

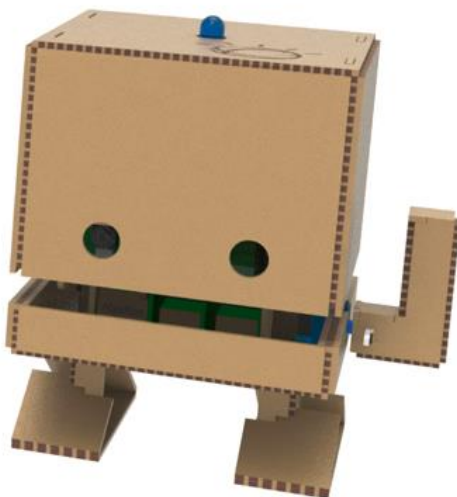


マイク・スピーカーを使って TJBOT と 会話

Watson 「Speech to Text」、「Conversation」、「Text to Speech」と Raspberry Pi、USB マイク、Bluetooth スピーカーを使って、Watson と簡単な会話をします。
会話を登録すれば、色々話せます。

<使用する Watson サービス>

「Speech to Text」、「Conversation」、「Text to Speech」



こんにちは、私の名前は TJBOT です！

私は、楽しい方法で Watson サービスを理解するために、お手伝いをするオープンソースプロジェクトです。

TJBOT は、IBM 最初の会長兼最高経営責任者（CEO）であるトーマス・J・ワトソンの名前を愛称としています。

TJBOT は、IBM Research の Maryam Ashoori によって、認知対象の設計と実装におけるベストプラクティスを見つけるための実験として作成されました。

もくじ

- 【1】Web Site 情報
- 【2】内容
- 【3】H/W セットアップ
- 【4】Watson「Speech to Text」、「Conversation」、「Text to Speech」サービスの作成
- 【5】Rasbian の最新化と Node.js、npm のインストール
- 【6】コードのダウンロードとインストール
- 【7】会話フローの作成
- 【8】「Speech to Text」、「Conversation」、「Text to Speech」、WorkspaceID 構成情報の設定
- 【9】Bluetooth スピーカの Pulse Audio が使えるようにの設定変更
- 【10】プログラムの実行
 - 【11】日本語対応に変更
 - 【12】日本語対応版プログラムの実行

材料 (価格は変わる可能性があります。消費税、郵送料は含まれません)

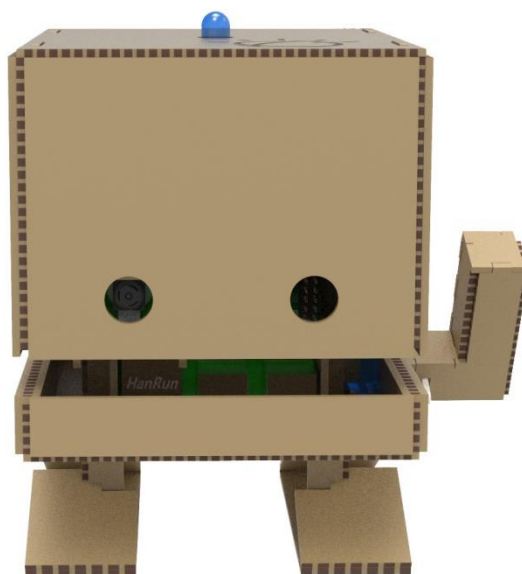
材料名	補足	価格 (消費税、送料含まず)	購入先例
厚紙	2mm厚	¥650	東急ハンズ
レーザーカット費用	1000円/10分	¥500	coromoza
Raspberry Pi3 Model B	Bluetooth、無線LAN含む。	¥5,600	秋月電子通商
電源 (Raspberry Pi3用)	スイッチングACアダプター5V2.5A AD-B50P250、USBケーブル Aオス-マイクロBオス 0.15m A-microB	¥1,210	秋月電子通商
microSD Card 8GB Class10	Transcend microSDHCカード 8GB Class10 (無期限保証)	¥1,200	アマゾン
Bluetoothスピーカー	Anker SoundCore mini コンパクト Bluetoothスピーカー	¥2,399	アマゾン
USBマイク	BU-Bauty PC Mac用USBマイク 超小型 超ミニ 22mmx18mmx5mm	¥599	アマゾン
	小計	¥12,158	
HDMIディスプレイ	Raspberry Pi3初期設定時のみ使用。テレビで代用		
HDMIケーブル	Raspberry Pi3初期設定時のみ使用。HDMI 1.4ケーブル	¥250	秋月電子通商
USBマウス	Raspberry Pi3初期設定時のみ使用。Logicool ロジクール 有線光学式3ボタンマウス M100r ブラック	¥475	アマゾン
USBキーボード	Raspberry Pi3初期設定時のみ使用。サンワサプライ USBキーボード(ブラック) SKB-L1UBK	¥664	アマゾン
	小計	¥1,389	
	合計	¥13,547	

【1】Web Site 情報

<https://www.instructables.com/id/Build-a-Talking-Robot-With-Watson-and-Raspberry-Pi/>

【2】内容

Watson 「Speech to Text」、「Conversation」、「Text to Speech」と Raspberry Pi、USB マイク、Bluetooth スピーカーを使って、Watson と簡単な会話をします



【3】H/W セットアップ

(1) USB マイク

BU-Bauty PC Mac 用 USB マイク 超小型 超ミニ 22mmx18mmx5mm を使用しました。



```
$ lsusb
```

lsusb コマンドで認識されているか確認します。

```
pi@raspberrypi:~$ lsusb
Bus 001 Device 004: ID 0d8c:0016 C-Media Electronics, Inc.
Bus 001 Device 003: ID 0424:ec00 Standard Microsystems Corp. SMSC9512/9514 Fast Ethernet Adapter
Bus 001 Device 002: ID 0424:9514 Standard Microsystems Corp.
Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
```

“Bus 001 Device 004: ID 0d8c:0016 C-Media Electronics, Inc”で認識されています。

以下のコマンドでハードウェアデバイスを調べます。

```
$ arecord -l
```

```
pi@raspberrypi:~$ arecord -l
**** ハードウェアデバイス CAPTURE のリスト ****
カード 1: Microphone [USB Microphone], デバイス 0: USB Audio [USB Audio]
サブデバイス: 1/1
サブデバイス #0: subdevice #0
```

カード番号は1です。

マイクの感度を調整します

```
$ amixer sset Mic 50 -c 1
```

(-cに続く番号は card 番号です)

```
pi@raspberrypi:~$ amixer sset Mic 50 -c 1
Simple mixer control 'Mic',0
Capabilities: cvolume cvolume-joined cswitch cswitch-joined
Capture channels: Mono
Limits: Capture 0 - 62
Mono: Capture 50 [81%] [16.59dB] [on]
```

81%の値がマイク感度です。


```
load-module module-always-sink

### Automatically suspend sinks/sources that become idle for too long
load-module module-suspend-on-idle

### Enable positioned event sounds
load-module module-position-event-sounds

### Automatically load driver modules for Bluetooth hardware
.ifexists module-bluetooth-policy.so
load-module module-bluetooth-policy
.endif

.ifexists module-bluetooth-discover.so
load-module module-bluetooth-discover
.endif

^G Get Help   ^O WriteOut  ^R Read File  ^Y Prev Page  ^K Cut Text   ^C Cur Pos
^X Exit       ^J Justify   ^W Where Is  ^V Next Page  ^U UnCut Text ^T To Spell
```

・pulseaudio サービスを起動

```
$ sudo systemctl start pulseaudio.service
```

・pulseaudio サービスを自動起動するよう設定

```
$ sudo systemctl enable pulseaudio.service
```

・root、pi ユーザが PulseAudio で音を出せるようにします

```
$ sudo gpasswd -a root pulse-access
```

```
$ sudo gpasswd -a pi pulse-access
```

```
pi@raspberrypi:~$ sudo systemctl start pulseaudio.service
pi@raspberrypi:~$ sudo systemctl enable pulseaudio.service
Created symlink from /etc/systemd/system/multi-user.target.wants/pulseaudio.service to /etc/systemd/system/pulseaudio.service.
pi@raspberrypi:~$ sudo gpasswd -a pi pulse-access
ユーザ pi をグループ pulse-access に追加
```

ここまでで PulseAudio 関連の設定が完了しましたので、次に bluetoothctl で Bluetooth デバイスを接続します。

スピーカーの電源をオン。


```
$ sudo bluetoothctl
```

```
[bluetooth]# power on
```

デバイスをスキャンし、スピーカーを探します。

```
[bluetooth]# scan on
```

Bluetoothスピーカーが見つかったら、スキャンを停止します。

```
[bluetooth]# scan off
```

```
pi@raspberrypi:~$ sudo bluetoothctl
[NEW] Controller B8:27:EB:C1:8F:AF raspberrypi [default]
[bluetooth]# scan on
Discovery started
[CHG] Controller B8:27:EB:C1:8F:AF Discovering: yes
[NEW] Device 00:E0:4C:58:6C:EC SoundCore mini
[NEW] Device CF:E9:B1:E0:D6:98 ID107 HR
[CHG] Device 00:E0:4C:58:6C:EC Class: 0x240404
[CHG] Device 00:E0:4C:58:6C:EC Icon: audio-card
[bluetooth]# scan off
[CHG] Device CF:E9:B1:E0:D6:98 RSSI is nil
[CHG] Device 00:E0:4C:58:6C:EC RSSI is nil
[CHG] Controller B8:27:EB:C1:8F:AF Discovering: no
Discovery stopped
[bluetooth]#
```

[NEW] Device 00:E0:4C:58:6C:EC SoundCore mini

で Bluetooth スピーカーのデバイス情報が入手できました。

```
[bluetooth]# trust 00:E0:4C:58:6C:EC
```

```
[bluetooth]# pair 00:E0:4C:58:6C:EC
```

```
[bluetooth]# connect 00:E0:4C:58:6C:EC
```

「ピロリン」という音が鳴ると、接続完了です。

```
[bluetooth]# exit
```

で bluetoothctl から抜けます。

・なかなか接続できない場合

<https://raspberrypi.stackexchange.com/questions/44497/having-an-issue-with-bluetooth-manager-pairing-on-the-pi>

Syslog を確認します。

```
$ cat /var/log/syslog
```

“bluetoothd[794]: a2dp-sink profile connect failed for 00:E0:4C:58:6C:EC: Protocol not available”というエラーの場合下記手順を試してください。

(1) PulseAudio Bluetooth ライブラリーがインストールされているか確認

```
$ sudo apt-get install pulseaudio-module-bluetooth
```

(2) Pulse Audio server が稼働しているか確認

まず停止します。

```
$ sudo killall pulseaudio
```

起動します。

```
$ pulseaudio --start
```

(3) bluetoothctl で connect します。

```
[bluetooth]# connect 00:E0:4C:58:6C:EC
```

【4】Watson「Speech to Text」、「Conversation」、「Text to Speech」サービスの作成

その作成

Bluemix にログイン

<https://console.bluemix.net/>



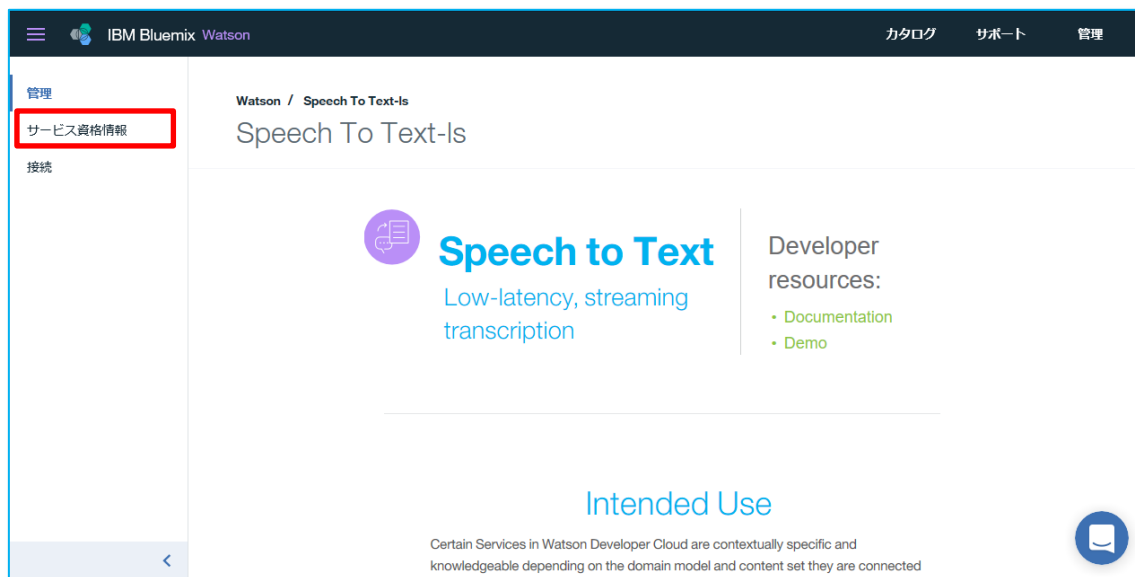
(1) カタログより「Speech to Text」を選択



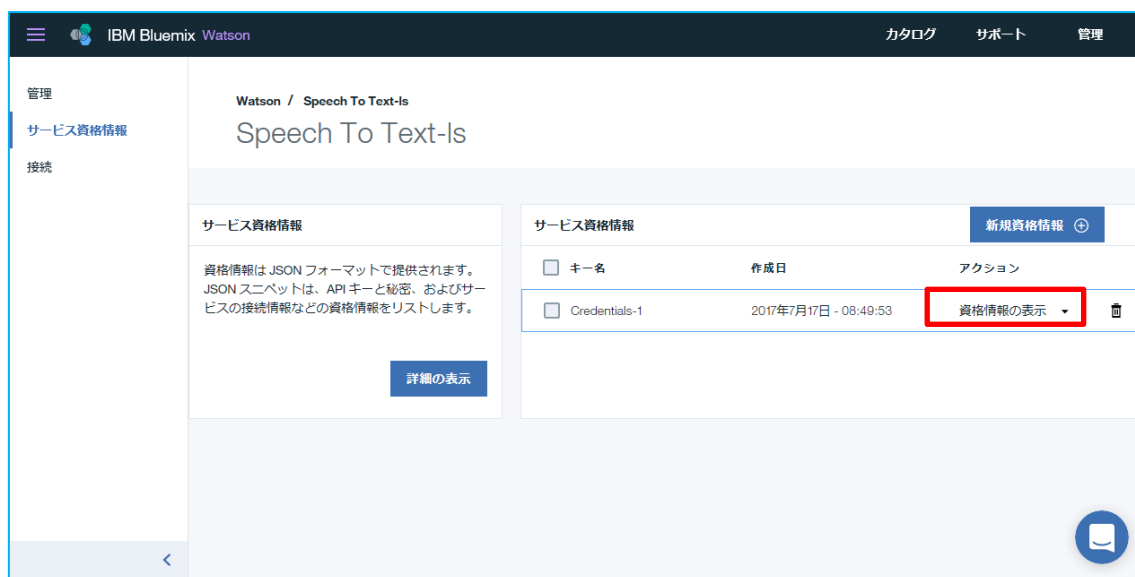
「作成」をクリック



「サービス資格情報」をクリック



「資格情報の表示」をクリック



「username」「password」を後程使用しますので、メモ帳などにコピー保存してください。



```
{
  "url": "https://stream.watsonplatform.net/speech-to-text/api",
  "username": "[redacted]",
  "password": "[redacted]"
}
```

(2) カタログより「Conversation」を選択



「作成を」クリック

IBM Bluemix カタログ

← すべて表示

Conversation

アプリケーションに自然言語インターフェースを追加してエンドユーザーとのやり取りを自動化します。共通アプリケーションには、任意のチャネルやデバイスとの統合や通信を行える仮想エージェントとチャットボットが含まれます。簡単に利用できる Web アプリケーションを使用して Watson Conversation サービスのトレーニングを行います。サービスは、アプリとユーザーの自然な対話の流れを迅速に組み立てられるようにデザインされ、スケーラブルで費用効率の高いソリューションをデプロイします。

サービス名:
Conversation-5z

資格情報名:
Credentials-1

イメージ

画面キャプチャー、スライダー、またはビデオの拡大と表示を行うには、イメージをクリックします。画面キャプチャーは、プロビジョンされた後のサービスのユーザー・インターフェースを示します。

接続:
ヘルプが必要ですか?
Bluemix 営業担当へのお問い合わせ

月額費用の計算
費用計算

作成

「サービス資格情報」をクリック

IBM Bluemix Watson

管理

サービス資格情報

プラン

接続

Watson / Conversation-5z

Conversation-5z

Conversation

Add a natural language interface to your application to automate interactions with your end users. Common applications include virtual agents and chat bots that can integrate and communicate on any

Launch tool ↗

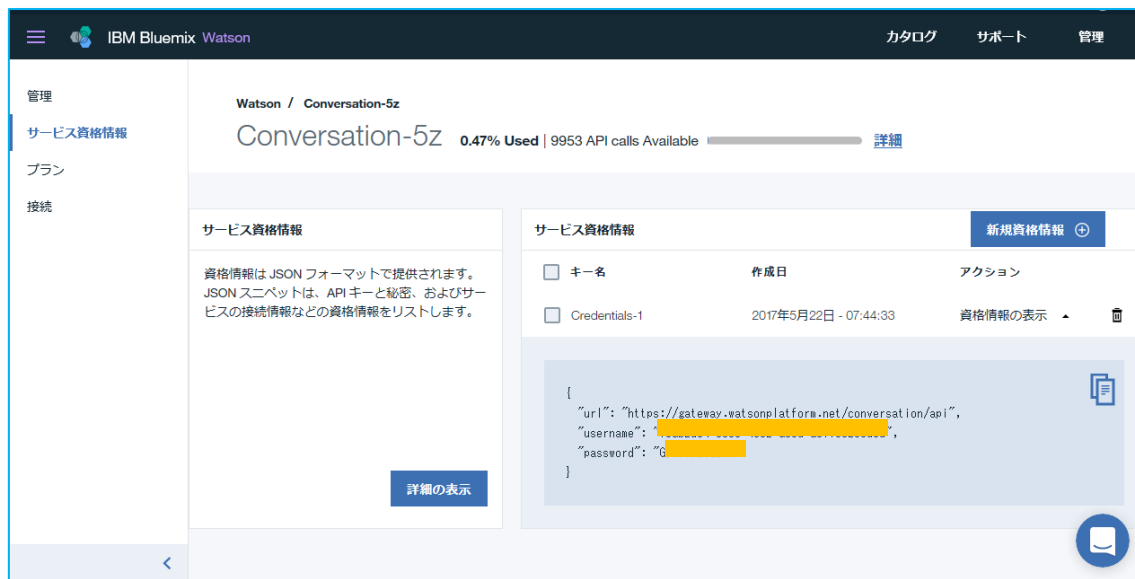
Developer resources:

- Documentation
- Demo

「資格情報の表示」をクリック



「username」「password」を後程使用しますので、メモ帳などにコピー保存してください。



```
{  
  "url": "https://gateway.watsonplatform.net/conversation/api",  
  "username": "G...",  
  "password": "G..."  
}
```

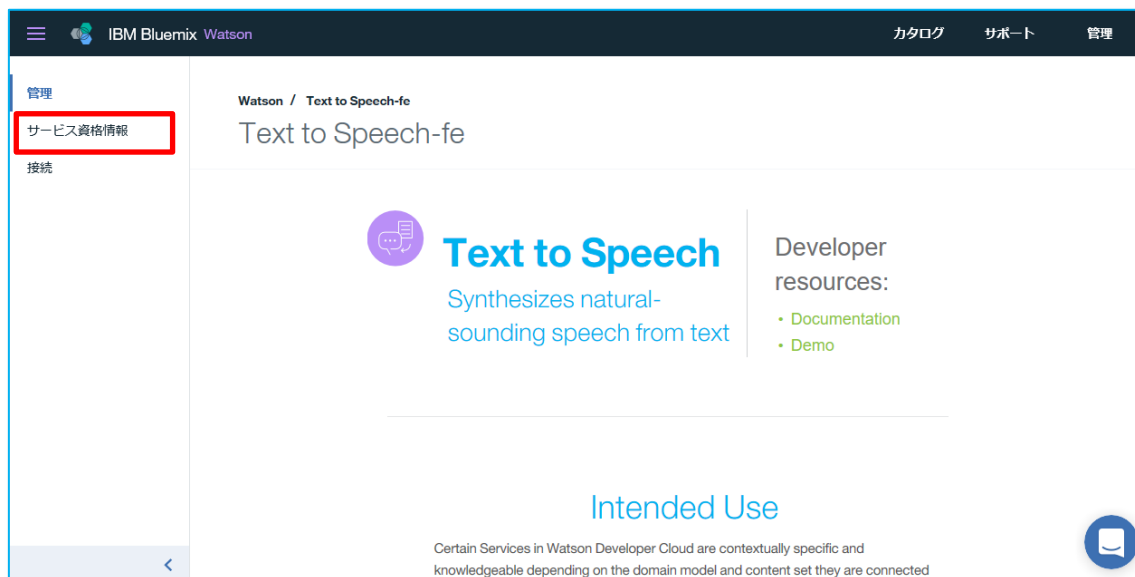

(3) カタログより「Text to Speech」を選択



「作成」をクリック



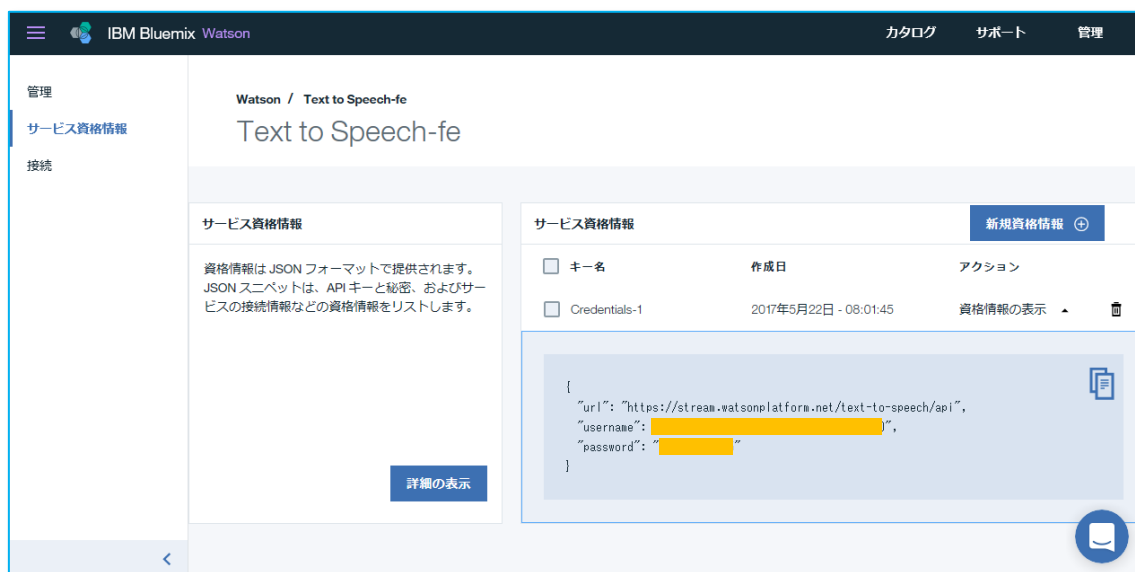
「サービス資格情報」をクリック



「資格情報の表示」をクリック



「username」「password」を後程使用しますので、メモ帳などにコピー保存してください。



The screenshot shows the IBM Bluemix Watson console interface. The main heading is 'Watson / Text to Speech-fe' and 'Text to Speech-fe'. On the left, there is a navigation menu with '管理' (Management), 'サービス資格情報' (Service Credentials), and '接続' (Connections). The main content area is divided into two panels. The left panel, titled 'サービス資格情報', contains a note: '資格情報はJSONフォーマットで提供されます。JSON スニペットは、API キーと秘密、およびサービスの接続情報などの資格情報をリストします。' (Credentials are provided in JSON format. JSON snippets list credentials such as API keys and secrets, and service connection information). Below this note is a '詳細の表示' (Show details) button. The right panel, also titled 'サービス資格情報', features a '新規資格情報' (New credential) button and a table with columns for 'キー名' (Key name), '作成日' (Created date), and 'アクション' (Action). The table lists one credential: 'Credentials-1' created on '2017年5月22日 - 08:01:45'. Below the table is a JSON snippet for the API configuration, with the 'username' and 'password' fields redacted with yellow boxes. The JSON is:

```
{
  "url": "https://stream.watsonplatform.net/text-to-speech/api",
  "username": "[REDACTED]",
  "password": "[REDACTED]"
}
```

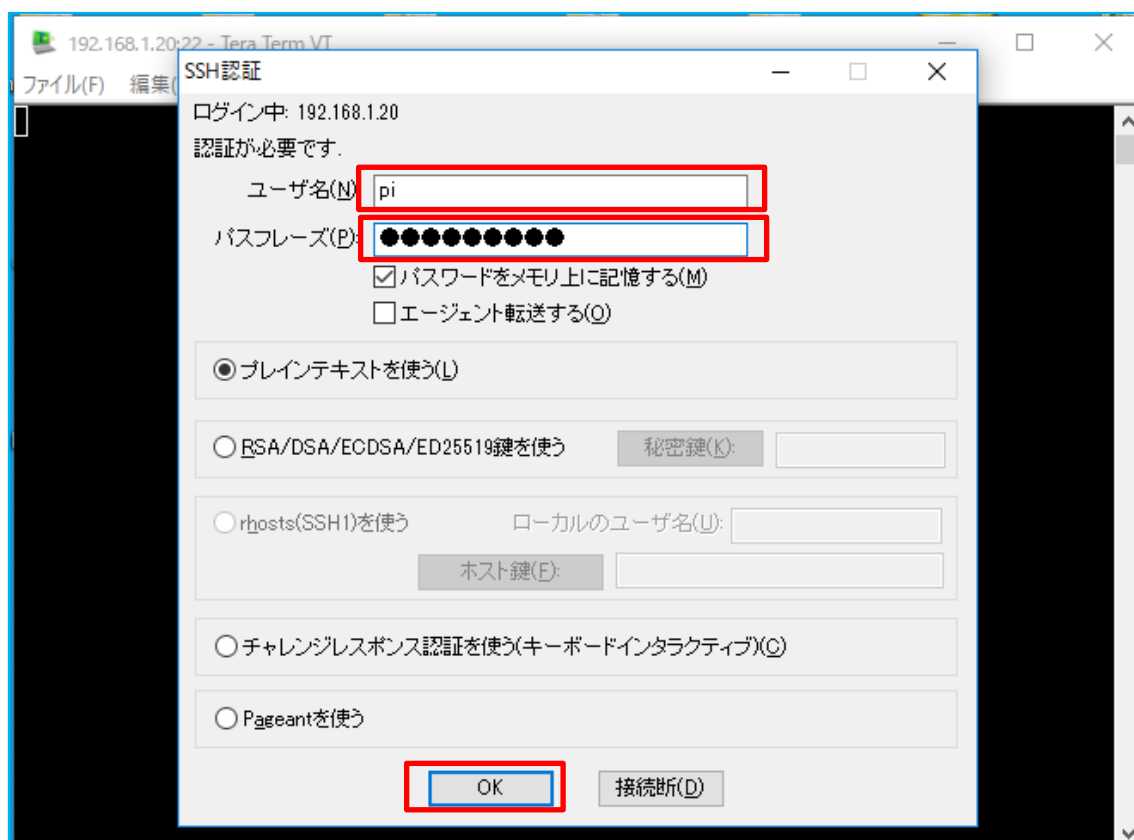
```
{
  "url": "https://stream.watsonplatform.net/text-to-speech/api",
  "username": "[REDACTED]",
  "password": "[REDACTED]"
}
```

【5】Raspbian の最新化と Node.js、npm、のインストール

Teraterm などで SSH 接続します。ID、パスワードの初期値は、

ID:pi

Password: raspberry です。



以下のコマンドを実行してください。

```
$ sudo apt-get update
```

```
$ sudo apt-get dist-upgrade
```

```
$ curl -sL https://deb.nodesource.com/setup_6.x | sudo -E bash -
```

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

【6】コードのダウンロードとインストール

```
$ git clone https://github.com/ibmtjbot/tjbot.git  
$ cd tjbot/recipes/conversation  
$ npm install
```

```
pi@raspberrypi: ~/tjbot/recipes/conversation $ ls -al  
合計 36  
drwxr-xr-x  3 pi pi 4096  7月 20 21:49 .  
drwxr-xr-x  5 pi pi 4096  7月 20 21:48 ..  
-rw-r--r--  1 pi pi  905  7月 20 21:48 .gitignore  
-rw-r--r--  1 pi pi 1719  7月 20 21:48 README.md  
-rw-r--r--  1 pi pi  942  7月 20 21:48 config.default.js  
-rw-r--r--  1 pi pi 2015  7月 20 21:48 conversation.js  
drwxr-xr-x 125 pi pi 4096  7月 20 21:51 node_modules  
-rw-r--r--  1 pi pi  632  7月 20 21:48 package.json  
-rw-r--r--  1 pi pi 3233  7月 20 21:48 workspace-sample.json
```

【7】会話フローの作成

ここでは作成済みのサンプルのデータを使用します。

データは“【6】コードのダウンロードとインストール”でインストールした
「~/tjbot/recipes/conversation/workspace-sample.json」です。
SCP などを使って、以降で操作するクライアント PC にコピーしてください。

Bluemix にログイン

<https://console.bluemix.net/>



Log into IBM Bluemix

IBMID またはメール・アドレスを入力してください
IBM ID をお忘れですか?

Email or IBMID

次へ

New to Bluemix? [Sign Up](#)



Log into IBM Bluemix

IBM ID:

パスワード パスワードをお忘れの場合

Password

Log in

別の IBM アカウントまたは E メールを使用してください

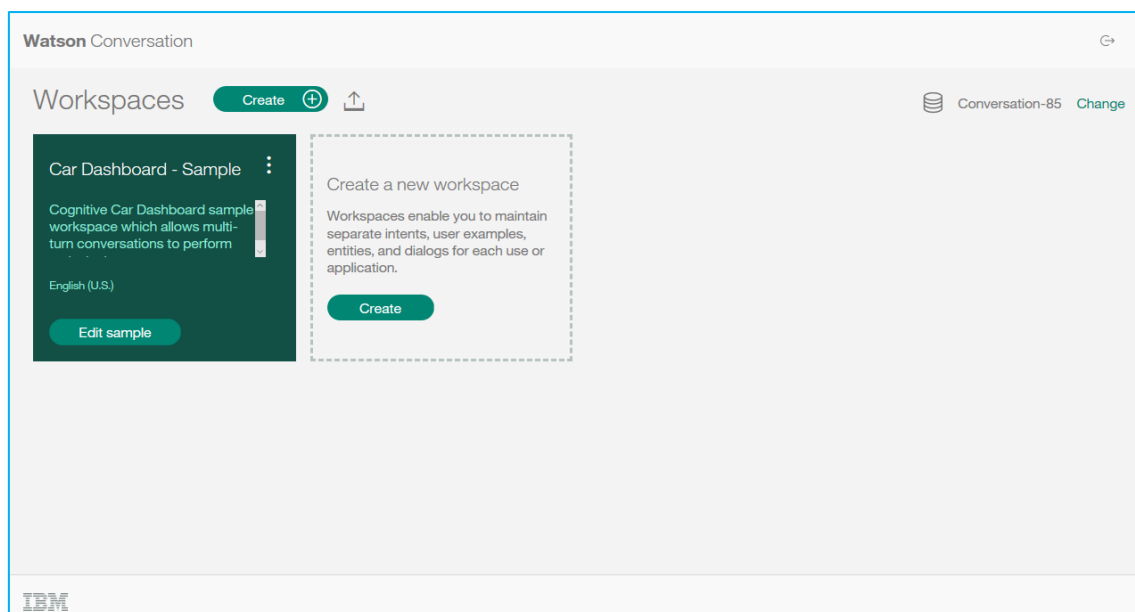
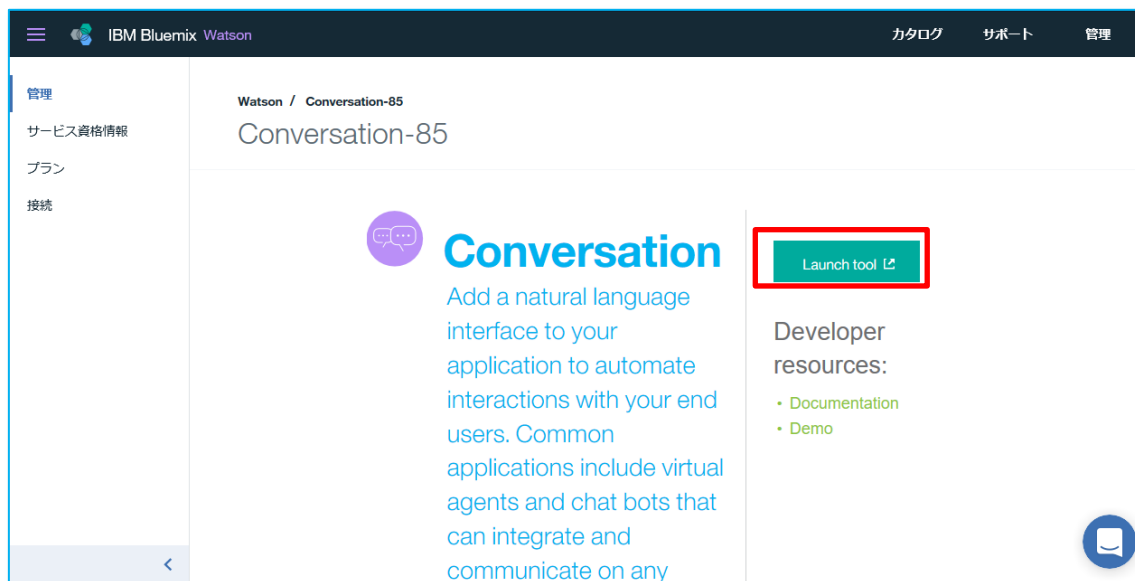
「ハンバーガーメニュー」をクリック



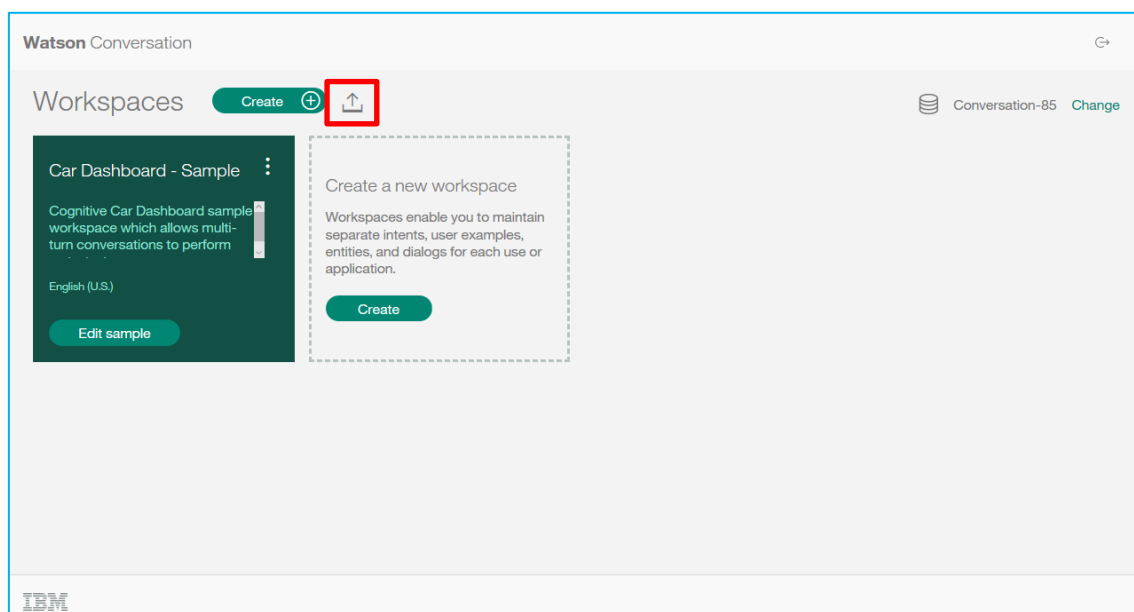
「ダッシュボード」をクリック



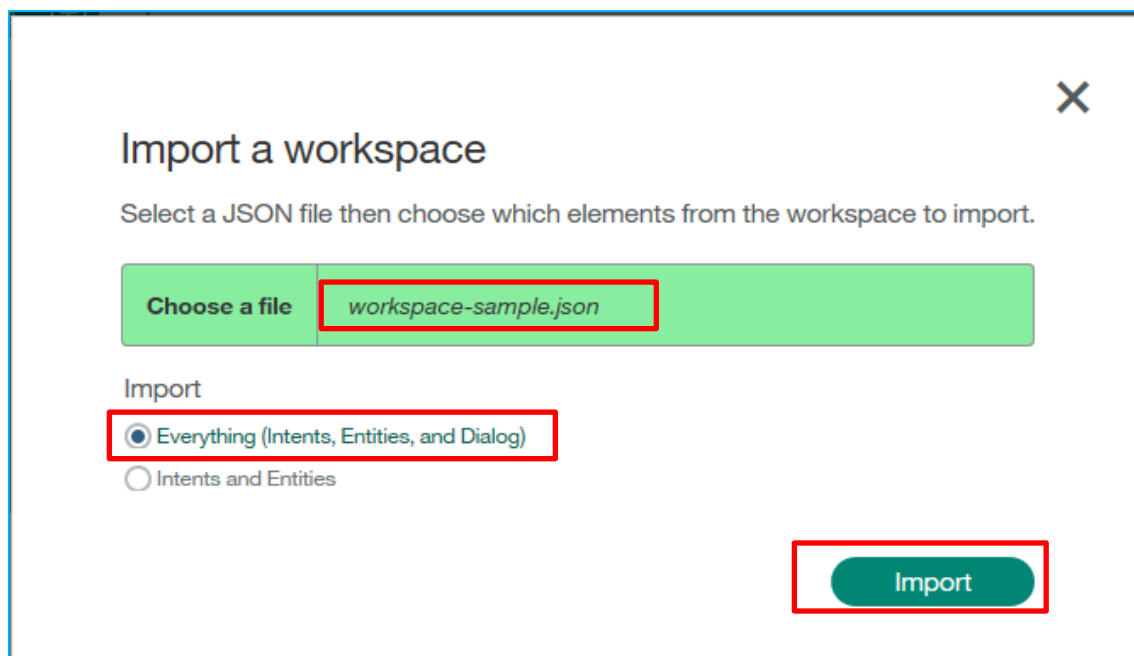
「Launch tool」ボタンを押す



「import workspace」をクリック

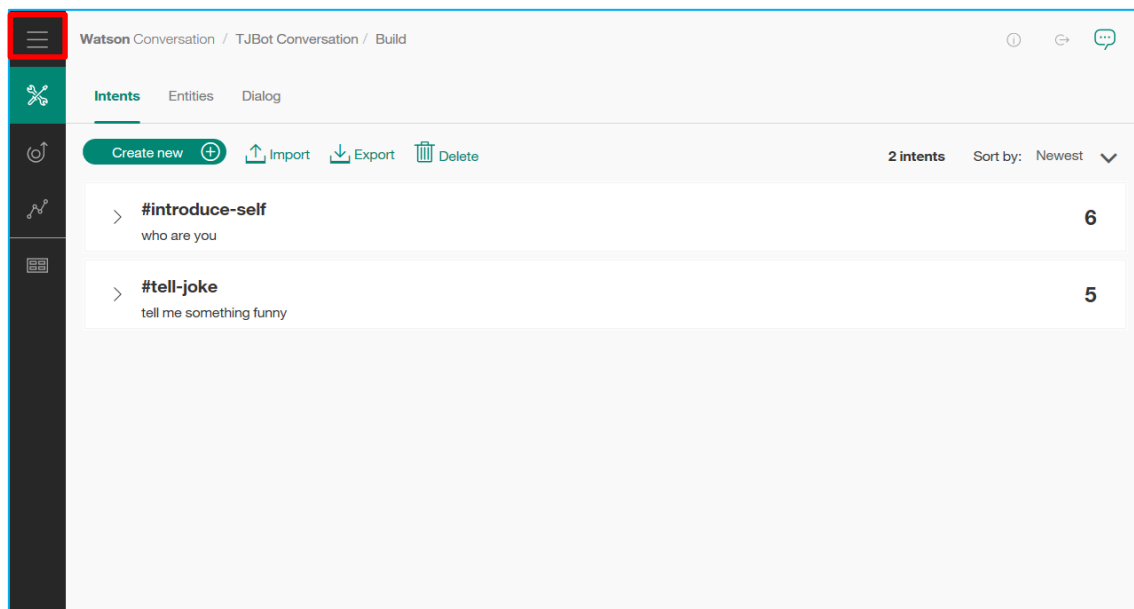


先程コピーしたファイル「workspace-sample.json」を選択
Import「Everything」を選択し、「Import」ボタンをクリック。

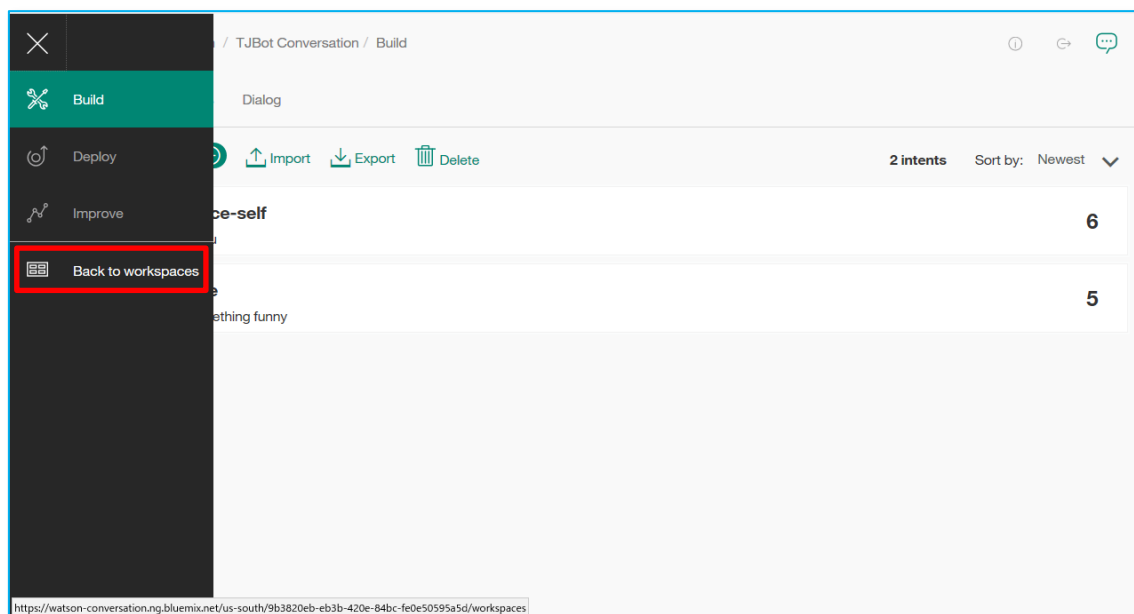


「workspace-sample.json」がインポートされました。

「ハンバーガーメニュー」をクリック

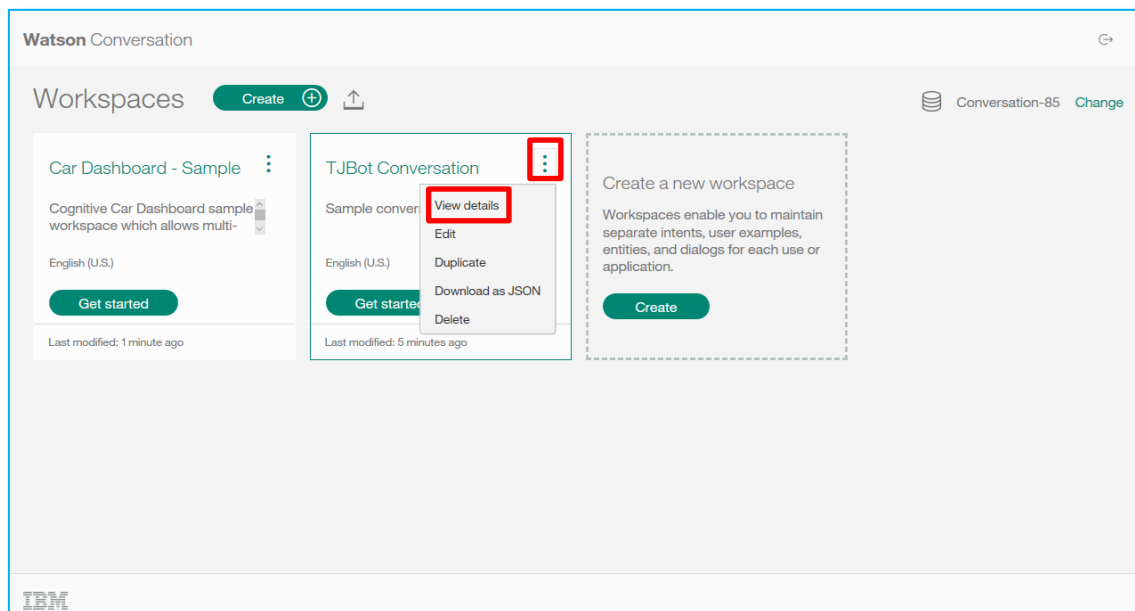


「Back to workspace」を選択してワークスペースに戻ります。

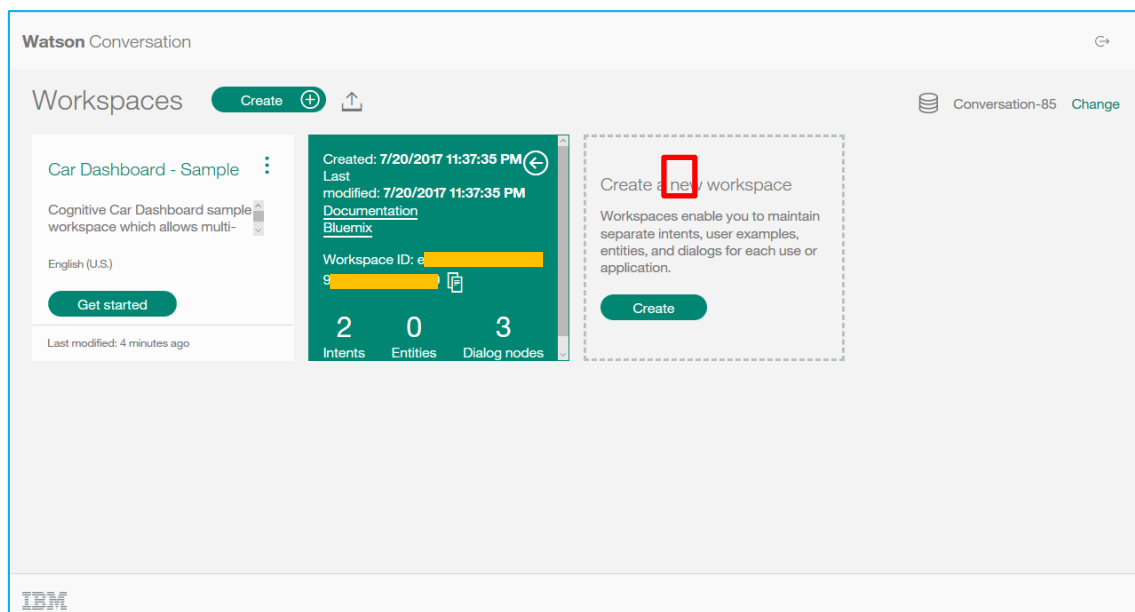


Workspace ID を取得します。

「View detail」を選択。



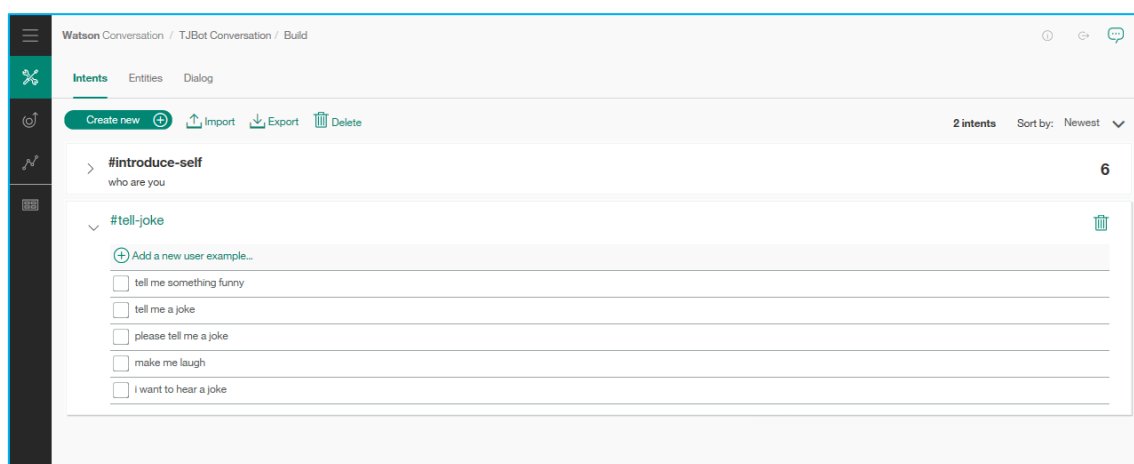
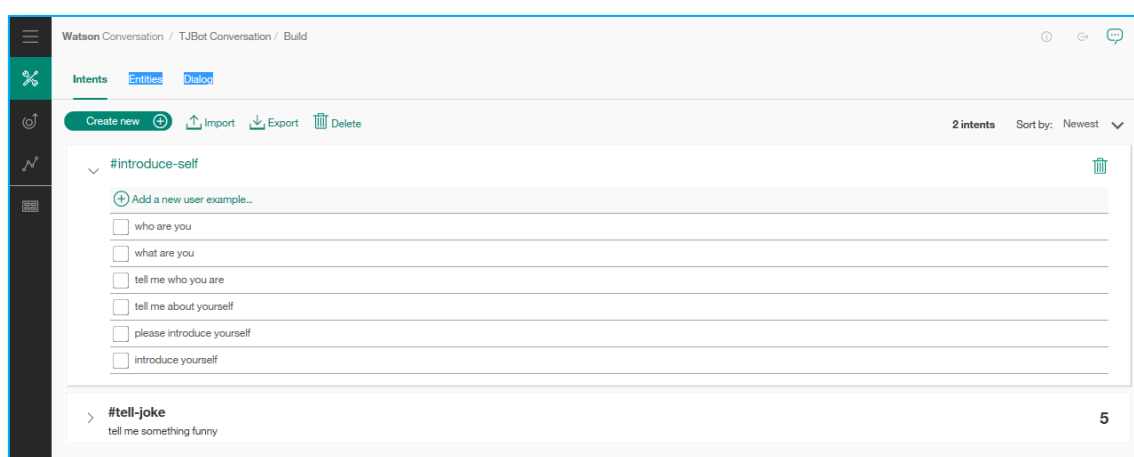
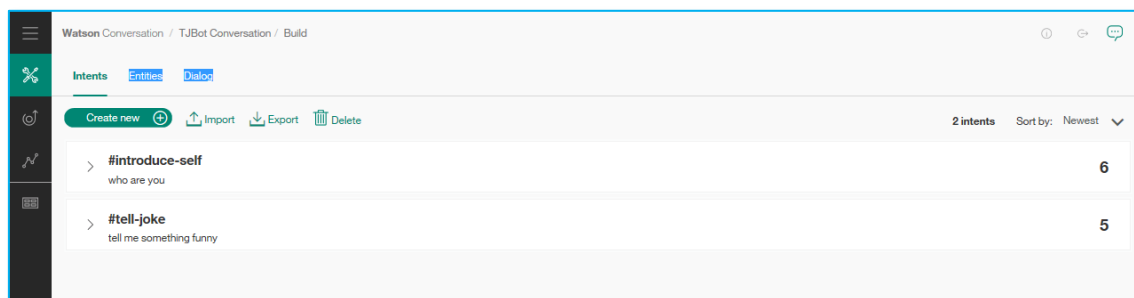
Workspace ID をメモ帳などにコピー。後程使用します。



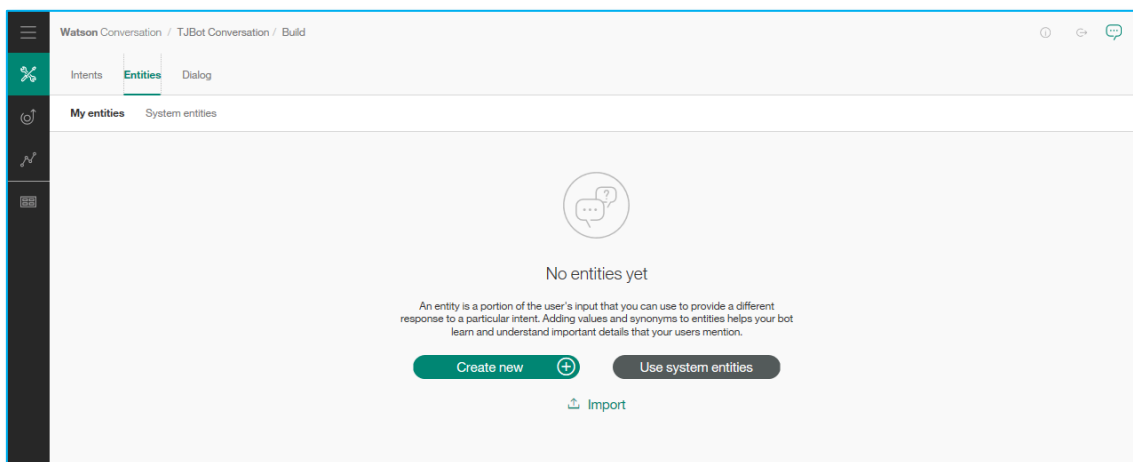
Workspace ID: e[redacted]

尚、サンプルの“Intents”、“Entities”、“Dialog”は下記の通りです。

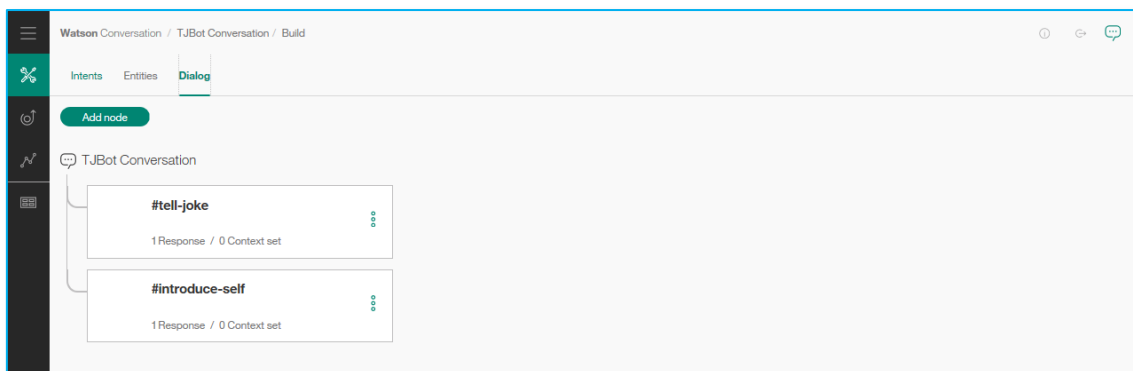
Intents

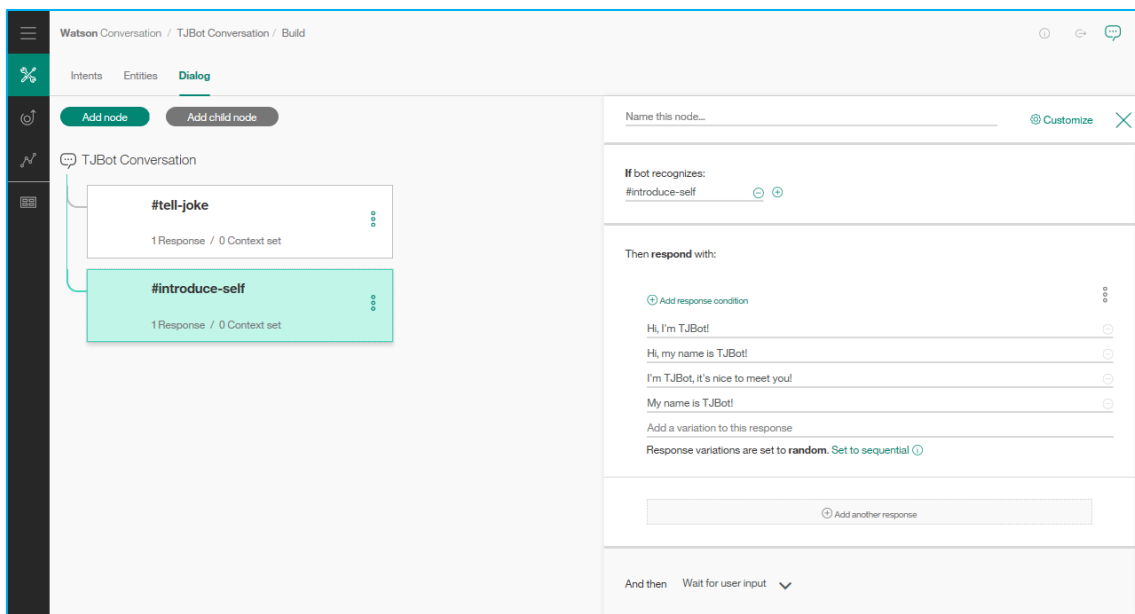
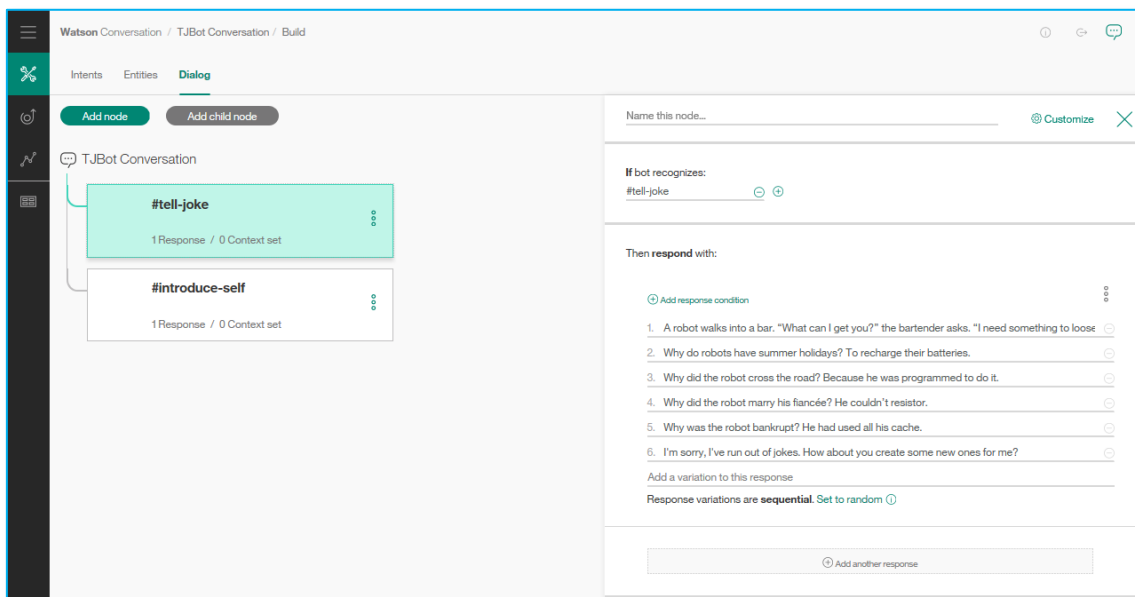


Entities



Dialog





【8】「Speech to Text」、「Conversation」、「Text to Speech」、

WorkspaceID 構成情報の設定

ディレクトリを `~/tjbot/recipes/conversation` に移動します。

```
$ cd ~/tjbot/recipes/conversation
```

インストールされたデフォルト構成ファイル「`config.default.js`」を「`config.js`」という名でコピーし、「`config.js`」を編集します。

```
$ cp config.default.js config.js
```

```
$ sudo nano config.js
```

■“【4】Watson「Speech to Text」、「Conversation」、「Text to Speech」サービスの作成”で取得した、「`username`」「`password`」

■“【5】会話フローの作成”で取得した、「`Conversation`」の `Workspace ID` を `exports.conversationWorkspaceId = ""` に登録します。

ここでは nano エディターを使用しました。

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

// Watson Conversation
// https://www.ibm.com/watson/developercloud/conversation.html
exports.credentials.conversation = {
  password: '',
  username: ''
};

// Watson Speech to Text
// https://www.ibm.com/watson/developercloud/speech-to-text.html
exports.credentials.speech_to_text = {
  password: '',
  username: ''
};

// Watson Text to Speech
// https://www.ibm.com/watson/developercloud/text-to-speech.html

^G Get Help   ^C WriteOut  ^R Read File ^Y Prev Page ^K Cut Text   ^C Cur Pos
^X Exit       ^J Justify   ^W Where Is ^V Next Page ^U UnCut Text ^T To Spell
```

各項目を登録し、 `Ctrl+x ⇒ y ⇒ ↵` で保存します。

【9】Bluetooth スピーカの Pulse Audio が使えるようにの設定変更

初期状態では、

~/tjbot/recipes/conversation/node_modules/tjbot/lib/tjbot.js

の tjbot.js は、USB スピーカの card 1, device 0 を使う定義になっています。

Tjbot.js の **160 行目あたり**の下記の

```
speakerDeviceId: "plughw:1,0" // plugged-in USB card 1, device 0; see aplay -l for  
a list of playback devices
```

```
    inactivityTimeout: -1, // -1 to never timeout or break the connection. Set this to a value in seconds e.g 120 to end connection after 120 seconds of  
    language: 'en-US' // see TjBot.prototype.languages.listen  
  },  
  wave: {  
    servoPin: 7 // corresponds to BCM 7 / physical PIN 28  
  },  
  speak: {  
    language: 'en-US', // see TjBot.prototype.languages.speak  
    voice: undefined, // use a specific voice; if undefined, a voice is chosen based on robot.gender and speak.language  
    speakerDeviceId: "plughw:1,0" // plugged-in USB card 1, device 0; see aplay -l for a list of playback devices  
  },  
  see: {  
    confidenceThreshold: {  
      object: 0.5,  
      text: 0.1  
    },  
  },  
  camera: {  
    height: 720,  
    width: 960,  
    vflip: false, // flips the image vertically, may need to set to 'true' if the camera is installed upside-down  
    hflip: false // flips the image horizontally, should not need to be overridden  
  }  
}  
};  
  
// List of all available configuration parameters  
TjBot.prototype.configurationParameters = Object.keys(TjBot.prototype.defaultConfiguration);  
  
// List of all available languages  
TjBot.prototype.languages = {};  
TjBot.prototype.languages.listen = ['ar-AR', 'en-UK', 'en-US', 'es-ES', 'fr-FR', 'ja-JP', 'pt-BR', 'zh-CN'];  
TjBot.prototype.languages.speak = ['en-GB', 'en-US', 'es-US', 'ja-JP', 'pt-BR'];  
TjBot.prototype.genders = ['male', 'female'];  
  
Get Help      WriteOut  
Exit          Justify  
line 160/1390 (11%), col 9/118 (7%), char 5604/45828 (12%)  
Read File    Prev Page    Cut Text  
Where Is     Next Page    UnCut Text  
Cur Pos  
To Spell
```

speakerDeviceId: "plughw:1,0"を speakerDeviceId: **"pulse"**に変更して、Bluetooth スピーカが使えるようにします。

```
speakerDeviceId: "pulse" // for bluetooth speaker Pulse Audio
```



```
    inactivityTimeout: -1, // -1 to never timeout or break the connection. Set this to a value in seconds e.g 120 to end connection after 120 seconds of
    language: 'en-US' // see TJBot.prototype.languages.listen
  },
  wave: {
    servoPin: 7 // corresponds to BCM 7 / physical PIN 26
  },
  speak: {
    language: 'en-US', // see TJBot.prototype.languages.speak
    voice: undefined, // use a specific voice; if undefined, a voice is chosen based on robot.gender and speak.language
    speakerDeviceId: "pulse" // for bluetooth speaker Pulse Audio
  },
  see: {
    confidenceThreshold: {
      object: 0.5,
      text: 0.1
    },
    camera: {
      height: 720,
      width: 960,
      vflip: false, // flips the image vertically, may need to set to 'true' if the camera is installed upside-down
      hflip: false // flips the image horizontally, should not need to be overridden
    }
  }
};

// List of all available configuration parameters
TJBot.prototype.configurationParameters = Object.keys(TJBot.prototype.defaultConfiguration);

// List of all available languages
TJBot.prototype.languages = {};
TJBot.prototype.languages.listen = ['ar-AR', 'en-UK', 'en-US', 'es-ES', 'fr-FR', 'ja-JP', 'pt-BR', 'zh-CN'];
TJBot.prototype.languages.speak = ['en-GB', 'en-US', 'es-US', 'ja-JP', 'pt-BR'];
TJBot.prototype.genders = ['male', 'female'];
```

Get Help WriteOut Read File Prev Page Cut Text Cur Pos
Exit Justify Where Is Next Page UnCut Text To Spell

【10】プログラムの実行

```
$ cd ~/tjbot/recipes/conversation/
```

```
$ sudo node conversation.js
```

```
pi@raspberrypi:~/tjbot/recipes/conversation $ sudo node conversation.js
verbose: TJBot initializing microphone
verbose: TJBot initializing conversation service
verbose: TJBot initializing speech_to_text service
verbose: TJBot initializing text_to_speech service
info: Hello from TJBot! My name is Watson.
verbose: TJBot library version v1.3.0
You can ask me to introduce myself or tell you a joke.
Try saying, "Watson, please introduce yourself" or "Watson, who are you?"
You can also say, "Watson, tell me a joke!"
verbose: TJBot initializing microphone
```

「Watson, please introduce yourself」

「Watson, who are you?」

「Watson, tell me a joke!」

と話しかけると答えてくれます。

発音が悪く、なかなか認識してくれませんが、Google 翻訳の音声を使うとよく反応しました。

【11】日本語対応に変更

追加・変更するのは3点です。

(1) 追加点1 :

Watson「Conversation」の会話フローを日本語で作成します。

日本語は、「【7】会話フローの作成」で作成した「workspace-sample.json」(TJBot Conversation) を翻訳しました。

Bluemix にログイン

<https://console.bluemix.net/>



Log into IBM Bluemix

IBMID またはメール・アドレスを入力してください IBM IDをお忘れですか?

Email or IBMID

次へ

New to Bluemix? [Sign Up](#)



Log into IBM Bluemix

IBM ID:

パスワード パスワードをお忘れの場合

Password

Log in

別の IBM アカウントまたは E メールを使用してください

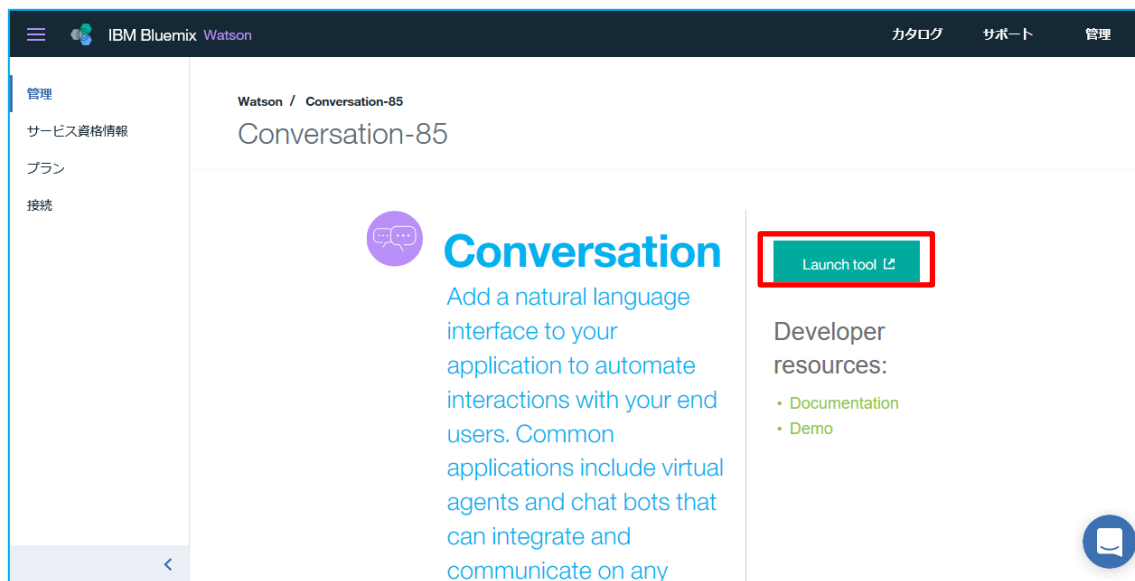
「ハンバーガーメニュー」をクリック



「ダッシュボード」をクリック

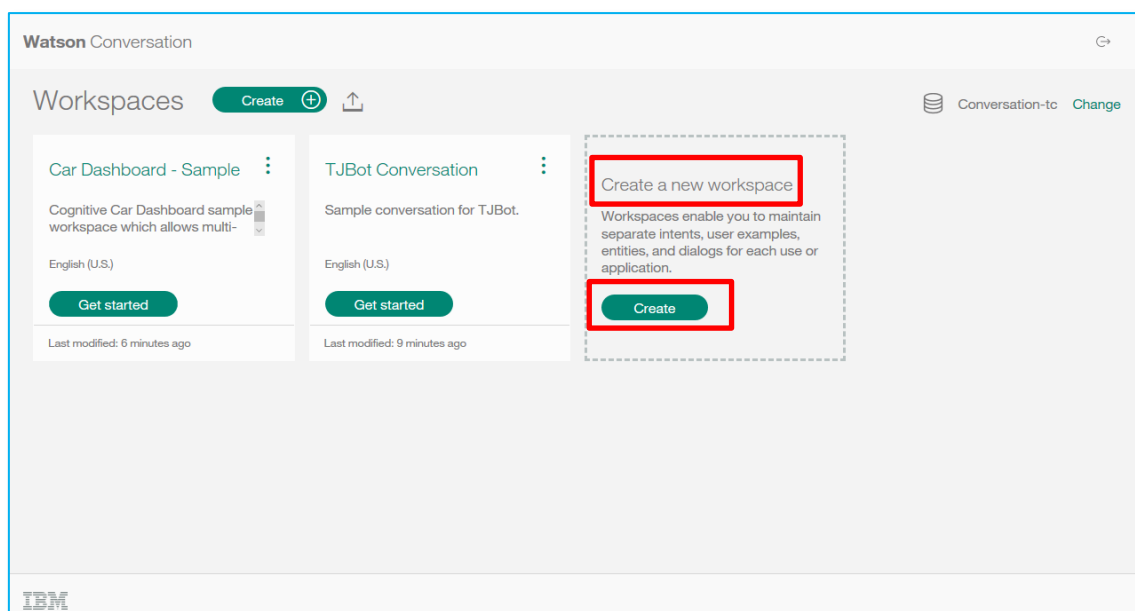


「Launch tool」をクリック



日本語対応用に新しい Workspace を作成します。

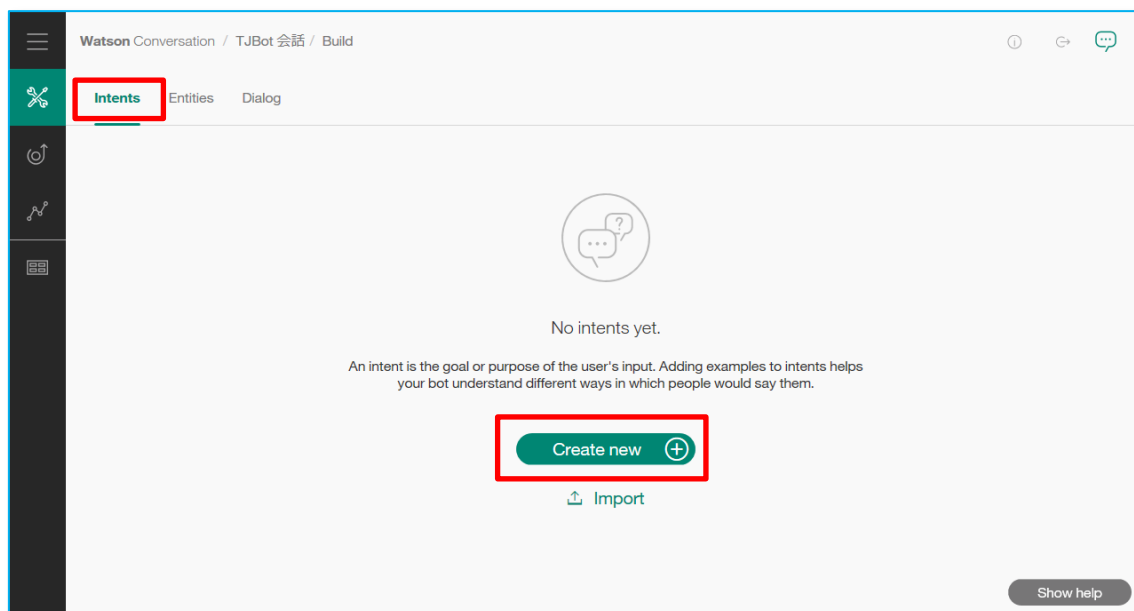
“Create a new workspace”の「Create」をボタンをクリック。



“Name”、“Description”、“Language:Japanese”を入力後「Create」ボタンを押す。
ここでは、“Name”:TJBot 会話、Description:TJBot の会話例 としました。

“Intents”の登録

「Intents」タブを選択後、「Create new」ボタンをクリック。

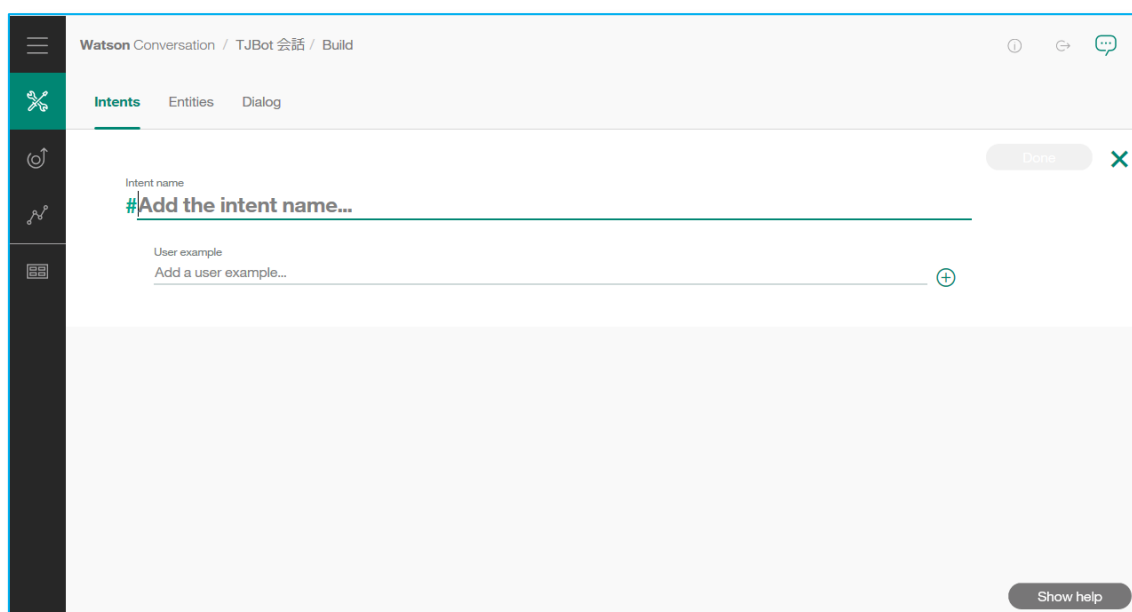


“Intent name”、“User Example”を登録します。

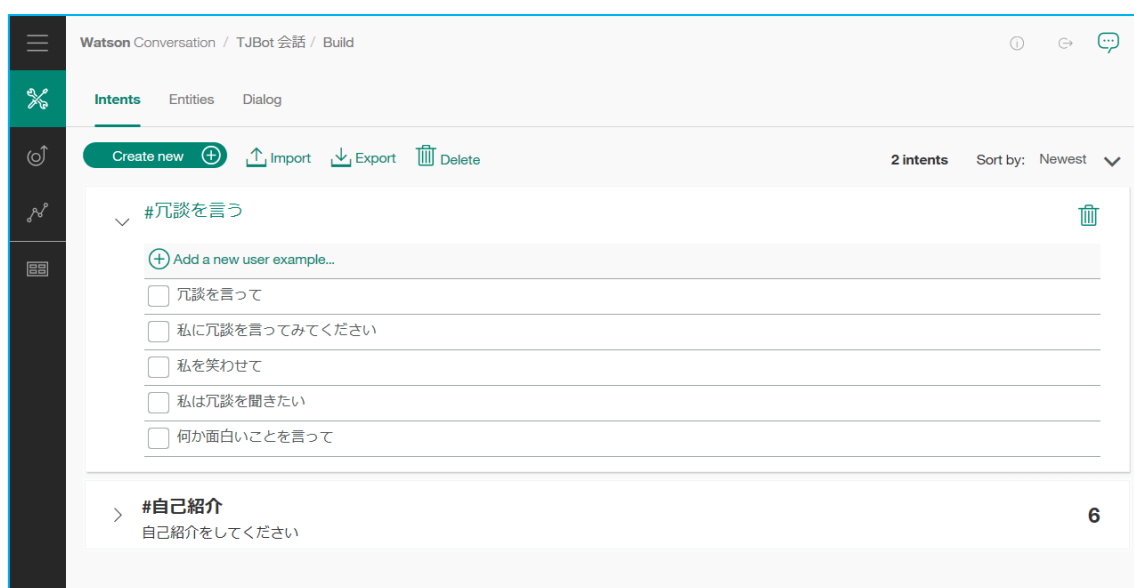
登録する Intents は2つで、下記の通りです。

尚、User Example を追加するには、「+ボタン」、Intent を追加するのは「Create new」ボタンを押します。

<u>Intent Name</u>	<u>User Example</u>
冗談を言う	冗談を言って 私に冗談を言ってみてください 私を笑わせて 私は冗談を聞きたい 何か面白いことを言って
自己紹介	自己紹介をしてください あなたは誰ですか あなたは何者ですか あなたが誰か教えてください あなたについて教えてください 自己紹介をお願いします



すべて登録したら下図のような構成になります。





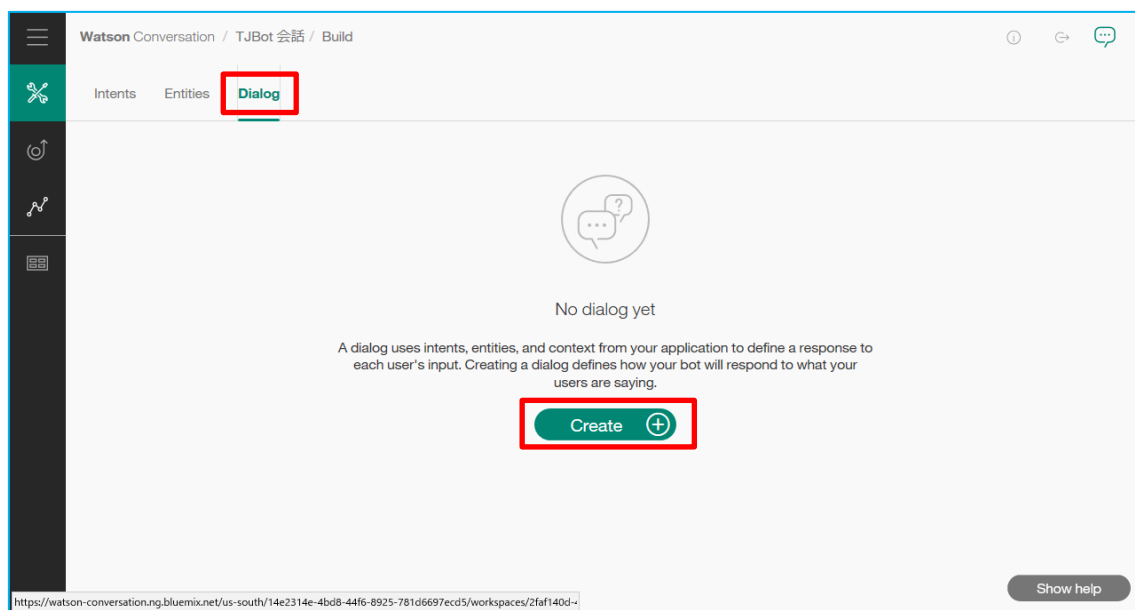
次に Entities は登録しないので、そのまま



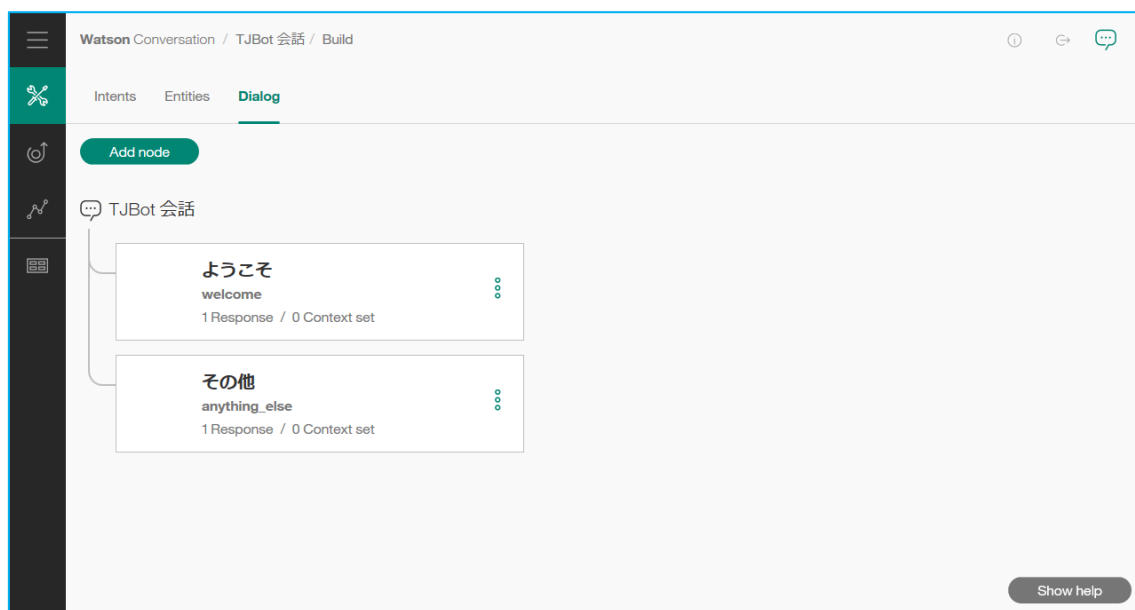
次に Dialog を登録します。

Dialog の登録

「Dialog」タブ選択後、「Create」ボタンを押す。

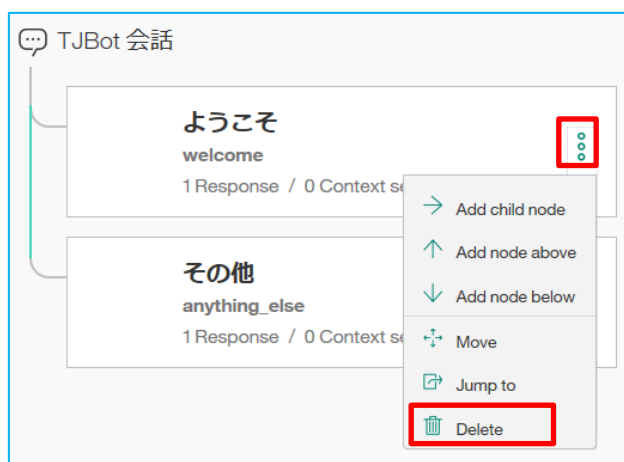


初期状態では、“ようこそ”、“その他”が登録されています。

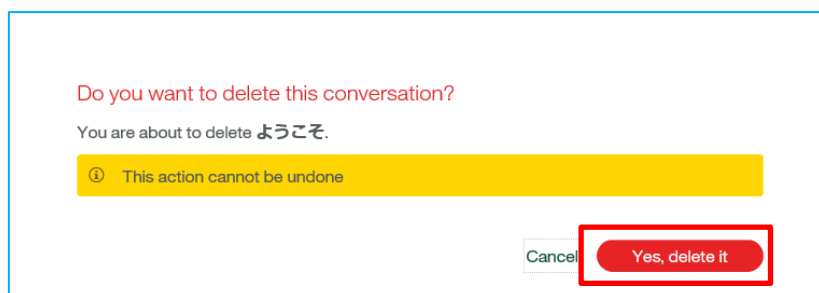


ここでは「ようこそ」node は使わないため削除しました。
削除方法は下記の通りです。

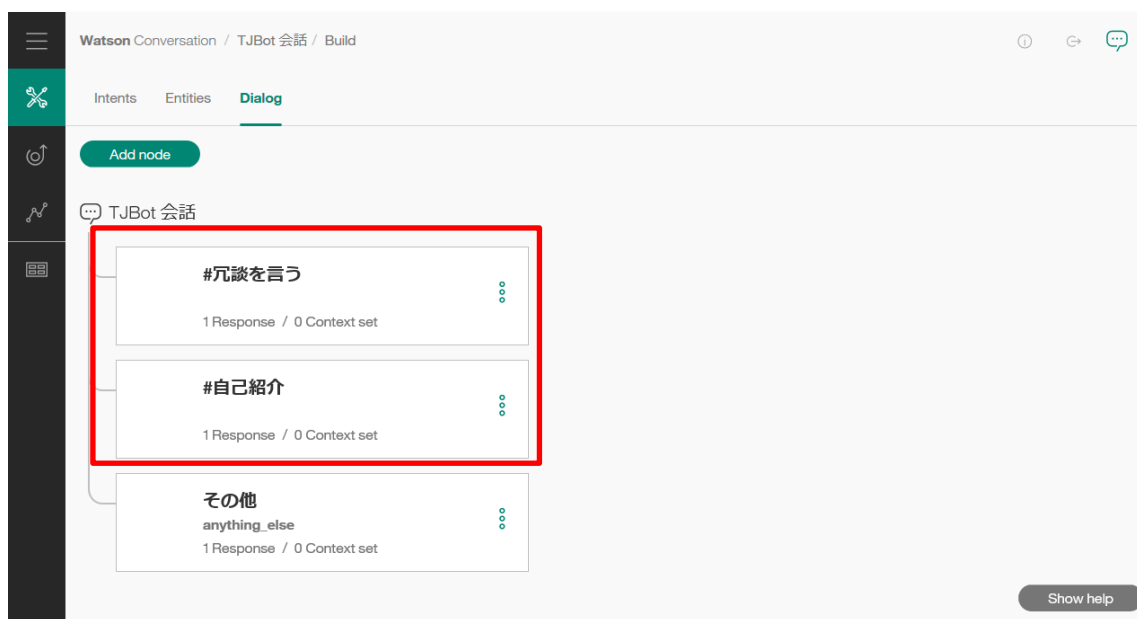
- マークを押し、「delete」で削除。



「Yes, delete it」ボタンを押す。



2つの node「冗談を言う」、「自己紹介」を登録します。



登録する node は以下の通りです。

If bot recognizes: Then respond with:

#自己紹介

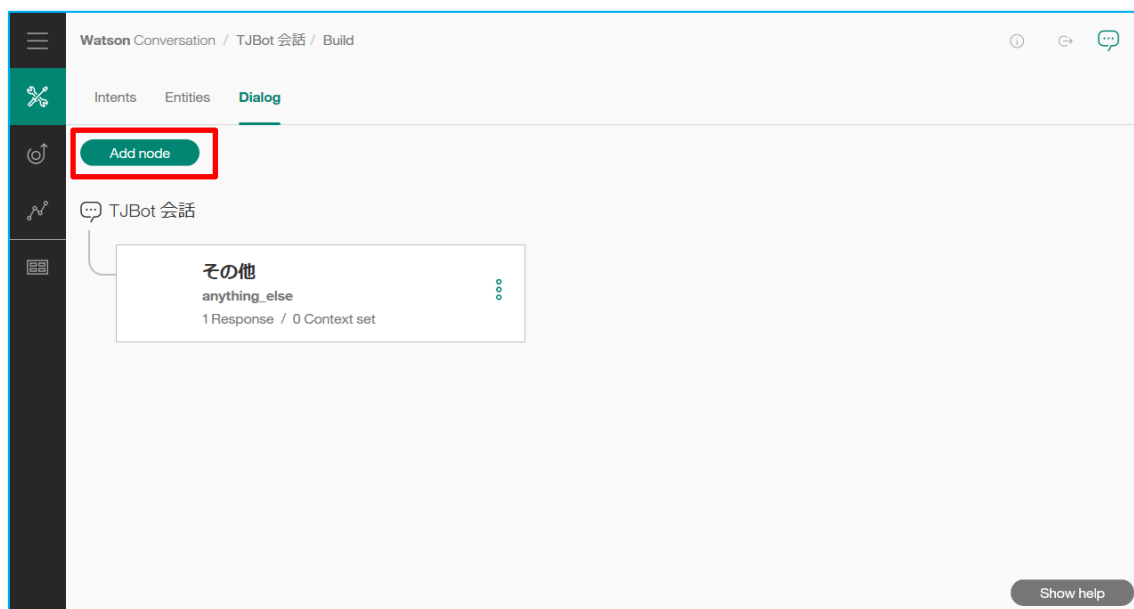
こんにちは、私は TJBot です！
こんにちは、私の名前は TJBot です！
私は TJBot です、お会いできてうれしいです！
私の名前は TJBot です！

#冗談を言う

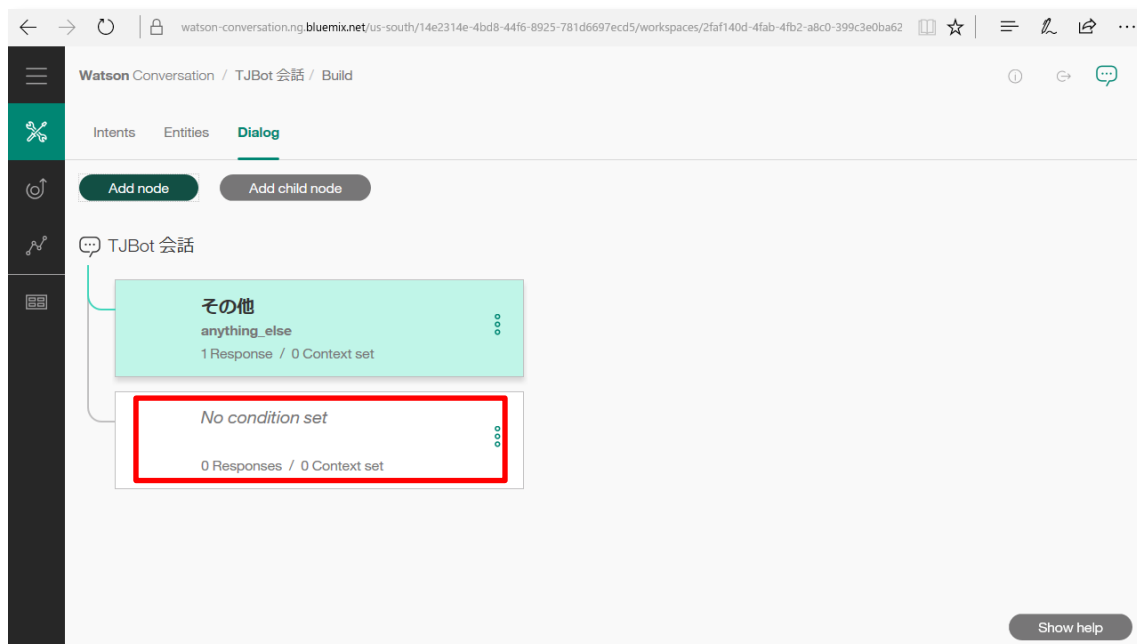
ロボットがバーに入る。「何ができるの？」バーテンダーが尋ねる。ロボットが返答する。バーテンダーは彼にスクロドライバーを務めます。
なぜロボットは夏休みをとるのか知ってますか？ 電池を充電するためです。
なぜロボットは道路を横切ったのですか？ 彼はそれをするようにプログラムされていたのです。
なぜロボットは彼の婚約者と結婚したのですか？ 彼は抵抗できませんでした。
なぜロボットが破産したのですか？ 彼はすべてのキャッシュを使いました。
申し訳ありません、私はジョークを使い果たしました。あなたは私のために新しいものを作りませんか？

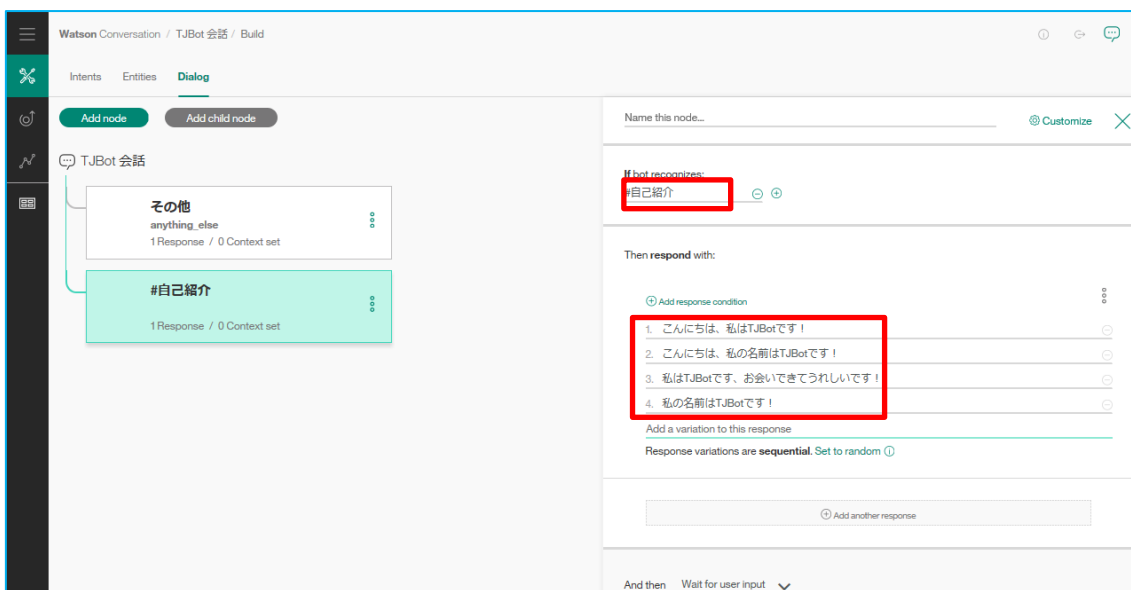
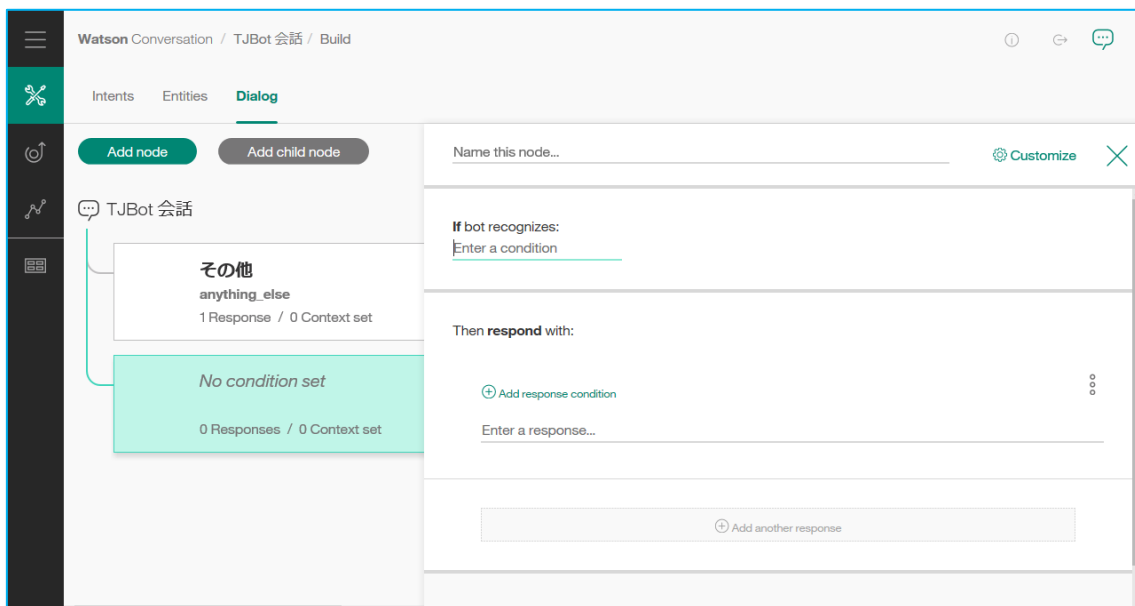
それでは、登録します。

「Add node」ボタンを押す。



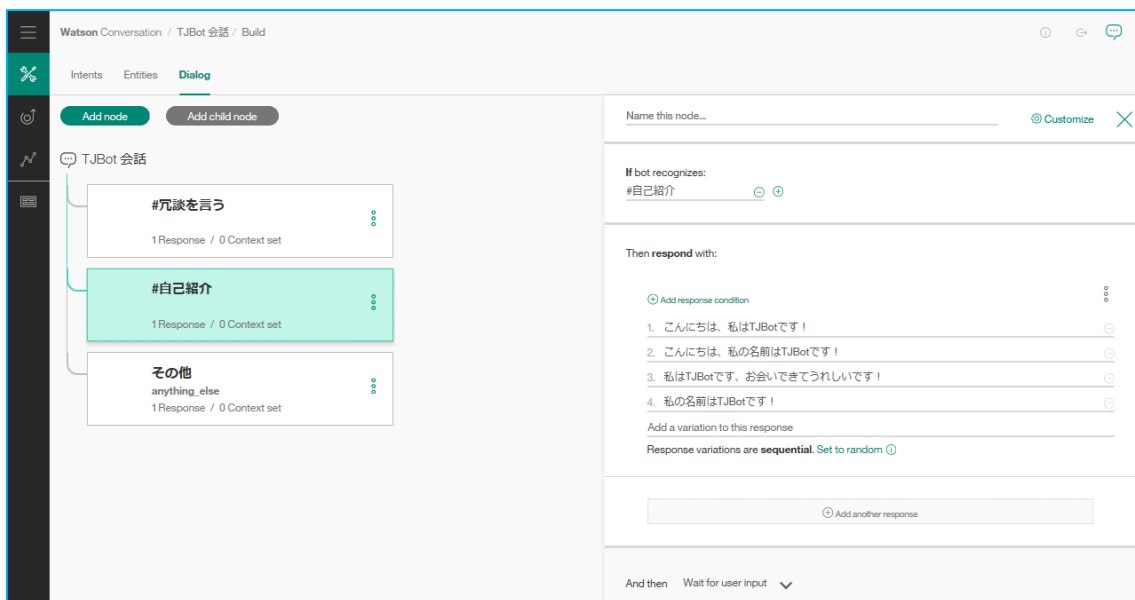
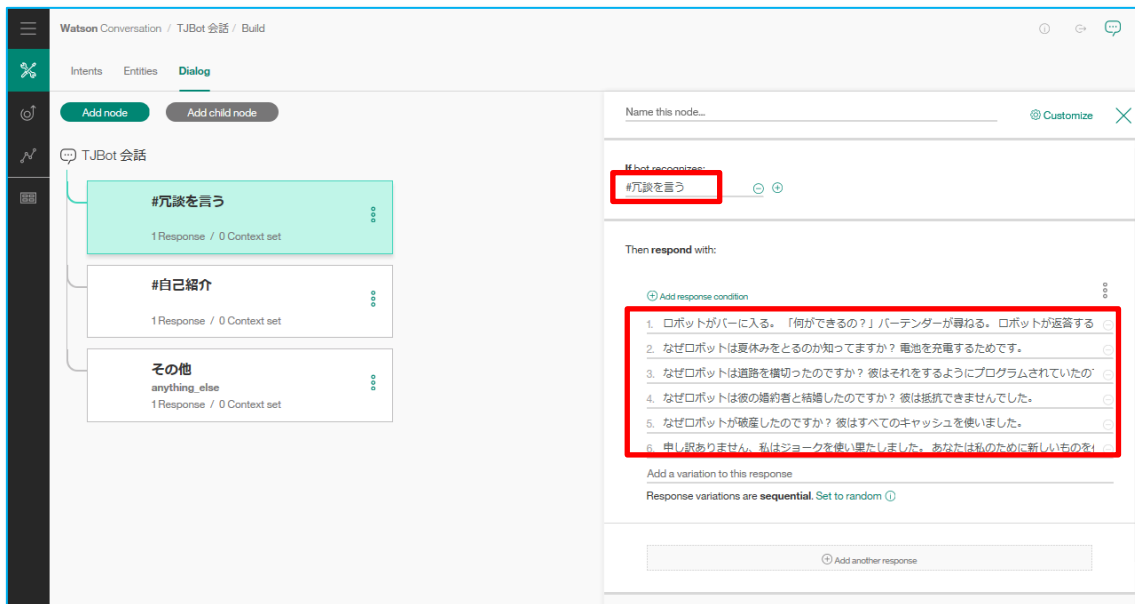
新しくできた「No condition set」の枠内をクリック





「冗談を言う」も「Add node」ボタンをクリックで登録します。

下図の通り、登録が完了しました。



(2) 変更点 1 :

Tjbot.js の「Speech to Text」、「Text to Speech」、言語を英語から日本語に変更する。

~/tjbot/recipes/conversation/node_modules/tjbot/lib/tjbot.js

tjbot.js の

152 行あたり

language: 'en-US' // see TJBot.prototype.languages.listen

language: 'ja-JP' // see TJBot.prototype.languages.listen

158 行あたり

language: 'en-US', // see TJBot.prototype.languages.speak

language: 'ja-JP', // see TJBot.prototype.languages.speak

```
},
listen: [
  microphoneDeviceId: "plughw:1.0", // plugged-in USB card 1, device 0; see arecord -l for a list of recording devices
  inactivityTimeout: -1, // -1 to never timeout or break the connection. Set this to a value in seconds e.g 120 to end connection after 120 seconds of
  language: 'ja-JP' // see TJBot.prototype.languages.listen
],
wave: [
  servoPin: 7 // corresponds to BCM 7 / physical PIN 26
],
speak: [
  language: 'ja-JP', // see TJBot.prototype.languages.speak
  voice: undefined, // use a specific voice; if undefined, a voice is chosen based on robot.sender and speak.language
  speakerDeviceId: "pulse" // for bluetooth speaker Pulse Audio
],
see: [
  confidenceThreshold: {
    object: 0.5,
    text: 0.1
  },
  camera: {
    height: 720,
    width: 960,
    vflip: false, // flips the image vertically, may need to set to 'true' if the camera is installed upside-down
    hflip: false // flips the image horizontally, should not need to be overridden
  }
]
];

// List of all available configuration parameters
TJBot.prototype.configurationParameters = Object.keys(TJBot.prototype.defaultConfiguration);

// List of all available languages
TJBot.prototype.languages = [];
TJBot.prototype.languages.listen = ['ar-AR', 'en-UK', 'en-US', 'es-ES', 'fr-FR', 'ja-JP', 'pt-BR', 'zh-CN'];
```

(3) 変更点 2 :

Tjbot 名称「Watson」を全角日本語に変更する。

~/tjbot/recipes/conversation/node_modules/tjbot/lib/tjbot.js

tjbot.js の

147 行あたり

name: 'Watson'

name: 'Watson'

必ず全角日本語にしないと呼びかけても認識しませんので、ご注意ください。


```
/**
 * List of TJBot hardware and services.
 */
TJBot.prototype.capabilities = ['analyze_tone', 'converse', 'listen', 'see', 'shine', 'speak', 'translate', 'wave'];
TJBot.prototype.hardware = ['camera', 'led', 'microphone', 'servo', 'speaker'];
TJBot.prototype.services = ['conversation', 'language_translator', 'speech_to_text', 'text_to_speech', 'tone_analyzer', 'visual_recognition'];

/**
 * Default configuration parameters.
 */
TJBot.prototype.defaultConfiguration = {
  log: {
    level: 'info' // valid levels are 'error', 'warn', 'info', 'verbose', 'debug', 'silly'
  },
  robot: {
    gender: 'male', // see TJBot.prototype.genders
    name: 'Watson'
  },
  listen: {
    microphoneDeviceId: "plughw:1,0", // plugged-in USB card 1, device 0; see arecord -l for a list of recording devices
    inactivityTimeout: -1, // -1 to never timeout or break the connection. Set this to a value in seconds e.g 120 to end connection after 120 seconds of
    language: 'ja-JP' // see TJBot.prototype.languages.listen
  },
  wave: {
    servoPin: 7 // corresponds to BCM 7 / physical PIN 26
  },
  speak: {
    language: 'ja-JP', // see TJBot.prototype.languages.speak
    voice: undefined, // use a specific voice; if undefined, a voice is chosen based on robot.gender and speak.language
    speakerDeviceId: "pulse" // for bluetooth speaker Pulse Audio
  },
  see: {
    confidenceThreshold: {
```

(4) 変更点3 :

デフォルトボイスの現「en-US_MichaelVoice」を日本の「ja-JP_EmiVoice」に変更する。

[~/tjbot/recipes/conversation/node_modules/tjbot/lib/tjbot.js](#)

tjbot.js の

1089 行あたり

```
var voice = "en-US_MichaelVoice";
```

```
var voice = "ja-JP_EmiVoice";
```

```
/** ----- */
/**
 * Speak the given message.
 *
 * @param {String} message The message to speak.
 */
TJBot.prototype.speak = function(message) {
  this._assertCapability('speak');

  // make sure we're trying to say something
  if (message == undefined || message == "") {
    winston.error("TJBot tried to speak an empty message.");
    return; // exit if theres nothing to say!
  }

  // default voice
  var voice = "ja-JP_EmiVoice";
}
```

Get Help WriteOut Read File Prev Page Cut Text Cur Pos
Exit Justify Where Is Next Page UnCut Text To Spell

(5) 変更点 4 :

“(1) 追加点 1 :”の Watson「Conversation」会話フロー日本語対応で作成した、[「Workspace Id」](#)に変更します。

~/tjbot/recipes/conversation/config.js

config.js の

9 行あたり

`exports.conversationWorkspaceId = ""`を変更登録する。

```

/*
User-specific configuration
** IMPORTANT NOTE ****
* Please ensure you do not interchange your username and password.
* Hint: Your username is the lengthy value ~ 36 digits including a hyphen
* Hint: Your password is the smaller value ~ 12 characters
*/

exports.conversationWorkspaceId = 'e2503b81-2a7f-47ab-97c5-4950e6e67a25'; // re$

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

// Watson Conversation
// https://www.ibm.com/watson/developercloud/conversation.html
exports.credentials.conversation = {
  password: 'GDvVrfdid3Vn',
  username: '13ab2dc4-e856-4992-a39d-a8710c2c5ded'
};

```

^G Get Help ^O WriteOut ^R Read File ^Y Prev Page ^K Cut Text ^C Cur Pos
 ^X Exit ^J Justify ^W Where Is ^V Next Page ^U UnCut Text ^T To Spell

【12】日本語対応版プログラムの実行

```
$ cd ~/tjbot/recipes/conversation
```

```
$ sudo node conversation.js
```

```
pi@raspberrypi:~/tjbot/recipes/conversation $ sudo node conversation.js
verbose: TJBot initializing microphone
verbose: TJBot initializing conversation service
verbose: TJBot initializing speech_to_text service
verbose: TJBot initializing text_to_speech service
info: Hello from TJBot! My name is W a t s o n .
verbose: TJBot library version v1.3.0
You can ask me to introduce myself or tell you a joke.
Try saying, "W a t s o n , please introduce yourself" or "W a t s o n , who are
you?"
You can also say, "W a t s o n , tell me a joke!"
verbose: TJBot initializing microphone
```

「W a t s o n」キーワードに続けて下記文章を言ってみてください。

Tjbot が答えます。

「自己紹介をしてください」

「あなたは誰ですか」

「あなたは何者ですか」

「あなたが誰か教えてください」

「あなたについて教えてください」

「自己紹介をお願いします」

「冗談を言って」

「私に冗談を言ってみてください」

「私を笑わせて」

「私は冗談を聞きたい」

「何か面白いことを言って」