

STEPS TO CREATE YOUR OWN TJ BOT

A Step-By-Step Manual

IBM

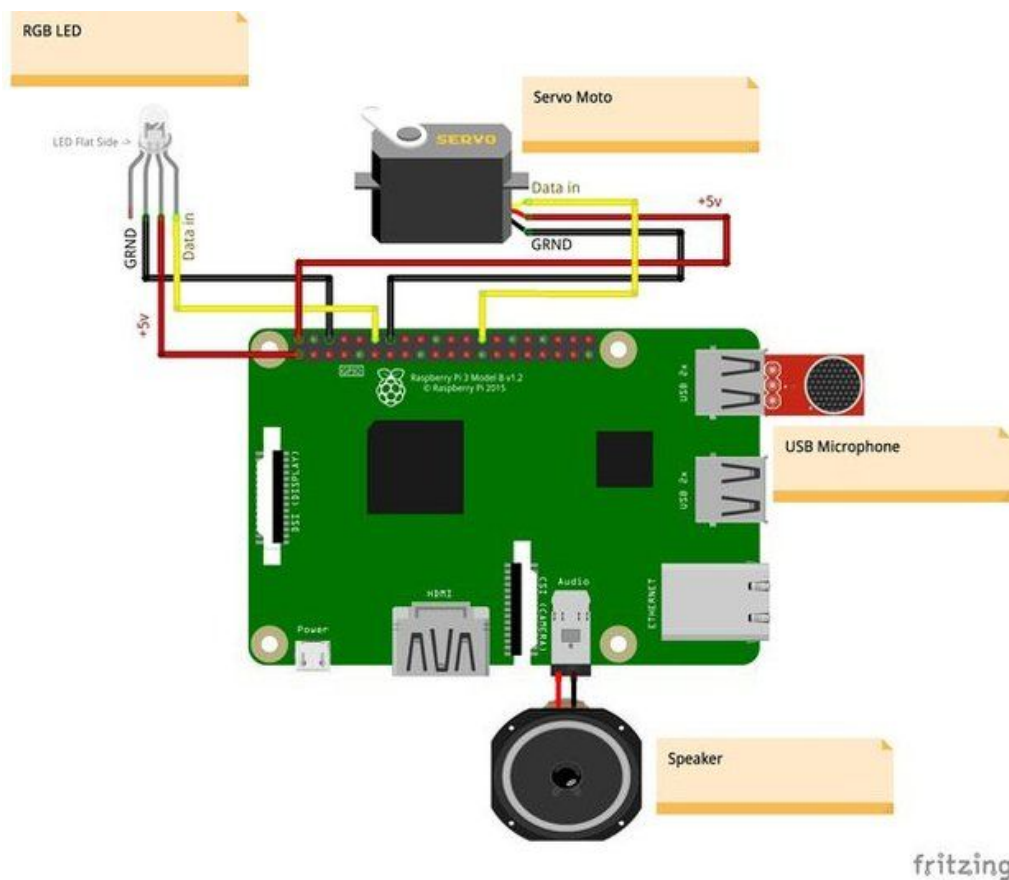
theWONDRY

TJBOT STEP-BY-STEP MANUAL

MODULE 1: BUILD THE TJBOT & BLUEMIX ACCOUNT

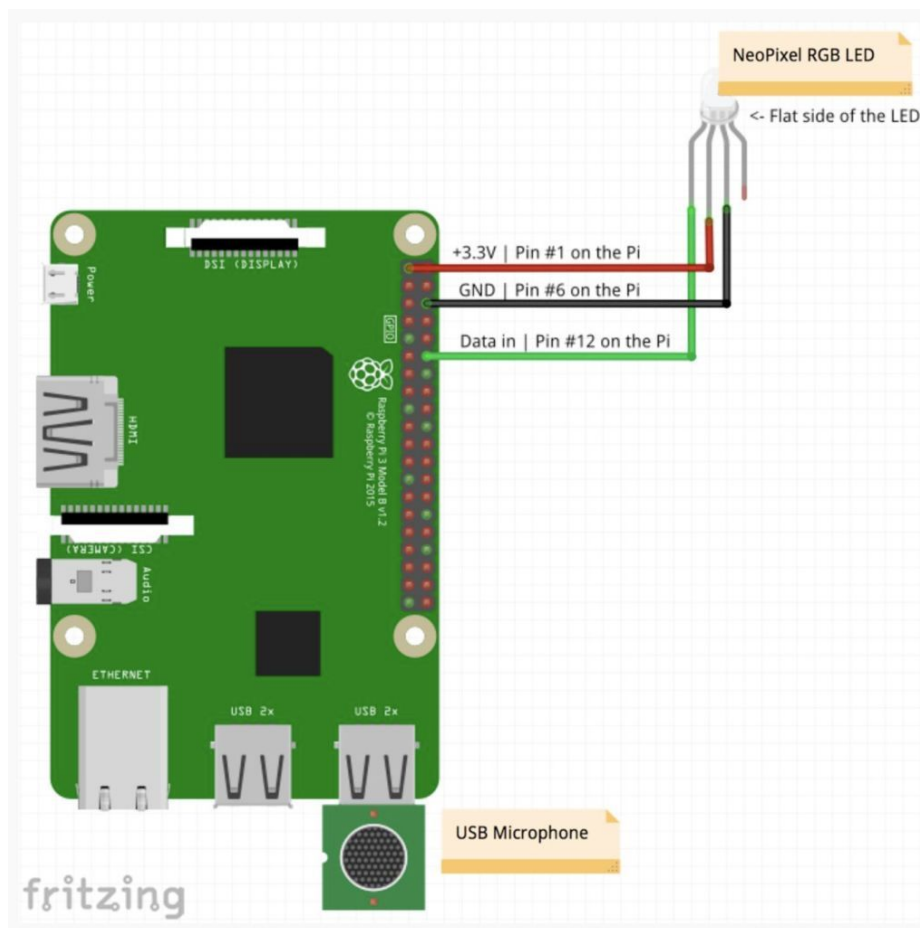
- You should have your TJbot laser-cut cardboard layout with you.
- Using this video, <https://www.youtube.com/watch?v=k928MQmD0oc>, you should follow a set of instructions to assemble and build a stable TJbot.
- Create a bluemix account using this link: <https://cloud.ibm.com/login>
- You will be needing credentials from your bluemix account to set up your TJbot.
- When building your TJbot make sure your wiring for the servo and the LED to the raspberry pi looks like this.

FOR THE SERVO:



- RED (+5V, PIN 2)
- BROWN (GROUND, PIN 14)
- YELLOW (DATA IN, PIN 26, GPIO7)

FOR THE LED:



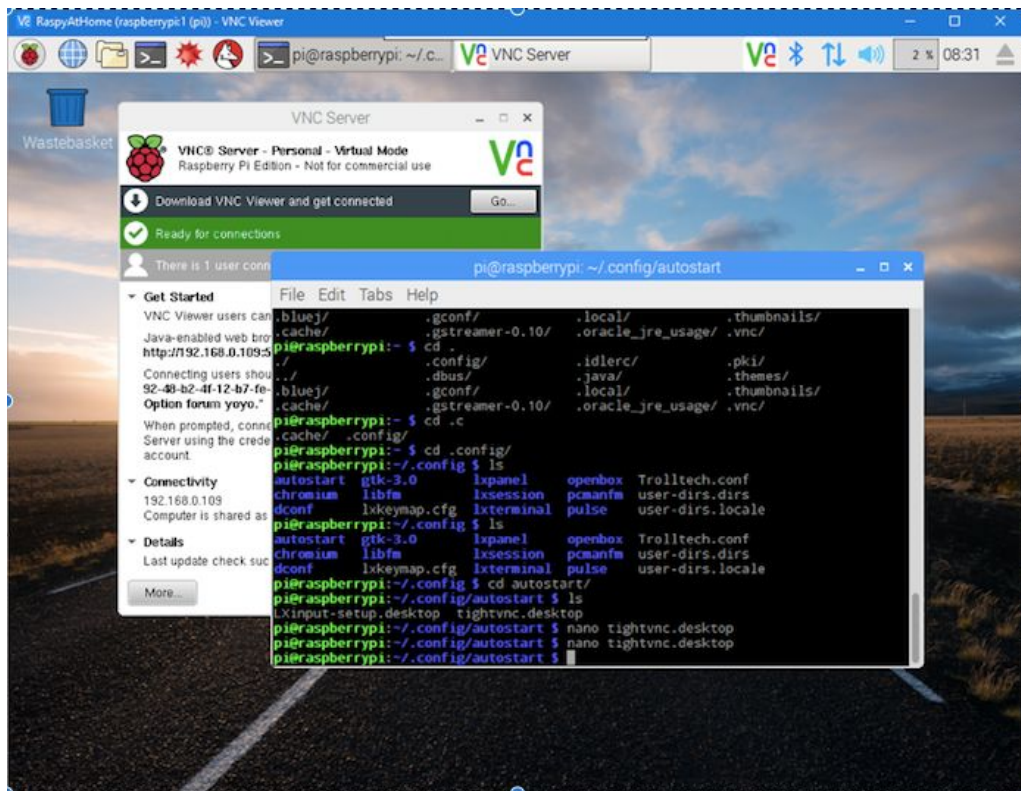
- RED (PIN 1)
- BLACK (PIN 6)
- GREEN (PIN 12)
- THE PIN CLOSEST TO THE FLAT SIDE OF THE LED IS NOT CONNECTED

MODULE 2: SETTING UP YOUR TJBOT

In order to connect your tjbots to a laptop, firstly, you have to connect it to a desktop, install the raspbian OS and run a few commands to be able to access it on a laptop. Below, is a step-by-step procedure for connecting it to the computer:

- Once you have assembled your TJBOT, you should connect the USB port to a power source.
- Connect the mouse and the keyboard from the desktop to the TJBOT.
- Project it on the desktop with the use of an HDMI cable.
- The raspberry pi will boot on the desktop screen.
- After booting, there will be a bunch of icons on the top of the screen including a wi-fi icon.
- Connect to the wi-fi network.
- Select and install only the raspbian OS (because installing other apps/OS could be detrimental to the raspberry pi's functionality)
- Then create a password for your raspberry pi. The custom password is raspberry.
- The coding terminal is a black icon coated with blue edge on the top left of the screen.
- Once installation is complete, open up the coding terminal and type: `sudo apt-get update`
- After running that command, run this one: `sudo apt-get install realvnc-vnc-server realvnc-vnc-viewer`
- Then run this command: `sudo raspi-config`
- Navigate to interfacing options.
- Then, scroll down to click on VNC, choose yes.
- You will be directed back to the homepage where you'll navigate and click on finish.
- Then, run this command: `vncserver`

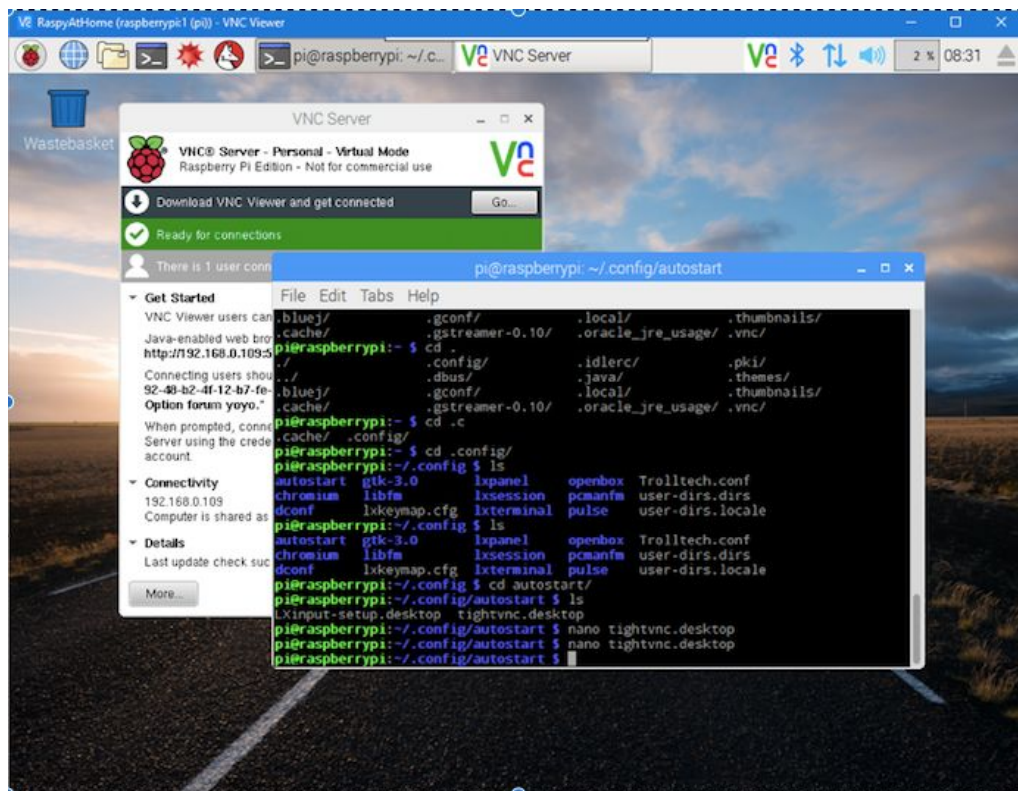
- After running the code, you shall see your IP address which will look like this: 192.167.5.149:1 but only take note of 192.167.5.149 and ignore :1
- On your laptop, use this link, <https://www.realvnc.com/en/connect/download/viewer/>, to download the RealVNC app. RealVNC is the application you will use to run the raspberry pi on your laptop.
- Once installed on your laptop, you shall see a part of the page demanding an IP address in the top search bar. Once, your TJobot is connected to a power source with the use of the USB cable, type in the IP address.
- The custom username is **pi** and the password is **raspberry**.
- From now on, you can now connect your TJobot to your laptops or any other devices that have VNC server.
- Once you've put in the username and the password, you get access to a screen that looks like this:



- Then click on the coding terminal (the black icon with a blue shell) on the top left of the screen and start coding.

MODULE 3: INSTALLING VNC VIEWER ON YOUR COMPUTER

- On your laptop, use this link,
<https://www.realvnc.com/en/connect/download/viewer/>,
to download the RealVNC app. RealVNC is the application you will use to run the raspberry pi on your laptop.
- Once installed on your laptop, you shall see a part of the page demanding an IP address in the top search bar. Once, your TJbot is connected to a power source with the use of the USB cable, type in the IP address and press enter.
- The custom username is **pi** and the password is **raspberrypi**.
- Once you've put in the username and the password, you get access to a screen that looks like this:



- From now on, you can now connect your TJbot to your laptops or any other devices that have VNC viewer with the **same Wi-Fi network** as the TJbot.
- If the computer does not have the **same Wi-Fi network** as the TJbot, you can use the instructions under **module2: setting up your TJbot** to configure it.
- Then click on the coding terminal (the black icon with a blue shell) on the top left of the screen and start coding.

NOTE FOR CODING

**** If you use a Mac, use command + C to copy from pdf and try either ctrl + shift + V or ctrl + V to paste into the coding terminal or files in the VNC viewer.**

MODULE 4: DOWNLOADING THE PACKAGES FOR RUNNING TJBOT

- For installation of nodejs, open up the coding terminal and type in this code:

```
curl -sL https://deb.nodesource.com/setup_9.x | sudo -E bash -
```

- Then, press the enter button on your keyboard and let it run. - After the previous command has finished running, type in:

```
sudo apt-get install -y nodejs
```

- type enter and let it run. - After the previous command has finished running, type in: *curl -sL http://ibm.biz/tjbot-bootstrap | sudo sh -* and type enter and let it run. While this code is running, you will be asked a bunch of yes/no questions in the format y/n.
- Type in y/n based on the recommendations of the coding platform.
- At some point it will ask you to clone your tjbot into a folder, just copy and paste what appears as an example of a default folder which is: /home/pi/Desktop/tjbot
- For some of the questions, I have put the right responses that will make the TJbot operate properly, the questions you don't see here should be replied to with "y":

disable ipv6: y

Enable google dns: y

sound kernels: n

Would you like to run hardware tests at this time?: n

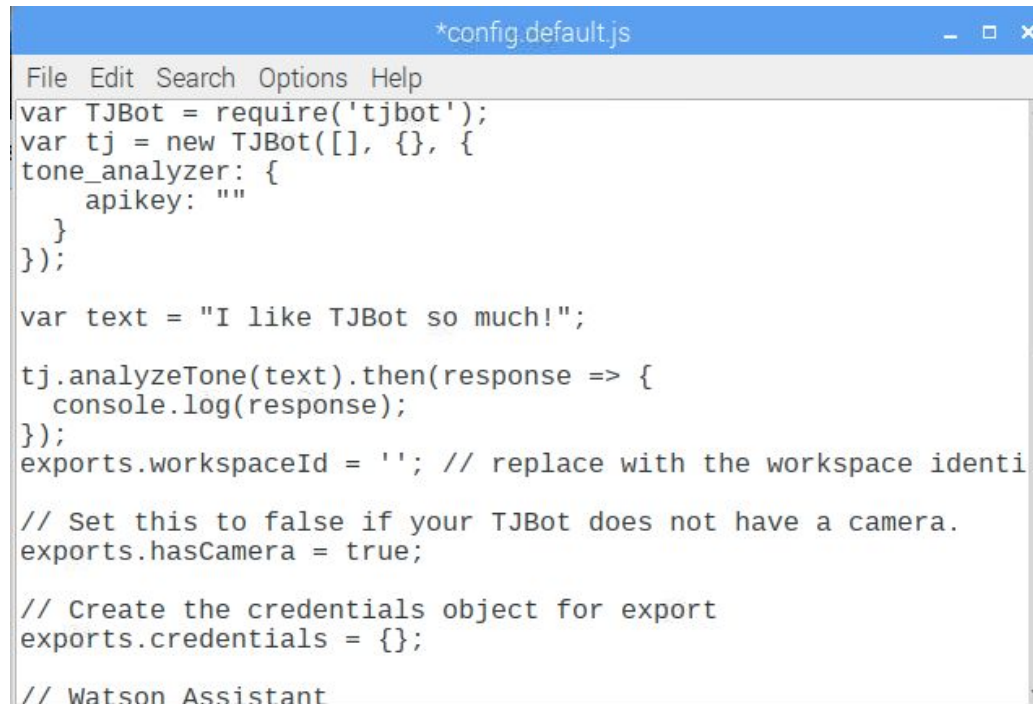
Reboot?: y

- The computer should reboot after the previous command

- Once it is done rebooting, type in: `git clone https://github.com/ibmtjbot/tjbot.git` into the coding platform.
- Then your TJbot should be ready to carry out specific tasks.

MODULE 5: CHATTING WITH YOUR TJBOT

- If you are interested in using earphones or loudspeaker to listen to your TJbot. go to this link:
<https://www.instructables.com/id/Build-a-Talking-Robot-With-Watson-and-Raspberry-Pi/> and carry out the command on **HDMI/3.5mm Audio Jack** part under step 3 and connect headphones or speaker into the audiojack. Once done with this set-up, do not do any of the following steps in the instructables, proceed to the next instructions on this manual
- Once you are done, do not do any of the following steps in the instructables, instead do the ones here.
- Open the coding terminal and run these codes one by one:
`cd tjbot/recipes/conversation`
`npm install`
`cp config.default.js config.js`
- Once you are done with running the codes, click on the file symbol with two files on the toolbar.
- Then, open the config.js file on the by navigating **/home/pi/tjbot/recipes/conversation/config.js**.
- This is what the config.js file should look like:



```

File Edit Search Options Help
var TJBot = require('tjbot');
var tj = new TJBot([], {}, {
  tone_analyzer: {
    apikey: ""
  }
});

var text = "I like TJBot so much!";

tj.analyzeTone(text).then(response => {
  console.log(response);
});

exports.workspaceId = ''; // replace with the workspace identi

// Set this to false if your TJBot does not have a camera.
exports.hasCamera = true;

// Create the credentials object for export
exports.credentials = {};

// Watson Assistant

```

- Replace the entire code on the file with the code in this link, <https://github.com/MUbarak123-56/tjbot/blob/master/recipes/conversation/config.default.js>.
- Once you have replaced the codes with the with the ones above, use this link to go on <https://console.bluemix.net/> and obtain the api keys. Here is a guide on how to do it.
- Once you've gotten on the page, login and you'll have access to a screen like the one in the next page:

The screenshot shows the IBM Cloud Dashboard interface. At the top, there's a navigation bar with the IBM Cloud logo, a search bar, and links to Catalog, Docs, Support, and Manage. Below the navigation bar, there's a 'Dashboard' section with a 'Customize' link and a 'Create resource' button. The main content area is divided into several panels:

- Resource summary:** A table listing resources with counts and status icons.

Resource	Count	Status
Cloud Foundry Apps	1	●
Cloud Foundry Services	9	
Services	11	
Storage	1	✓
Developer Tools	1	
- Planned maintenance:** A section showing upcoming events.

Next event: Mon, Jun 24, 2019 5:00 AM
 PLANNED: Update the TLS certificates (some clusters only)

Upcoming
 PLANNED: Update the TLS certificates (some clusters only)
 PLANNED: Upgrade the Blockchain firewall (Sao Paulo da...)
 PLANNED: Deploy a new Secure Gateway service release
- Location status:** A table showing the status of different regions.

Location	Status
Asia Pacific	✓
Europe	✓
North America	✓
South America	✓
- Apps:** A section with a code icon and a message: "You can view your apps here after you create them. Learn more about how to get started." Below this is a "Create an app" button.
- Support cases:** A section with a support icon and a message: "You can view a summary of your support cases here after you submit them. Learn more about how to get support."

- For each apikey, you'll notice there is a code like

```
exports.credentials.whatever = {
  apikey: "" };
```

- Search for the service that is put in place of **whatever** in the search bar on the top of the page and you'll get access to a page like this:

IBM Cloud Catalog Docs Support Manage Search for resource... MUBARAK GANIYU's A...

View all Knowledge Catalog Lite • IBM

Simplify data science and data compliance with IBM Watson Knowledge Catalog. Make your data easy to find and share while controlling access to ensure appropriate use.

[View Docs](#) [Terms](#)

AUTHOR IBM
PUBLISHED 01/11/2019
TYPE Service

Service name: Knowledge Catalog-wo

Choose a region/location to deploy in: Dallas

Select a resource group: 1 Default

Features

- Catalog**
Create a 360-degree view of your data, no matter where (or in what format) it is stored.
- Find**
With the right tools, discover the data you need, and collaborate to discover fresh insights.
- Govern**
Control data access by defining policies and monitoring enforcement.

Images

Need Help? [Contact IBM Cloud Support](#) Estimate Monthly Cost [Cost Calculator](#) [Create](#)

- Click on create and it'll take a few seconds to create. Do not change anything, just click on create.
- Once created, click on the manage button on the top left corner right on top of getting started. You'll have access to a page like this:

IBM Cloud Catalog Docs Support Manage Search for resource... MUBARAK GANIYU's A...

Manage

Dashboard / Speech to Text-4v

Resource Group: Default Location: Dallas

Get started with the service. Plan: Lite [Upgrade](#)

[Getting started tutorial](#) [API reference](#)

Credentials [Download](#) [Show Credentials](#)

API Key:

URL: <https://stream.watsonplatform.net/speech-to-text/api>

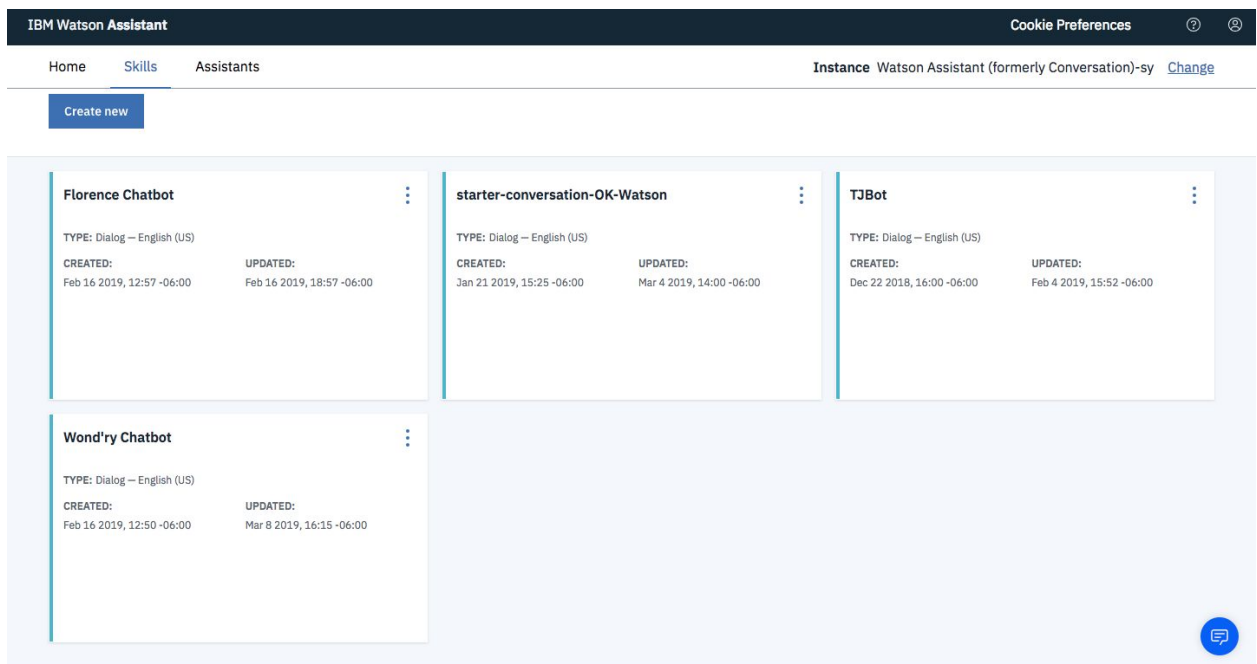
Try an API call

Call the `POST /v1/recognize` method to request a basic transcription of a FLAC audio file with no additional request parameters. First, download the sample audio file [audio-file.flac](#). Then, issue the following command to call the service's `/v1/recognize` method for basic transcription with no parameters. The example uses the `Content-Type`

- Then, click on the **show credentials** to show your credentials and plug your api key into the code like:

```
exports.credentials.tone_analyzer = { apikey:
"TfL6kx8Vy9dCGyaP1YsTOak_guM1jynq8buZd9NDMrVr"};
```

- Repeat this process for every service/credential and obtain its apikey.
- For the workspace id, you'll need to use the browser on the vnc viewer by clicking on the web browser logo which is on the top left corner.
- Then, go on this website, <https://www.ibm.com/cloud/watson-assistant/> and log in using your bluemix credentials.
- Once logged in, click on **skills** and navigate to **create new**. Then, click on **import skill**, then **choose JSON file** and navigate your way through **/home/pi/tjbot/recipes/conversation/** and you shall see two .json files.
- Choose the one that says **workspace-sample.json** and open it. Your webpage should look like this with the **TJbot folder visible** (that's what we need).



- Then, click on the three dots on the TJbot folder and click on **view api details**.
- This will take you to a page with a bunch of credentials. Then copy and paste the **workspace id** credential into the config.js file.

- Then, save your changes by exiting the config.js file.
- Then, run the code on the coding terminal using: `sudo node conversation.js`
- Follow the **Troubleshooting** instructions under Step 7 in this link, <https://www.instructables.com/id/Build-a-Talking-Robot-With-Watson-and-Raspberry-Pi/> if the codes are not working.
- Next time, you run the code, just plug in:


```
cd tjbot/recipes/conversation
npm install
sudo node conversation.js
```
- You can speak to the TJbot by saying:

Watson, hello

Watson, tell me a joke

Watson, what can you see?

Watson, what can you do?

Watson, wave your arms.

Watson, how are you doing.

Note: Watson is its trigger word just like Siri for apple

MODULE 6: CHANGING THE COLOUR OF THE LED LIGHT

- Open the coding terminal and run these codes one by one:


```
cd tjbot/recipes/speech_to_text
npm install
cp config.default.js config.js
```
- Once you are done with running the codes, click on the file symbol with two files on the toolbar.
- Then, open the config.js file on the by navigating **/home/pi/tjbot/recipes/speech_to_text/config.js**
- This is what the config.js file should look like:

```

*config.default.js
File Edit Search Options Help
var TJBOT = require('tjbot');
var tj = new TJBOT([], {}, {
  tone_analyzer: {
    apikey: ""
  }
});

var text = "I like TJBOT so much!";

tj.analyzeTone(text).then(response => {
  console.log(response);
});
exports.workspaceId = ''; // replace with the workspace identi

// Set this to false if your TJBOT does not have a camera.
exports.hasCamera = true;

// Create the credentials object for export
exports.credentials = {};

// Watson Assistant

```

-
- Replace the code on the file with this code in this link,
https://github.com/MUbarak123-56/tjbot/blob/master/recipes/speech_to_text/config.default.js.
- Once you have replaced the codes with the with the ones above, use this link to go on <https://console.bluemix.net/> and obtain the api keys. Here is a guide on how to do it.
- Once you've gotten on the page, login and you'll have access to a screen like this web page:

The screenshot shows the IBM Cloud Dashboard interface. At the top, there's a navigation bar with 'IBM Cloud' and a search bar. Below the navigation bar, the 'Dashboard' is visible with a 'Customize' link and a 'Create resource' button. The main content area is divided into several sections:

- Resource summary:** A table listing resources with counts and status indicators.

Resource	Count	Status
Cloud Foundry Apps	1	●
Cloud Foundry Services	9	
Services	11	
Storage	1	✓
Developer Tools	1	
- Location status:** A table showing the status of different regions.

Location	Status
Asia Pacific	✓
Europe	✓
North America	✓
South America	✓
- Apps:** A section with a code icon and a message: 'You can view your apps here after you create them. Learn more about how to get started.' Below it is a 'Create an app' button.
- Support cases:** A section with a support icon and a message: 'You can view a summary of your support cases here after you submit them. Learn more about how to get support.'
- Planned maintenance:** A section with a calendar icon and a message: 'Next event: Mon, Jun 24, 2019 5:00 AM'. It lists several planned events: 'PLANNED: Update the TLS certificates (some clusters only)', 'PLANNED: Upgrade the Blockchain firewall (Sao Paulo da...)', and 'PLANNED: Deploy a new Secure Gateway service release'.

- For each apikey, you'll notice there is a code like


```
exports.credentials.whatever = {
  apikey: "" };
```

- Search for the service that is put in place of whatever in the search bar and you'll get access to a page like this:

The screenshot shows the IBM Cloud Knowledge Catalog service page. The top navigation bar includes 'IBM Cloud', 'Catalog', 'Docs', 'Support', and 'Manage'. A search bar is present with the text 'Search for resource...'. The user's name 'MUBARAK GANIYU's A...' is visible in the top right corner.

The main content area is titled 'Knowledge Catalog' with a subtitle 'Lite • IBM'. It includes a description: 'Simplify data science and data compliance with IBM Watson Knowledge Catalog. Make your data easy to find and share while controlling access to ensure appropriate use.' Below this are links for 'View Docs' and 'Terms'.

Metadata is displayed: AUTHOR: IBM, PUBLISHED: 01/11/2019, TYPE: Service.

Configuration options include:

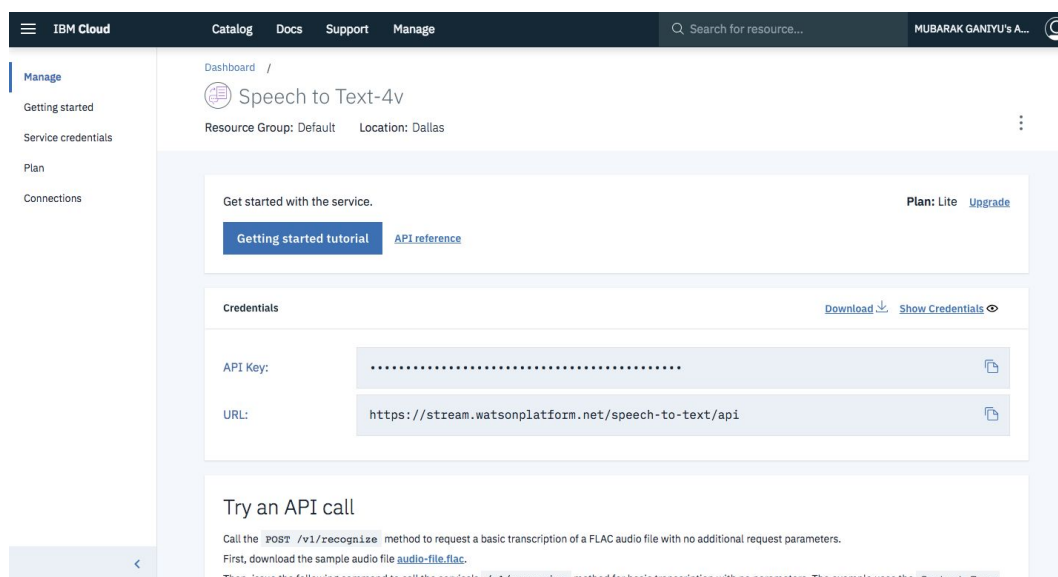
- Service name:** Knowledge Catalog-wo
- Choose a region/location to deploy in:** Dallas (dropdown menu)
- Select a resource group:** Default (dropdown menu)

Features:

- Catalog:** Create a 360-degree view of your data, no matter where (or in what format) it is stored.
- Find:** With the right tools, discover the data you need, and collaborate to discover fresh insights.
- Govern:** Control data access by defining policies and monitoring enforcement.

At the bottom, there are links for 'Need Help? Contact IBM Cloud Support' and 'Estimate Monthly Cost Cost Calculator', along with a 'Create' button.

-
- Click on create and it'll take a few seconds to create. Do not change anything, just click on create.
- Once created, click on the **manage** button on the top left corner right on top of **getting started**. You'll have access to a page like this:



- Then, click on the **show credentials** to show your credentials and plug them into the code like:

```
exports.credentials.tone_analyzer = { apikey:
"HyL6kx5HO9dC9yb21YsTpaj_guW1jrnqp9uZdUNDMrVe"};
```

- Repeat this process for every service/credential and obtain the other api keys.
- Then, save your changes by exiting the config.js file.
- Then, run the code on the coding terminal using: `sudo node stt.js`
- If you are having problem with turning on the LED light, you should move thered wire from pin 1 to either pin 2/pin 3. - There are other instructions for fixing the LED light if it is not working well under **Troubleshooting** in Step 6 on this link:

<https://www.instructables.com/id/Use-Your-Voice-to-Control-a-Light-With-Watson/>

- Next time you run the entire sequence, Just type in the following commands:

```
cd tjbot/recipes/speech_to_text
npm install
sudo node stt.js
```

- You can speak to the TJbot by saying:
turn the lights on

turn the lights off
turn the lights red
turn the lights blue
turn the lights [to any other color]

MODULE 7: MODIFYING/CREATING YOUR OWN CONVERSATION STREAM WITH THE TJBOT

- If you want to learn how to create your own conversation flow with the TJbot, you will go ahead to this website: <https://cognitiveclass.ai/>.
- Login using your bluemix credentials or you might have to create an account for the website.
- Once you have logged in, you will see an icon called courses on the top of the page. Select it and it will direct you to another page.
- In this new page, you shall click on the **Build your own chatbot** and enroll in it. Then, navigate to courseware, which is on the top left corner of the page and take the course from Module 1 to Module 4. Those are the modules that are needed for building a chatbot for the TJbot.
- With the knowledge you have gained from this online class, you can create your own chatbot and use its credentials on TJbot to chat with it.