Universiteit Antwerpen

Lab report week 3

Michiel Aernouts michiel.aernouts@student.uantwerpen.be
Mats De Meyer mats.de.meyer@student.uantwerpen.be
Dennis Joosens dennis.joosens@student.uantwerpen.be

April 26, 2017

1 Progress

1.1 OpenHAB & MQTT

By default, mosquitto comes without any listeners. instructions to add a listener:

Edit the following file:

```
$ sudo nano /etc/mosquitto/mosquitto.conf
```

Add **listener 1883** to the file.

Restart the mosquitto service:

```
$ sudo service mosquitto restart
```

Confirm if the listener runs:

```
$ sudo netstat -nlp | grep mosquitto
```

Source:
`https://openhardwarecoza.wordpress.com/2016/06/25/raspbian-jessie-as-openhab-mqtt-server`

1.2 OpenHAB tweaks

To make sure the OpenHAB page gets updated we need to change the first line of the **teammmd.sitemap** as:

```
sitemap <filename> label="any name" {
```

In our case:

```
sitemap teammmd label="any name" {
```

1.2.1 OpenHAB (Basic UI): Date and time

To get the date and time working in openHAB we need to install the NTP package:

- Go to **143.129.37.79:8080** and go to the **PAPER UI**.
- Then go to **Add-ons** and select **BINDINGS**.
- Search for **NTP** and install it, a reboot might be needed.

Add the following to the **teammmd.sitemap** file:

```
Frame label="Date"{
        Text item=Date2
}
```

Add the following to the **sensors.items** file:

```
DateTime Date2 "Date and time [%1$tA, %1$td.%1$tm.%1$tY %1$tT]"
<calendar> { channel="ntp:ntp:sensors:dateTime" }
```

Add the following to the **sensors.things** file:

```
ntp:ntp:sensors [ hostname="nl.pool.ntp.org",
refreshInterval=1, refreshNtp=30 ]
```

1.2.2 OpenHAB (Basic UI): Temperature graph

To use basic charts in OpenHAB we need to install the RRD4J PERSISTENCE tool.

- Go to **143.129.37.79:8080** and go to the **PAPER UI**.
- Then go to **Add-ons** and select **PERSISTENCE**.
- Search for **RRD4J PERSISTENCE** and install it, a **reboot** might be needed.

Add the following to the **teammmd.sitemap** file:

```
Frame {
        Switch item=Pitemp_Period label="Chart Period"
        mappings=[0="Hour", 1="Day", 2="Week"]
        Chart item=piTemp2 period=h refresh=6000
        visibility=[Pitemp_Period==0, Pitemp_Period=="Uninitialized"]
        Chart item=piTemp2 period=D refresh=30000
        visibility=[Pitemp_Period==1]
        Chart item=piTemp2 period=W refresh=30000
        visibility=[Pitemp_Period==2]
        }
```

Add the following to the **sensors.items** file:

```
Group All
Number Pitemp_period <chart>
Number piTemp2 (All) {mqtt="<[myBroker:sensors/temp:state:default]"}
```

REMARK: 'All' can be any name as long as you change it at the correct place(s).

In the folder **/etc/openhab2/persistence/** create a file called **rrd4j.persist** and add the following:

```
Strategies {
everyMinute : "0 * * * * ?"
everyHour : "0 0 * * * ?"
everyDay : "0 0 0 * * ?"
```

```
default = everyChange
}

Items {
* : strategy = everyChange, everyMinute, restoreOnStartup
}
```

Finally a **reboot** is needed:

```
$ sudo reboot -h now
```

Source: `http://frederickvandenbosch.be/?p=162`

1.3 OpenHAB picture refresh problem

In **/etc/openhab2/sitemaps/teammmd.sitemap** change **refresh=5000** to **refresh=5** (for 5 seconds refresh interval).

1.4 MongoDB

In order to install and run a MongoDB server on Windows, we followed the tutorial from `http://stackoverflow.com/questions/20796714/how-do-i-start-mongo-db-from-windows`.

We installed a MongoDB server on the Raspberry Pi by following these steps:

- `sudo apt-get update`
- `sudo apt-get upgrade`
- `sudo apt-get install mongodb-server`

Starting the service is achieved by executing:

- `sudo service mongodb start`

1.5 Installing OpenCV

While OpenCV can be compiled on the pi directly, this gave us problems. The compilation would get stuck at 85% and took very long. However, a precompiled version of 3.1.0 can be found on github at `https://github.com/jabelone/OpenCV-for-Pi`. Following these instructions, OpenCV is installed in a few minutes.

1.6 Raspberry Pi (2 Model B) SD card backup

To backup the SD card we will be using a Linux Ubuntu PC (or VM).When you insert the SD card into the Linux machine it should automatically be mounted.

1.7 Backup SD card

To show the correct devicename of the SD card you can use the following commands:

- `lsblk | grep disk`
- `df -h`

In my case the SD card path is **/dev/sdc**, make sure you have the right one!

Writing the image to the home folder of the Linux PC:

- `sudo dd bs=4M if=/dev/sdc of=/home/dennis/backup_pi_04_03_2017.img status=progress`

It wil take a while (10-15 min) to backup the image, so grab a coffee.

1.7.1 Restore the backup

Make sure the SD card is FAT32 formatted. Unmount the partition(s) and format the SD card (just in case):

- `umount /dev/sdc1`
- `umount /dev/sdc2`
- `sudo mkdosfs -I -F32 /dev/sdc`

Start the restore process:

- `sudo dd bs=4M if=/home/dennis/backup_pi_04_03_2017.img of=/dev/sdc status=progress`

It wil take a while (10-15 min) to restore the image, so grab a coffee. It will tell you when it's done.

1.8 Backup with gzip compression

Save the image file to the Linux PC and compress it with gzip. Using gzip decreases the filesize immensely (in my case 8 GB to 1.6 GB). This process is not much faster than the previous one but it saves a lot of disk space.

- `sudo dd bs=4M if=/dev/sdc status=progress |`

  `gzip > /home/dennis/image_rpi_04_03_2017.gz`

It wil take a while to make the compressed image, so grab a coffee. It will tell you when it's done.

1.8.1 Restore the backup

- `sudo gzip -dc /home/dennis/image_rpi_04_03_2017.gz`

  `| sudo dd bs=4M of=/dev/sdc status=progress`

To show the restore progress you need to open an new terminal and install **pv**:

- `sudo apt-get install pv`
- `sudo pv -tpreb /home/dennis/backup_pi_04_03_2017.gz | dd of=/dev/sdc`

Source:
`https://diyhacking.com/raspberry-pi-backup/`
`https://www.raspberrypi.org/forums/viewtopic.php?f=91&t=46911`

1.9 Useful commands

- `df -T`
- `df -h`
- `du -hs`
- `free -h`
- `htop`

2 Planning

- Calibrating and testing MEMS sensor
- Add MEMS sensor data to openHAB (ok? uitbreiding?)
- Get OpenCV working with camera (busy)
- Backup SD card on blanco SD

3 Extra info

- Raspberry Pi address: 143.129.37.79