



IBM Watson

Integrate voxibot with chatbot:

In next steps we illustrate how to integrate with Watson engine which is the same way for other engines, just changing some credentials to validate whatever your engine is.

Our chatbot here will be based on Watson

First of all, create an account for Watson engine to get some credential that we need to set and connect between voxibot and chatbot engine, click here to create your account

<https://watson-conversation.ng.bluemix.net/login> ,

When you succeed to create an account next step is to create a workspace with your account and next create the first chatbot to test your system

1. After you log in, click on proceed to start.

About your IBMid Account Privacy

This notice provides information about accessing your IBMid user account (Account). If you have previously been presented with a version of this notice, please refer to "Changes since the previous version of this notice" below for information about the new updates.

+ Changes since the previous version of this notice

+ What data does IBM collect?

+ Why IBM needs your data

+ How your data was obtained

+ How IBM uses your data

+ How IBM protects your data

+ How long we keep your data

Your rights

Our Privacy Statement provides more information about your personal data rights. It also provides contact information if you have questions or concerns regarding our handling of your personal data.

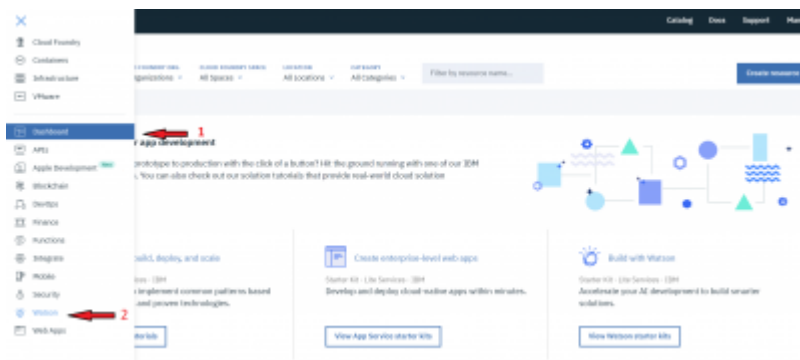
Acknowledgement

I acknowledge that I understand how IBM is using my Basic Personal Data and I am at least 16 years of age.

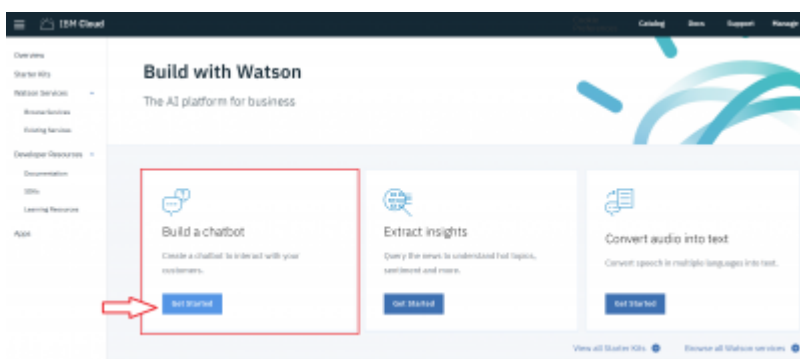
1

2. We need to find a proper tool to build our chatbot, go to Dashboard on left the choose Watson in

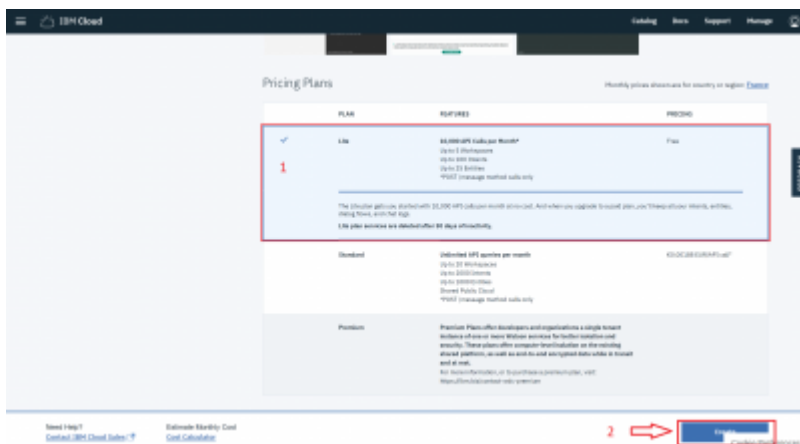
the bottom.



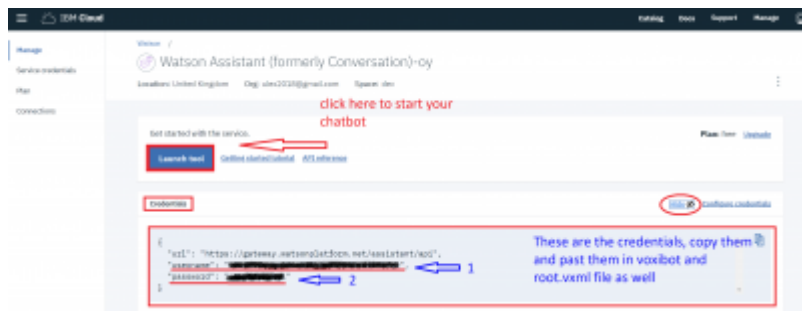
3. Here as you see click **Get Start** to launch your own chatbot.



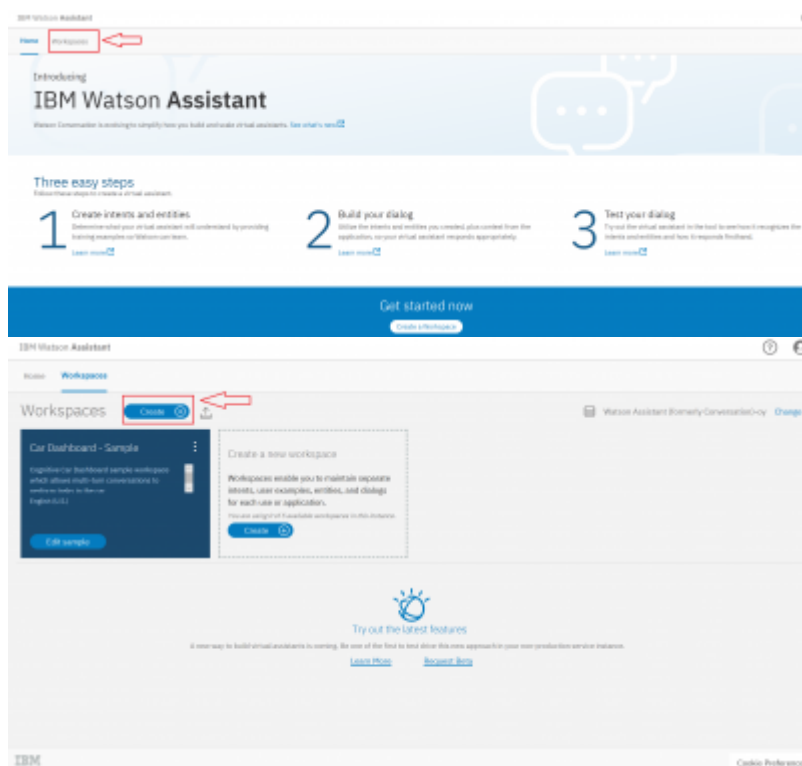
4. We will use Lite option in our example it is limited and it has 30 days to keep your work up. On right bottom click on Create.



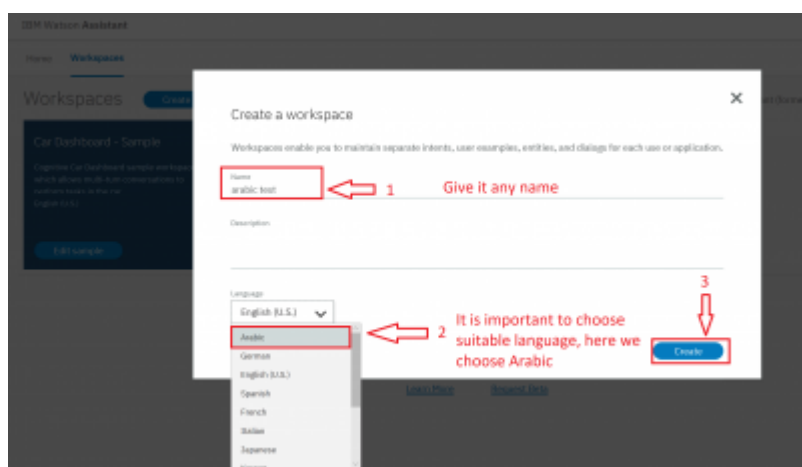
5. Before you launch the platform, have a look to your credential.copy your credential Username/password and paste them on proper places in Root.vxml file (you can see at the end of this Doc). After that press on Launch test.



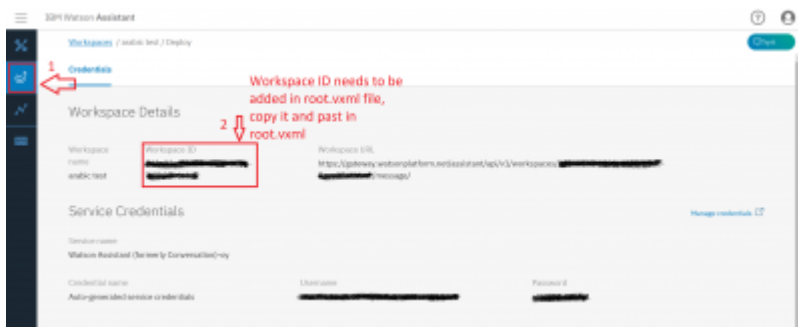
6. Create a Workspace to communicate with during your test, by this workspace you will get the response, and you will understand how voximal assistant helpful and easy to use.



7. Put the language of your chatbot, it is up to you. **A remark**, choose the right language which you want to interact with.



8. On the left click on Deploy and as you can see, we need Workspace ID to add it in voxibot file "root.vxml". this ID to push us to specific chatbot that we choose.



9. The way to set the rules in chatbot is same for any language, so you can follow Watson documentation on this website <https://www.ibm.com/blogs/nordic-msp/get-started-building-chatbots-using-watson-bluemix/>

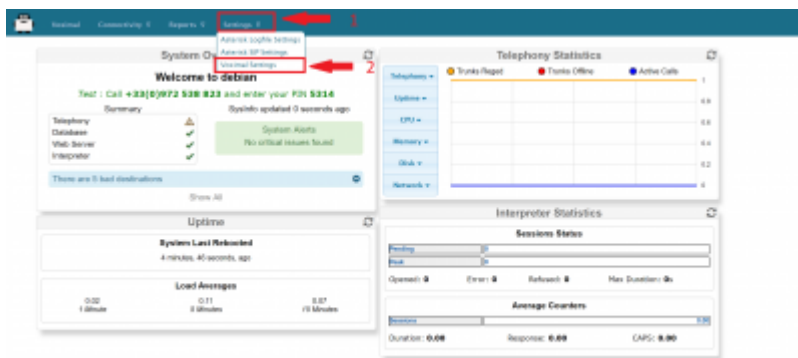
Now form voxibot side:

In this side we need to adapt our watson credential in easy way inside Voxibot, then by finishing these next steps we will be able to talk to our chatbot thanks to voximal voice assistant. **Note**, We deal with Arabic language as we will see.

Let's start step by step

Note : Each step you edit or add something new you should do save/submit the apply config

1. To start configuring our voxibot firstly, go to **settings** then **voximal settings** to adjust our settings.



2. As shown below choose your API, in our case **HTTP or MRCP** which easy and support Arabic. Just copy and paste the link in **URI** <http://tts6.i6net.org/tts/acapela/tts.php>. **Pay attention** to format (alaw), save and apply config after finishing.

Voximal Settings

General Synthesis Recognition License Call Test

Synthesis

API HTTP or MRCP

URL http://ml.dnet.org/voiceapi/its.php

Method POST GET ASYNCHRONOUS

Format wav mp3 pcm AAC aac raw s16 s16le

ID

Key

User

Password

Voice

SSML Yes No

Cache aging 1

Call prompt Yes No

Save Discard Cache clear

3. We need API recognition engine to print out and translate our input as a text to be understood by the chatbot. You need to have a credential key for Google or your favorite provider. There are other engines can provide recognition features that depend on the quality and the language support.

Voximal Settings

General Synthesis Recognition License Call Test

Recognition

API Google Speech API

Key AlzaSyCbgSyNqJl3uEE

Save Discard

4. Create an account to assign it to a Number, this account will connect to the database to communicate with the chatbot, **URL** is the source where our essentials file located. set the speech to **Automatic** to avoid any issues,

Voximal Application

Accounts Editor

Edit Application

The fields marked with * can not be left in blank.

Name* demo_server

URL* https://source file /vxml/chatbot Select ...

Max Sessions

Dial Format

Mark

Speech Emulation No Yes Automatic

Speech Provider

Max time (s)

Session parameter

Start delay (ms) 500

Save Changes Delete

5. Set a call number to our account that just created, give it any number and name, the important

Last update: 2018/05/31 14:03 developer_guide:connect_your_chatbot:watson:start https://wiki.voximal.com/doku.php?id=developer_guide:connect_your_chatbot:watson:start

part here is to set **destination**. it will be **voximal application** and link it to the account name that should appear on the list because we just made it ****(demo server)****

The screenshot shows the MikroTik WinBox interface with the 'Static Route' configuration window open. The 'Route' tab is selected, and the 'Static Route' configuration form is displayed. The 'Description' field is set to 'demo chatbot', the 'CID Number' is '3800', and the 'Static Application' is set to 'demo_server'. The 'Apply Config' button is highlighted in the top right corner.

6. There are two files need to adapt to our case **root.vxml** and **conversation_watson.vxml** For the root file set the language that you want and supported by the provider (engine). also Because we use Watson so our credential will be Watson credential, we explain how to get these credential from Watson in the top of this page. in addition, By setting your email address you can have the results of your calls and your communication logs with the chatbot.

```

1  #!/usr/bin/perl
2
3  my $url="http://10.10.10.10:8080/";
4
5  my $property name="timeout" value="10s"/>
6
7  <property name="connectTimeout" value="10s"/>
8
9  <property name="connectTimeout" value="5s"/>
10
11 <property name="keepAliveTimeout" value="10s"/>
12
13
14
15
16 <!-- GML configuration -->
17
18 <!-- name="url" value="https://session.sessionize.local:8080/" -->
19
20 <!-- name="url" value="https://session.sessionize.local:8080/" -->
21
22 <!-- name="id" value="telephone.id"/>
23
24 <!-- name="param" value="telephone.param"/>
25
26 <!-- name="language" value="fr"/>
27
28 <!-- name="username" value="Boupa/Faria"/>
29
30 <!-- name="password" value="123456"/>
31
32 <!-- name="url" value="https://session.sessionize.local:8080/" -->
33
34
35 <!-- MAIL configuration -->
36
37 <!-- name="url" value="https://session.sessionize.local:8080/" -->
38
39 <!-- name="url" value="https://session.sessionize.local:8080/" -->
40
41 <!-- name="url" value="https://session.sessionize.local:8080/" -->
42
43 <!-- name="url" value="https://session.sessionize.local:8080/" -->
44
45 <!-- name="url" value="https://session.sessionize.local:8080/" -->
46
47 <!-- name="url" value="https://session.sessionize.local:8080/" -->
48
49 <!-- name="url" value="https://session.sessionize.local:8080/" -->
50
51 <!-- name="url" value="https://session.sessionize.local:8080/" -->
52
53 <!-- name="url" value="https://session.sessionize.local:8080/" -->
54
55 <!-- name="url" value="https://session.sessionize.local:8080/" -->
56
57 <!-- name="url" value="https://session.sessionize.local:8080/" -->
58
59 <!-- name="url" value="https://session.sessionize.local:8080/" -->
60
61 <!-- name="url" value="https://session.sessionize.local:8080/" -->
62
63 <!-- name="url" value="https://session.sessionize.local:8080/" -->
64
65 <!-- name="url" value="https://session.sessionize.local:8080/" -->
66
67 <!-- name="url" value="https://session.sessionize.local:8080/" -->
68
69 <!-- name="url" value="https://session.sessionize.local:8080/" -->
70
71 <!-- name="url" value="https://session.sessionize.local:8080/" -->
72
73 <!-- name="url" value="https://session.sessionize.local:8080/" -->
74
75 <!-- name="url" value="https://session.sessionize.local:8080/" -->
76
77 <!-- name="url" value="https://session.sessionize.local:8080/" -->
78
79 <!-- name="url" value="https://session.sessionize.local:8080/" -->
80
81 <!-- name="url" value="https://session.sessionize.local:8080/" -->
82
83 <!-- name="url" value="https://session.sessionize.local:8080/" -->
84
85 <!-- name="url" value="https://session.sessionize.local:8080/" -->
86
87 <!-- name="url" value="https://session.sessionize.local:8080/" -->
88
89 <!-- name="url" value="https://session.sessionize.local:8080/" -->
90
91 <!-- name="url" value="https://session.sessionize.local:8080/" -->
92
93 <!-- name="url" value="https://session.sessionize.local:8080/" -->
94
95 <!-- name="url" value="https://session.sessionize.local:8080/" -->
96
97 <!-- name="url" value="https://session.sessionize.local:8080/" -->
98
99 <!-- name="url" value="https://session.sessionize.local:8080/" -->
100

```

7. In **conversation_watson** put the language as you configure in root file. This depends on which workspace ID you want to communicate with.

```

1 <!-- session = "1.0" -->
2 <!-- next change I need in this line language en, en-GB, en-UK, en, or regarding what is the TTS provider -->
3 <xml version="2.0" xmlns="http://www.w3.org/2001/xml" application="root.xml" xml:lang="en" />
4 <var name="start" expr="true"/>
5
6 <var name="prompt" expr=""/>
7
8 <var name="url_to_invoke" expr=""/>
9 <var name="url_args" />
10 <var name="url_result" expr=""/>
11
12 <script>
13 var context = { conversation_id: id };
14 var input = { text: '' };
15 </script>
16
17 <var name="answer" expr="VIDE" />
18 <var name="sessionId" expr="session.connection.remote.uri"/>
19
20 <form id="main">
21 <!-- Init for Watson -->
22 <block cond="start">
23 <assign name="input.text" expr="SESSION"/>
24 <log MATHSON : input: <value expr="input"/> context: <value expr="context"/> </log>
25 <data name="api" expr="https://gateway.watsonplatform.net/conversation/api/v1/workspaces/'workspace'" />
26 <log MATHSON : result: <value expr="api"/> </log>
27 <assign name="start" expr="false"/>
28 <assign name="prompt" expr=""/>
29 <foreach item="text" array="api.output.text">
30 <assign name="prompt" expr="prompt+text"/>
31 </foreach>
32 <script>log("Watson (init): " + prompt);</script>
33 </block>
34
35 <!-- Get the user input -->
36 <field cond="true" name="text">
37 <property name="timeout" value="15s"/>
38 <property name="compleatetimeout" value="705ms"/>
39 <property name="interdigittimeout" value="300ms"/>
40 <property name="sensitivity" value="0.5"/>
41 <grammar expr="builtin:grammar/text" +lang"/>
42 <grammar expr="alias" +lang" />
43 <grammar expr="phrase" +lang" />
44 <script cond="lang" expr="fr" xml:lang="fr-FR"><value expr="prompt"/></script>

```

From:

<https://wiki.voximal.com/> - **Voximal documentation**

Permanent link:

https://wiki.voximal.com/doku.php?id=developer_guide:connect_your_chatbot:watson:start

Last update: **2018/05/31 14:03**

