Rudra Modi

ruraina@microsoft.com

Abstract

Learn how to leverage the Dynamics 365 Health Accelerator integration with a virtual agent through this Setup Guide

Dynamics 365   
health accelerator   
Healthbot INTEGRATION

Microsoft Corporation

Contents

No table of contents entries found.

# Overview

The Microsoft Healthcare Bot service empowers healthcare organizations to build and deploy an AI-powered, compliant, conversational healthcare experience at scale. The service combines built-in medical intelligence with natural language capabilities, extensibility tools and compliance constructs, allowing healthcare organizations such as Providers, Payers, Pharma, HMOs, Telehealth to give people access to trusted and relevant healthcare services and information.

This document outlines the setup guide for enabling the Healthcare Bot service to light up scenarios on top of the Healthcare Accelerator. Once complete, you can use Healthcare bot in cohesion with the CDS to build scenarios leveraging transactional Patient, Practitoner and clinical data. You can also get the bot service deployed as a WebChat and show it up on your portal.

If you are looking for information on the health accelerator such as the API reference or examples for extending the model, please look [here](https://community.dynamics.com/365/b/healthaccelerator/dashboard).

For information on Healthcare Bot service, please look [here](https://www.microsoft.com/en-us/research/project/health-bot/).

# Components of the Accelerator

The accelerator consists of several components as shown below. The scope of this document is to outline how MS Teams can be used in context of the HealthCare Accelerator.

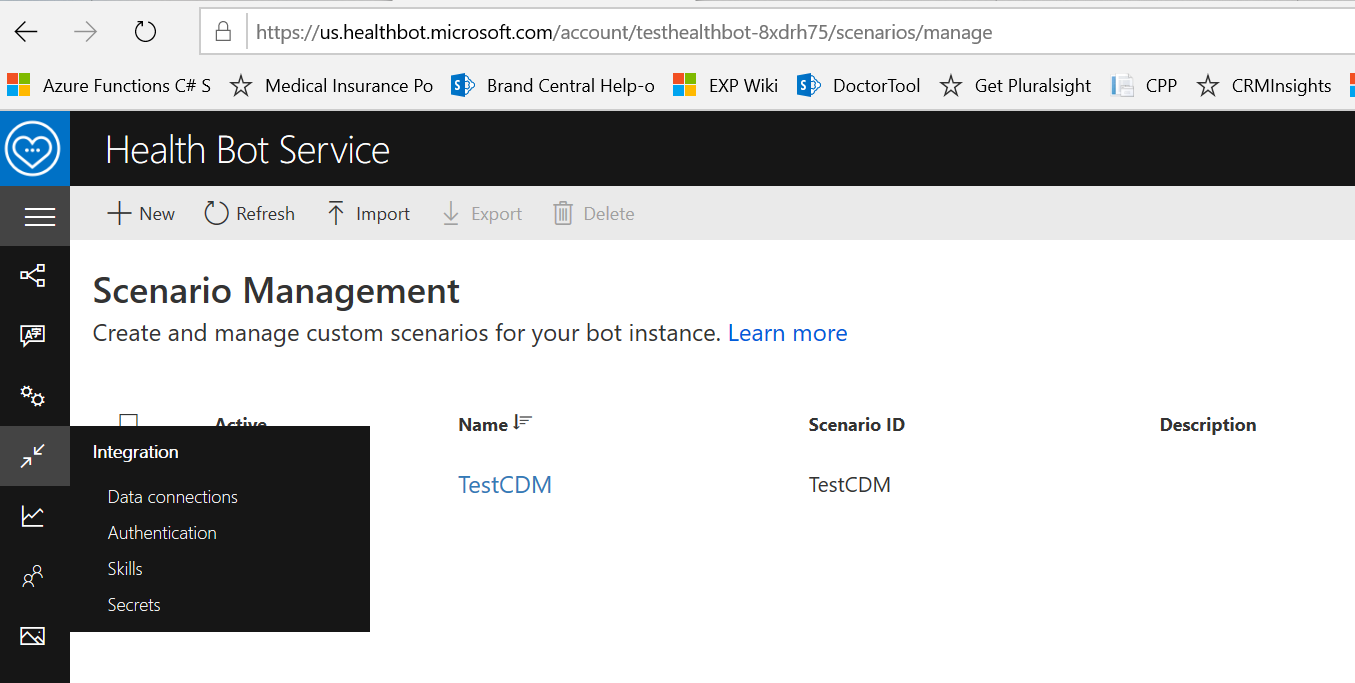
* Entity Model: extensions to the Microsoft Common Data Model
* UCI App: Customer Engagement sample UCI App
* Legacy App: Customer Engagement Legacy App
* Office: How MS Teams can be used in context of the Healthcare Accelerator
* Power BI: Power BI extensions and samples
* Portals: Patient experience thru Dynamics Portals
* \*IOT: Current guide outlines IOT scenarios and samples
* \*AI / Azure: We are working on AI / Azure extensions and samples

# Pre-requisites

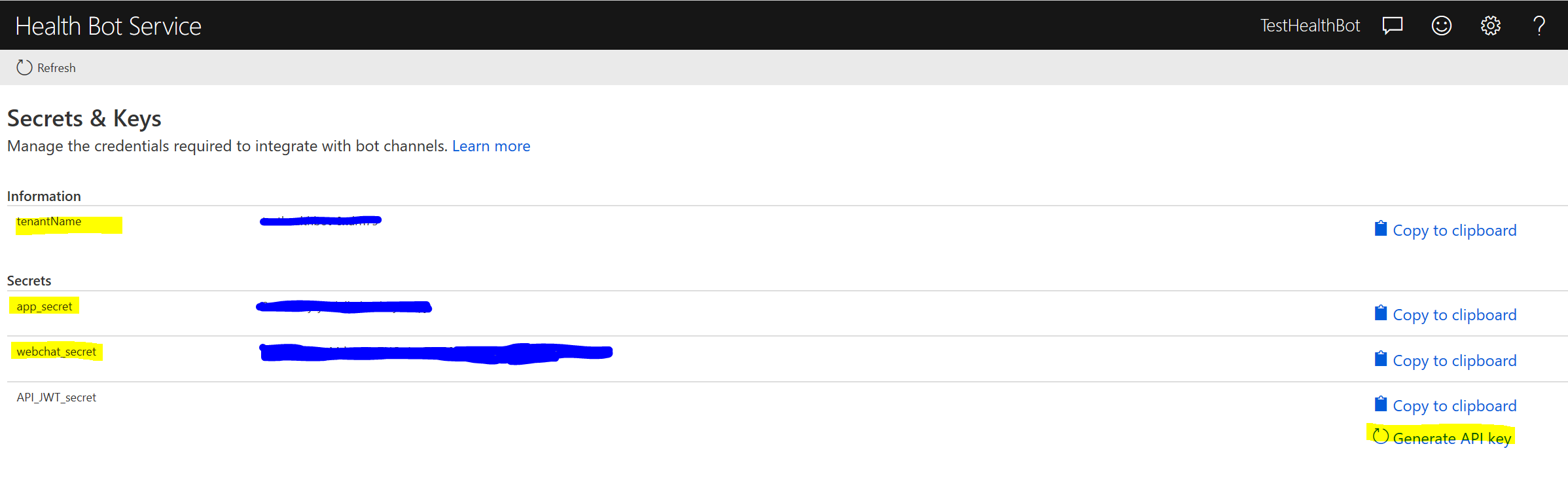
* Required Roles - Office Global Administrator
* Licenses – Active Azure subscription required.

# Setup – Health Bot Instance

* Navigate to the link @ <https://azuremarketplace.microsoft.com/en-us/marketplace/apps/microsoft-hcb.microsofthealthcarebot> and click on ‘Get it Now’ using your tenant credentials. That should be same tenant that have the Healthcare Accelerator deployed.
* Once ready, click on Configure account and begin deployment for the SaaS.
* After the deployment is done, navigate to the deployed SaaS and click on ‘Manage Account’. This should take you to the health bot page.
* In the health bot home page, navigate to ‘Secrets’ as shown below.

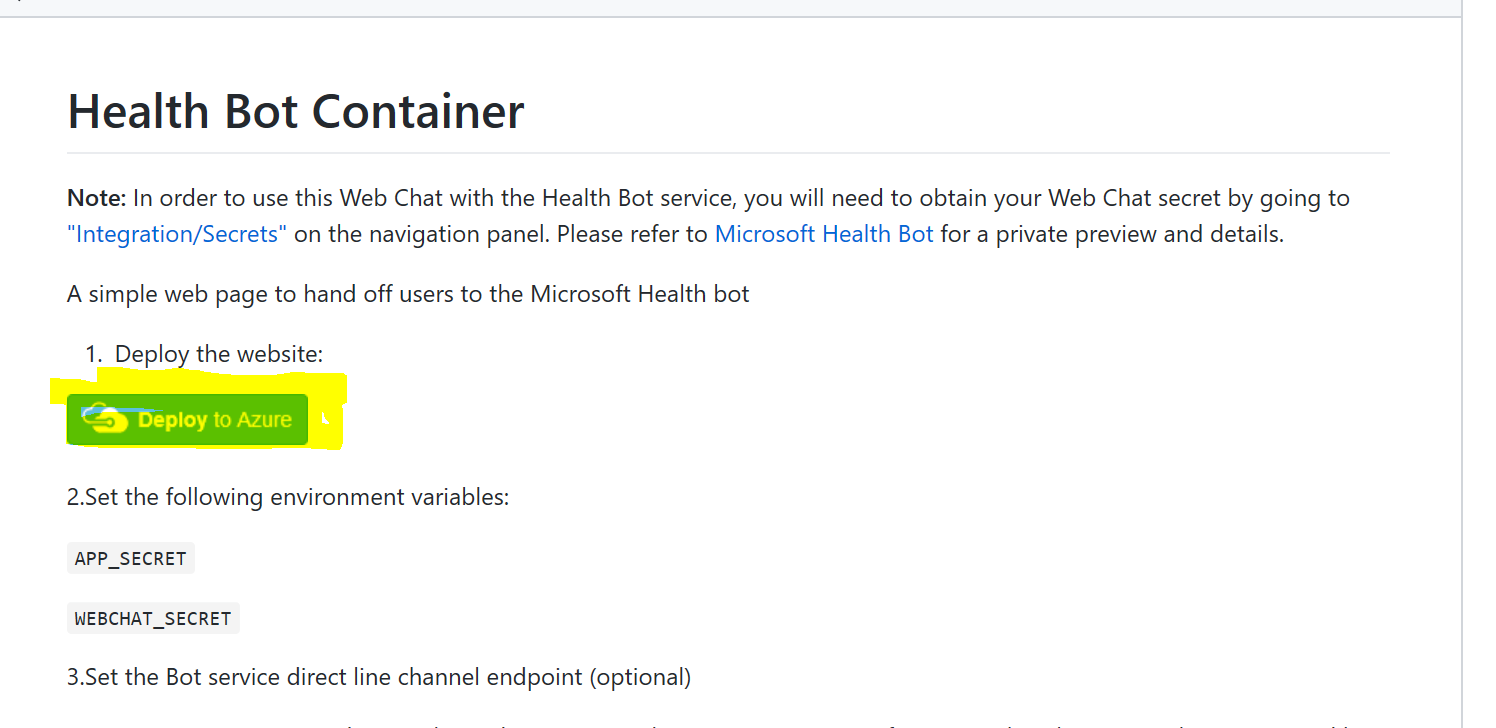


* Copy the highlighted values and save for configuration in the next steps. For API\_JWT\_secret, generate the API key first and then copy the secrets.

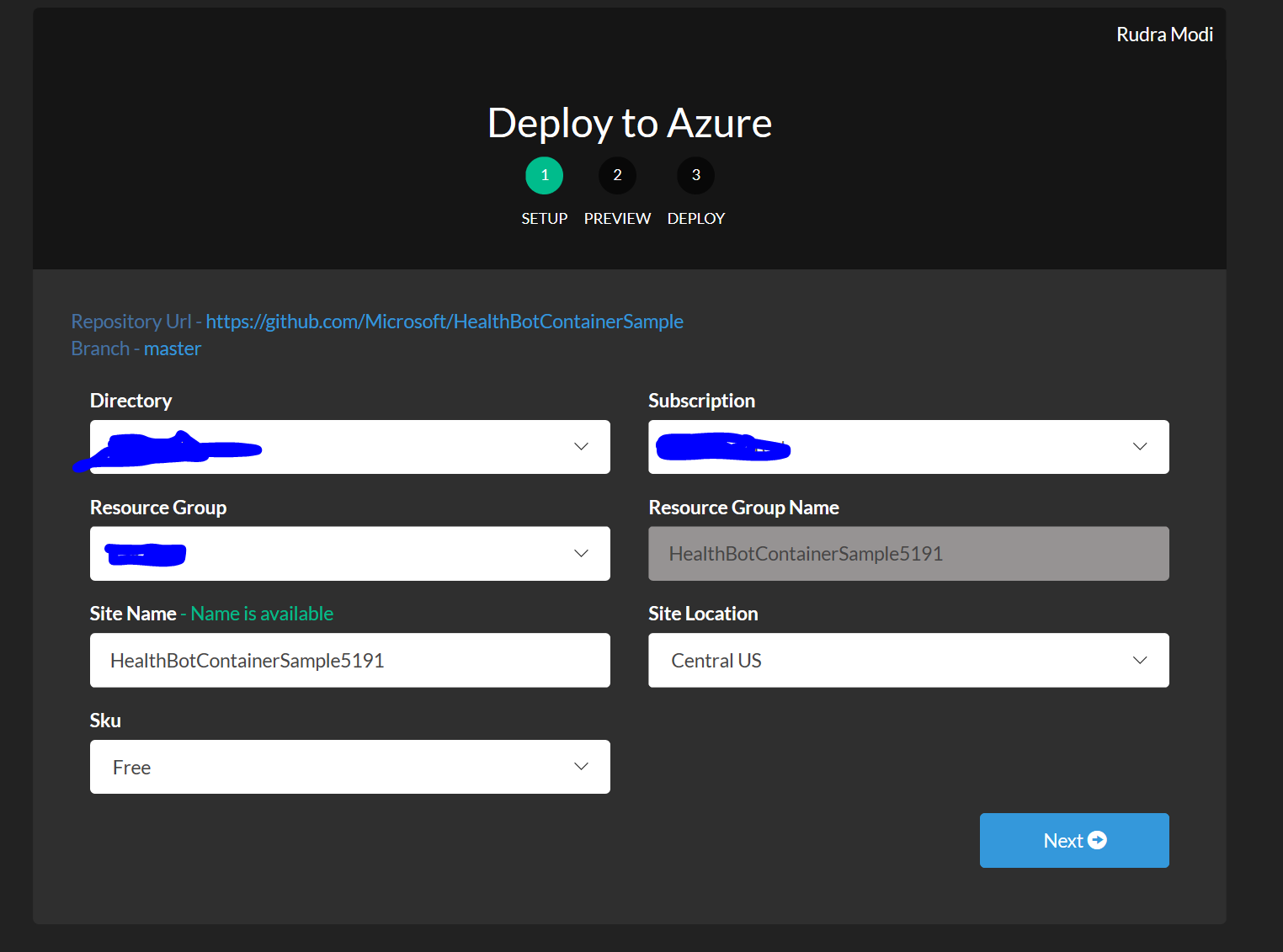


# Setup – Deploy Health Bot Website

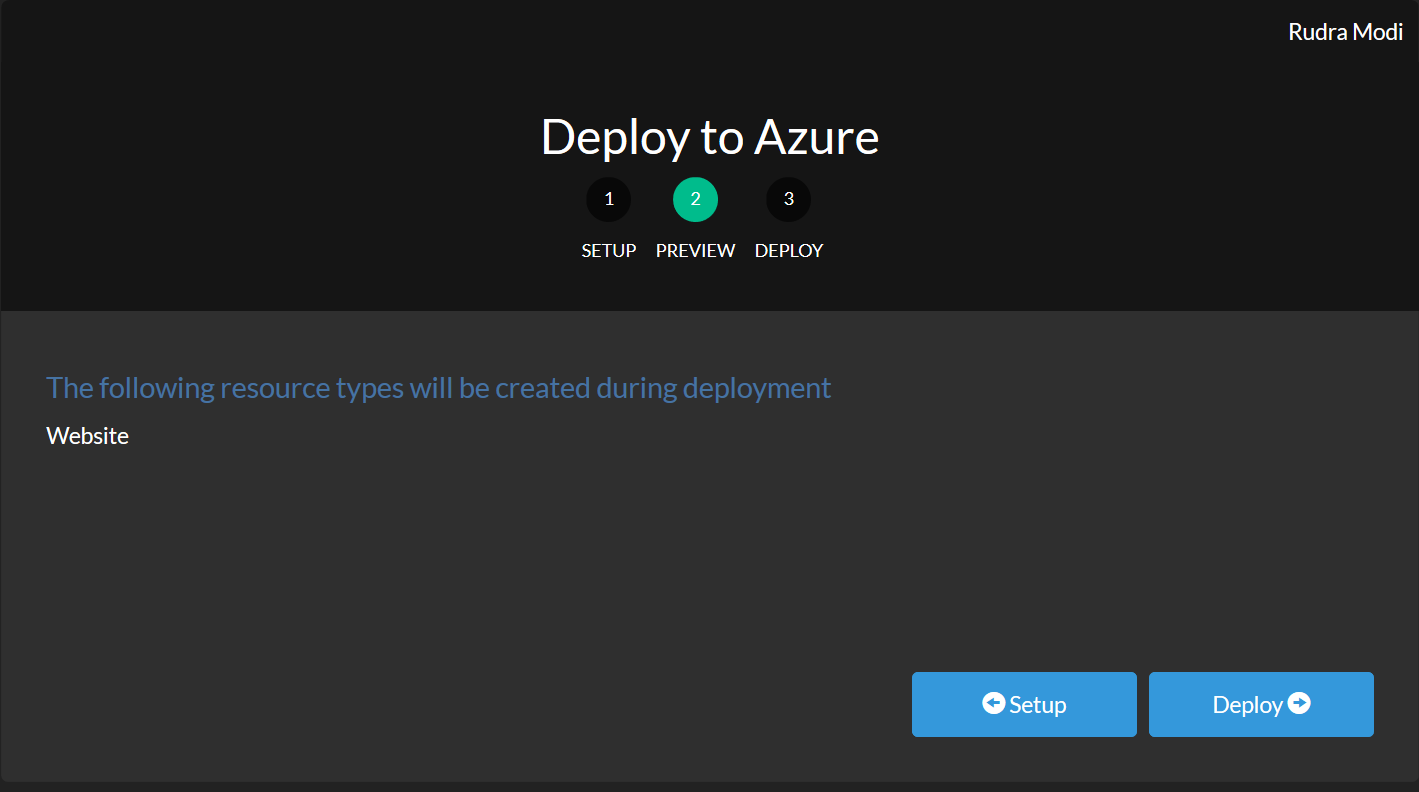
* Navigate to the link @ <https://github.com/Microsoft/HealthBotContainerSample> and click on ’Deploy to Azure’



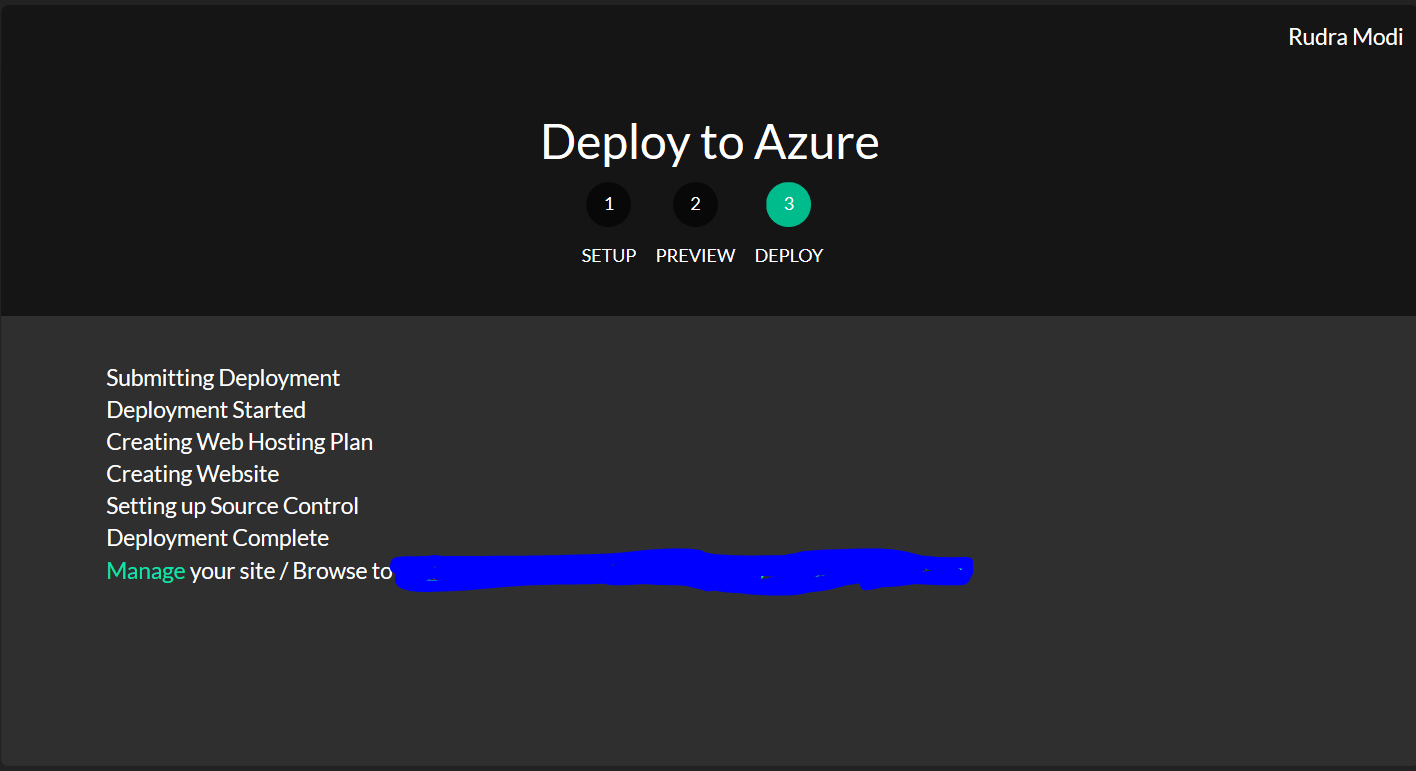
* Select the Azure directory, subscription and resourcegroup in which you want your bot to be deployed.



* Note the resources that get created and click on ‘Deploy’



* Once ready, the website link is shown (grayed out)as below. Navigate to the website that got deployed.



* Navigating to the site will throw the following error. This is because the secrets are not yet configured.



* Navigate to the Azure portal and locate the App Service that you deployed above. Go to Settings->Configuration and add new Connection Strings called APP\_SECRET and WEBCHAT\_SECRET. Set the values from what we got in from secrets above.
* Relaunch the site now and note a webchat shows up. You have successfully deployed the healthbot as a webchat that can be embedded in a portal of your choice. If you choose to embed it as an iframe, heres how it can show up –

<iframe src="https://<deployedwebsitename>.azurewebsites.net/" width="500" height="500" seamless></iframe>

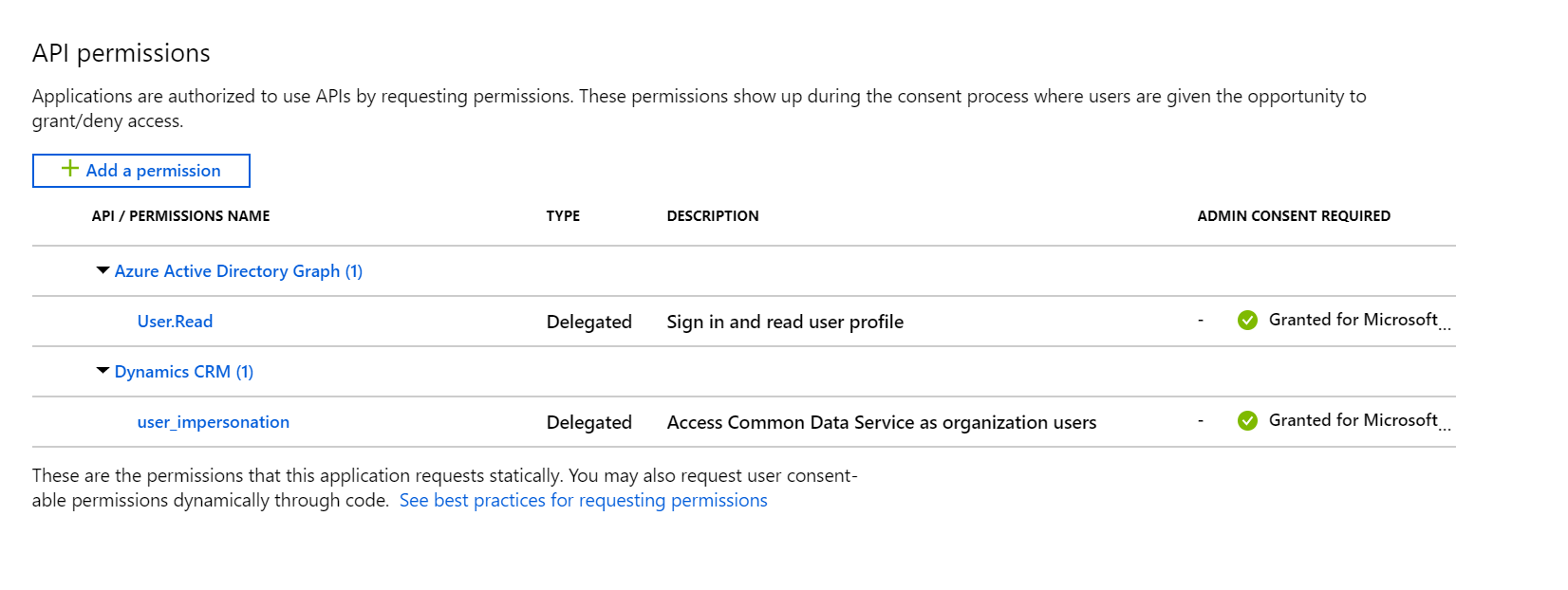
# Setup – Create scenarios in bot consuming CDS

## Prerequisite

* + Since the CDS requires tenant credentials, we can only build scenarios that can be consumed by the CDS users. In other words, build scenarios that can be consumed by users who have a license to access CDS. If you plan to build patient scenarios, each patient would have to be given a CDS license.
  + For the sake of this document, it is assumed that all the Practitioners are licensed users in the CDS org. Hence they would list as ‘Contacts’(by virtue of the Healthcare CDM) as well as ‘Users’ (by virtue of the license) in the CDS org. They would be identified with same emailaddresses in Contacts and Users records.

## Deploy Azure App for authentication –

* + Go to the Azure Portal and navigate to your Active Directory. Note that it should be the same as the one that hosts the CDS org.
  + Create a new ‘Application Registration’. Call it ‘Health-CDS App’. Set signin url to https://localhost.
  + Set Authentication -> Create new Redirect URI and set to **‘https://healthagentbotprod.trafficmanager.net/redirect/oauth2’**
  + Set Client Secrets -> Create new client secret. Save and copy it for next steps.<<Will be used as **ClientSecret** later>>
  + Copy and save the ApplicationID of the application <<Will be used as **ClientID** later>>
  + Set API Permissions as follows. Grant permissions once configured.



## Build CDS scenario-

* + Navigate to your healthbot home page link like [https://us.healthbot.microsoft.com/account/<healthbot link here>/scenarios/manage](https://us.healthbot.microsoft.com/account/%3chealthbot%20link%20here%3e/scenarios/manage)
  + Setup Authentication ->Navigate to Integration->Authentication. Create a new End User Authentication Provider called ‘CDM Endpoint’. Set the following properties.

*Client ID = <<Application ID copied from Azure App in above steps>>*

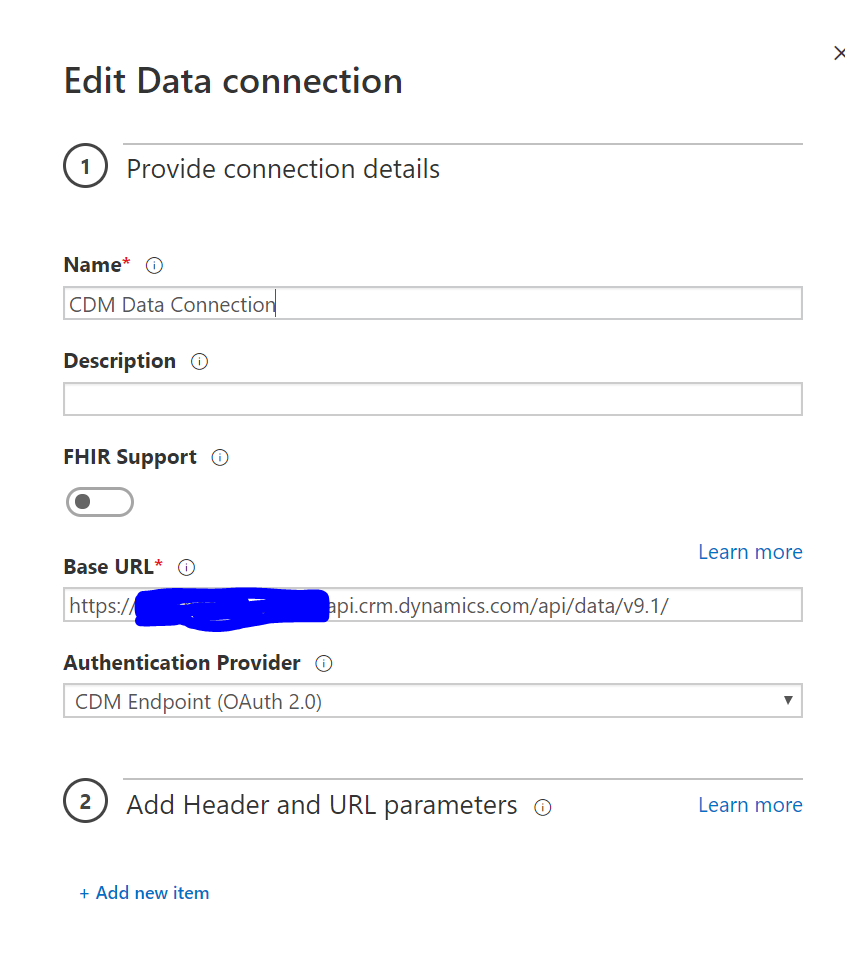
*Client Secret = <<Client secret copied from Azure App in above steps>>*

*Authorization URL =*  [*https://login.microsoftonline.com/common/oauth2/authorize*](https://login.microsoftonline.com/common/oauth2/authorize)

*Access Token URL =* [*https://login.microsoftonline.com/common/oauth2/token*](https://login.microsoftonline.com/common/oauth2/token)

*Scope = &resource = https://<<your instance org>>.crm.dynamics.com*

* + Setup data connections -> Create a new data connection as follows. The BaseURL should be the WebAPI endpoint for the CDS org in the tenant.



* + Import the *CDMScenario.json* from the repo and edit the scenario to use the above data connections. Run the scenario to test out the doctor experience.