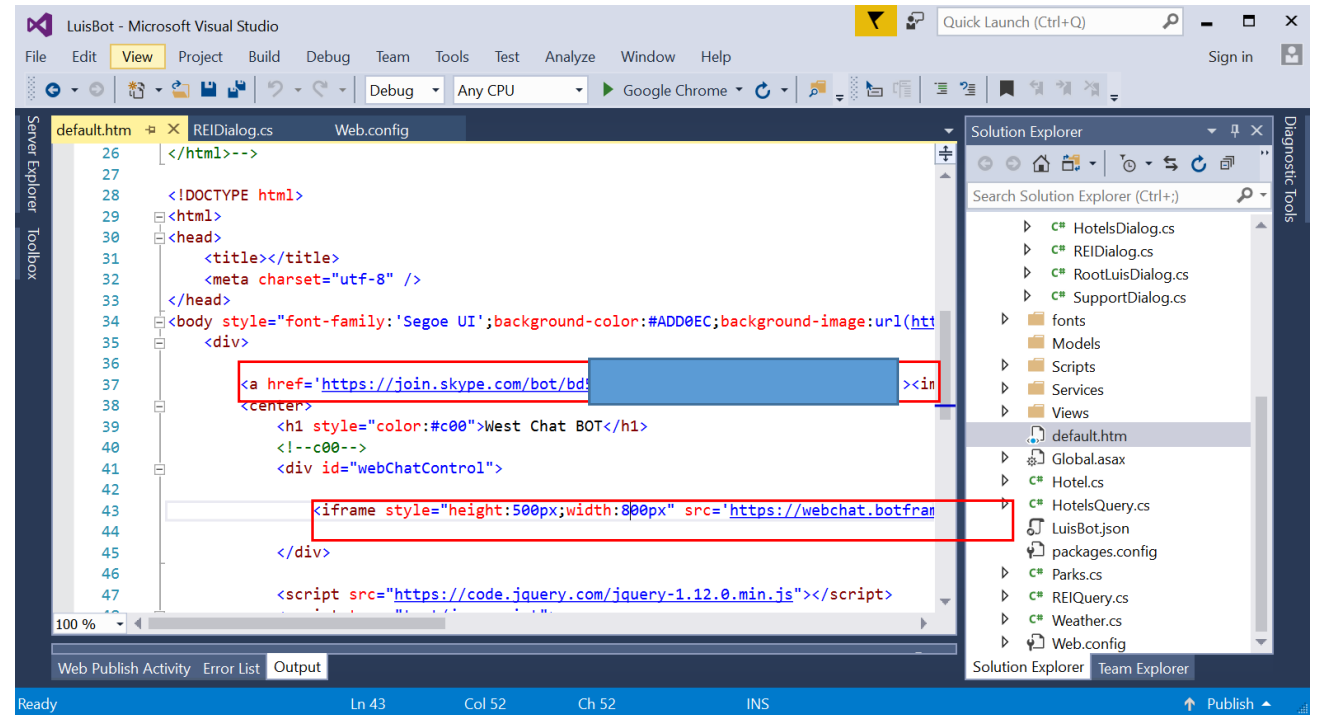


```
1 <?xml version="1.0" encoding="utf-8"?>
2 <!--
3 For more information on how to configure your ASP.NET application, please visit
4 http://go.microsoft.com/fwlink/?LinkId=301879
5 -->
6 <configuration>
7   <appSettings>
8     <!-- update these with your BotId, Microsoft App Id and your Microsoft App Password-->
9     <add key="BotId" value="WestDemo2017" />
10    <add key="MicrosoftAppId" value="bd50cc" />
11    <add key="MicrosoftAppPassword" value="REDACTED" />
12    <add key="WebChatSecret" value="vnCi6CsNC30.cwA.1TE.1L8ZK-U2nkR4XCC1hL1HTYR5jIx1FCzo3A-60fzc6EI" />
13    <!--
14 This is a free trial Microsoft Bing Spell Check key with limited QPS.
15 Please subscribe to create your own key and try it out further.
16 Subscription URL: https://www.microsoft.com/cognitive-services/en-us/subscriptions?productId=/products/56ec2df6dbe2d91
17 -->
18    <add key="BingSpellCheckApiKey" value="da7b8d56645e4adba07fc26eee6c6d8a" />
19    <!-- Boolean value to enable correcting the text before processing it-->
20    <add key="IsSpellCorrectionEnabled" value="false" />
21    <add key="webpages:Version" value="3.0.0.0" />
22    <add key="webpages:Enabled" value="false" />
23    <add key="ClientValidationEnabled" value="true" />
24    <add key="UnobtrusiveJavaScriptEnabled" value="true" />
25  </appSettings>
26 <!--
27 For a description of web.config changes see http://go.microsoft.com/fwlink/?LinkId=235367.
28 -->
```

Bot Publication Web

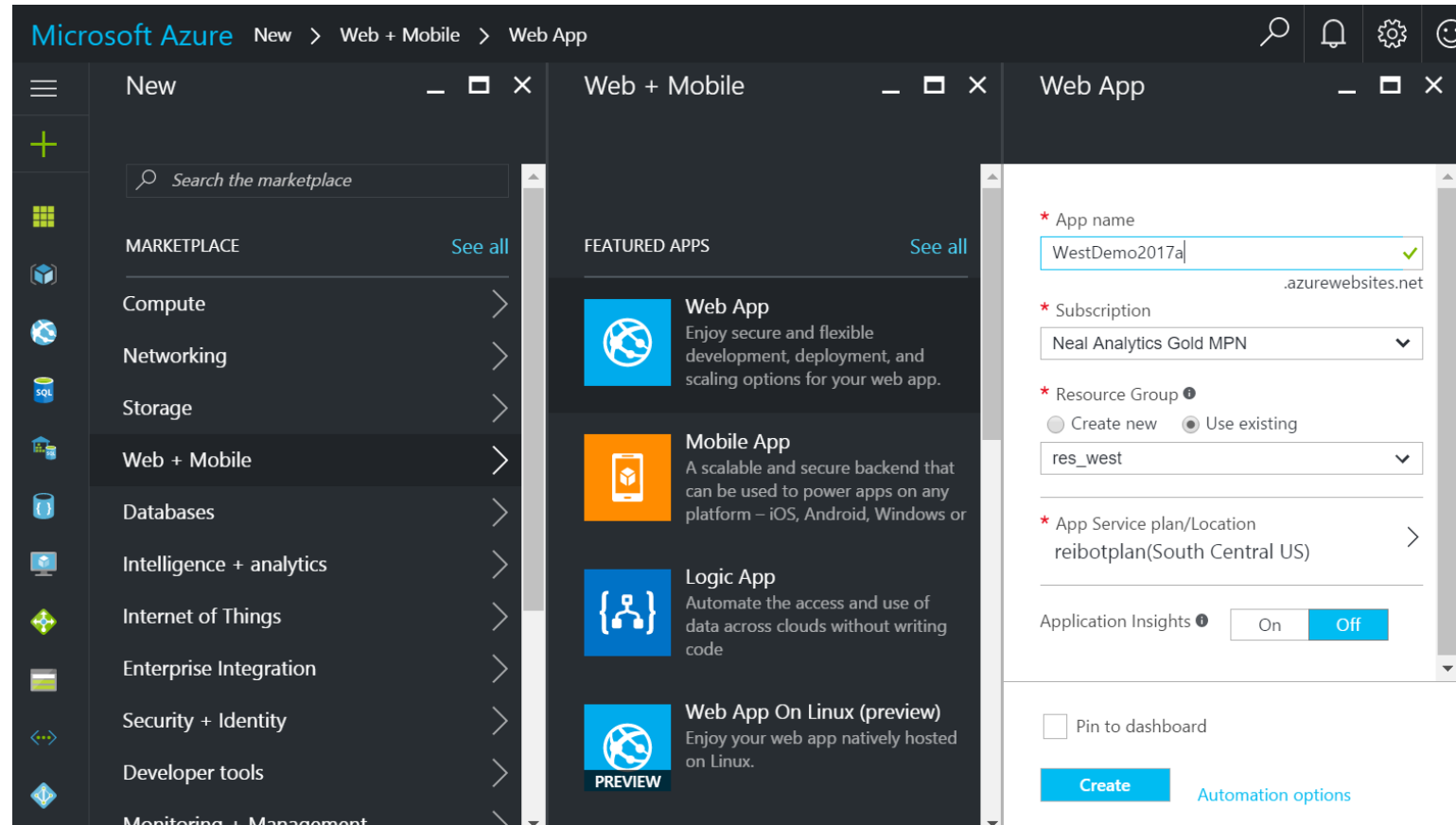
- Navigate to default.htm in Visual Studio
- Find the <a href> tag and replace with the <a href> tag for Skype in your notepad
- Find the <iframe> tag and replace it with the iframe tag in your notepad

This will enable Skype and the web interface for your bot



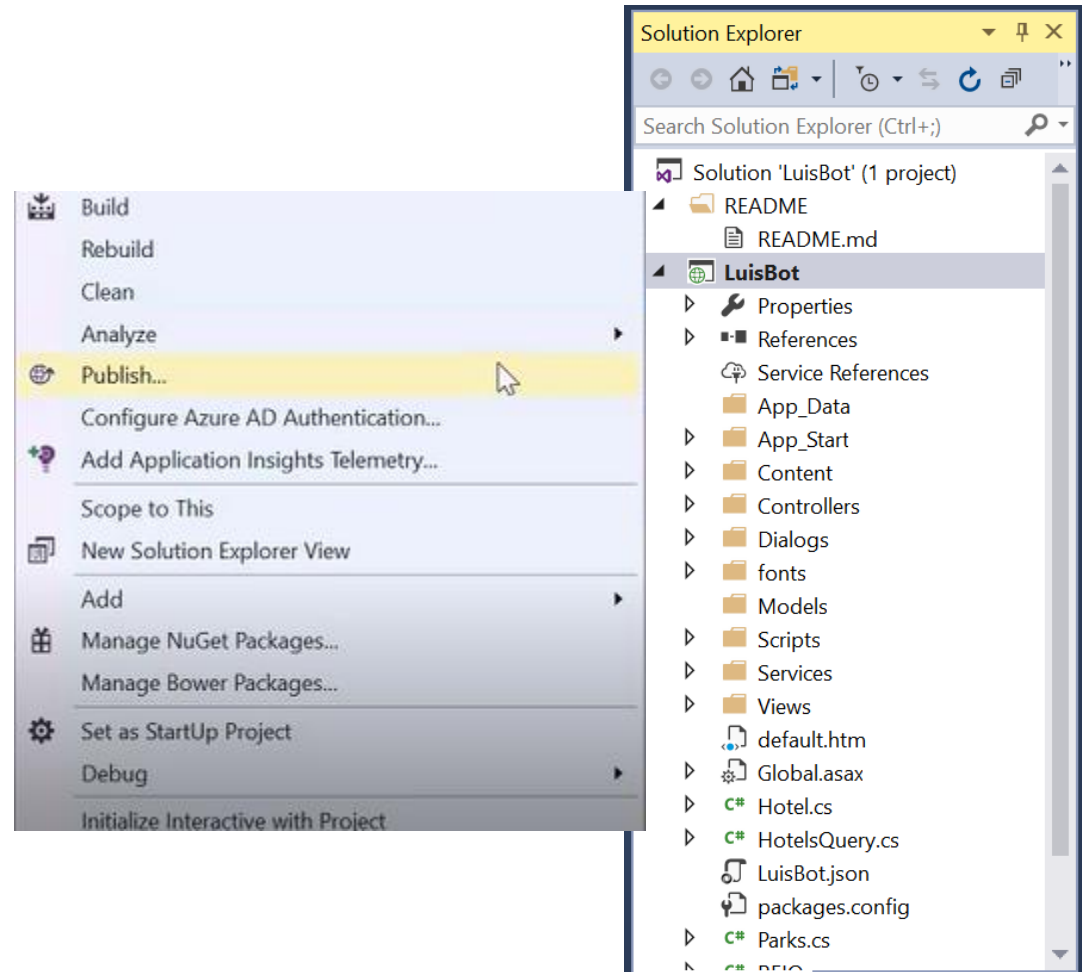
Bot Publication Azure Portal

- <https://portal.azure.com/>
- Click +
- Click “Web + Mobile”
- Web App



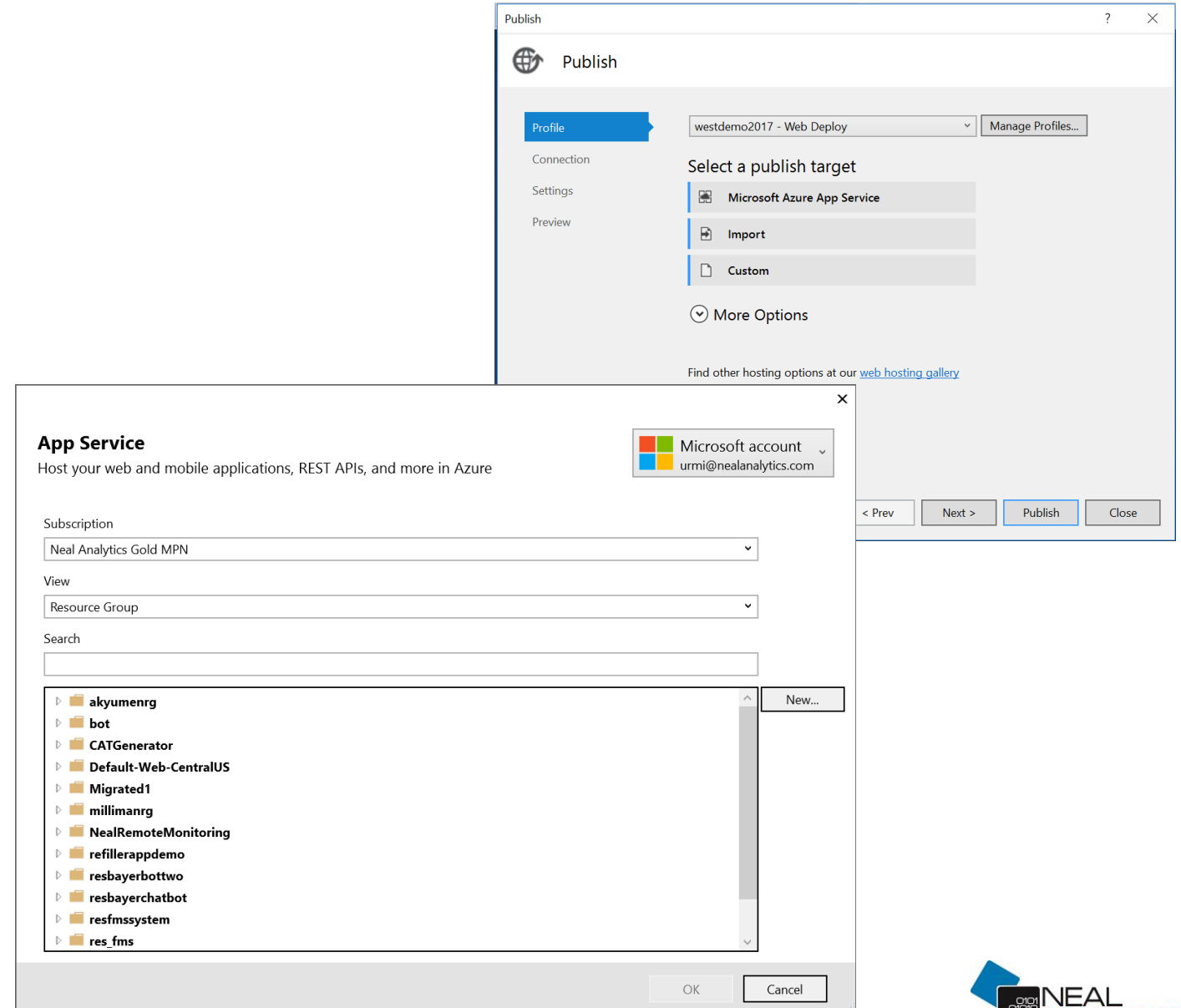
Bot Publication Web

- Right Click on your solution
- Click “Publish”



Bot Publication Web

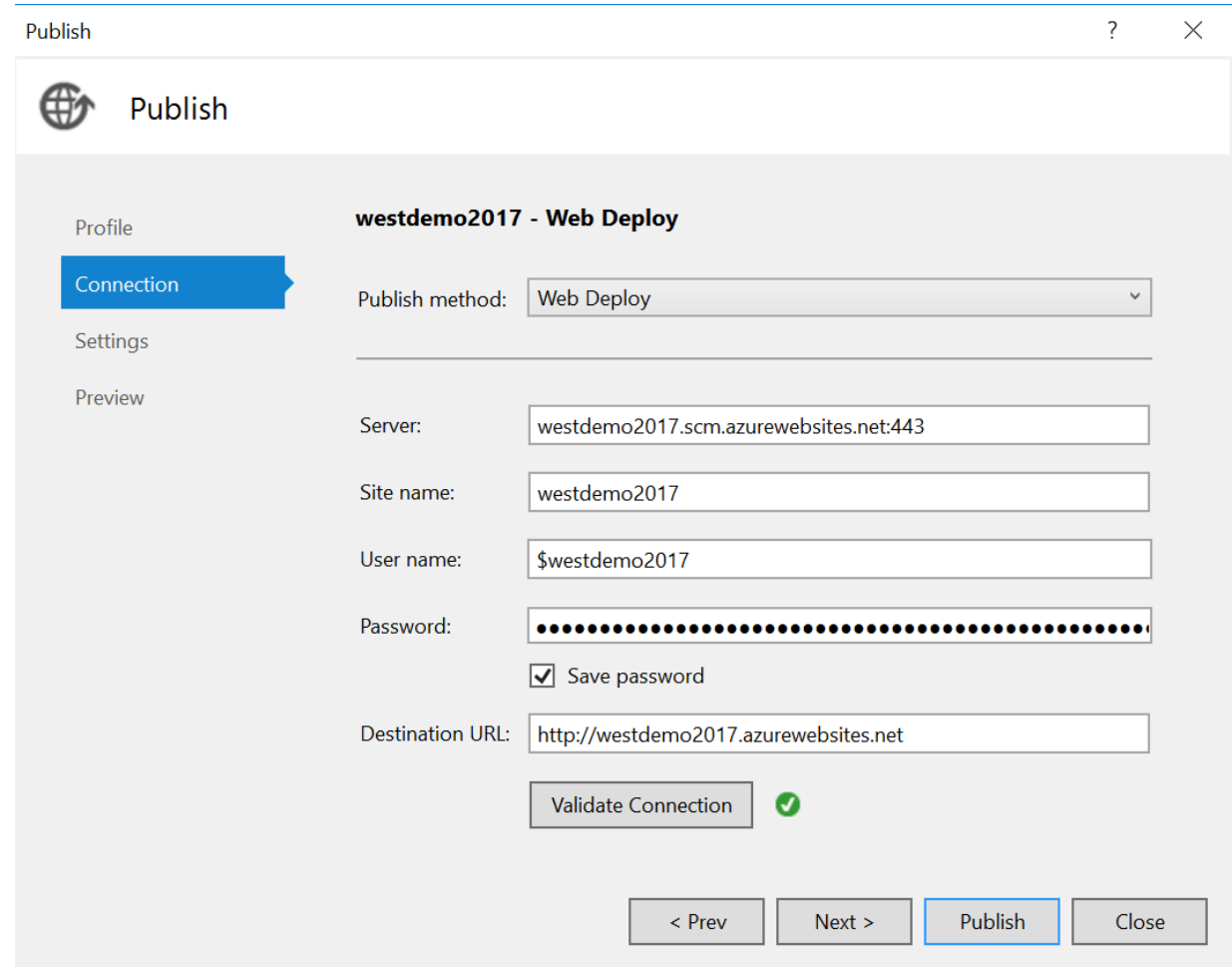
- Click Profile
- Click Manage Profiles
- Remove any profiles (these are unique to the machine)
- Click “Microsoft Azure App Service”
- Select an account and Sign in
- Select the resource group you just created or selected



Bot Publication Web

- Click “Validate Connection”
- Click “Publish”

Security Feature: You can only publish this app from this machine from now on!!!



The screenshot shows the 'Publish' dialog box in Visual Studio. The title bar says 'Publish'. On the left, there is a sidebar with 'Profile', 'Connection' (highlighted in blue), 'Settings', and 'Preview'. The main area is titled 'westdemo2017 - Web Deploy'. It contains the following fields and controls:

- Publish method:** A dropdown menu set to 'Web Deploy'.
- Server:** A text box containing 'westdemo2017.scm.azurewebsites.net:443'.
- Site name:** A text box containing 'westdemo2017'.
- User name:** A text box containing '\$westdemo2017'.
- Password:** A text box filled with dots, with a 'Save password' checkbox checked below it.
- Destination URL:** A text box containing 'http://westdemo2017.azurewebsites.net'.
- Validate Connection:** A button with a green checkmark icon next to it.
- Navigation buttons:** '< Prev', 'Next >', 'Publish' (highlighted in blue), and 'Close'.


Bot is Live

- Click “Validate Connection”
- Click “Publish”

Security Feature: You can only publish this app from this machine from now on!!!

- Return to <https://dev.botframework.com/bots?id=erichullandertest>
- Click “Edit”
- Specify an endpoint

⚠ You cannot test or publish your bot until you specify an endpoint

 EricHullandertest

Details

Bot handle
erichullandertest

Bot Framework Version
3.0

Messaging endpoint
Not set

Microsoft App ID
c6851ae8-631c-4911-ae11-44a236edc3ad

[Edit](#)

Test connection to your bot

[Test](#)

Configuration

Messaging endpoint:

Register your bot with Microsoft to generate a new App ID and password

[Manage Microsoft App ID and password](#)

Paste your app ID below to continue

c6851ae8-631c-4911-ae11-44a236edc3ad