**Setting up Watson-z/Os bot on local workstation**

**Links:**- GitHub repository: <https://github.ibm.com/dfsms-devicesupport/watson-zos>

- Watson Assistant Document: <https://console.bluemix.net/docs/services/assistant/getting-started.html#gettingstarted>

- IBM Cloud: <https://cloud.ibm.com/>

**Content:**

Chapter 1: [Run the WA using the Watson Assistant workspace of Device Support team](#Chapter1)

Part A – Option 1 - [Set up local repository by downloading from GitHub.](#Chapter1A1)

Part A– Option 2 - [Set up local repository by cloning from GitHub.](#Chapter1A2)

Part B - [Installing Node.js and npm](#Chapter1B)

Part C - [Run the app on local host](#Chapter1C)

Chapter 2: [Run the WA using the Watson Assistant workspace of yours](#Chapter2)

Part A – [Launching Watson Assistant service](#Chapter2A)

Part B – [Set up Watson Assistant workspace](#Chapter2B)

Chapter 1: Run the WA using the Watson Assistant workspace of Device Support team

Note: Please use **Part A – Option 1** if you only need to have a working local copy of the project and don’t need to publish your changes to GitHub. Please use **Part A – Option 2** if you want to use GitHub client and contribute with your updates.

Part A – Option 1 - Set up local repository by downloading from GitHub.

- Go to <https://github.ibm.com/dfsms-devicesupport/watson-zos>, click on “Clone or download” button, download zip file and extract it to a folder of your choice.

- Environment file:  
Please create one file with the content below in a text editor and save it with the name **.env** into the watson-zos folder.

# Environment variables

SKILL\_ID=10caecc2-869a-4790-a31e-ac532e185df4

WORKSPACE\_ID=5e97314d-2fb8-4376-b67a-e6c3cab93f7c

# You need to provide either username and password

ASSISTANT\_USERNAME=apikey

ASSISTANT\_PASSWORD=tqyvQgHq6dM11mPxPaSI8fJdporvwKvOp3-3Dl1JWKRW

# And IAM API key and URL

ASSISTANT\_APIKEY=tqyvQgHq6dM11mPxPaSI8fJdporvwKvOp3-3Dl1JWKRW

ASSISTANT\_URL=https://gateway.watsonplatform.net/assistant/api

USE\_CLIENT\_ACTIONS=true

Part A– Option 2 - Set up local repository by cloning from GitHub.

1 - GitHub for Desktop client

Installing: The GitHub for Desktop app is available for download at [desktop.github.com](https://desktop.github.com) and the PC@IBM App Store.

Logging in:

- The app will ask two options of Sign in: GitHub and Enterprise. Please Sign in for Enterprise.

- Enterprise server address: <https://github.ibm.com>

- Continue with IBM ID and password.

You’ll get an email from GitHub <noreply@github.ibm.com> to your IBM email:

“A third-party OAuth application (GitHub Desktop) with repo and user scopes was recently authorized to access your account.”

- If the app complains that you have not had Git, download and install the latest version of Git:   
<https://git-scm.com/downloads>

2 - Cloning watson-zos repository into your local folder

a. Open GitHub for Desktop  
b. File -> Clone repository… -> URL

Repository URL: <https://github.ibm.com/dfsms-devicesupport/watson-zos.git>

Local path: A folder in your computer that you choose to store the repository and work on. Let’s say C:\WatsonPlay

After cloning process is done, you’ll have watson-zos folder at C:\WatsonPlay\watson-zos.

c. Environment file:

Please create one file with the content below in a text editor and save it with the name **.env** into C:\WatsonPlay\watson-zos folder.

# Environment variables

SKILL\_ID=10caecc2-869a-4790-a31e-ac532e185df4

WORKSPACE\_ID=5e97314d-2fb8-4376-b67a-e6c3cab93f7c

# You need to provide either username and password

ASSISTANT\_USERNAME=apikey

ASSISTANT\_PASSWORD=tqyvQgHq6dM11mPxPaSI8fJdporvwKvOp3-3Dl1JWKRW

# And IAM API key and URL

ASSISTANT\_APIKEY=tqyvQgHq6dM11mPxPaSI8fJdporvwKvOp3-3Dl1JWKRW

ASSISTANT\_URL=https://gateway.watsonplatform.net/assistant/api

USE\_CLIENT\_ACTIONS=true

Refs: GitHub installation info above can be found in this IBM link: <https://w3.ibm.com/help/#/article/github_ent_ibm/github_setup?requestedTopicId=github_setup>. However, link should be out of date. Several steps are not necessary.

Part B - Installing Node.js and npm

On Mac:

1 – Installing Homebrew (a package manager for the Mac):  
In Terminal, type and wait for it to be done:  
ruby -e "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/master/install)"

2 – Installing Node.js including npm:  
In Terminal, type and wait for it to be done:  
brew install node

Reference: <https://blog.teamtreehouse.com/install-node-js-npm-mac>

On Windows:

Download .msi file and run from the link: <https://nodejs.org/en/>. Restart computer.

Reference: <https://blog.teamtreehouse.com/install-node-js-npm-windows>

Part C - Run the app on local host

- In terminal window, change directory to the folder of the repository, e.g. C:\WatsonPlay\watson-zos.

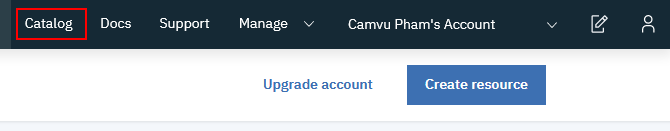
- Type: npm install watson-developer-cloud  
- Type: npm install zos-node-accessor  
- Type: npm install express  
- Type: npm install   
- Type: npm start  
- View the application in a browser at localhost:3000  
- Start a conversation with the bot. You’ll see the terminal print out the log of how WA works with the user input.

Chapter 2: Run the WA using the Watson Assistant workspace of yours

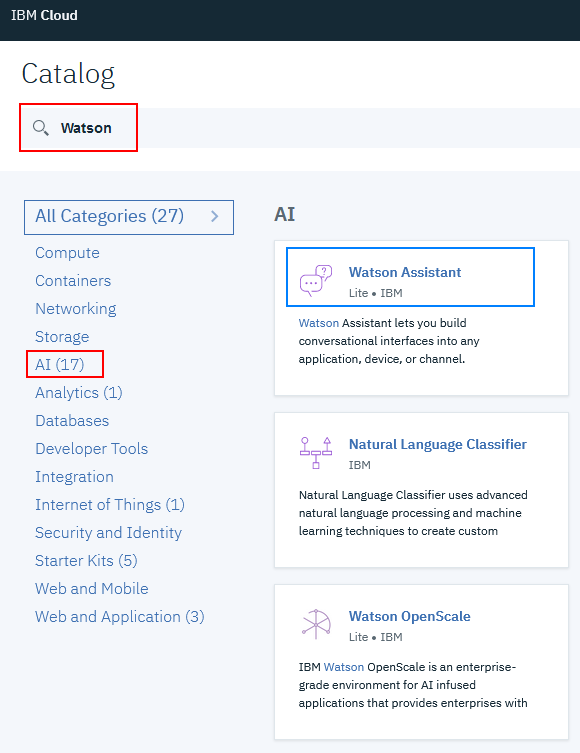
Part A – Launching Watson Assistant service

1. Sign up and log in with IBM ID at <https://cloud.ibm.com>

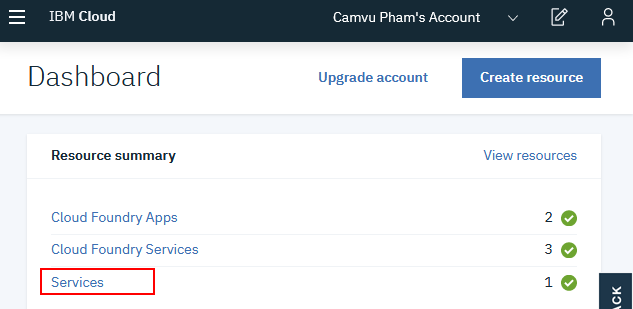
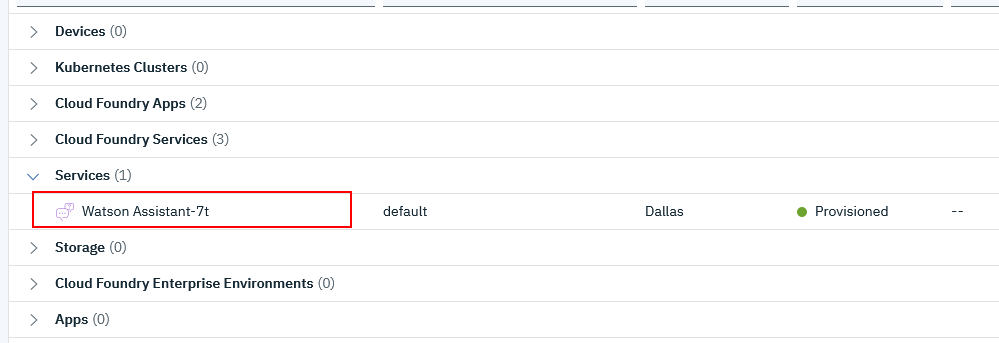
2. Go to Watson Assistant service:

Right top corner, click on “Catalog” 

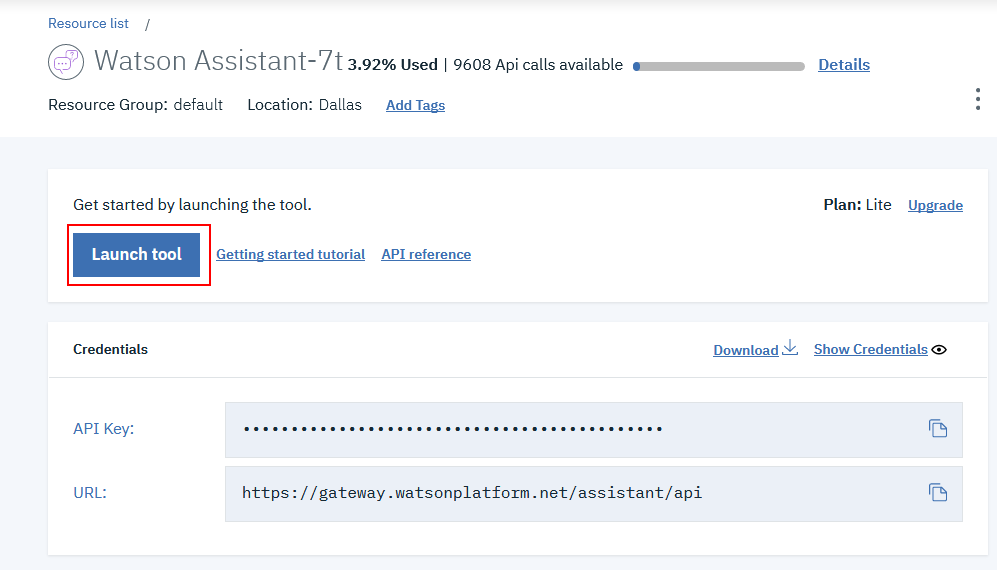
In Catalog page, either click on AI or type “Watson” in the search bar. Click on “Watson Assistant”.



At bottom right, click “Create” button to create a new Watson Assistant service.

This service can be found any time later when you log in to your Cloud account and view your Dashboard   
( Click on the three bar icon top left  then pick Dashboard.   
  
  
Watson Assistant is in “Services.”  


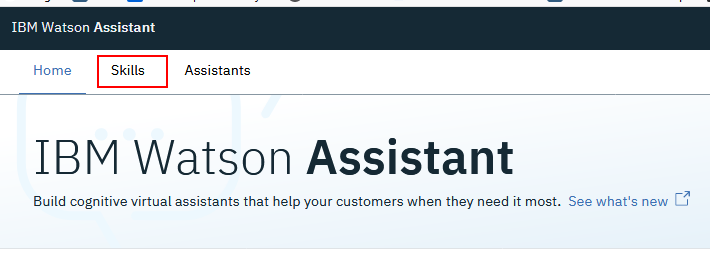
Click on the service, then button “Launch tool”



Now you are in!

Part B – Set up Watson Assistant workspace

1 – Create a new Skill:

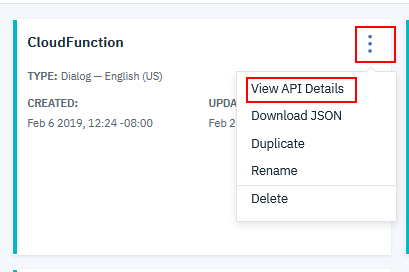
Click on **Skills**Click on **Create new**

Choose **Import skill** and **Choose JSON file**  
Navigate to your watson-zos folder, look in **watson-zos\lib** to pick the JSON file **skill-CloudFunction.json**.

Click **Import**

Now you have a new Skill on your Skills page!

2 – Create environment file that stores your WA credentials so the project can use the services in the skill.

Click on the 3 dot icon of the skill, View API Details  


Using these credentials to create the .env file:

Example:

**Skill Details**

Skill Name:CloudFunction

Skill ID:10caecc2-869a-4790-a31e-ac532e185df4

Workspace ID:5e97314d-2fb8-4376-b67a-e6c3cab93f7c

Legacy v1 Workspace URL:**https://gateway.watsonplatform.net/assistant/api**/v1/workspaces/5e97314d-2fb8-4376-b67a-e6c3cab93f7c/message

**Service Credentials**

Service Credentials Name:auto-generated-apikey-70165a55-f795-48fc-bfea-95db7d527420

Username:**apikey**

Password:tqyvQgHq6dM11mPxPaSI8fJdporvwKvOp3-3Dl1JWKRW

That gives .env file:

# Environment variables

SKILL\_ID=10caecc2-869a-4790-a31e-ac532e185df4

WORKSPACE\_ID=5e97314d-2fb8-4376-b67a-e6c3cab93f7c

# You need to provide either username and password

ASSISTANT\_USERNAME=**apikey**

ASSISTANT\_PASSWORD=tqyvQgHq6dM11mPxPaSI8fJdporvwKvOp3-3Dl1JWKRW

# And IAM API key and URL

ASSISTANT\_APIKEY=tqyvQgHq6dM11mPxPaSI8fJdporvwKvOp3-3Dl1JWKRW

ASSISTANT\_URL=**https://gateway.watsonplatform.net/assistant/api**

USE\_CLIENT\_ACTIONS=true

Place this .env file in watson-zos folder, the same place with app.js file.

Now when you run the app (terminal window, in the watson-zos directory, command npm start, go to browser and get to <http://localhost:3000>), the bot will use the set up in your newly created WA skill.   
You can click on the skill, revise the setup, and the bot will reflect your new updates.