CSEE5590-0005

IoT/ Robot Programming

(2018 Fall)

*Lab Assignment 4*

**Integrating IOT platform with NAO Robot**

Submitted On:

02 December, 2018

Submitted By:

Name: Farid Uddin Ahmed

Class ID: 01

Name: Mihir Manoj Pitale

Class ID: 09

Name: Zarin Tasnim Sandhie

Class ID: 10

Name: Kenton William Hanifl

Class ID: 12



LIST OF THE DOCUMENTATIONS

1. AUTHORS
2. OBJECTIVE
3. INTRODUCTION
4. MATERIALS REQUIRED
5. PLATFORM USED
6. CIRCUIT CONNECTION
7. NODE RED FLOW
8. MIT APP DESIGN
9. METHODOLOGY
10. ANDROID APP VISUALIZATION
11. CONCLUSION
12. LINKS

AUTHORS

This is the report containing the documentations of the Assignment #4. The contributors of the assinments are: Farid Uddin Ahmed (Class ID: 01), Mihir Manoj Pitale (Class ID: 09), Zarin Tasnim Sandhie (Class ID: 10) and Kenton William Hanifl (Class ID: 12)

OBJECTIVE

The objective of the project can be listed as follows:

1. Create a dialog-flow system with NAO Robot.
2. Explore ways to Integrate your project with NAO Robot and report if any failures or success in the attempt.

INTRODUCTION

In the past three weeks of IoT/Robot Programming class, we learned how to communicate with NAO Robot using “Choregraphe” software. And our project is to design an IoT smart home application with a fore fighting robot. For this assignment, we have integrated our project with NAO robot in such a way that the NAO robot will be able to say the weather condition at home and also if there is a fire at home, it will be able to say it. The assignment #4 is a combination of all the previous three ICPs and the project we did for hackathon. The assignment can be subdivide into the following sections:

* Integrate a sensor tag with raspberry Pi.
* The data from sensor tag will be uploaded to node-red via raspberry Pi.
* The weather data from node-red will be transferred to thingspeak.
* Those data from theingspeak will be fetched by “Choregraphe” software through appropriate flow which in turn will be said by NAO robot.
* Again, in case of a fire, the fire information will be send to thingspeak via Arduino Uno board.
* That data also will be fetched by “Choregraphe” software through appropriate flow which in turn will be said by NAO robot.

MATERIALS REQUIRED

* Sensor tag
* Arduino Uno
* Raspberry Pi
* Connectors
* NAO robot
* Bread board

PLATFORM USED

* Node-red
* Arduino
* Choregraphe
* Thingspeak

CIRCUIT CONNECTON

The circuit is connected as per figure:

NODE-RED FLOW

The flow made in node-red is given below:

CHOREGRAPHE FLOW

The flow made in choregraphe is given below:

METHODOLOGY

* At first, a sensor tag is connected with raspberry Pi via Bluetooth.
* The sensor tag sends the weather data such as temperature, humidity and pressure inside of the home to the raspberry Pi node-red via Bluetooth.
* Through node-red flow, those data are sent to thingspeak channel.
* Flow is created in “choregraphe” software which can fetch data form a website, in this case thingspeak website.
* Through dialog flow in choregraphe, a conversation is created with NAO robot where it can say the specified weather data.
* For the firefighting part, if there is any fire at the home, the fire detector will detect the fire and send the notification to thingspeak via Arduino board.
* That data also will be fetched from thingspeak to choregraphe flow in suach a way that NAO robot will be able to say “There is a fire”.

CONCLUSION

This assignment integrates of all the ICPs done during the previous three weeks along with the project. During this assignment, we were able to create a IoT smart home application with the NAO robot. Also the fire detection was also integrated with NAO robot.

LINKS

GitHub Link:

https://github.com/farid7666/CS5690-IoT-Robot/tree/master/Assignment\_3

GitHub Wiki Link:

<https://github.com/farid7666/CS5690-IoT-Robot/wiki/Lab-%233>

Video link:

https://www.youtube.com/watch?v=YGa\_81UVn70&fbclid=IwAR1twwCk74GYmtI\_ccECp0TOpFD2MwKFsLsqg2iIRXs2qdyGYwXStb1HofA

Source Code Link:

<https://github.com/farid7666/CS5690-IoT-Robot/tree/master/Assignment_3/Source%20code>