Building a Bot for IBM Watson Workspace using Node-RED on IBM Bluemix

Author: Benedikt Müller <benedikt.mueller@de.ibm.com>

Introduction 3

Prerequisites 3

Step by Step Guide 4

Create Space in IBM Watson Work Space 4

Access IBM Watson Work Services 4

Create a new IBM Watson Work Services App 5

Create a Node-RED instance on IBM Bluemix 6

Accessing Node-RED 10

Installing IBM Watson Work Services Nodes 10

Sending a Message to IBM Watson Workspace 12

Receiving Messages from IBM Watson Workspace 15

# Introduction

IBM Watson Workspace on the surface is a persistent group chat platform available in the browser and as mobile apps for iOS and Android. But looking beyond the surface, it is far more than that. IBM Watson Workspace is based on IBM Watson Work Services, an extensive set of extensibility points and APIs provided to organizations and developers, enabling them to leverage IBM Watson Workspace as a conversation based user interface to surface cognitive capabilities in an extremely easy to user manner. Those cognitive capabilities can provide access to internal or external backend systems, surface relevant information or integrate with business processes.

Node-RED[[1]](#footnote-1), an open source project that allows “flow-based programming for the Internet of Things” provides a very easy way to rapidly build prototypes of cognitive solutions for IBM Watson Work Services. And by using the collection of predefined nodes specifically built for IBM Watson Work Services (node-red-contrib-wws[[2]](#footnote-2)), this process is simplified even further.

# Prerequisites

This guide will walk you through creating a bot app that can be added to spaces in IBM Watson Workspace and is able to listen and react to conversations happening in those spaces. The app will be created using Node-RED running on IBM Bluemix.

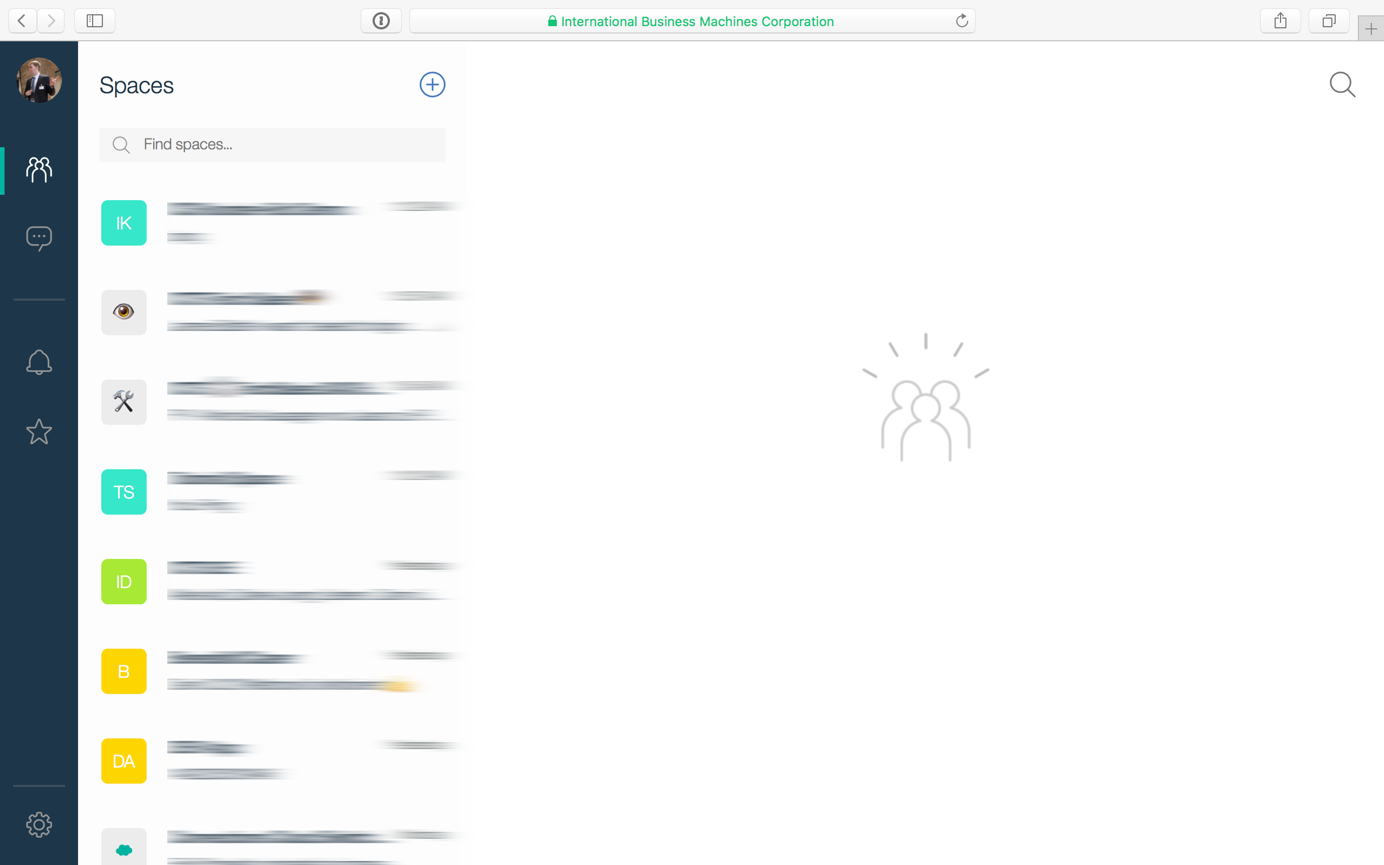
What you need to follow this guide:

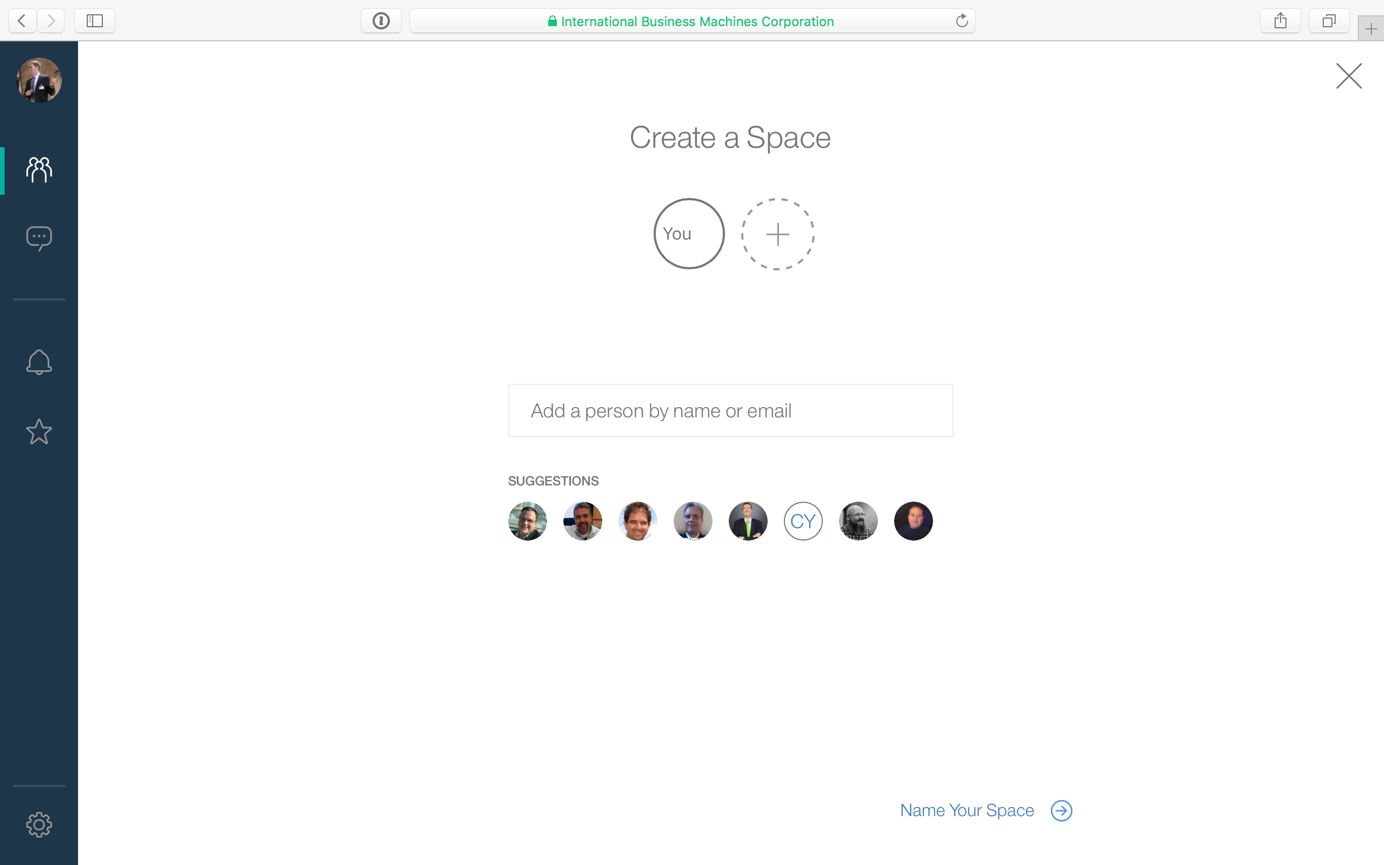
* An IBM ID[[3]](#footnote-3) with access to IBM Watson Workspace and IBM Bluemix
* Basic knowledge of JavaScript

# Step by Step Guide

## Create Space in IBM Watson Work Space

Access IBM Watson Workspace via <https://workspace.ibm.com> and create a new space to that will be used to test the app we’re developing.

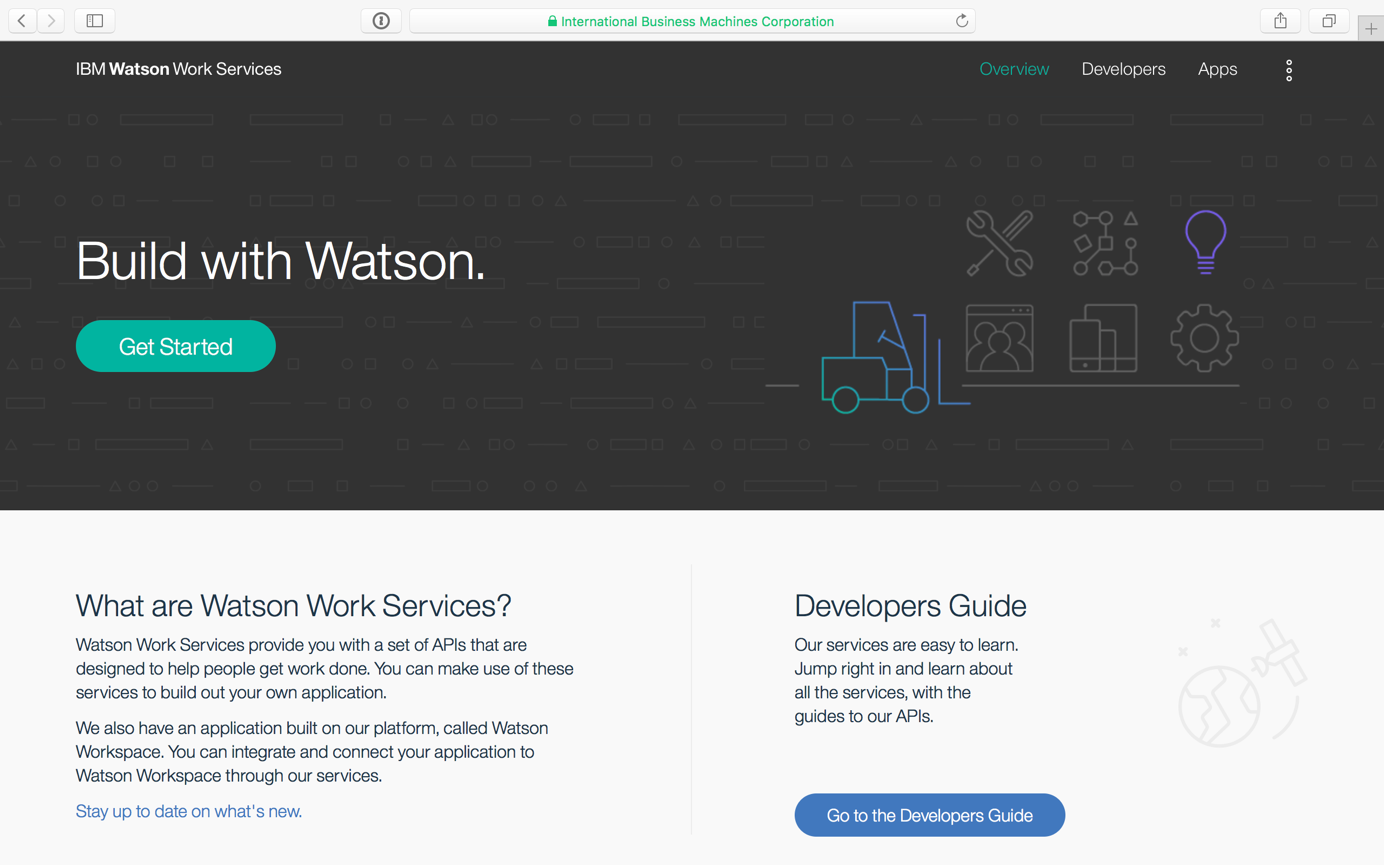




## Access IBM Watson Work Services

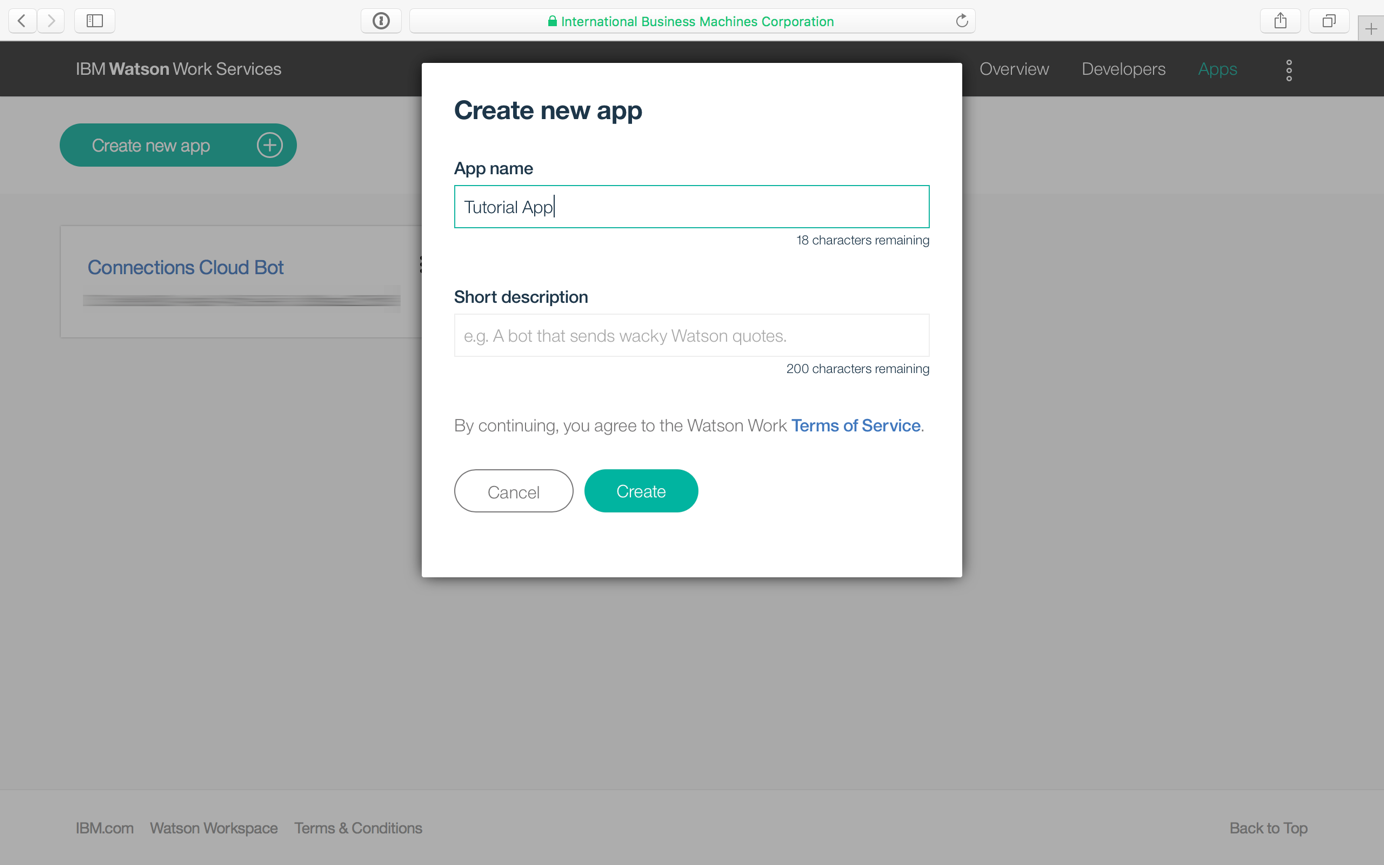
Within the settings menu, click on “Developers” to access IBM Watson Work Services.



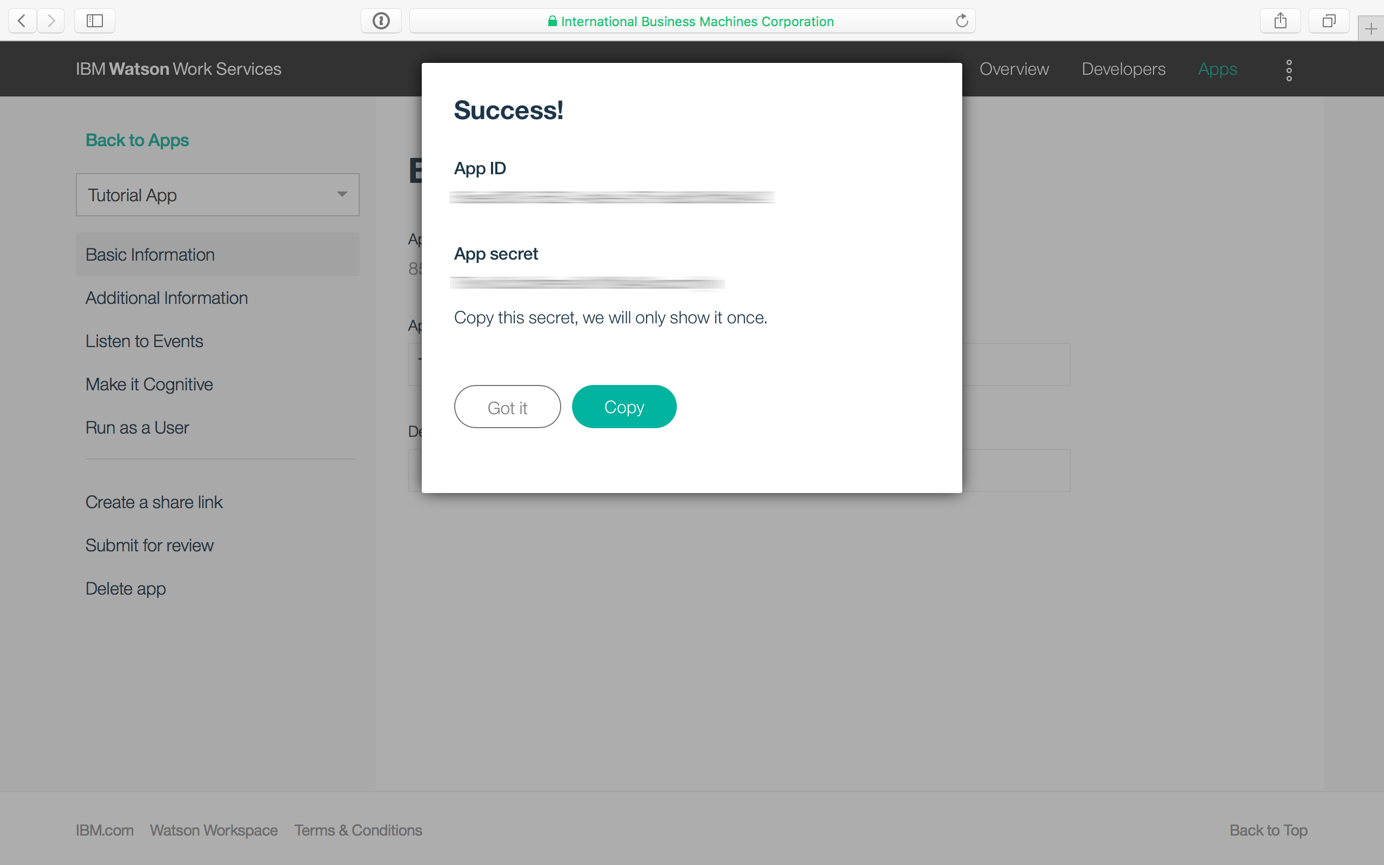


## Create a new IBM Watson Work Services App

Click the button “Create new app” to create a new IBM Watson Work Services app. Enter a name and description for the app.

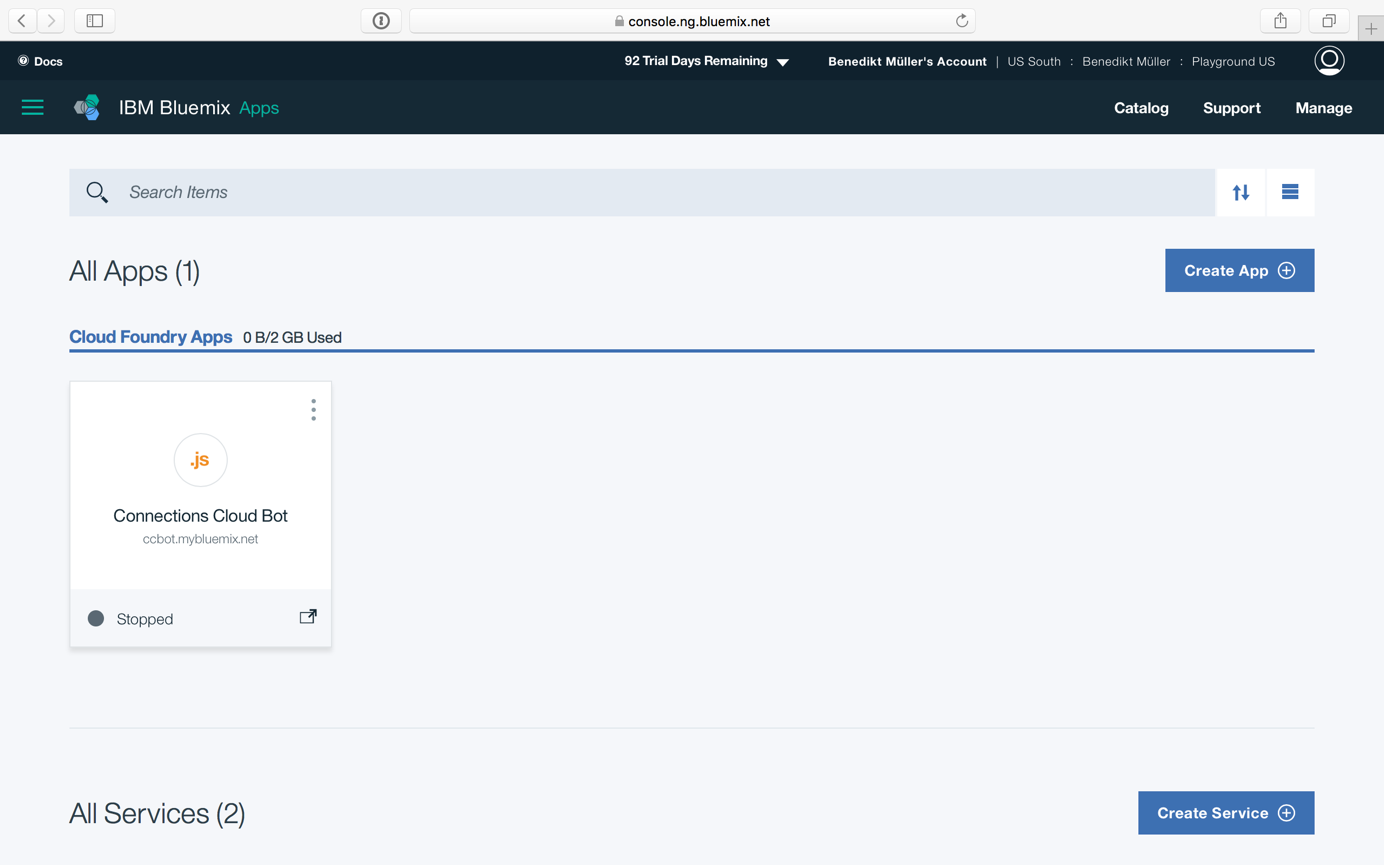


Note down the App ID and the App Secret. Saving the App Secret somewhere is especially important, because you will not be able to view it after closing this dialogue.

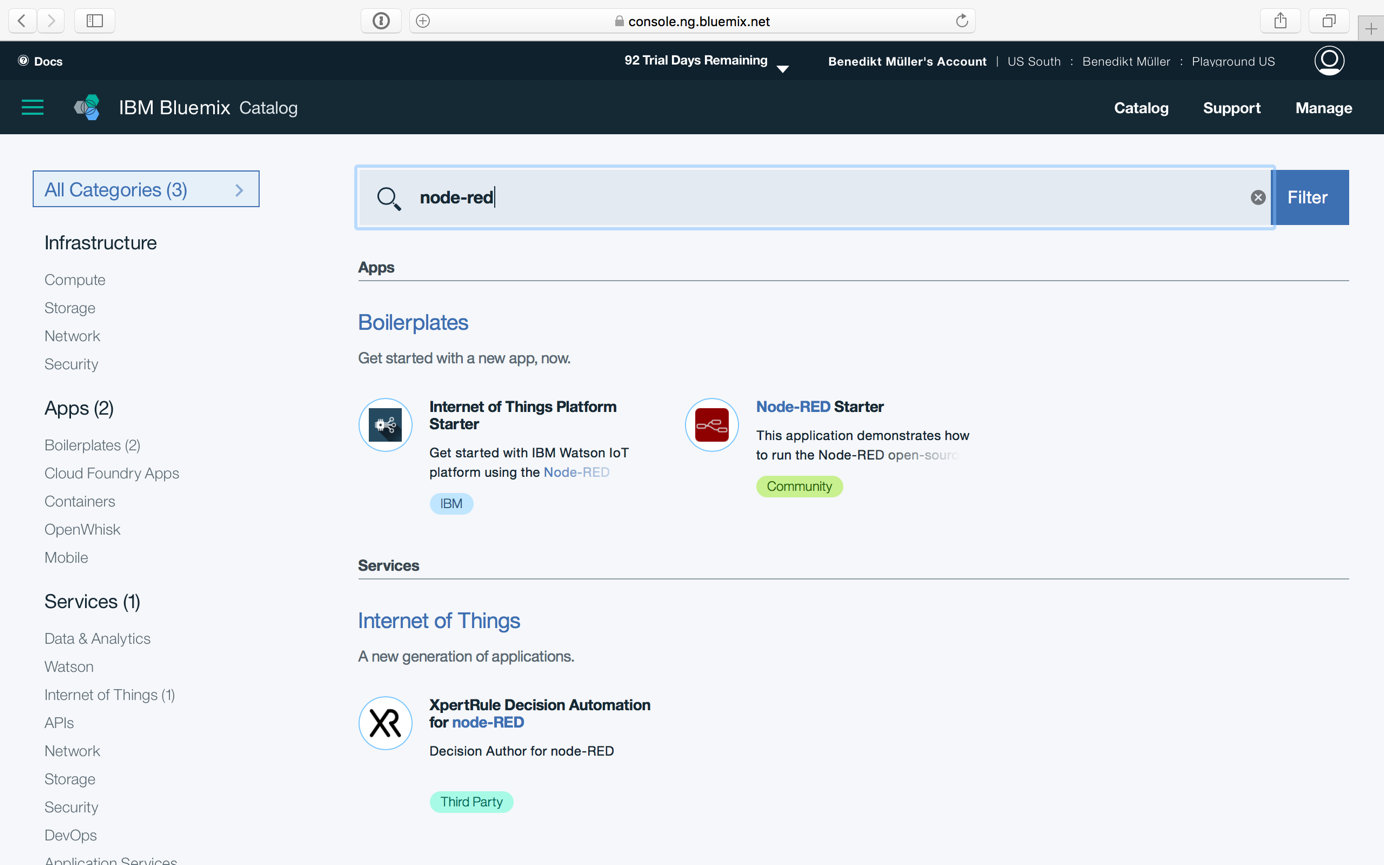


## Create a Node-RED instance on IBM Bluemix

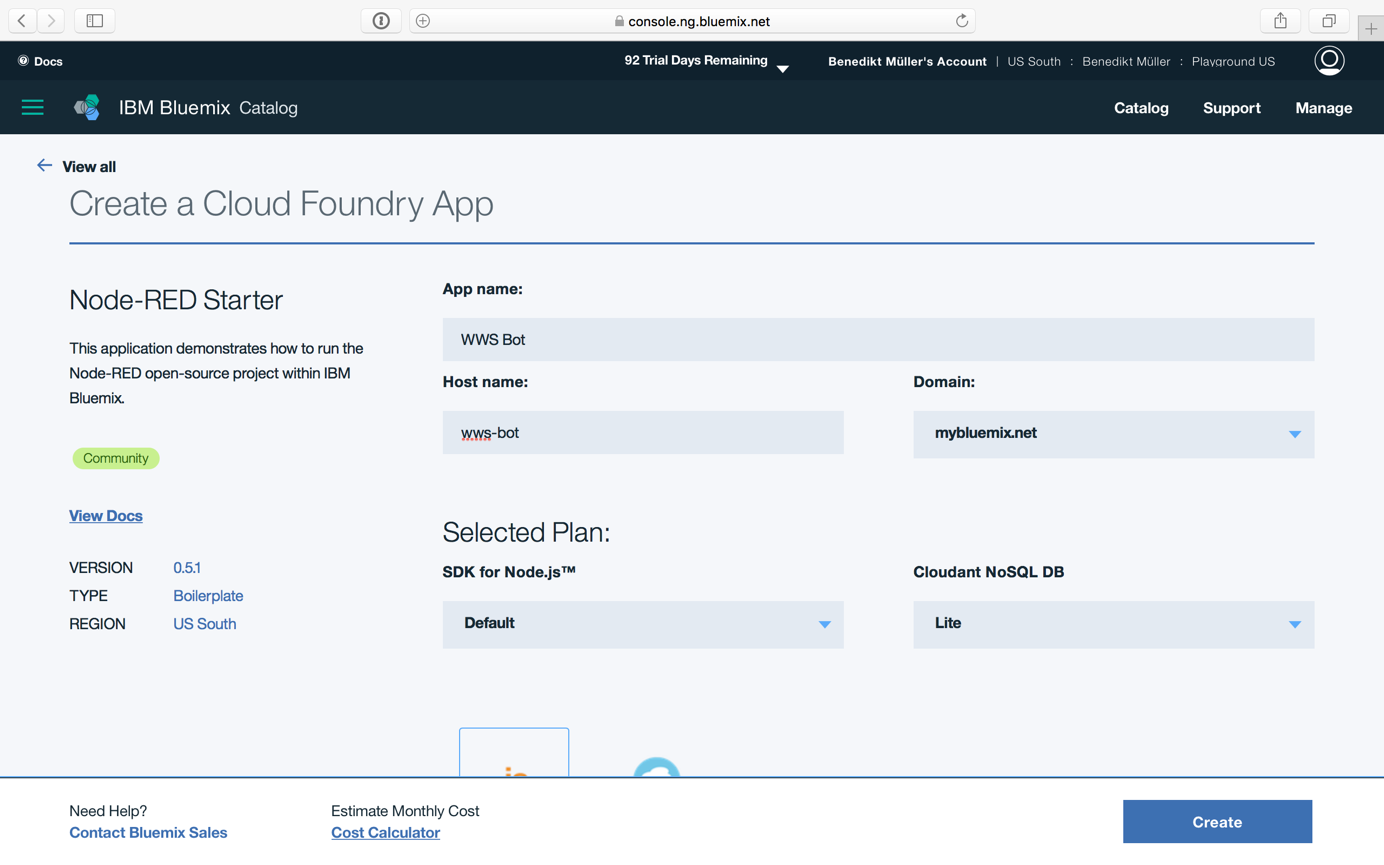
Log into the IBM Bluemix console via <https://console.ng.bluemix.net>.



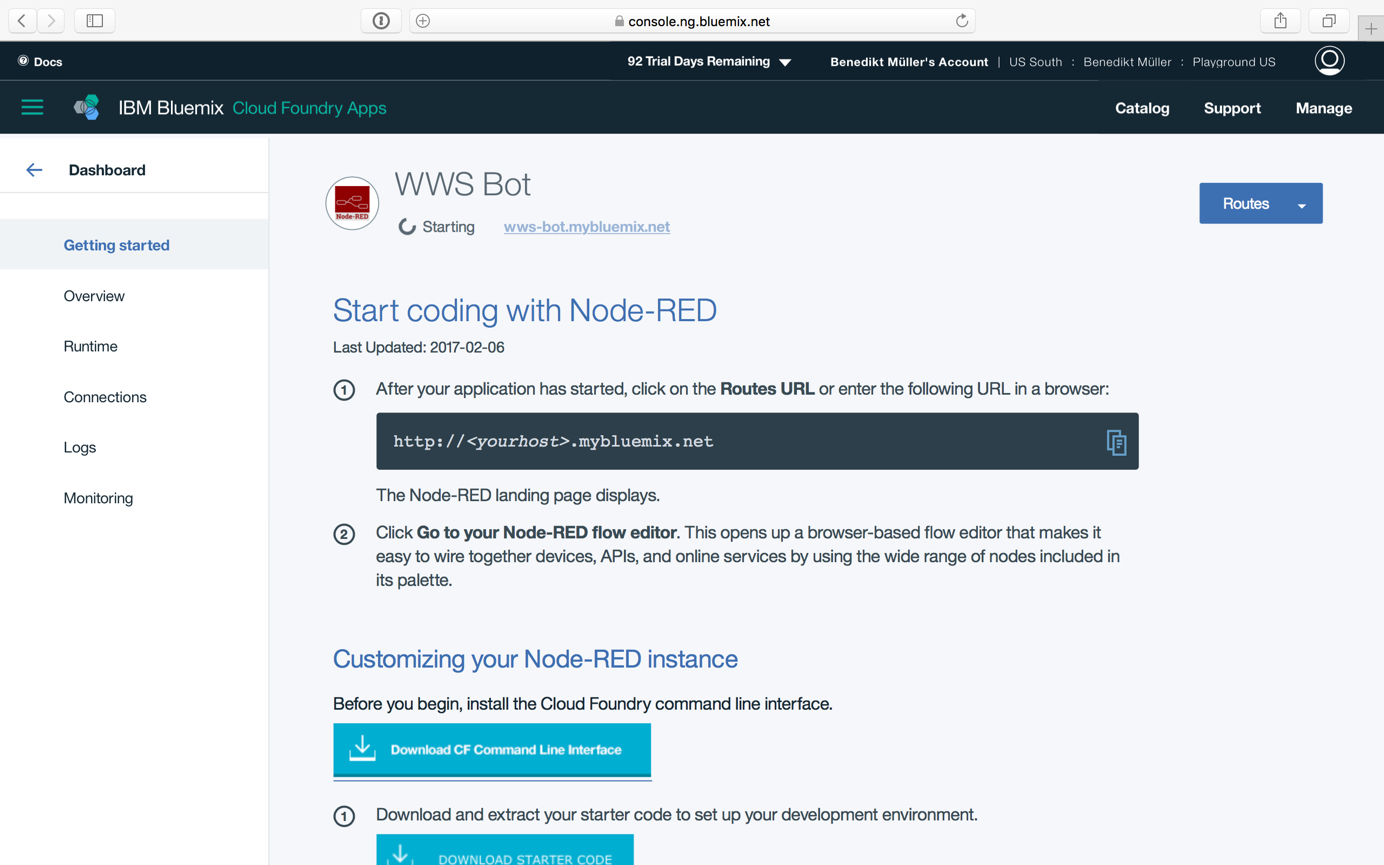
In the overview of your Cloud Foundry Apps, click the “Create App” button to view the catalog.



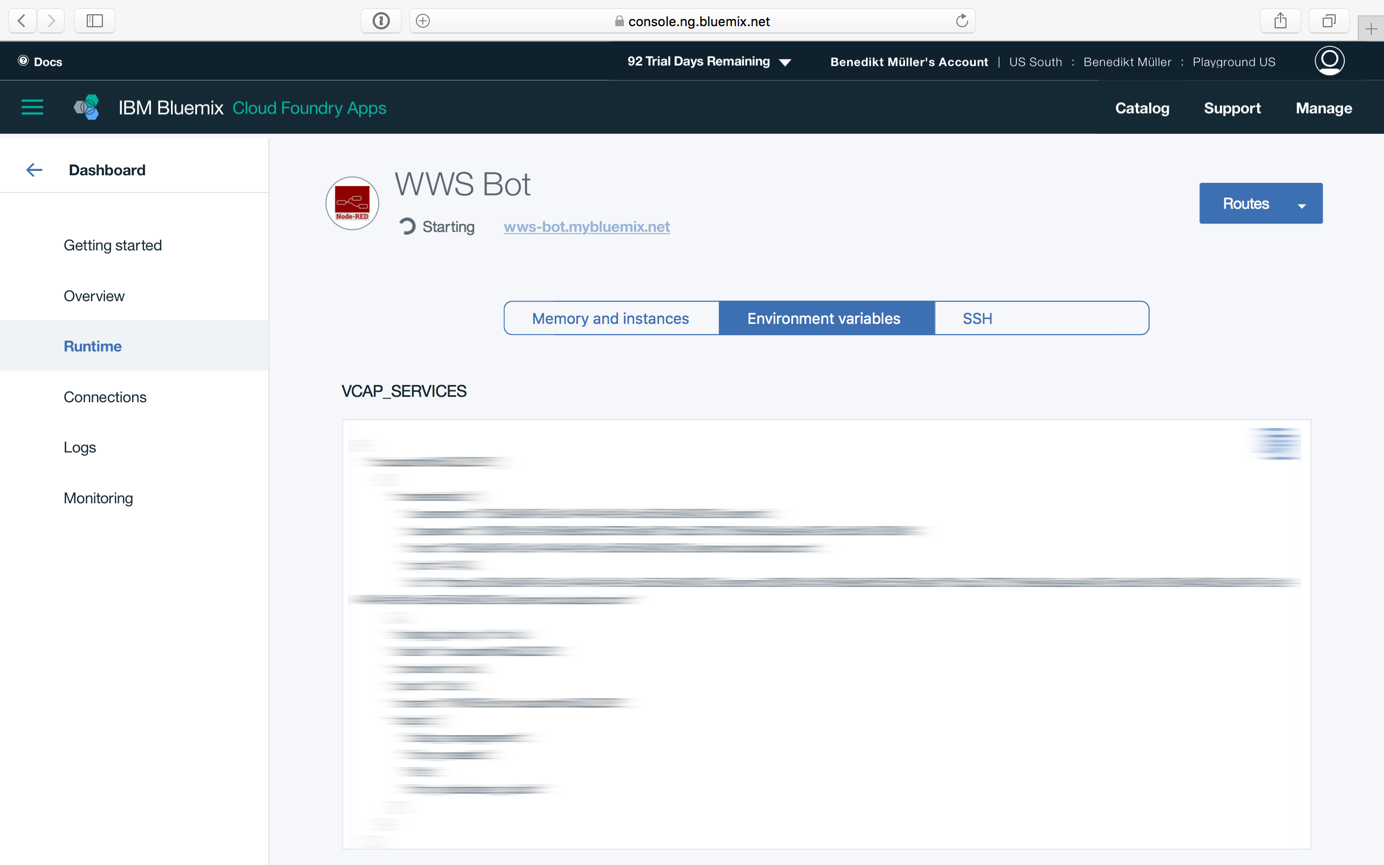
Search for “node-red” and click the “Node-RED Starter” boilerplate.



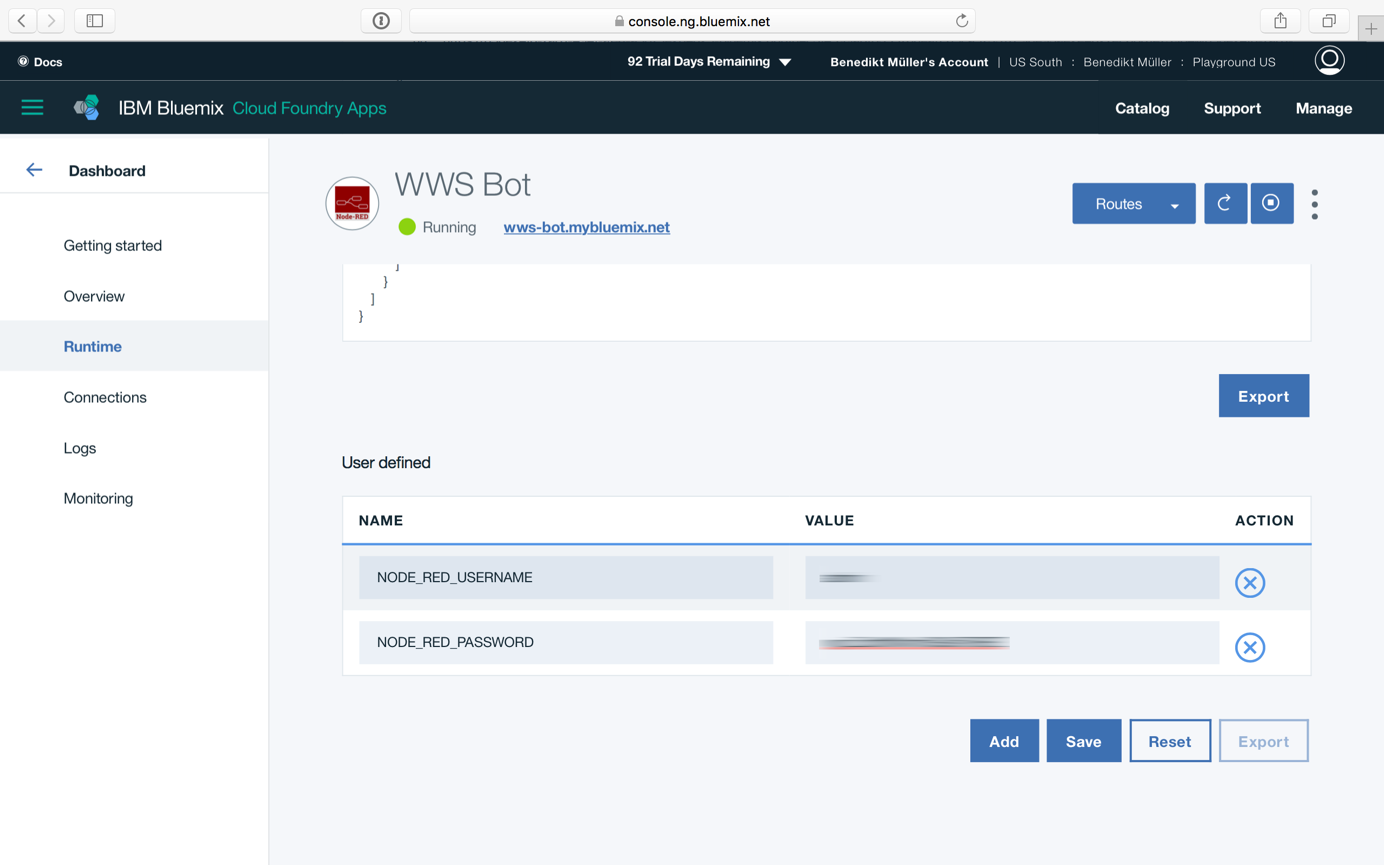
Choose an app name and a host name and create a Cloud Foundry App.



One the Cloud Foundry App is deployed, click on “Runtime” in the left hand navigation and then on “Environment variables”:



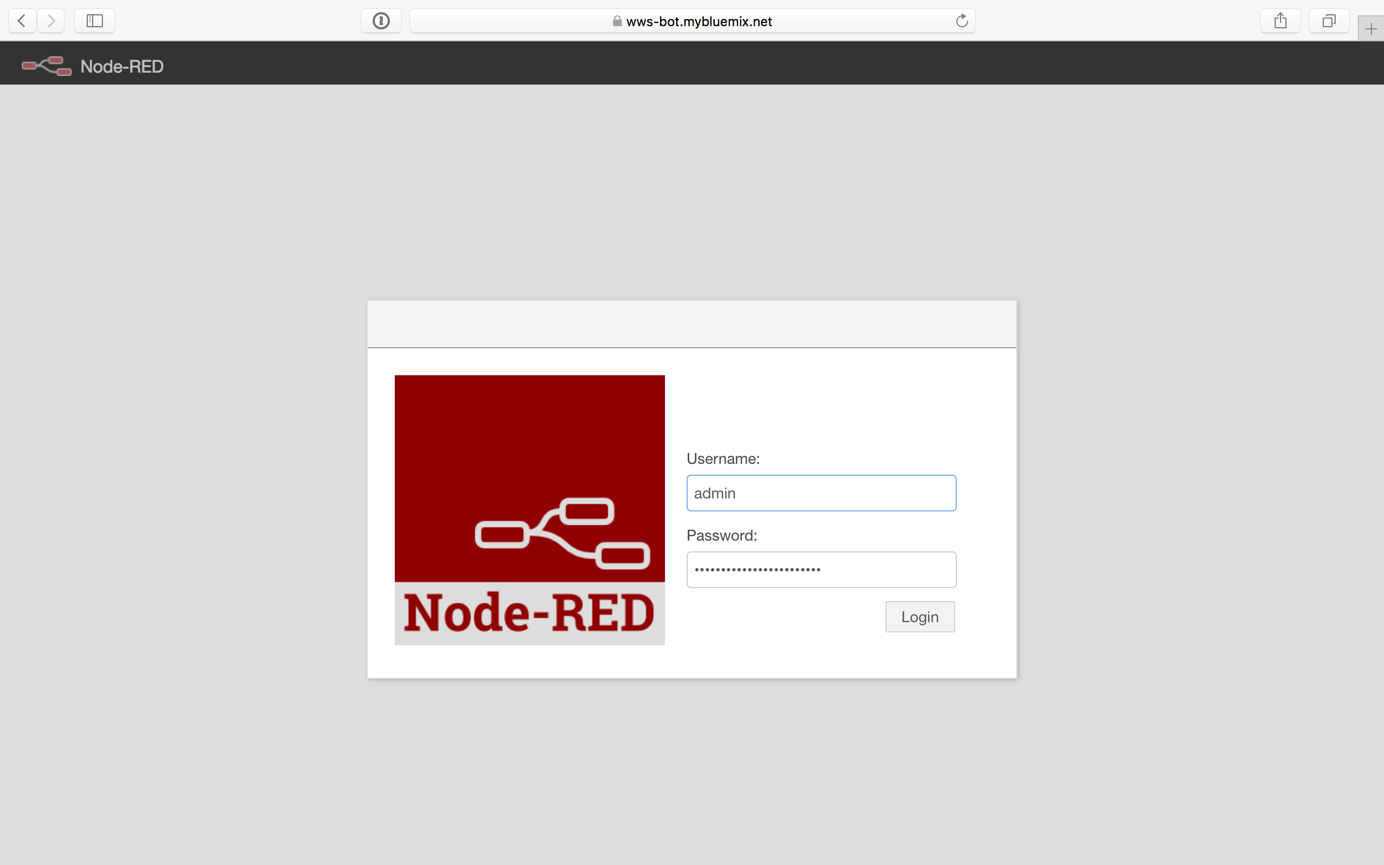
Scroll down to create two new environment variables to password protect your Node-RED instance. Enter a username and password of your choice.



Click “Save” and restart the app afterwards.

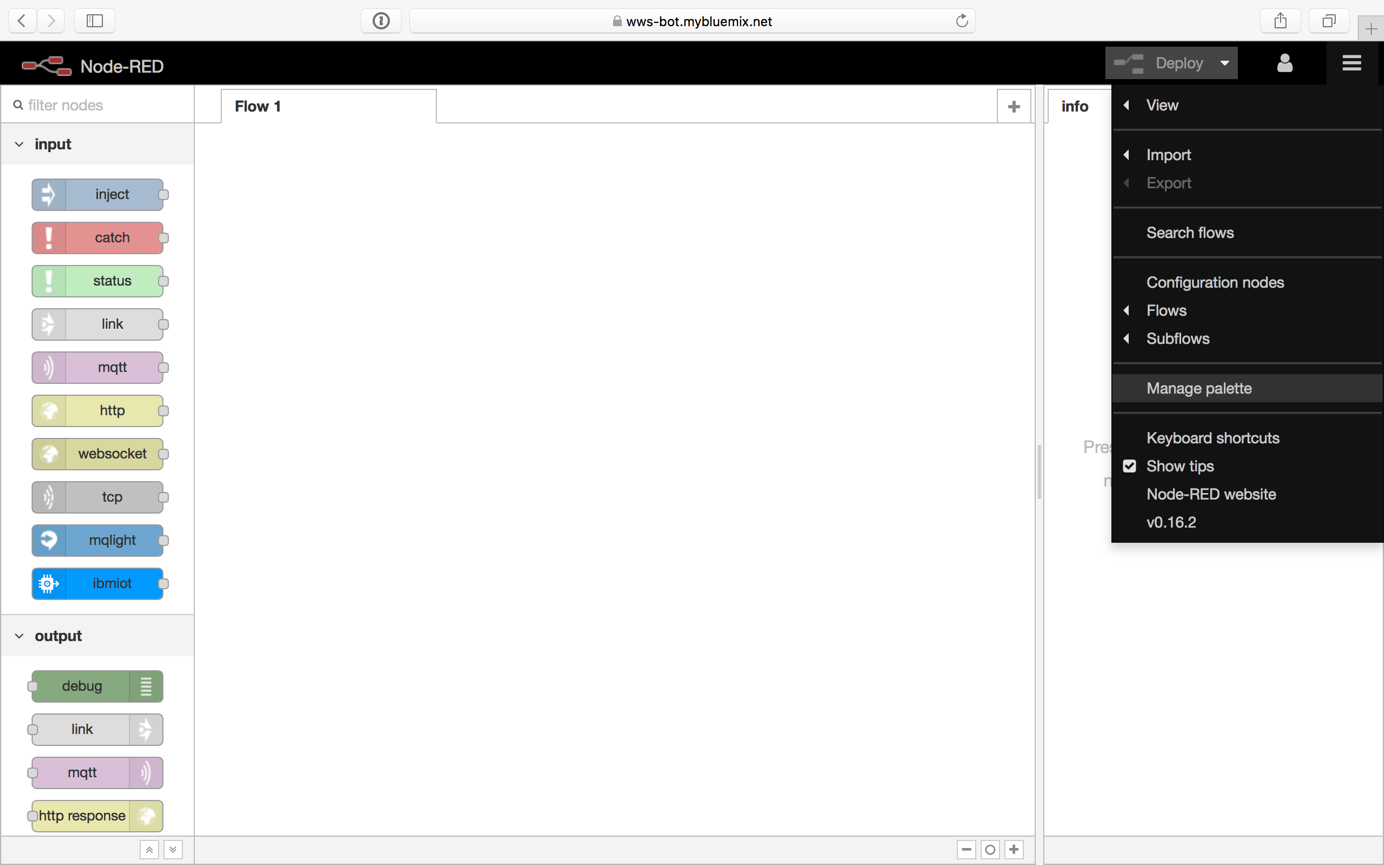
## Accessing Node-RED

Go to your Node-RED instance using the host name you provided earlier (you can click the link to your app within the IBM Bluemix console).



## Installing IBM Watson Work Services Nodes

Using the menu in the upper right corner, go to the “Manage palette” option.



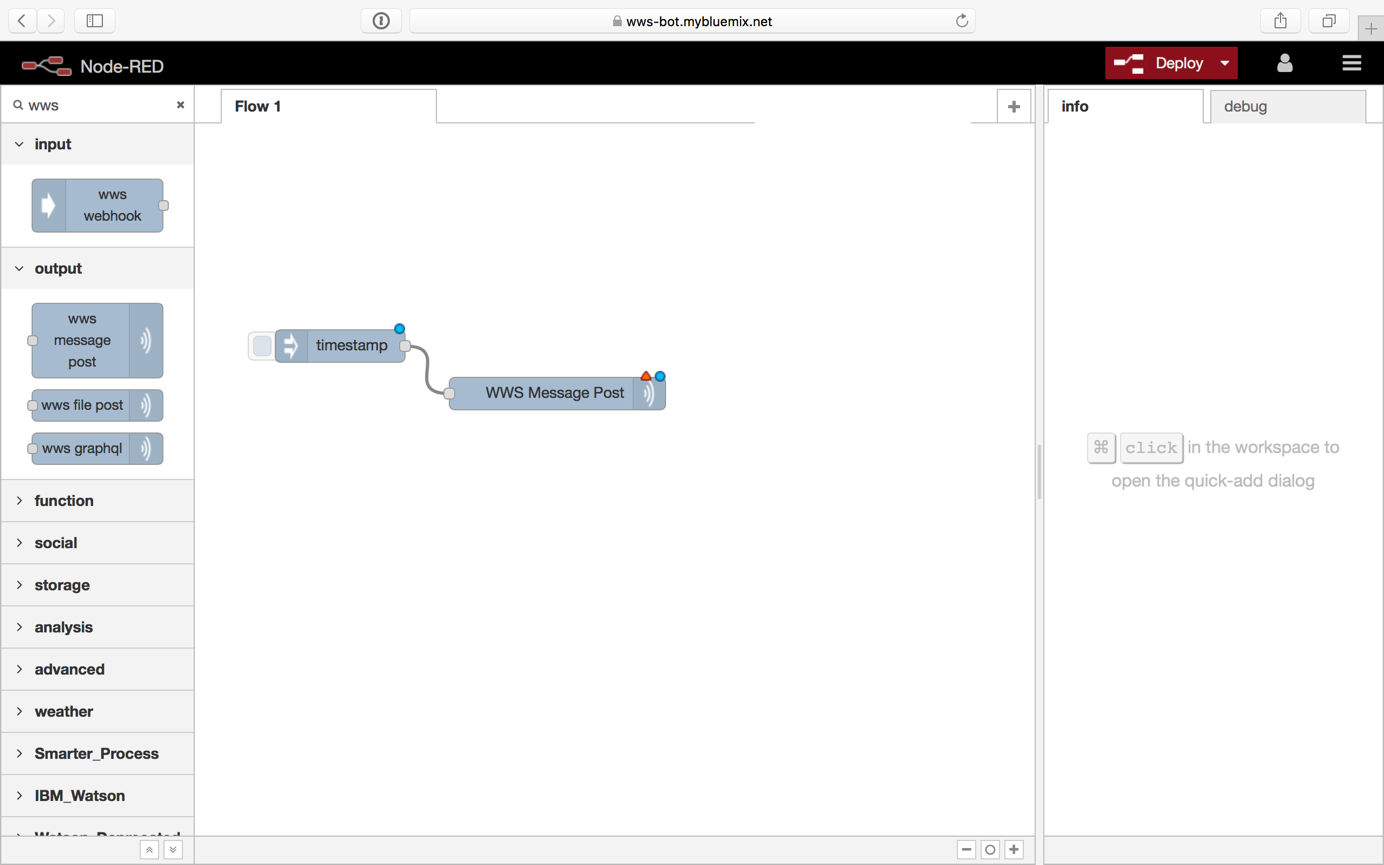
In the “Install” tab, search for “wws” and install the “node-red-contrib-wws” package.



Note: There is no documentation in this package itself (yet), that is what this document is for. I’m working on documenting the nodes themselves so people can use them without an accompanying guide like this one.

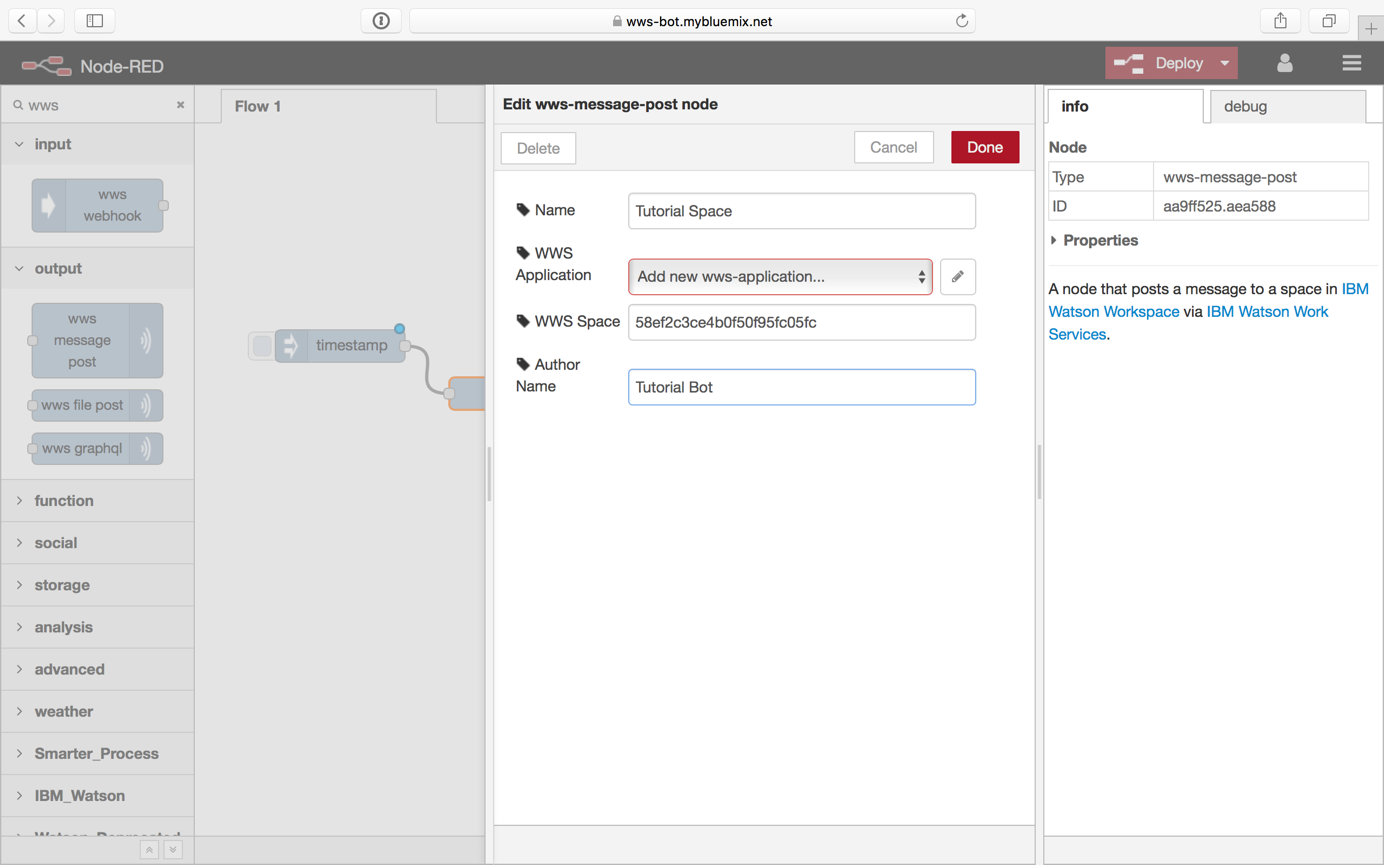
## Sending a Message to IBM Watson Workspace

After installing the package, you can add WWS nodes to your flow. Add an “Inject” node and the “wws meesage post” node to test posting into a space within IBM Watson Workspace.

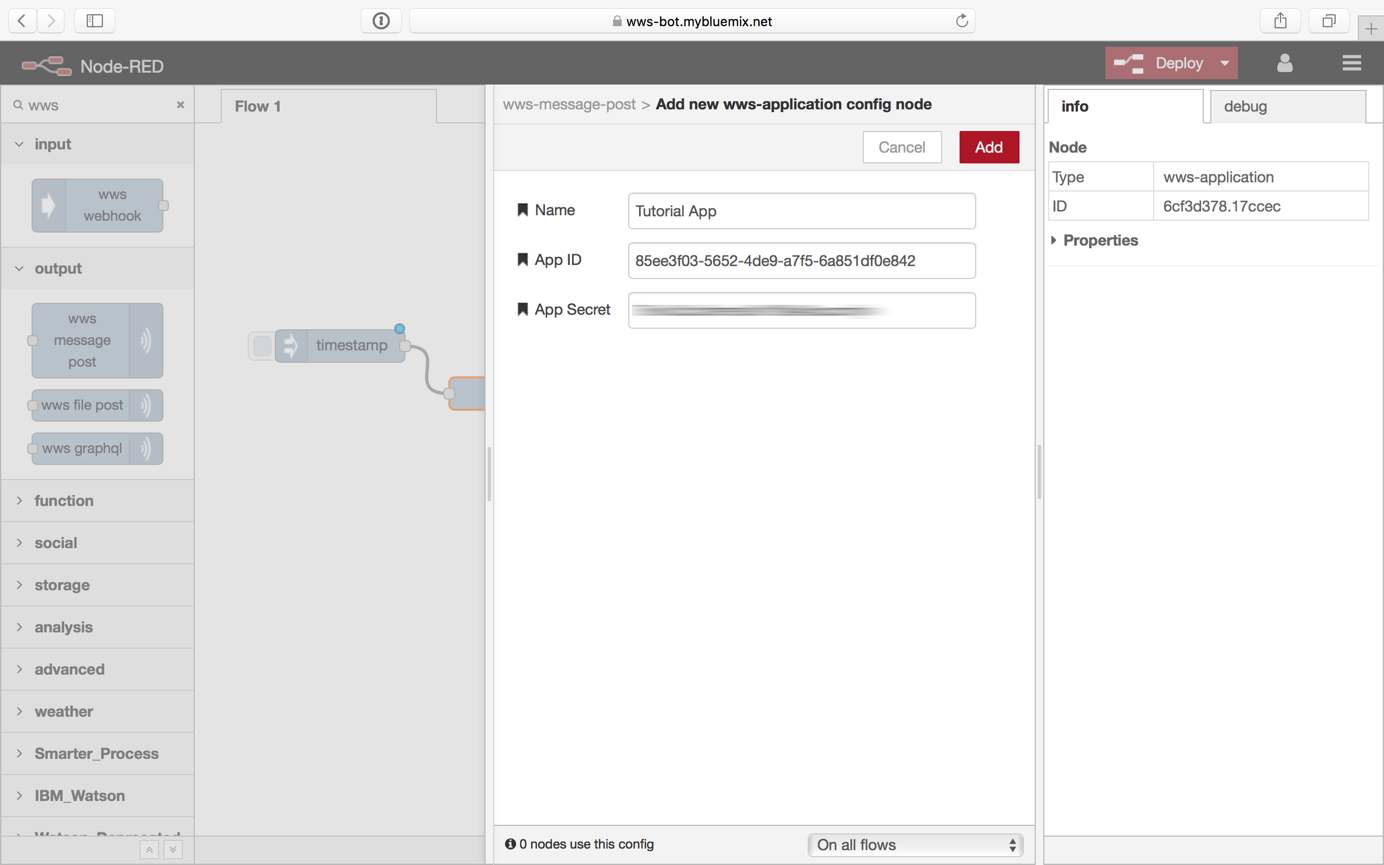


After drawing a connection between the two nodes, double click the “WWS Message Post” node to open its configuration pane.

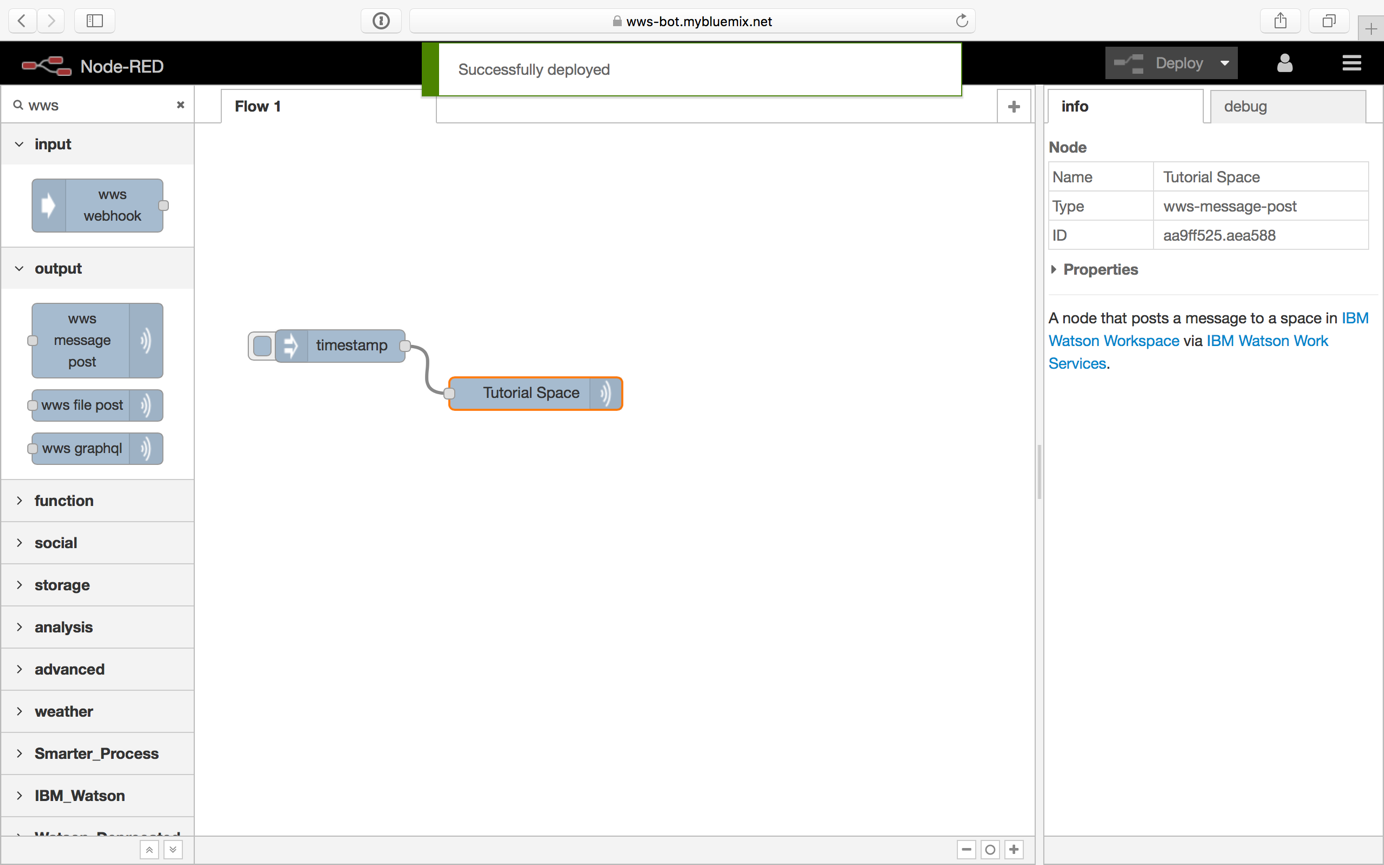
Enter a name for the node as well as a space ID and an author name. The author name will be shown as the author of the message we’re going to post into the space. The space ID itself can be drawn from the URL when accessing a space via the browser (e.g. 58ef2c3ce4b0f50f95fc05fc is the space ID of <https://workspace.ibm.com/space/58ef2c3ce4b0f50f95fc05fc)>.



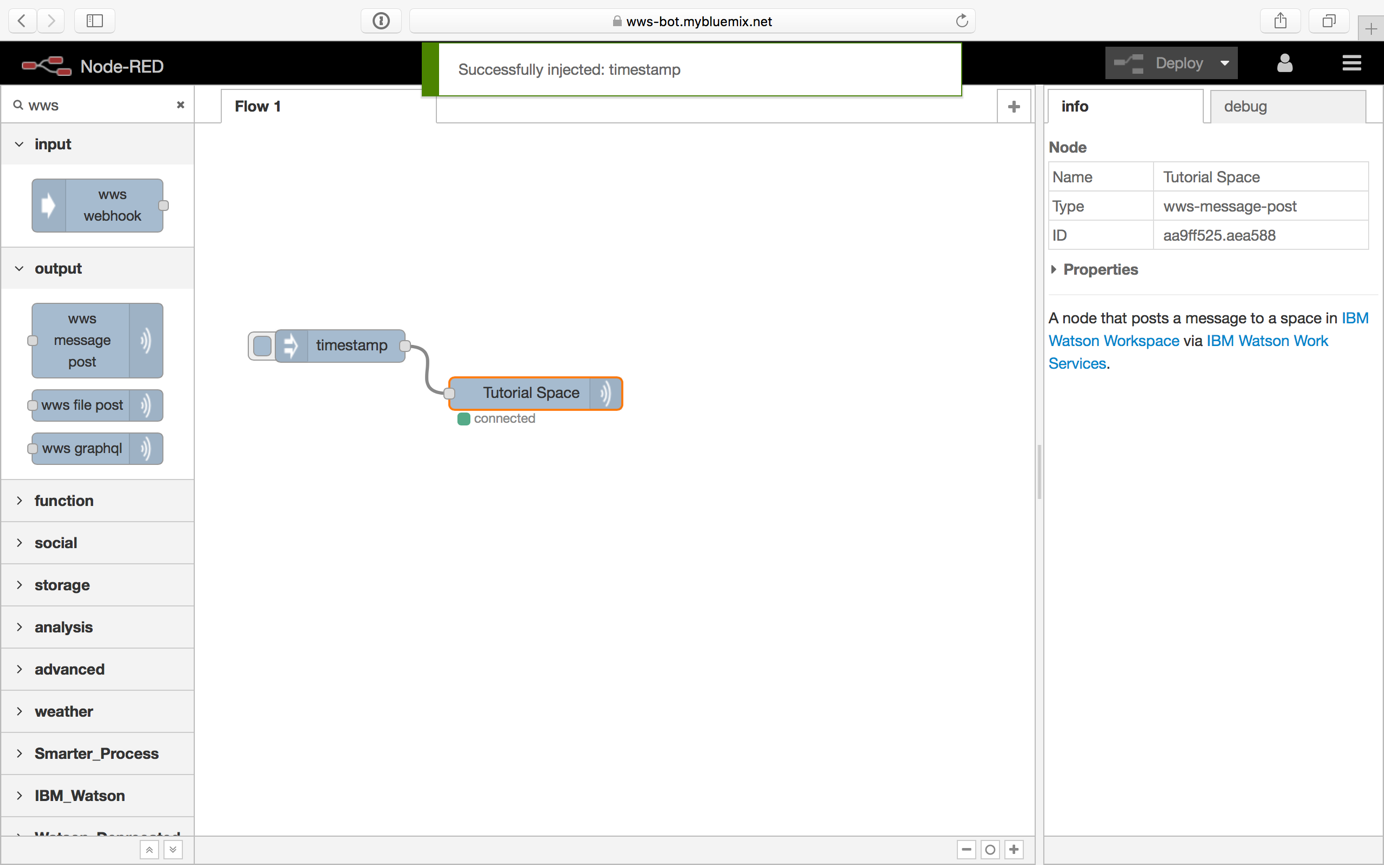
Click the pencil button next to the “WWS Application” dropdown. In the following form, enter the App ID and App Secret of the IBM Watson Work Space app you created earlier.



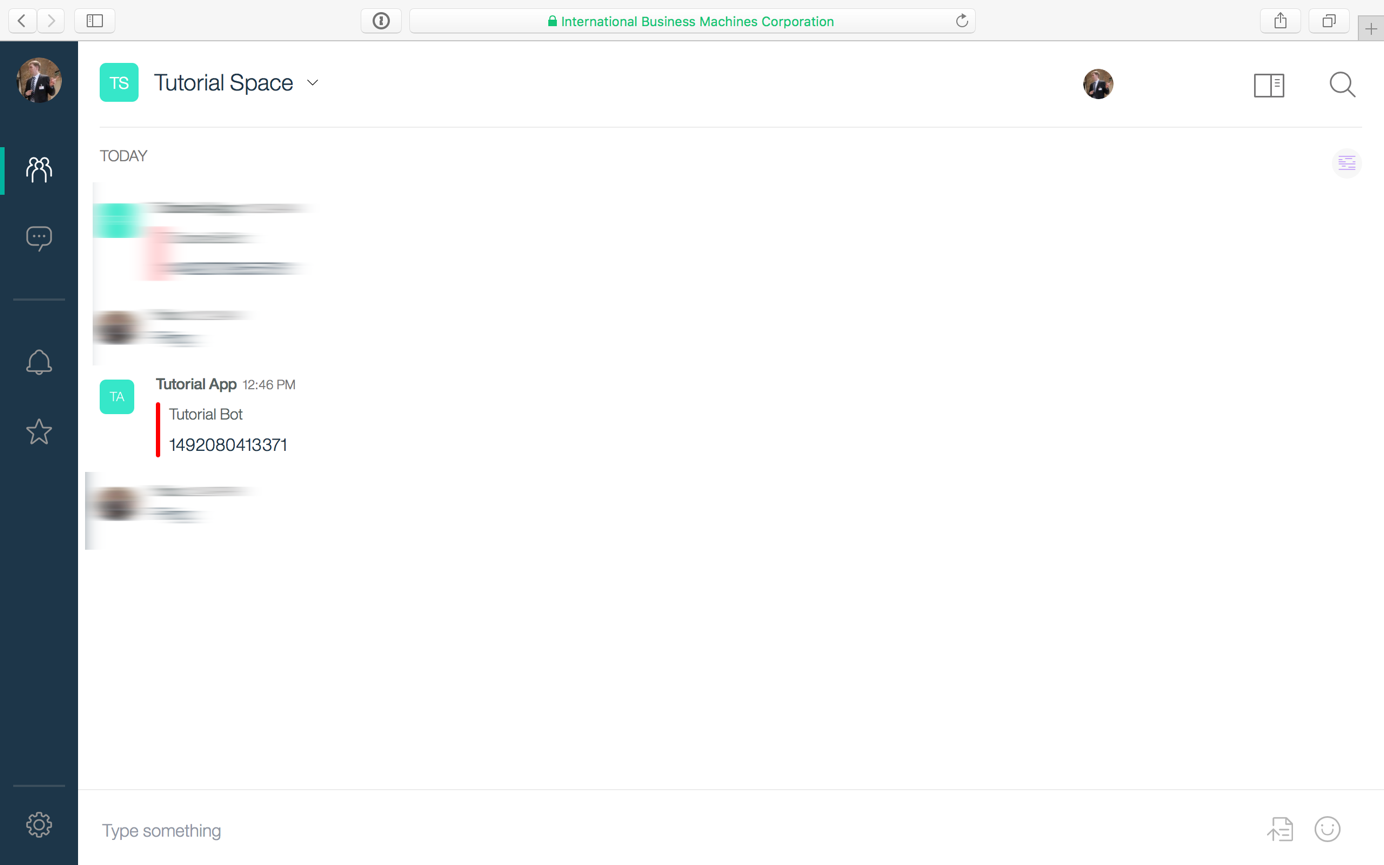
Click “Add” in this screen and then “Done” in the next screen. Your node is now configured and you can deploy your flow by clicking the “Deploy” button in the top right.



Now, clicking the little square left of the “Inject” node (by default labeled “timestamp” because it will send a timestamp) will send a message to the “Tutorial Space” node, which will then pass on this message to IBM Watson Workspace.

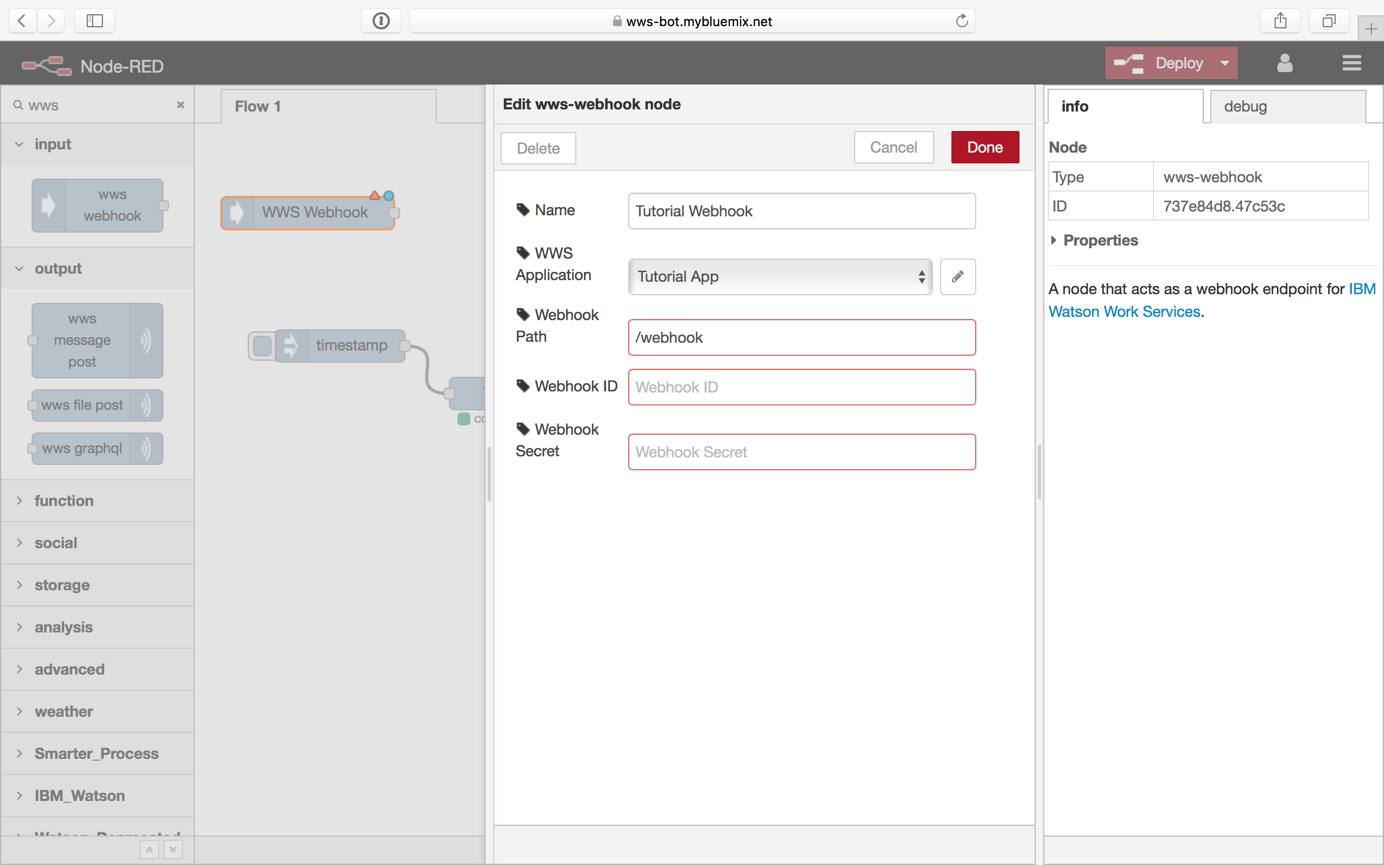


After clicking the small square, the “Tutorial Space” node should show a small green indicator saying “connected”. Also, you should be able to see the message within IBM Watson Workspace.

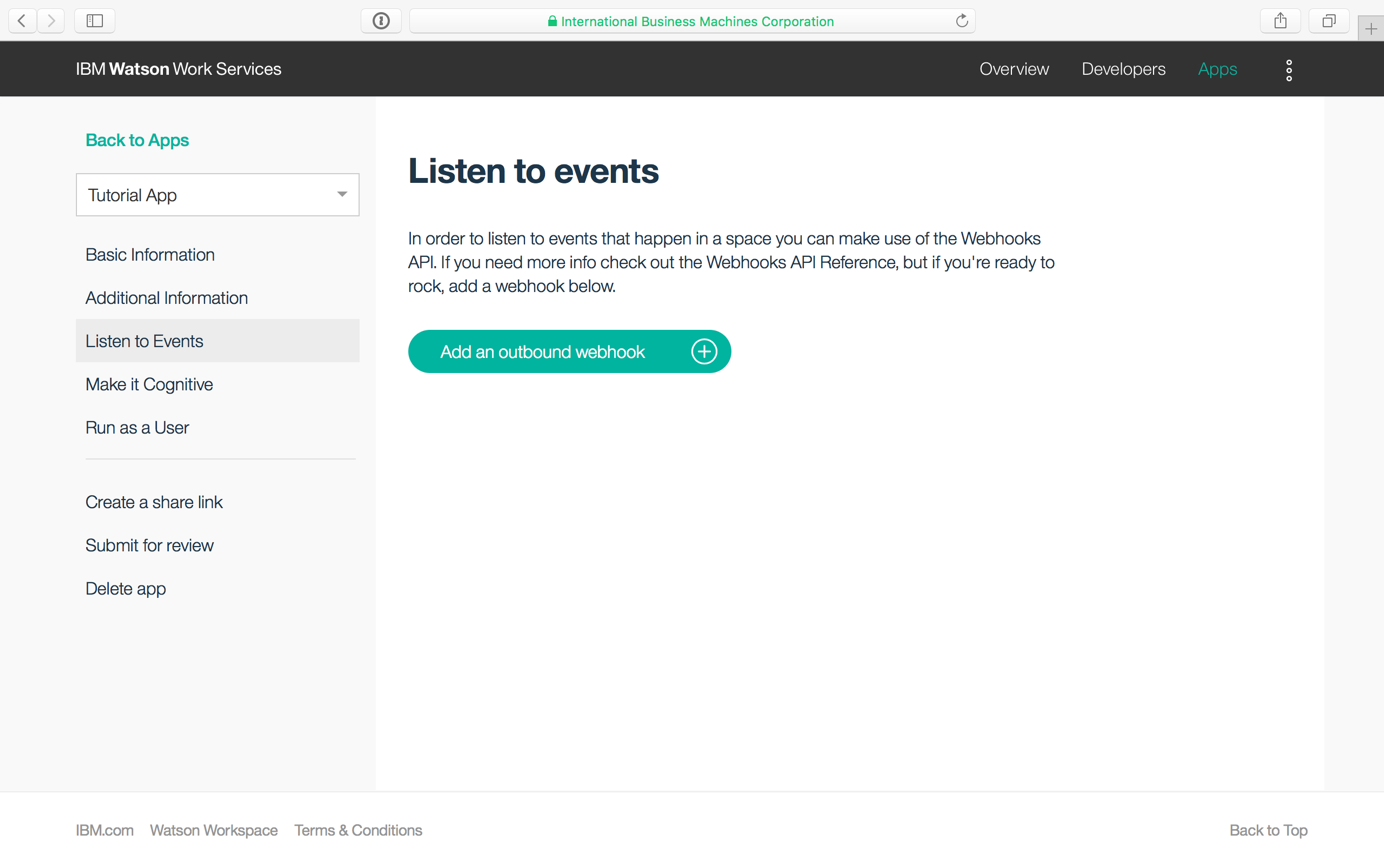


## Receiving Messages from IBM Watson Workspace

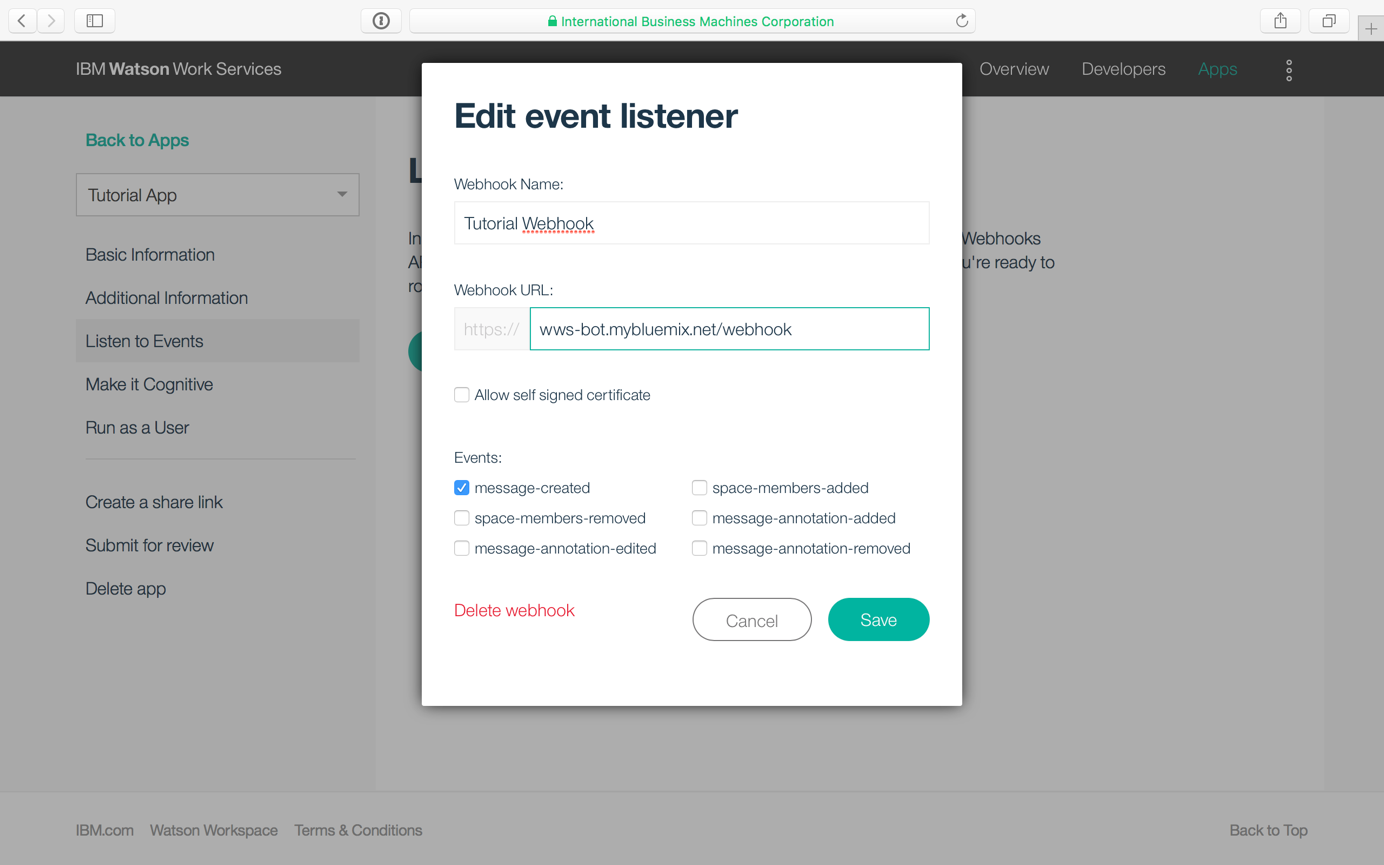
Now that sending a message into a space works, let’s concentrate on receiving events from IBM Watson Workspace. In order to do so, add the “wws webhook” node to your Node-RED flow, give it a name, select the “WWS Application” you configured for the “wws message post” node earlier (yes, you only have to define that configuration once) and choose a webhook path (e.g. “/webhook”).



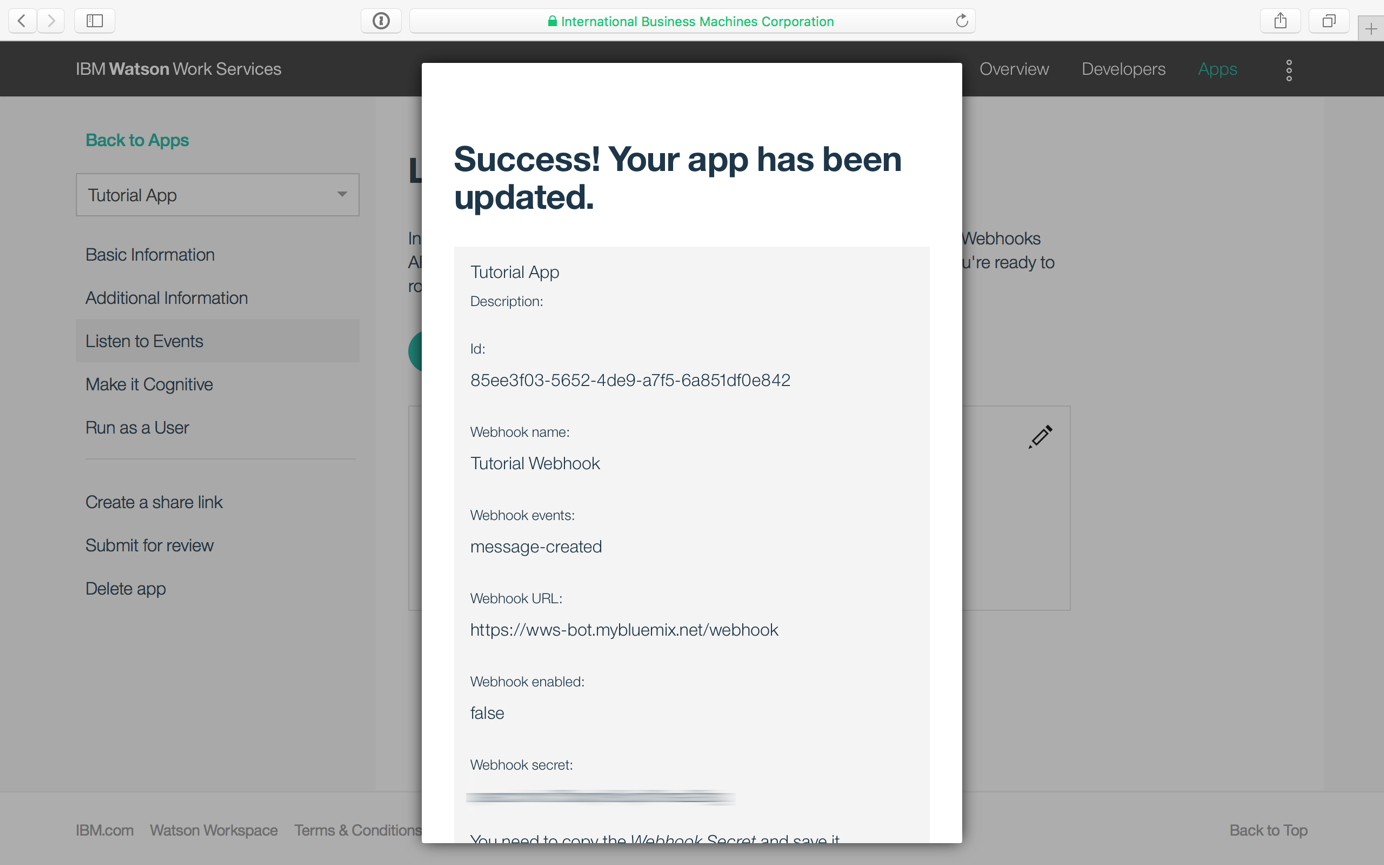
In order to fill in the “Webhook ID” and “Webhook Secret” fields, you need to create a webhook in your IBM Watson Work Services App. Go to IBM Watson Work Services and access the app you created earlier. Then select the “Listen to Events” link from the menu:



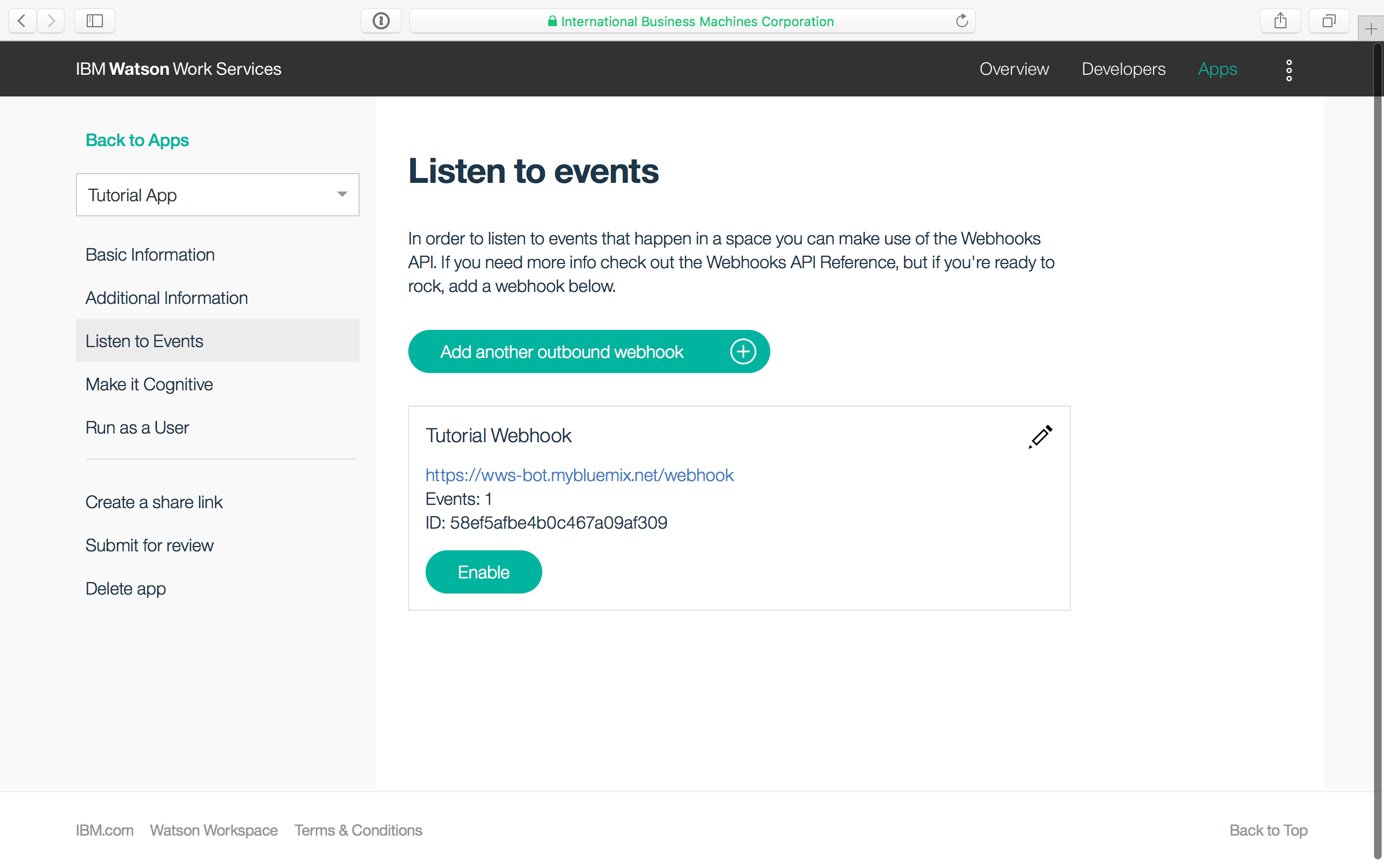
Click “Add an outbound webhook” and enter a name for your webhook as well as the webhook URL. The URL consists of the hostname of your Node-RED app and the path you chose when configuring the webhook node (e.g. <https://wws-bot.mybluemix.net/webhook> when “wws-bot.mybluemix.net” was the hostname you chose and “/webhook” was the path you chose. Also, select which events this webhook should listen to.



After clicking “Save”, note down the webhook secret shown in the confirmation screen. Just like with the app secret, the webhook secret will not be accessible after closing this information.



Caution: The ID shown in this confirmation screen is not the webhook ID. Instead this screen is showing the app ID again (I don’t know why). To access the webhook ID, close the confirmation and there it is:

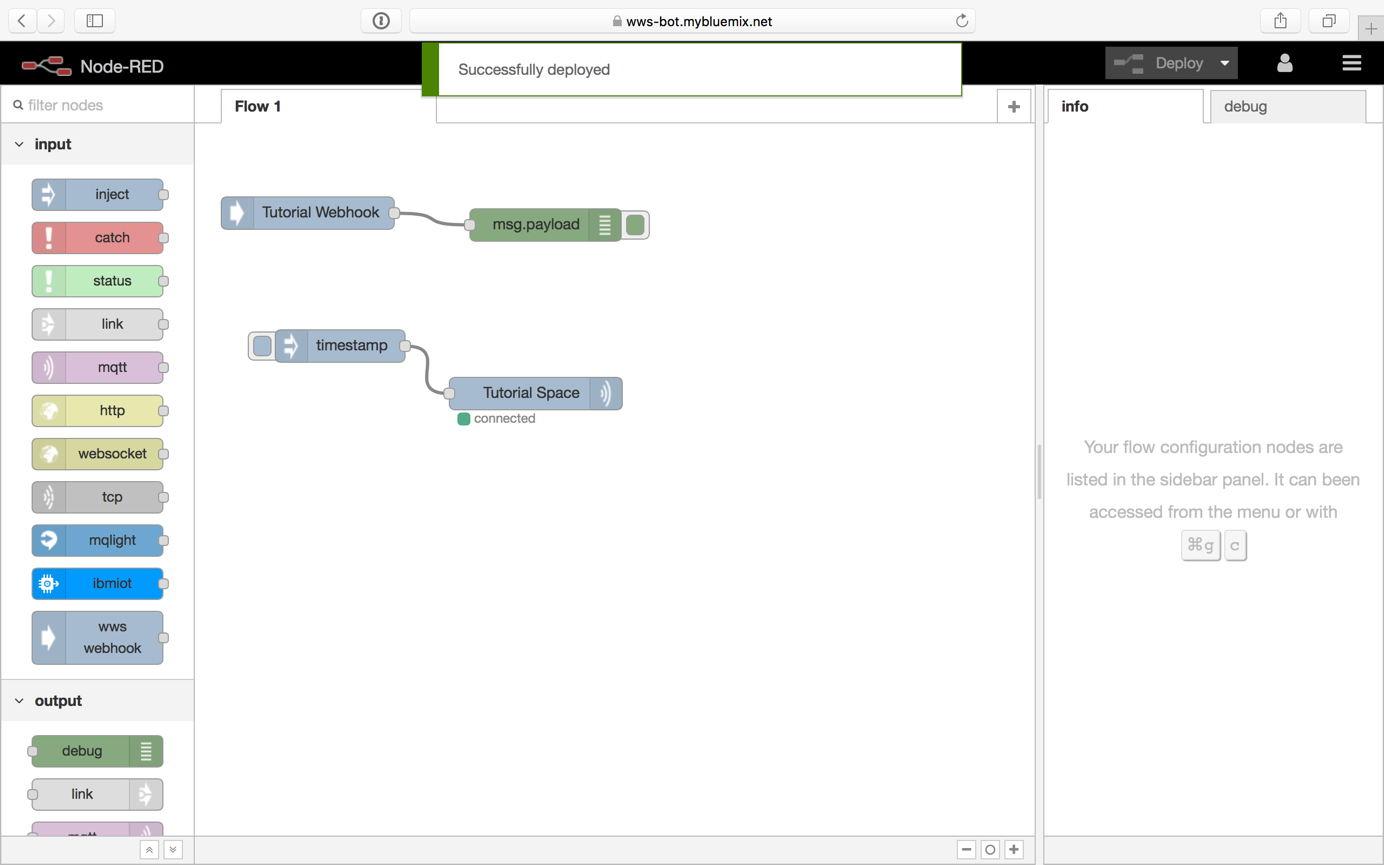


Caution: Do not click “Enable” yet.

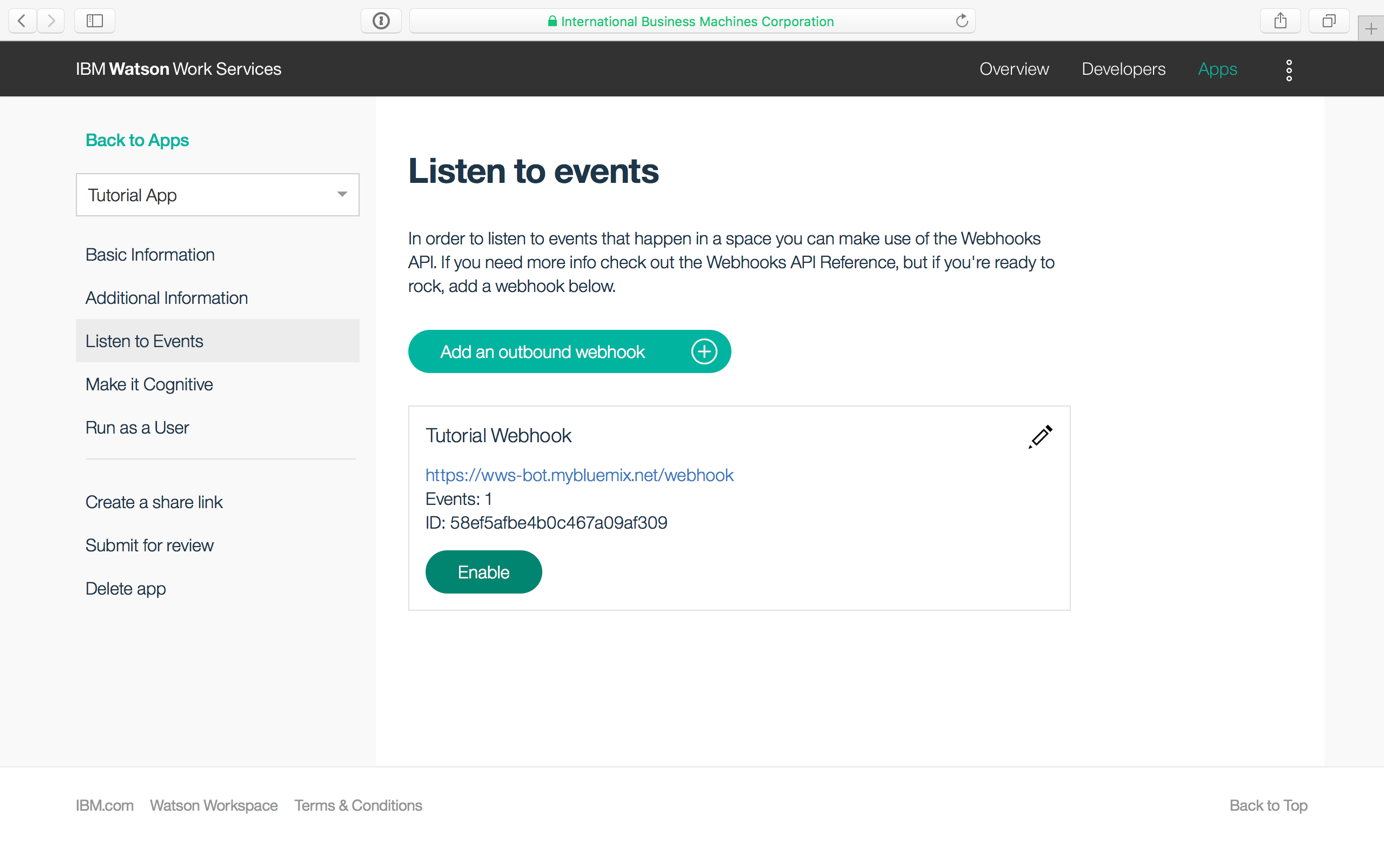
Now, go back to your Node-RED instance and enter the webhook ID and secret:



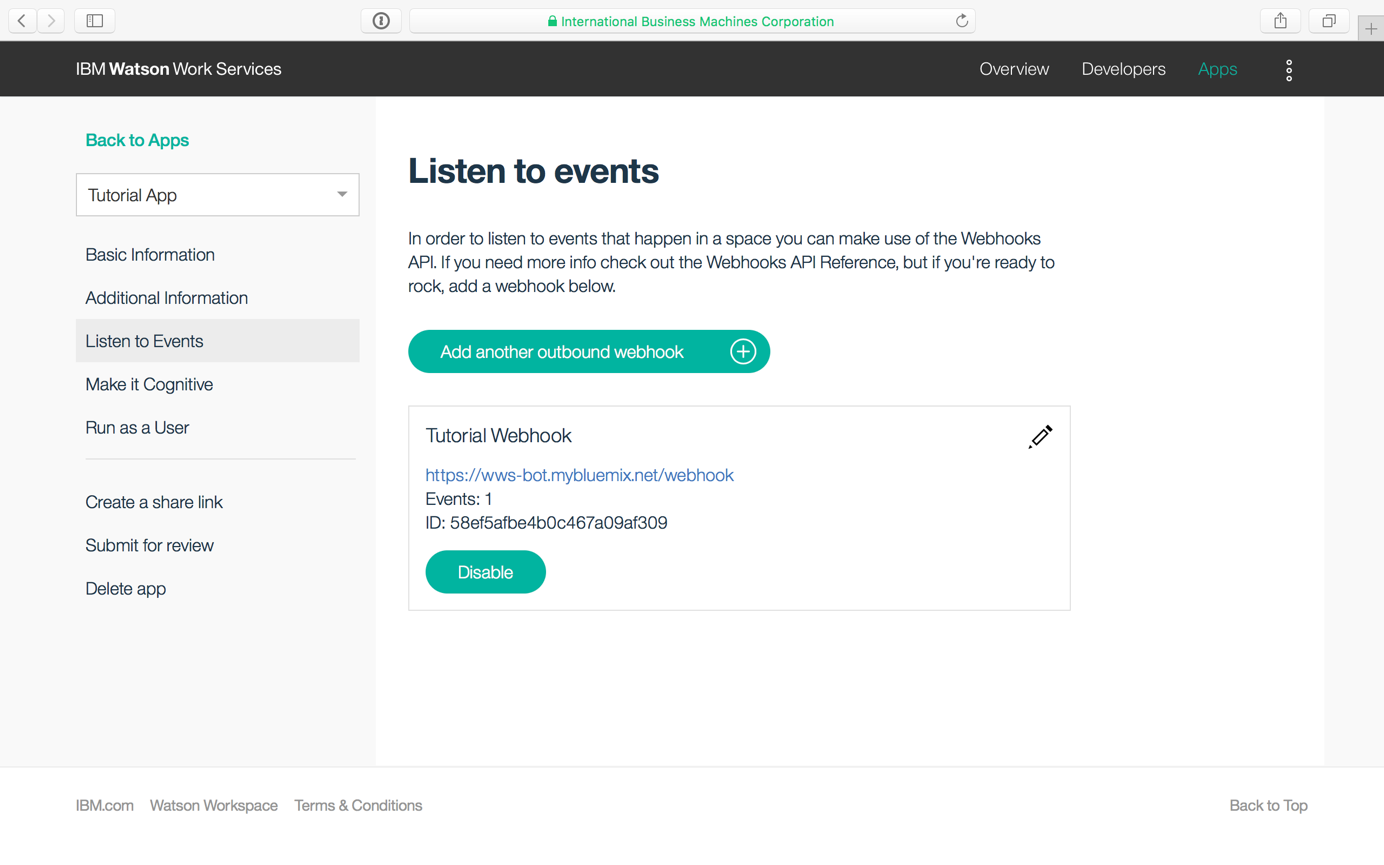
Then proceed to click “Done”, add a “Debug” node, connect it to the webhook node you just configured and deploy your flow.



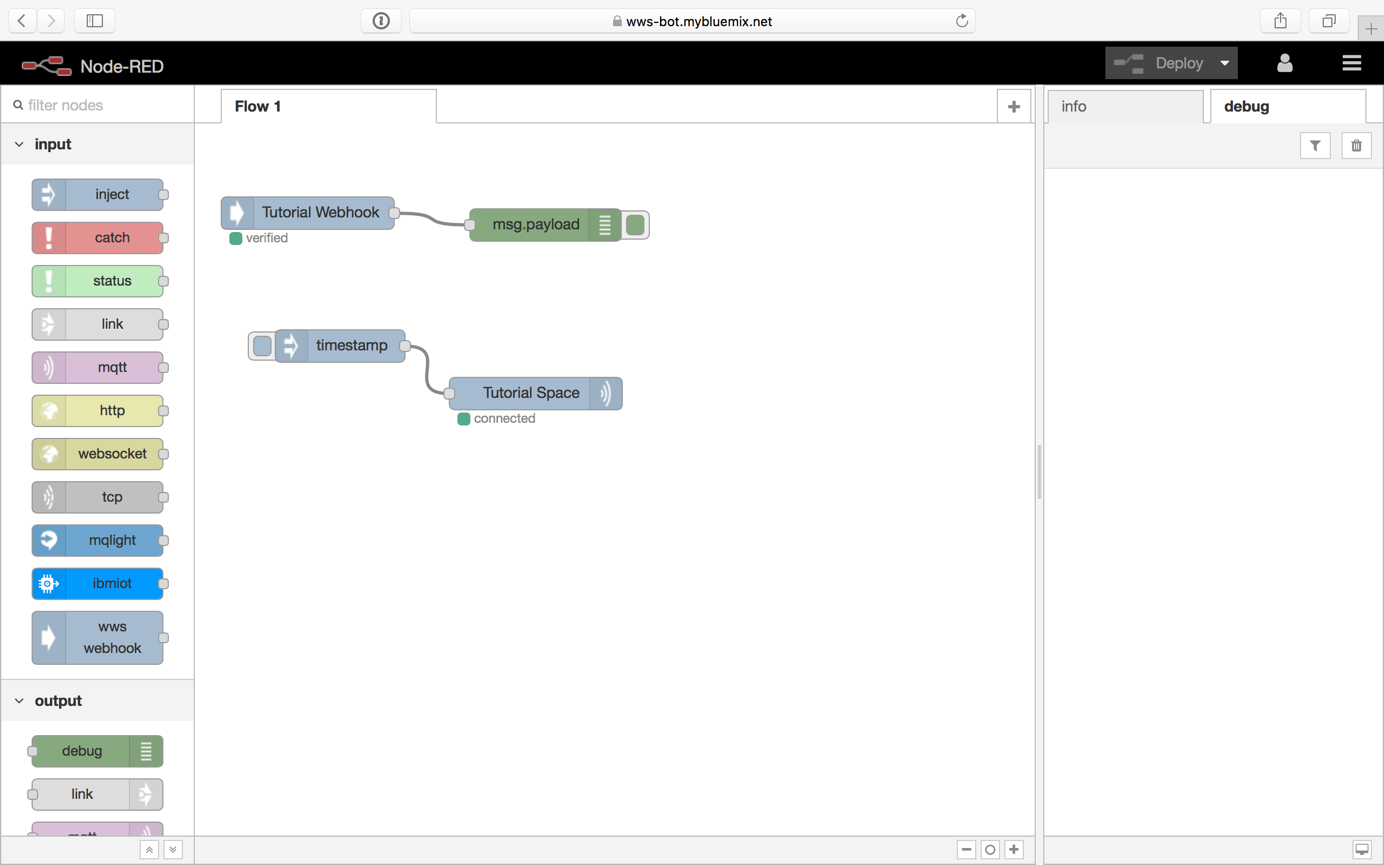
Now go back to IBM Watson Work Services and enable the webhook by clicking “Enable”:



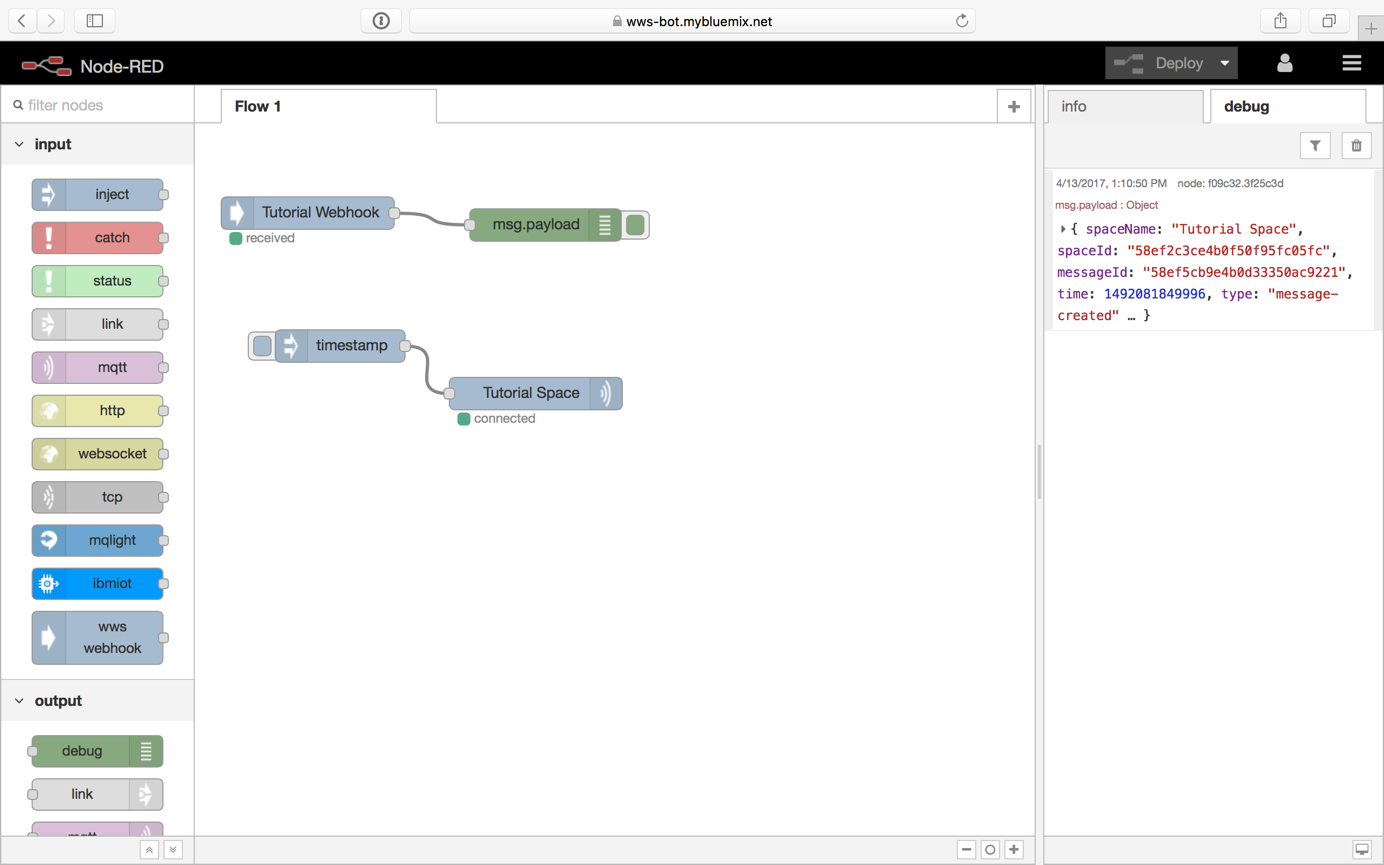
After enabling, the button should now read “Disable”:



And in Node-RED you should see a little green indicator saying “verified” right under your webhook node:



Now when you type a message and send it into IBM Watson Workspace, it should show up in the debug pane of your Node-RED instance:



Please note: Do not use the inject node we created earlier to test this. The webhook node will ignore messages created by the app itself to prevent bots from talking to themselves and thereby creating an infinity loop ;)

Now we can receive messages from and send messages into spaces in IBM Watson Workspace. The only thing left to do is to make some magic happen in between.

1. https://nodered.org [↑](#footnote-ref-1)
2. https://flows.nodered.org/node/node-red-contrib-wws [↑](#footnote-ref-2)
3. https://www.ibm.com/account/us-en/signup/register.html [↑](#footnote-ref-3)