**Botium CLI Installation**

**Step1: Installation** & **Configuration**

1. Install Node.js first if your machine does not have. Botium cli would be installed as Node.js module.
2. Now Run this in terminal window ---> ***npm install –g botium-cli***

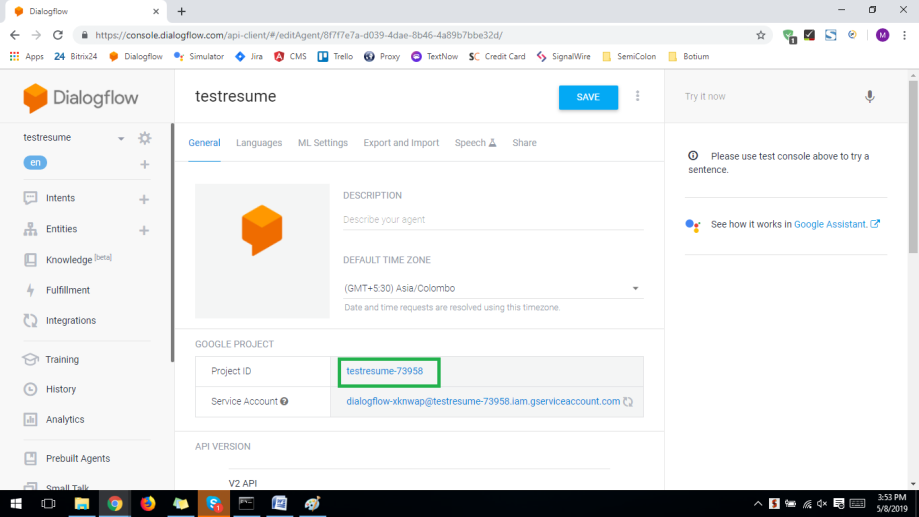
It will install botium cli in machine

1. In next step we have to configure connection between dialogflow agent and botium cli
2. The connection to the Dialogflow Agent is configured in a JSON-file. Create a file called “botium.json”, containing :

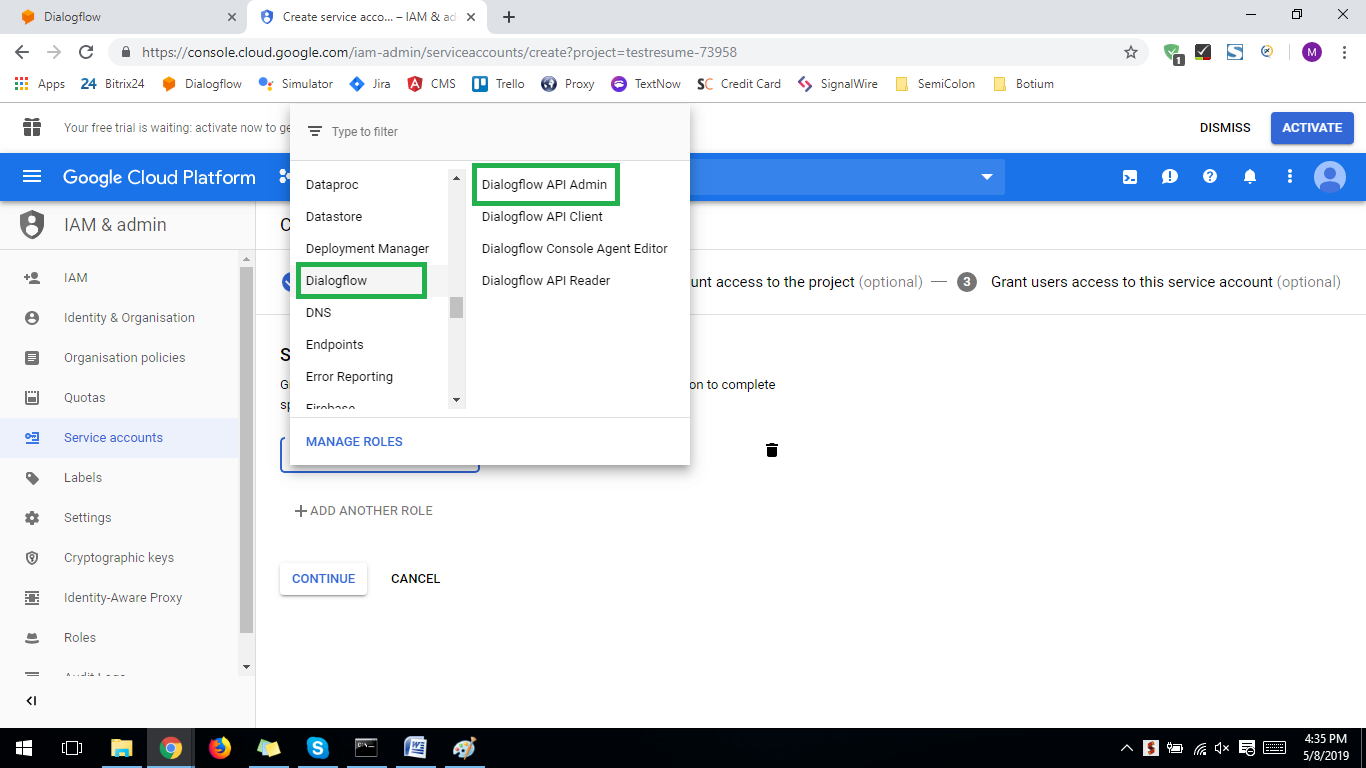
{  
 "botium": {  
 "Capabilities": {  
 "PROJECTNAME": "Dialogflow Sample",  
 "CONTAINERMODE": "dialogflow",  
 "DIALOGFLOW\_PROJECT\_ID": "<google project id>",  
 "DIALOGFLOW\_CLIENT\_EMAIL": "<service credentials email>",  
 "DIALOGFLOW\_PRIVATE\_KEY": "<service credentials private key>",  
 "DIALOGFLOW\_USE\_INTENT": false  
 }  
 }  
}

**How we can get this information of our dialogflow agent**

* We can find “google project id” in the setting icon of our agent in the dialogflow cosole



* For receiving service credentials to connect bottium to your diaogflow agent. Click on “Google Project ID” and it will redirect to Google cloud platform
* Now from side menu, go to IAM & Admin 🡪 Service Accounts
* Create a new service account and make sure that you have assigned “Dialoflow API Admin” role for botium (Admin role is required to run a full dialogflow agent)



* In next step is about to create private key for service account. Click on “Create Key” and select JSON as key type and download file. Open it in text editor, there is an JSON attribute is called “client\_email” and “private \_key” included, these are the values to copy to our botium.json file

**Note:** Private key should be included “Begin Private Key” and “End Private Key” texts with double quotes. Ex. "-----BEGIN PRIVATE KEY-----\....................\n-----END PRIVATE KEY-----\n"

**Complete and verify configuration:**

To complete the configuration, we set an environment variable pointing Botium the configuration file we just composed:

Linux/Bash:  
> export BOTIUM\_CONFIG=path/to/your/botium.json

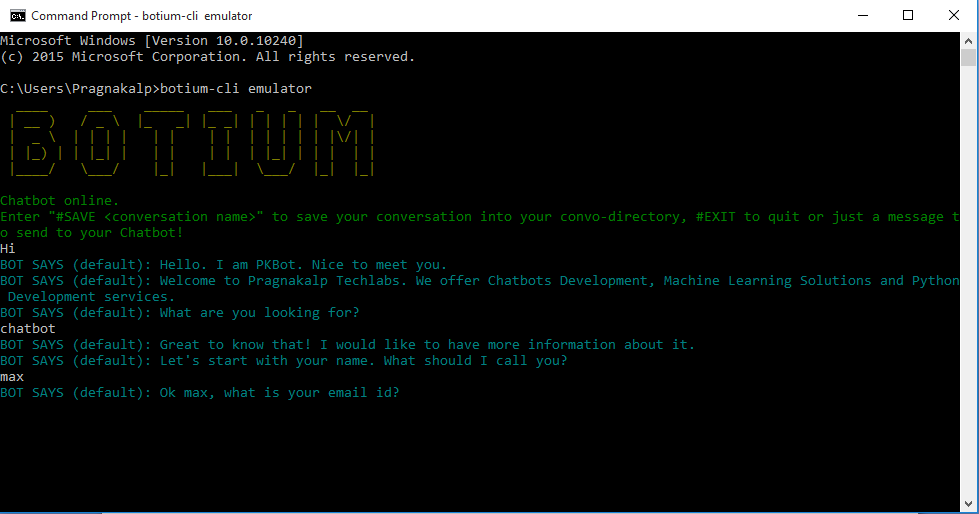
Windows cmd:  
> set BOTIUM\_CONFIG path/to/your/botium.json

**Note:** If above command does not work for you. Try to add environment variable manually by following these simple steps:

* Search for “Edit the environment variables for your account ”
* Under “User Variable” click on New and then fill variable name and give path of the botium.json file and save it

Now validate the configuration by running botium-cli emulator in terminal. It will connect to your diaogflow agent and will give simple chat interface in the terminal

* Botium-cli emulator

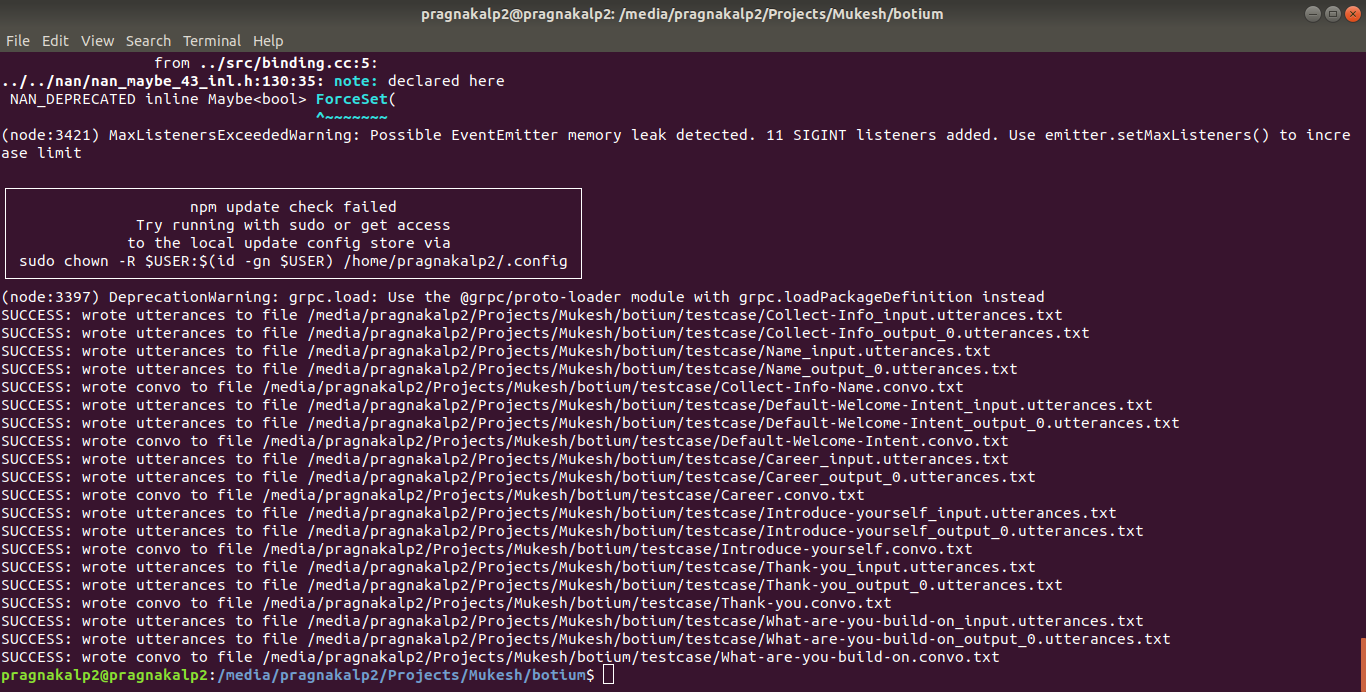


**Step 2: Generate Test Cases**

Now we would go for test case generation for dialogflow agents. In this phase botium will connect to your dialogflow agent, analyze the conversation structure and extract the intent resolution, the trained phrases and the conversation flow into test case automatically. We can create test case(called convo file in botium speak) for all intents, utterances(phrases) and conversation steps by using these simple command

export BOTIUM\_CONVOS=path/to/some/directory/holding/test/cases

botium-cli import dialogflow-conversations

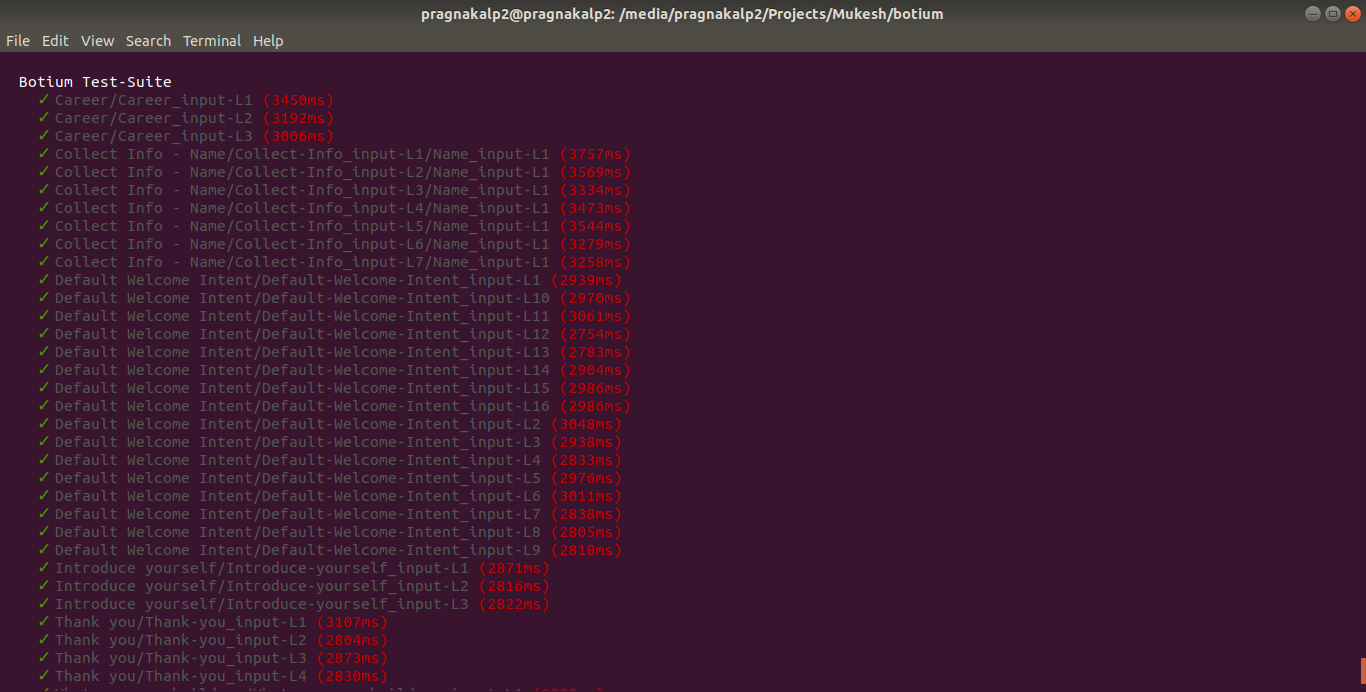


Note: Botium does a static analysis of your conversation flow. The Dialogflow Agent won’t be activated during this analysis, no fulfillment action or anything else will happen. The generated convo files should be seen as starting point for manually refining your test cases.

**Step 3: Verify Test Output:**

To execute all generated test cases, run botium again in “run” mode

botium-cli run

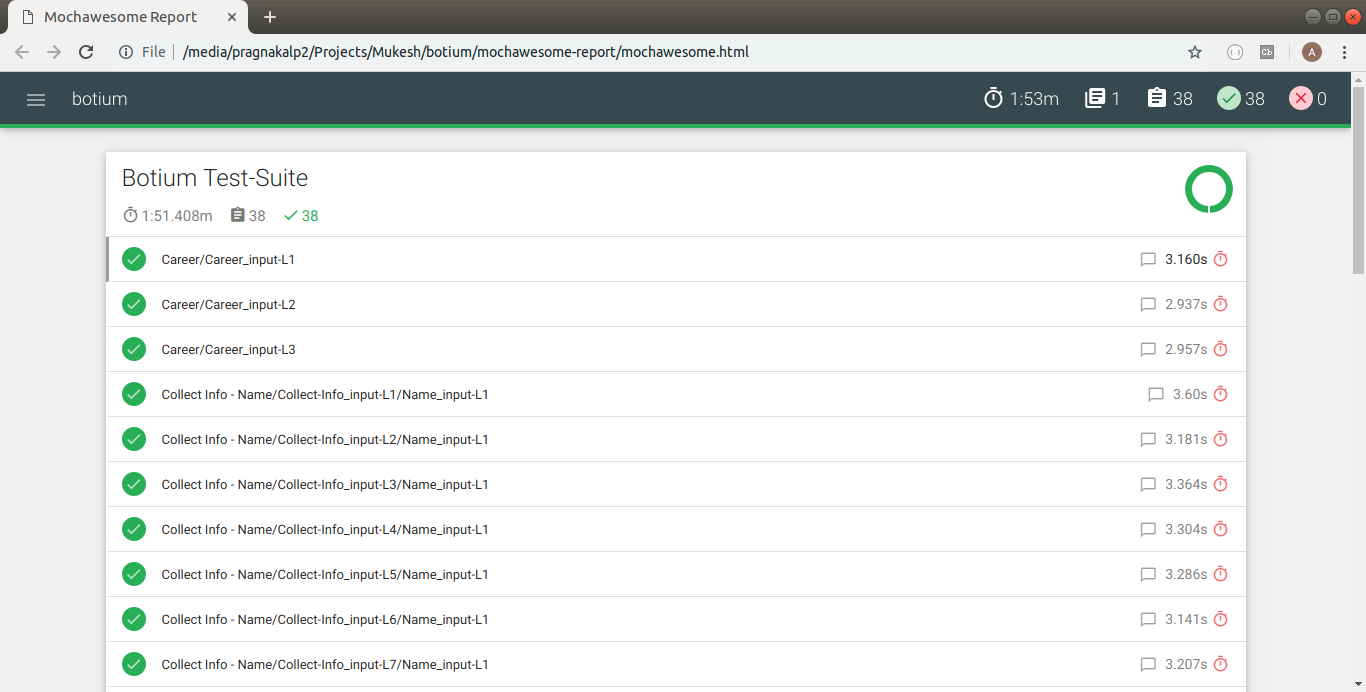


**Test Report Generation**: Once test case execution finish. All the test results display as PASSED and FAILED in CMD screen which green tick mark and red highlighted texts of particular test case.

But imagine what if we can see test report in HTML template. Sounds good, right? Of course YES !!!

So, Mochawesome repor generator is inbuilt functionality for botium-cli. So, We do not need to install it separately to your machine. We can generate test report in HTML and JSON by using simple command.

botium-cli run mochawesome



**Important Note**: For Linux users, Terminal’s commands are different. So whenever we need to run command starts with “botium”, command should be starts with “npx”. For example we want to run the test in the linux machine, that command would be something like this

npx botium-cli run