

COE 4DN4 Lab1 - A Simple ‘Internet-of-Everything’ Smart-Home Server

Lab 1 Posted: Monday, Jan. 20, 2014

Labs: M,T,W,Th - JHE-234 EOW

TA Supervision & Demo Times (for Labs 1 and 2):

Labs: M,T,W,Th - JHE-234 EOW

TA Supervision & Demo Times (for Labs 1 and 2):

(These labs will be supervised if 2+ groups reserve time on each day)

Wed, Thu - Jan 29, 30

Wed, Thu - Feb 5, 6

(or by appointment)

Lab 1 Brief Report Due: Mon, Feb. 10, 2014 in Avenue-to-Learn (by 10 am)

(check email for updates on this lab)

Goal:

The Internet is undergoing a fundamental transformation and evolving into the “Internet-of-Things” (IoT) and more recently the “Internet-of-Everything” (IoE). The IoE will interconnect people, processes and devices, to yield a better and more efficient world. It will allow for Smart-Homes, Smart-Buildings, Smart-Cities, Smart-Manufacturing, Smart-Supply-Chains, etc. You will not find this material in any textbook since it is too new, but it was the hottest topic at the 2014 CES and other industry tradeshows. Please see these URLs for nice easy-read summaries:

<http://blogs.cisco.com/ioe/beyond-things-the-internet-of-everything-takes-connections-to-the-power-of-four/>

http://www.cisco.com/web/about/ac79/docs/innov/IoT_IBSG_0411FINAL.pdf

At the 2014 CES, the Cisco CEO estimated the commercial value of the IoE to be \$16 Trillion US over the period 2014-2022. To realize this value, the world will need to create a tremendous amount of easy-to-use intuitive software, to enable all people to easily interact with the IoE to achieve their goals.

Please download the TCP Echo Client-server code from the Donahoo site. Change the TCPServercode to act as a IoE Smart-Home Server.

No organization like IEEE has yet defined a standard for what a Smart-Home Server might look like since the IoE is so new, so we will define our own simple server. The Smart-Home server will allow any smart devices to 'register themselves' to the server. Once registered, the home-owner can then read the current-read-value and write the current-write-value. The IoE is expected to interconnect 50 billion smart-devices by 2022. The startup NEST builds smart thermostats, and was just bought by Google for \$3.2 Billion. The CES demonstrated smart thermostats, smart slow-cookers, smart refrigerators, smart moisture-sensors for your lawn/garden, smart-cars, etc. Next year there will be even more smart-devices.

The Smart-Home server maintains a data-base with many rows (say 64) and 3 columns: In column A is a text string for the smart-device name, with lengths = 128 bytes. In column B is another text string current-read-value, with length = 32 bytes. In column C is another text string current-write-value, with length = 32 bytes. You need to create a C or Java data-structure for this data-base.

You may initialize the first 2 rows of this data-base in your C or java code for testing as follows (here I use commas to separate the fields):

```
Thermostat-Main,      19, 23
Thermostat-Living-Room, 18, 22
```

The server responds to 4 commands from the client. Each command is a single transaction on a TCP socket. The server is running on a given IP-address and port, associated with the Smart-Home. The client must connect to the server, and then it can send several commands:

```
CONNECT  IP_address, Port
ADD       "device-name"
REMOVE    "device-name"
READ      "device-name"
WRITE     "device name"
QUIT
```

The READ command will return 2 values associated with the smart-device, the read value and the write-value. The WRITE commands will write a value to the write-value.

What to hand in:

You may work in teams of 1-4 students. The working system should be demonstrated to a TA during the laboratory demonstration hours, who will record a presentation mark. The TA(s) will ask questions during the demonstration to assess your software.

The lab report should be brief and have these sections:

- (1) Objective and brief introduction - a few sentences (1 page tops)
- (2) Server flow-chart* (thorough self-explanatory flow-charts)
- (3) Client flow-chart*

- (4) Well-documented C or Java code, referring back to flow-chart when necessary
- (5) Experimental results (include a few sentences and a few screen snap-shots to demonstrate working software)
- (6) Issues / Problems that were encountered & needed to be solved. Enumerate any issues. If we know common problems, we can address them for next year's class.
- (7) Record name of TA to whom you demo-ed the system
- (8) Conclusion - brief summary

If you have demo-ed the software, then the report is very brief and simply documents what you have demoed. The flow-charts are important and worth marks, so please follow our flow-chart format. (My written summary of my demo at CES was 2 pages - nice and brief.)

Flow-chart* format: The key to an easy-to-use IoE is easy-to-understand flow-charts, with sufficient technical depth to enable a computer-engineer to quickly understand and edit the software with confidence. The flow-charts are worth marks. The better flow-charts will get better marks. Every socket API function call should be written in a flow-chart box, along with all its parameters. Every data-structure item should be explained and initialized. Every operating-system function call (if any) should be written in a flow-chart box, along with all its parameters. All major operations on data-structures should be visible on the flow chart. I will present a few examples in class. You may use nested flow-charts, i.e., call other flow-charts for complex operations.

Please submit a electronic-copy of the lab report along with professionally documented software on Avenue-to-Learn.

Create a folder called 4DN4-Lab1-Submission-XX-YY (xx-yy are the initials of your group members). The folder contains the brief report (PDF format), and the flow-charts (PDF format). Also create a folder for the Server code and a folder for the Client code, and include the documented C or Java code in appropriate folders. Only 1 team member needs to upload a report, but the report should mention all team members and student numbers in the front page.

I encourage everyone to start and finish the labs early. They should be fun to do and there is no need to wait until the last week before they are due. The time-slots for demos are shown above. Each team needs to reserve a 15 minute demo-slot with the TA before the lab. session, on a FCFS basis. If you finish lab 1 early, please start work on lab 2. The prof or TAs should be able to help you with either lab.