LabW06 – Internet of Things

Objectives:

1. Understand how to query data from cloud service
2. Get familiar with how to connect IoT devices to Azure Cloud service

Tasks:

1. Connect backend server to database on the Azure SQL Server by using Node.js
2. Setup your Raspberry Pi
3. Send temperature data to Azure cloud service

The Internet of Things (IoT) is the network of physical devices, vehicles, buildings and other item-embedded with electronics, software, sensors, actuators, and network connectivity that enable these objects to collect and exchange data. The IoT allows objects to be sensed and controlled remotely access existing internet frameworks.

In this tutorial, we will use a Raspberry Pi 3 to capture the temperature data and then display this data on your mobile device.

Task 1: Connect backend server to SQL Database by using Node.js

Please refer to the following link step-by-step to create a API service to access the Azure SQL server.
https://azure.microsoft.com/en-us/documentation/articles/sql-database-develop-nodejs-simple/

Connect to your server by using the command line tool, and then install the following libraries by typing the following codes in command line tool:

```
npm install express
npm install tedious
npm install promise
npm install forever
```

Then add an API link in your backend by updating the code in **app.js**:

```javascript
app.get('/gettemp', function (req,res) {
    var data = [];
    var sqlline = "SELECT TOP 20 createdAt,temperature FROM [dbo].[Pitemperature] ORDER BY createdAt DESC";

    // make a promise
    var p = new Promise(function (resolve,reject){
        // initalize a connection
        var connection = new Connection(constring_comp);
        connection.on('connect',function(err){
            if (err) {
                console.log("err: "+ err);
            } else {

                    // make a request
                var request= new Request(sqlline, function(err1,rowCount, rows) {
                    if (err1) {
                        console.log("err1: " + err1);
                    }
                }).on('doneInProc', function(rowCount, more, rows){
                    for (var i = 0; i < rowCount; i++) {
                        p = {
                            createdAt: rows[i][0].value,
                            temperature: rows[i][1].value,
                        };
                        data.push(p);
                    };
                    resolve(data);
                });
                connection.execSql(request);
            }
        });
    }).then(function (data){
        res.json(data);
    });
});
```

Please also update your database information in **app.js**. (The final content of your backend would be similar to app.js in your lab material.)

Once you finish updating the **app.js**, launch your backend by typing the following code in command line tool:

```
forever start app.js
```

Now you finish building your own backend. At this stage, if all the steps are done in a proper way, you can access data in the SQL server by visit:
http://***yourservername***.azurewebsites.net/gettemp

Task 2: Setup your Raspberry Pi

This task will help to setup Raspberry Pi development environment.

If you want to do this task, you need to have a Raspberry Pi and a LM75B temperature Sensor.

Please read the following link to connect your Raspberry Pi with temperature sensor:
http://wiki.52pi.com/index.php/Raspberry_Pi_LM75B_temperature_Sensor_v1.0_SKU:EP-0031

For your convenience, please also refer to **pi_setup.pdf** in your lab files.

Task 3: Send Temperature data to Azure cloud service

After the PI development environment are set, the next step is to send the temperature data on the Cloud. By doing this, we need to setup the SQL Server to save these data. Please visit the following link for detailed instruction:
http://www.instructables.com/id/Sending-Temperature-Sensor-Data-to-Azure-Database/?ALLSTEPS