IOT pulse measuring system

Project: IOT pulse measuring system
Deltager : Tobias Valbjørn
Projekt startdato: 27-aug-2018
Projekt slutdato: 14-dec-2018

Introduction

This report covers the 5th semester IOT project at EDE Herning in the fall by Tobias Valbjørn. It describes the development of a prototype of an IOT pulse measuring device, that can give you a general idea of your fitness level based on your resting heart rate. The project scope and requirements are described in Aarhus University course catalog for the course E5IOT.

Problem statement

The resting heart rate is the speed of the heart beat, when you are rested.

"As you get fitter, your resting heart rate should decrease. This is due to the heart getting more efficient at pumping blood around the body, so at rest more blood can be pumped around with each beat, therefore less beats per minute are needed. "[1]

It could be interesting to track the resting heart rate over a period of time, to get a general indication of fitness.

[1] https://www.topendsports.com/testing/heart-rate-resting.htm

“The measurement of resting heart rate or pulse rate (the number of heart beats per minute) should be taken after a few minutes upon waking whilst still lying in bed. Give your body some time to adjust to the change from sleeping before taking your pulse (2-5 minutes). “

The measurement of the pulse first thing in the morning can be a challenge if you don’t have a wrist heart rate monitor, which can be expensive. If you use a chest strap, your resting heart rate will increase as you put it on. You can also use your fingers and a stopwatch, which can be difficult and inaccurate. In most of the cases you will have to remember the resting heart rate and note it manually.

Project description

The project focuses on improving and automating the process of keeping track of the resting heart rate.

The idea is that upon waking up, you would turn on your simple IOT resting heart rate monitoring system consisting of a particle photon, simple arduino heart rate sensor, a status RGB LED, and a vibrator with tactile feedback. The whole system is connected to your home Wifi.

Within a minute, as your body adjusts to change from sleeping, the system will be ready. Using the heart rate monitor your resting heart rate would be monitored, and uploaded to a webservice. You would get tactile feedback about when the test is done, and the results are uploaded to the web. Throughout the proces you could also check the status LED for the system state.

You could afterward see what your average heart rate is, if it has been increasing or decreasing, and it would place you into a basic category according to your sex and age, based on figure 1.

Resting Heart Rate for MEN

Age	18-25	26-35	36-45	46-55	56-65	65+
Athlete	49-55	49-54	50-56	50-57	51-56	50-55
Excellent	56-61	55-61	57-62	58-63	57-61	56-61
Good	62-65	62-65	63-66	64-67	62-67	62-65
Above Average	66-69	66-70	67-70	68-71	68-71	66-69
Average	70-73	71-74	71-75	72-76	72-75	70-73
Below Average	74-81	75-81	76-82	77-83	76-81	74-79
Poor	82+	82+	83+	84+	82+	80+

figure 1. General classification of fitness based on resting heart rate.

The webinterface can be used any time to change the information or watch simple charts of the progress. The system could be used to evaluate a fitness routine. The webservice is responsible for accepting basic user information, storing the information, being available for the IOT system throughout the day, and displaying the results, ie as shown in figure 2.

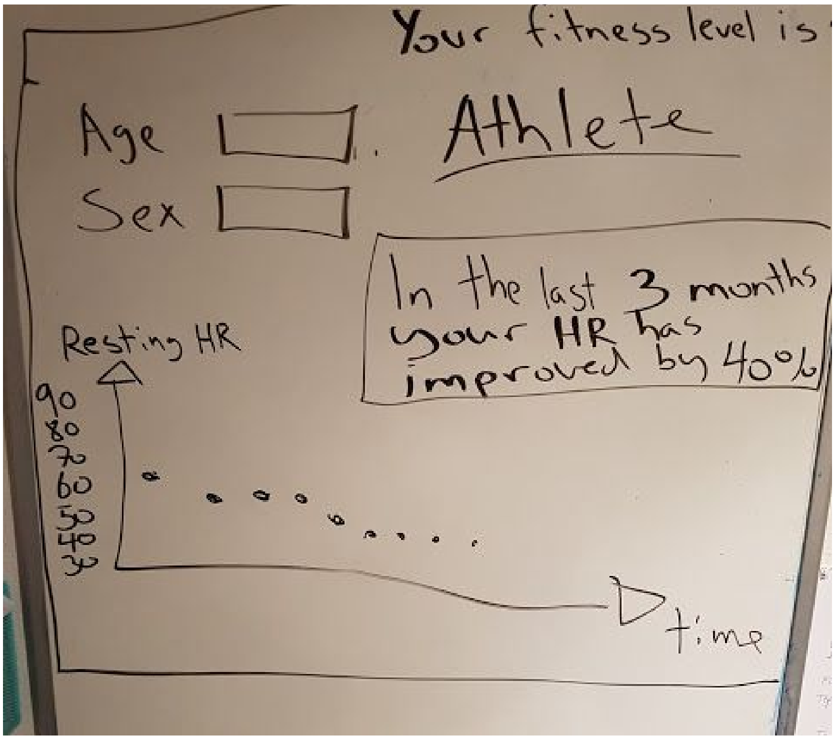

Figure 2. Demo webinterface.

Design requirements

The system will be turned on during the measurements. When the results have been read, the system will be turned off immediately. In this project scope to make a simple prototype and keep the budget low, USB power from the wall socket can be used instead of a battery powered system. Most people have a socket close to the bed for charging the phone or a night lamp for reading.
It does not need to take any sleep-mode functionality into account.
It only needs to start up fast.
It needs to be small to be in the bedroom, near the bed, supposedly at a bedside table.
The hardware needs to be organised in a way, that it can be easily equipped upon waking up.

Functional requirements

1. The platform has Wifi connectivity, and the system will connect over Wifi.
2. When the Wifi has been set up, the system should automatically connect to the internet when power is turned on.
3. After power is on, the application should start running automatically.
4. The platform has available digital or analog I/O to connect sensors and actuators.
5. The system reads the resting heart rate from a heart rate sensor local to the device.
6. The system should be able to send the heart rate locally from the particle photon to the webservice.
7. The web service should use the data to evaluate the resting heart rate monitor.
8. The device controls actuators in the form of LED and vibrator.
9. The device uses data from the webservice to augment what it does, by giving the user feedback on the LED and vibrator.
10. The webservice should be able to show the results of the heart rate measurements visually on a website.
11. The webservice should be able to store each measurement.
12. The webservice should be able to compare the heart rate data with the heart rate table.
13. The webservice should be able to compare the heart rate data with previous data.
14. The system should be able to measure the heart rate accurately. Within 2 bps difference from a polar chest strap heart rate monitor.
15. The system should be able to store the necessary information of 1 user from the website. (age, sex, previous heart rate measurements, category, and date)
16. The whole system should be able to fit into a box with the maximum dimensions of length 15 cm, width 10 cm, height 8 cm.
17. The system should not take more than 30 seconds to initialize upon power on.
18. After the system is ready, the user should be able to start measuring the heart rate within 15 seconds.
19. The device will be able to connect to AU's “AU Gadget network”

Non-functional requirements

- User interaction will be kept on a minimum to keep the system simple.
- The system functionality will be focused on the task of measuring the resting heart rate.
- Existing standards and protocols will be used wherever possible.
- The system will be built following the maxim; small pieces, loosely joint.

- The system needs to be simple, with as few user interactions as possible.
- The system will be calm when everything is working fine. It will not bring attention to itself beside when giving the user instruction or feedback.

Hardware

The particle photon can be used, since the project has simple needs, and a small form factor would be prefered. There needs to be Wi-fi and a few GPIO's. We don't need a lot of GPIO's, memory, high processing speed, ports or other communication modules. Furthermore the device does not have to be able to deliver power to a lot of sensors and actuators, only one heart rate sensor, a rgb-LED, and a vibrator, where the particle photon will be sufficient. The particle photons tasks will be to measure an analog value from the heart rate, publish the analog value, get basic response from the internet, and control the LED and vibrator.

I can use the particle photon from the class set.

Heart rate sensor

There is a well documented plug and play heart rate sensor for arduino, SEN-11574[2], with a lot of code to get started. It includes velcro, ear plugs, and other stuff to help with the measurements.
The 3 pins are:

RED wire = +3V to +5V
BLACK wire = GND
PURPLE wire = Signal

The price is: $24.95 = 161 dkk.

I have a simple RGB-LED and arduino vibrator I can use. The vibrator only needs power and ground, it might need a simple driver circuit as well, based on simple components from the lab. Since I am designing this project for myself, I will personally handle the expenses for the heart rate monitor.

[2] https://www.sparkfun.com/products/11574

Diagrams

Figure 3 shows the architecture of the system. Figure 4 shows a simple use case diagram, with the flow through the system. The diagrams can be open to changes as more information is gathered, especially figure 4, and the implementation of the webservice. A webhook might be used as middleman between the local system and the webservice, and maybe there will be more than one web interface to store the information and represent the results.

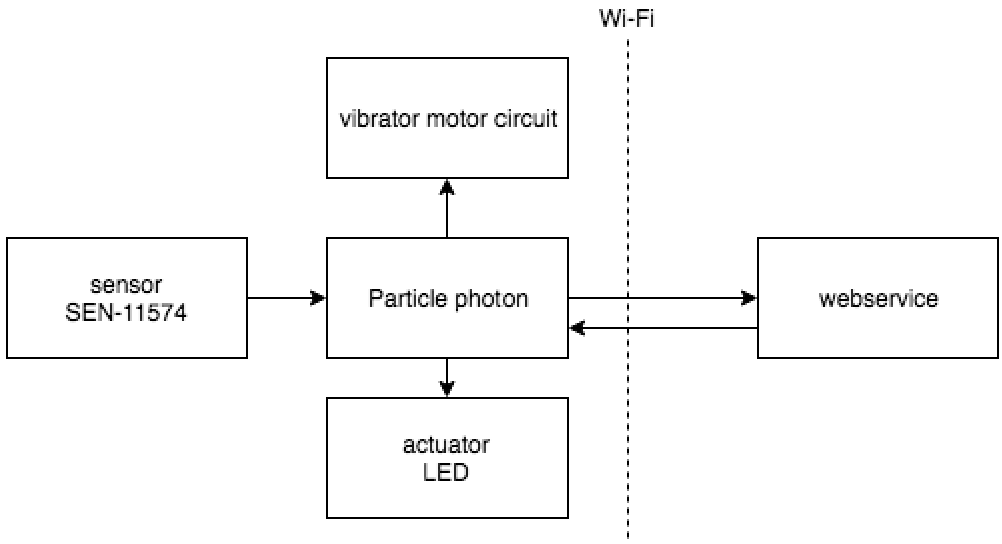

figure 3. architecture of the system.

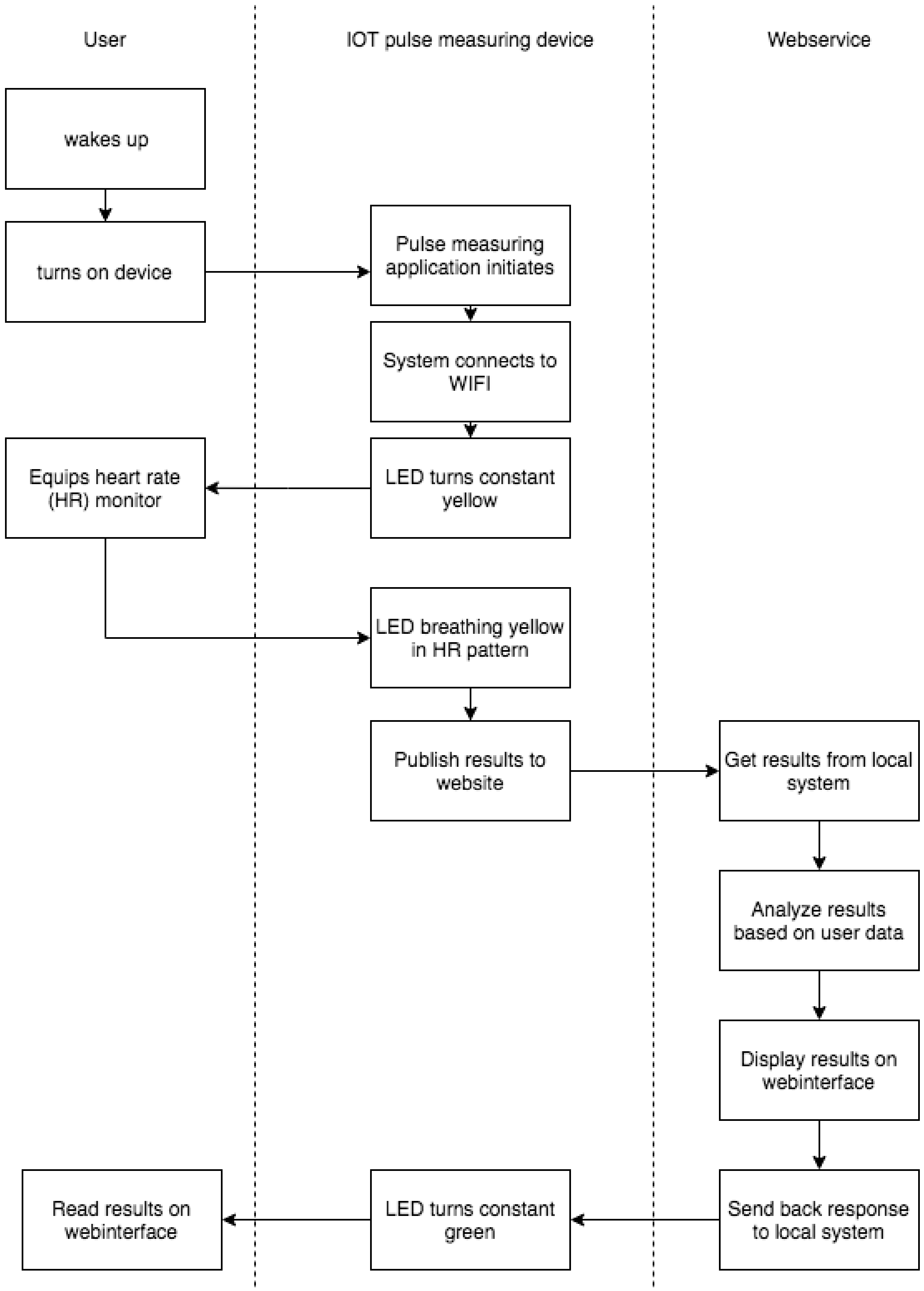

Figure 4. use case diagram for the system

Plan

Uge	Assignment	Deadline
35-37	Project selection Project description Purchasing hardware	14-sep
38	Implementation of heart rate sensor code Presentation	21-sep
39	Implementation of heart rate sensor code	28-sep
40	vibrator and RGB-LED implementation user experience testing.	5-oct
41	Get the whole heart rate measuring system working locally	12-oct
43	Analysis and design of the webservice part. Presentation	26-oct
44	Send local data to webserver Response from webserver Implement webservice to store and display resting heart rate.	2-nov
45	Implement analysis of user data	9-nov
46	Implement analysis of user data	16-nov
47	Represent user data Presentation	23-nov
48	Implementation of input	30-nov

	user data	
49	Final adjustments, report	7-dec
50	Report Upload via wiseflow	14-dec