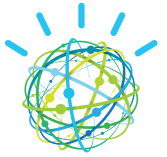


This presentation explains how you
can quickly create a Web UI for
Watson Conversation Service using a
sample posted on GitHub



Creating the Web UI 1/10



We will use a sample node.js application and slightly modify it for our purpose.

1. Download the sample node.js application from github

- Go to: <https://github.com/watson-developer-cloud/conversation-simple>
- Click on button **Clone or Download**, then click on **Download ZIP**

The screenshot shows the GitHub repository page for `watson-developer-cloud / conversation-simple`. The repository has 51 watches, 246 stars, and 751 forks. It is a simple sample application demonstrating the conversation API, with a link to `http://conversation-simple.mybluemix...`. The repository statistics show 199 commits, 3 branches, 0 releases, 21 contributors, and the Apache-2.0 license. The 'Clone or download' button is highlighted, and a dropdown menu is open, showing the 'Download ZIP' option.

Clone with HTTPS ⓘ Use SSH

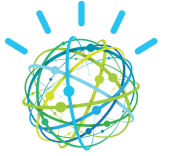
Use Git or checkout with SVN using the web URL.

`https://github.com/watson-developer-cl`

Open in Desktop Download ZIP



Creating the Web UI 2/10



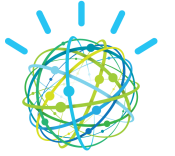
2. Obtain your Watson Conversation service's credentials:
 - a) Go to: <https://console.bluemix.net/>, then login with your IBM Cloud ID
 - b) Under **Cloud Foundry Services**, find your Watson Conversation service
 - c) Click on it, then open **Service credentials** on the left hand side
 - d) Click on **View credentials** under **ACTIONS**.

✓ KEY NAME	DATE CREATED	ACTIONS
✓ Credentials-2	Aug 3, 2017 - 08:21:43	View credentials ▲
<pre>{ "url": "https://gateway.watsonplatform.net/conversation/api", "username": "XX", "password": "XX" }</pre>		

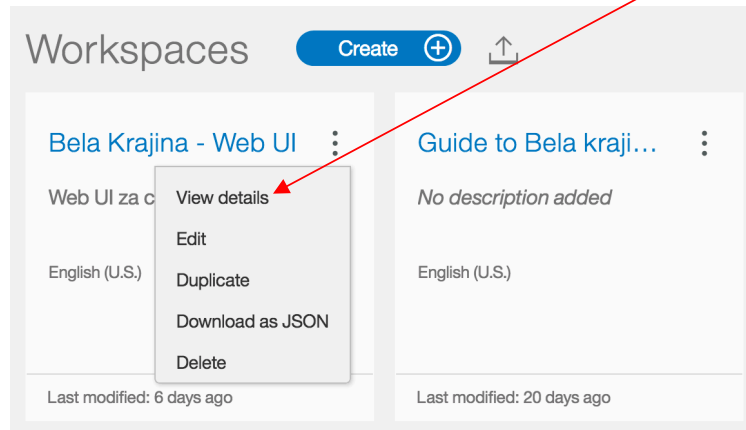
- e) Copy the **username** and **password** and save them in convenient place (i.e. some file). You will need them later.



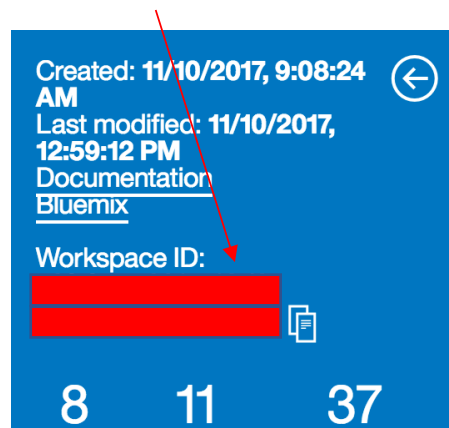
Creating the Web UI 3/10



3. Obtain the ID of the Workspace where your Conversation was created.
 - a) On the left hand side, click on **Manage**, then click on **Launch tool** button, then click on **three vertical dots** in your Workspace , then click on **View details**.

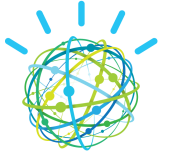


- b) Copy the Workspace ID and save it in convenient place (i.e. some file). You will need it later.





Creating the Web UI 4/10

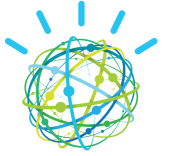


4. Install Node.js and Cloud Foundry to your workstation

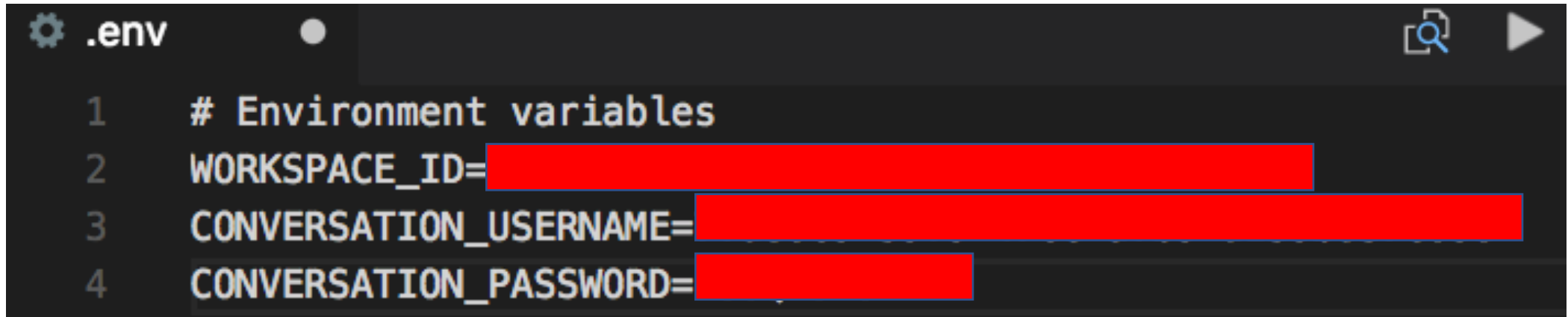
- Node.js is mandatory for testing and development the app on your workstation. You can install node.js from here: <https://nodejs.org/en/#download>
- Cloud Foundry is a multi cloud application PaaS. We will use it to deploy our application code to IBM Cloud. You can install Cloud Foundry from github at <https://github.com/cloudfoundry/cli#downloads>



Creating the Web UI 5/10



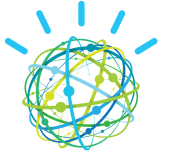
5. Modify the apps environment file so your app will be able to connect with your Conversation service on Cloud
 - a) Unzip the sample app from **conversation-simple-master.zip**
 - b) In the newly created folder **conversation-simple-master** rename the hidden file **.env.example** into **.env**
 - c) Open the **.env** file with a text editor and update all three environment variables with your Workspace ID, your Conversation username and your Conversation password.



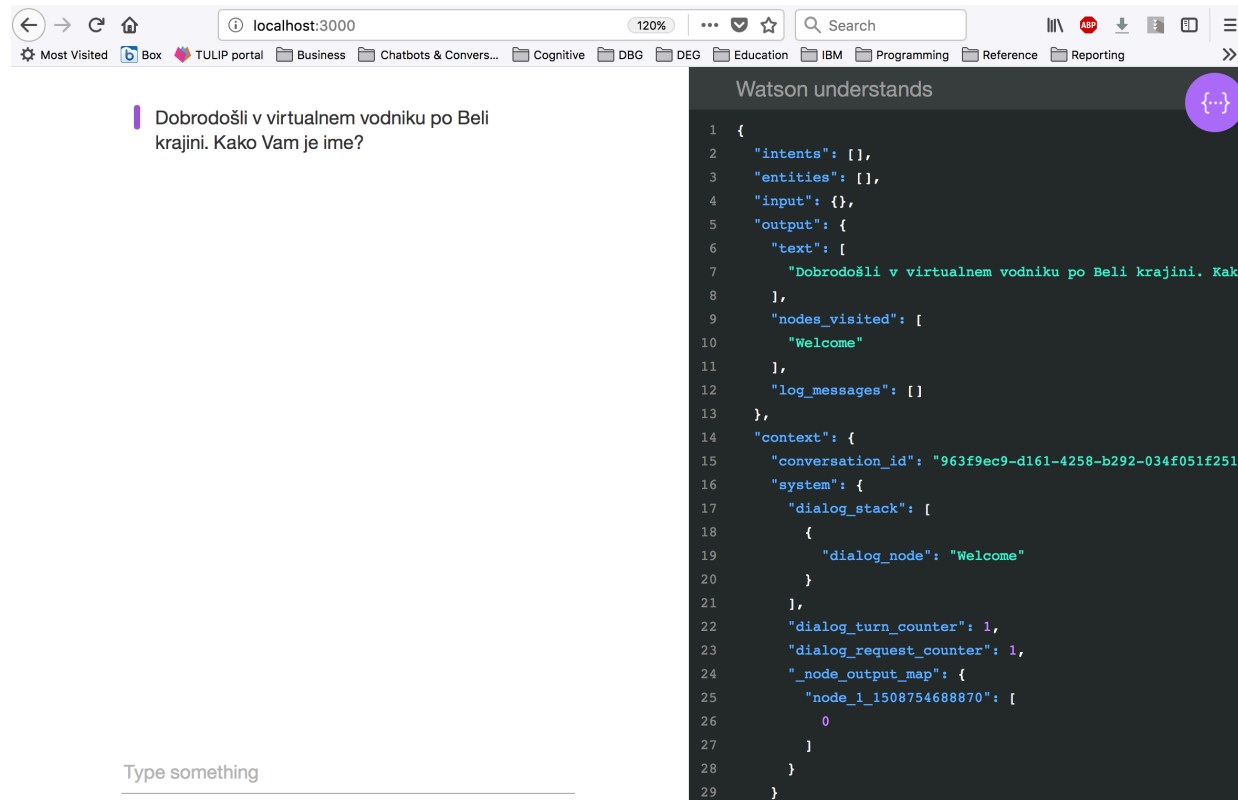
```
1 # Environment variables
2 WORKSPACE_ID=[REDACTED]
3 CONVERSATION_USERNAME=[REDACTED]
4 CONVERSATION_PASSWORD=[REDACTED]
```



Creating the Web UI 6/10

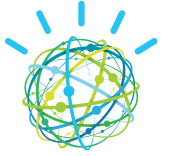


6. Install the node.js project modules on your computer
 - Open the **conversation-simple-master** folder in terminal and type in *npm install*
7. Start your application with *npm start* command, write down the port number (usually it is 3000)
8. Test the app locally in your web browser with: <http://localhost:3000>

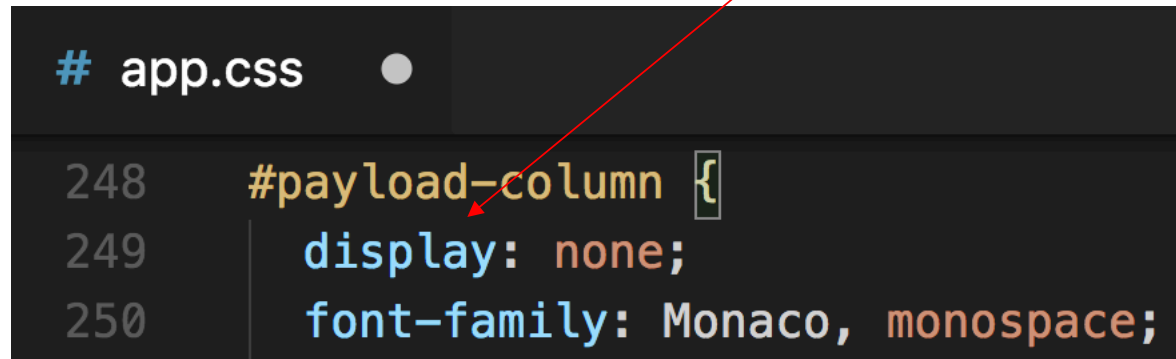




Creating the Web UI 7/10



9. If you do not want to see JSON data with Conversation responses on the right hand side, you can hide it this way:
- In your text editor open the file app.css (it is in subfolder public/css)
 - Insert following into line 249: *display: none;* (right below the line with “#payload-column {”)

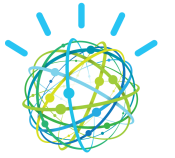


```
# app.css
248 #payload-column {
249     display: none;
250     font-family: Monaco, monospace;
```

- Refresh the browser to see the updates



Creating the Web UI 8/10



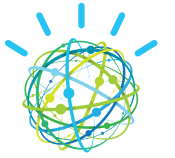
10. Deploy the app from your workstation to IBM Cloud

- a) In your text editor, open file **manifest.yml**
- b) Change two variables:
 - a) *name*: any-name-you-want (but it must be unique)
 - b) *services*: name-of-your-Watson-Conversation-service

```
! manifest.yml x
1  ---
2  declared-services:
3    my-conversation-service:
4      label: conversation
5      plan: free
6  applications:
7  - name: [REDACTED]
8    command: npm start
9    path: .
10   memory: 256M
11   instances: 1
12   services:
13   - [REDACTED]
14   env:
15     NPM_CONFIG_PRODUCTION: false
```



Creating the Web UI 9/10



11. In your terminal window, navigate to folder **conversation-simple-master** and:
- Log in to Cloud Foundry with *cf login* command
 - When prompted, enter your IBM Cloud user ID and password
 - Deploy the application with *cf push* command
12. This will take few minutes. When done, see messages in terminal. Near the bottom you will see your apps' URL.

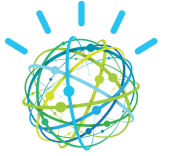
```
requested state: started
instances: 1/1
usage: 256M x 1 instances
urls: workshop-zagreb.eu-gb.mybluemix.net
last uploaded: Thu Nov 16 16:23:39 UTC 2017
stack: cflinuxfs2
buildpack: SDK for Node.js(TM) (ibm-node.js-6.11.4, buildpack-v3.15-20171024-1528)
```

	state	since	cpu	memory	disk	details
#0	running	2017-11-16 05:27:25 PM	0.0%	0 of 256M	0 of 1G	

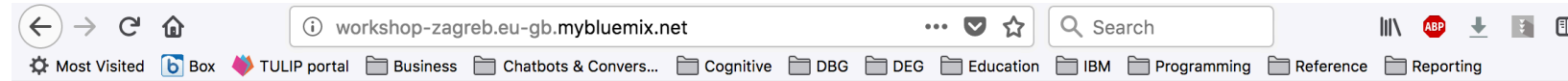
acos:conversation-simple-master Aco\$



Creating the Web UI 10/10



13. You can now access your app in Cloud.



Dobrodošli v virtualnem vodniku po Beli krajini. Kako Vam je ime?