

# Lab4

[Edit](#)[New Page](#)[Jump to bottom](#)

MonkeyCanCode edited this page 4 minutes ago · 4 revisions

---

[Source Code](#)[Video](#)

## 🔗 Introduction

---

For lab 4, it contains two parts:

1. Create a dialog flow system with NAO robot that will allow user to interact with the NAO robot (e.g. ask the temperature to NAO robot and get the temperature back via voice)
2. Integrate NAO robot with ThinkSpeak and report data back via voice

## Objectives

---

The objective for this lab is for us to use what we learned in previous classes and combined them into something that we didn't done in the class. From this lab, I learned how to play with NAO and learn more about advantages and disadvantages for NAO robot.

## Member Contribution

---

We did this one together, so each member will have same contribution.

## Approaches/Methods

---

- For part 1, as the physical robot we used in the class has issue with built-in microphone, we are using virtual robot instead. For virtual robot, we wrote Python code to pull temperate with openweather API then connect this Python module with NOA. In order to interact with NOV in a dialog method, we sent message to NOA virtual robot and NOA robot responded back with the current temperature.

- For part 2, as NOV robot doesn't work with HTTPS. We wrote a Python web service to pull data from ThinkSpeak. The data in ThinkSpeak is populated from the sensors. Then NOA will just pull the latest record from Python web service and responds back with the last value read from the Python web service.

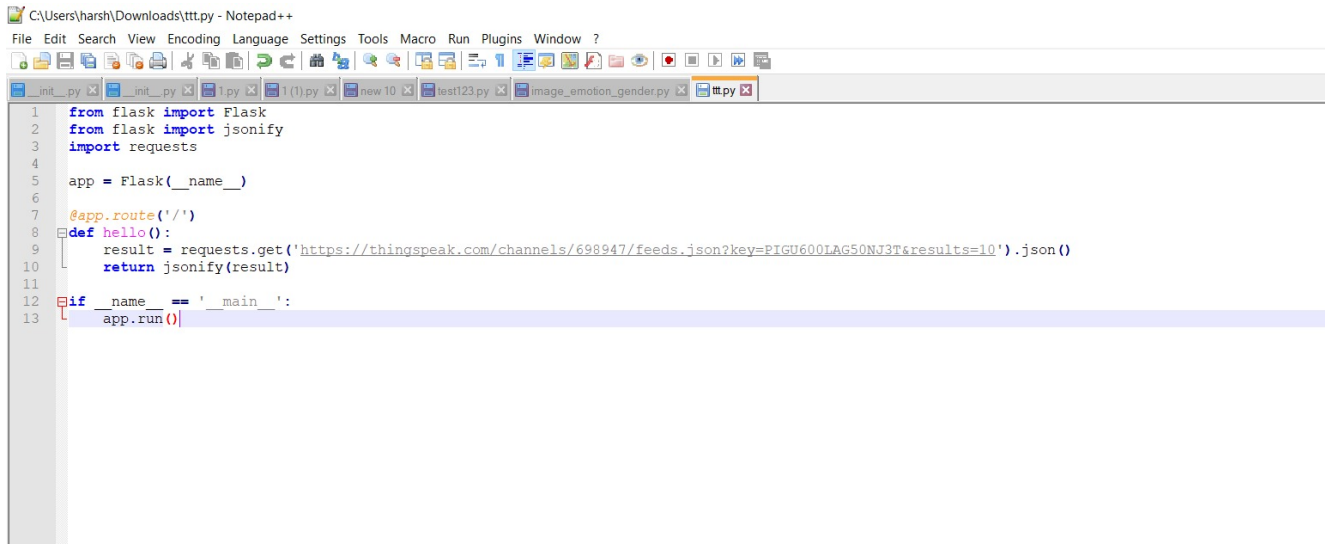
## Workflow

- Wrote Python code for module and web service.
- Connect Python code with NOA
- Interact with NOA

## Circuit Diagram

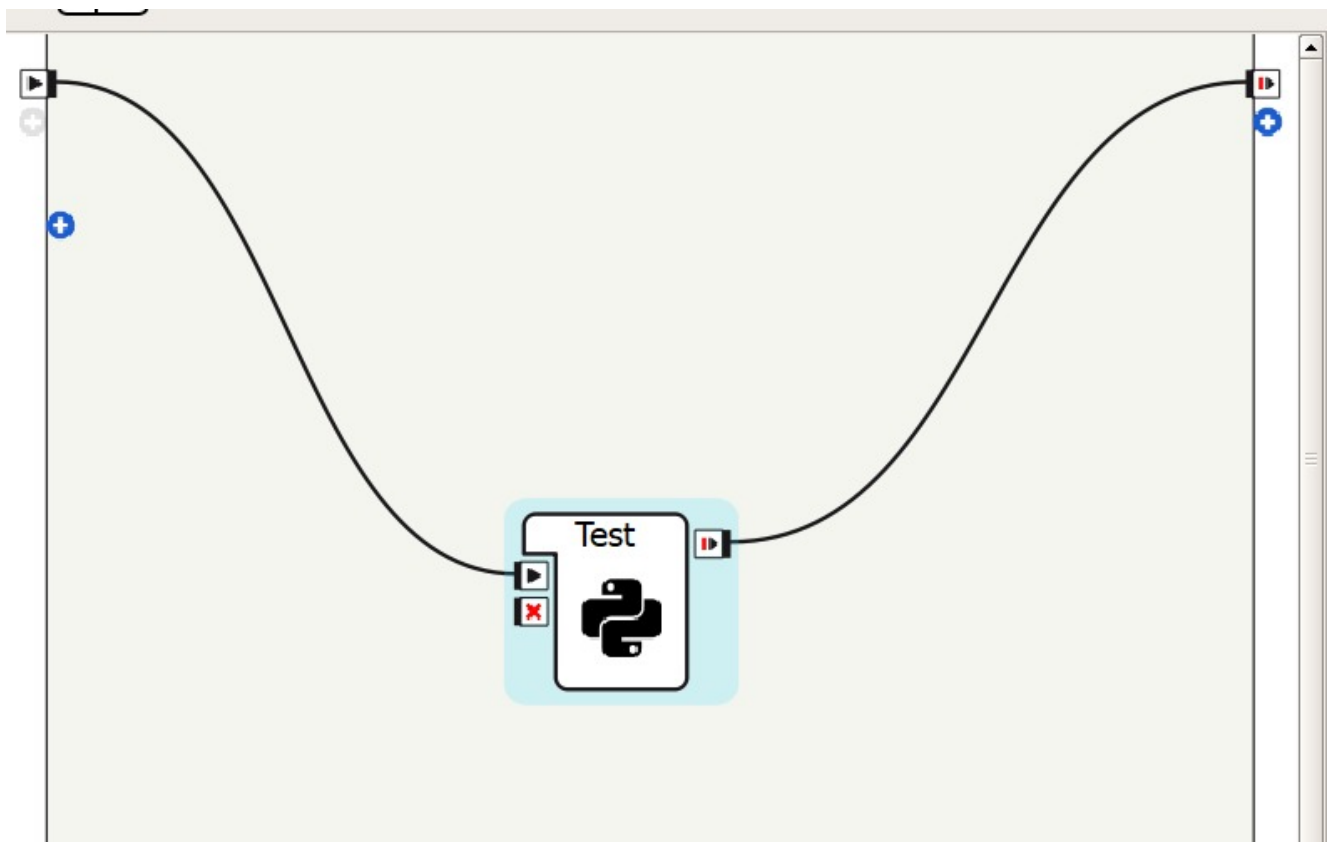
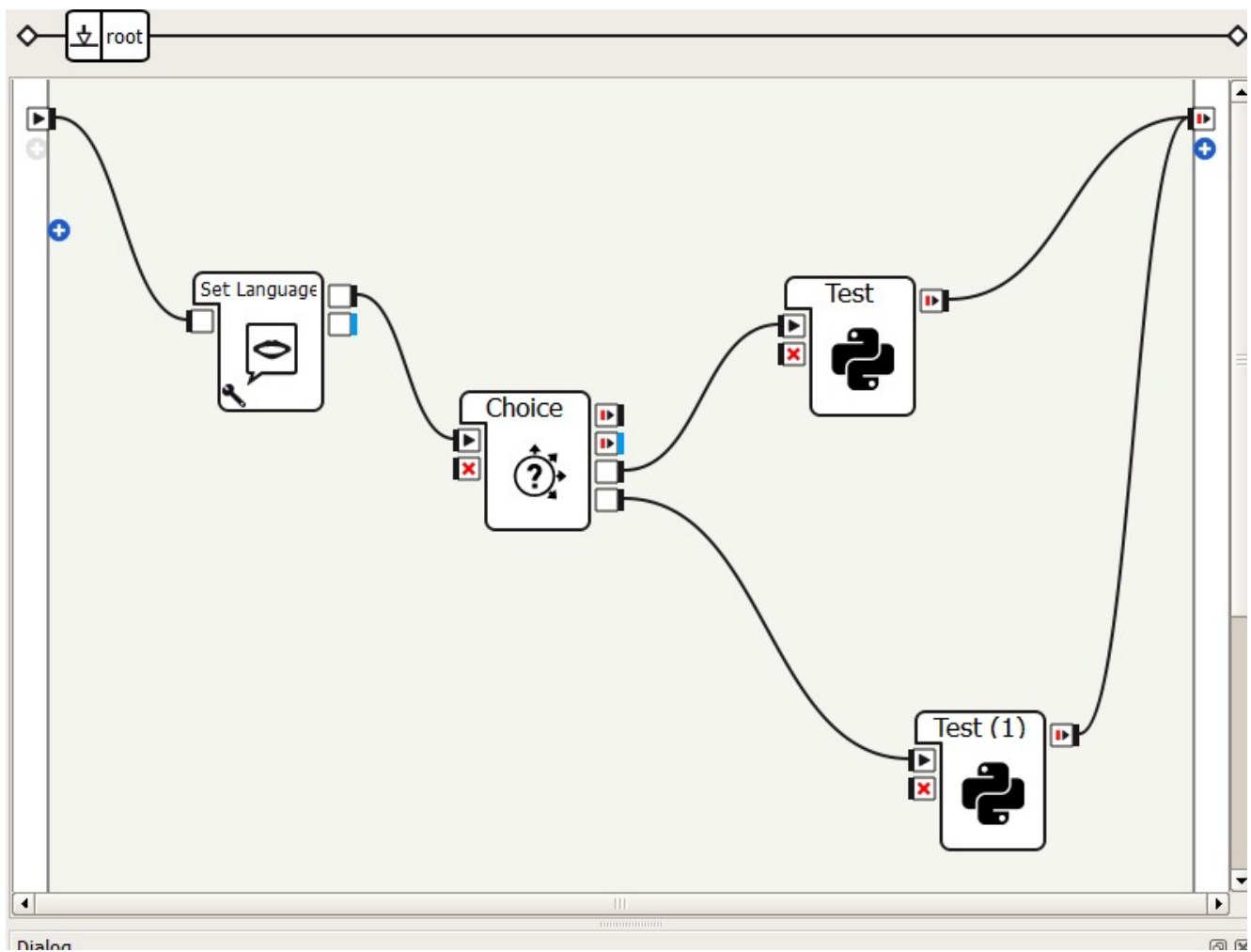
Same as previous Lab/ICPs.

## Additional images



```
C:\Users\harsh\Downloads\ttt.py - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?

1 from flask import Flask
2 from flask import jsonify
3 import requests
4
5 app = Flask(__name__)
6
7 @app.route('/')
8 def hello():
9     result = requests.get('https://thingspeak.com/channels/698947/feeds.json?key=PIGU600LAG50NJ3T&results=10').json()
10    return jsonify(result)
11
12 if __name__ == '__main__':
13     app.run()
```



The screenshot shows a web browser displaying a JSON feed of sensor data. The data includes fields for pressure, altitude, concentration, light, humidity, temperature, vibration, BPM, and location (latitude, longitude, name). The feed is updated at 2019-02-09T07:55:52Z.

Below the browser, a Notepad++ editor is open, showing a Python script named `delete.py`. The script uses `urllib2` to fetch data from `http://127.0.0.1:5000/` and displays it in a web application. The script includes methods for `onLoad`, `onUnload`, `onInput_onStart`, and `onInput_onStop`.

```

1  import json
2  import urllib2
3  class MyClass(GeneratedClass):
4      def __init__(self):
5          GeneratedClass.__init__(self)
6
7      def onLoad(self):
8          #put initialization code here
9          pass
10
11     def onUnload(self):
12         #put clean-up code here
13         pass
14
15     def onInput_onStart(self):
16         tts = ALProxy("ALTextToSpeech")
17         tts.say("hello")
18         result = json.loads(urllib2.urlopen(urllib2.Request("http://127.0.0.1:5000/")).read())
19         output_str = "There are " + str(len(result["feeds"])) + " feeds and latest value is " + str(result["feeds"][-1]["field1"])
20         tts = ALProxy("ALTextToSpeech")
21         tts.say(str(output_str))
22
23     def onInput_onStop(self):
24         self.onUnload() #it is recommended to reuse the clean-up as the box is stopped
25         self.onStopped() #activate the output of the box

```

## Evaluation & Discussion

NAO programming is pretty interesting. However, we didn't get a lot time to play with the physical robot as we only have one robot in the class.

## Conclusion

From this lab, I reviewed all of the materials I learned for previous lectures.

+ Add a custom footer

Find a Page...

Home

ICP1

ICP11 and ICP12

ICP13

ICP14

ICP2

ICP3

ICP4

ICP5

ICP6

ICP7

ICP8

ICP9

Lab1

Lab2

Show 3 more pages...

+ Add a custom sidebar

Clone this wiki locally

https://github.com/MonkeyCanCode/IOT.wiki.git

