Graz University of Technology

Institute for Technical Informatics

Computer Systems and Networks UE, WS2019/20
Konrad Diwold
kdiwold@tugraz.at

Assignment 4: The Bits go Global (5pt)

Deadline: January 16, 2020; 23:59 CET

In this assignment, you will connect your Raspberry Pi to the rest of the world using a variety of protocols. To make things more straightforward (and since you've already seen in detail how exactly the Raspberry obtains the sensed values), we will leave the bare metal realm for this assignment and use Raspbian instead. Otherwise, you are free to use a programming language of your choice to implement the tasks in this assignment.

① Switching Gears and Setting Up (0.5pt)

Switch to Raspbian on the RPi (or, in fact, to any OS of your choice) and make sure that your temperature sensor code is still running. This should not require more than a few rather simple changes to your program (e.g., timers). Also, make sure that your RPi is reachable over the network.

② Going Global and Mashing Up (1.5pt)

Although you'll only be creating a simple Web server in this task, please have a look at the current IRTF recommendations for *RESTful Design for Internet of Things Systems*[1]. Next, create a simple Web server that enables to read the most recently measured temperature value via the HTTP protocol. In particular, we recommend that you use the *OpenAPI* specification for creating your API and then use the *swagger* tool to automatically create a server. Integrate the (generated) server with your sensor access code and verify that your system works correctly by accessing the RPi's Web server remotely from your browser.

③ Mind your Footprint! (1.5pt)

In addition to the HTTP server, create another server that uses the *Constrained Application Protocol* (CoAP) to provide measured temperature values. In particular, make sure that your server supports the CoAP `OBSERVE` method[2]. We recommend that you use the Copper plugin[3] for debugging and to verify that `OBSERVE` is working as expected. Add to your report a brief, opinionated, paragraph about benefits and drawbacks of CoAP with respect to HTTP.

④ Beyond Client-Server: Chatty Sensors (1.5pt)

Select **one** messaging protocol out of [MQTT, AMQP, XMPP] and implement it on your RPi – your RPi should be able to deliver temperature values via these protocols. Briefly motivate your choice of protocol in your report, and also briefly describe the ecosystem you needed to create around your RPi to make it work.

⑤ Hand-in Instructions

By the deadline, hand in via TeachCenter: a **zipfile** that contains (i) your **code** and (ii) a **short report** that states the team members of your team and their roles when doing the assignment, pitfalls you encountered while working on the assignment, and instructions about how to run your code (if non-obvious).

[1] https://datatracker.ietf.org/doc/draft-irtf-t2trg-rest-iot/

[2] https://tools.ietf.org/html/rfc7641

[3] https://addons.mozilla.org/en-US/firefox/addon/copper-270430/