Gezamenlijke opleiding Universiteit Hasselt en KU Leuven

2014•2015
FACULTEIT INDUSTRIËLE INGENIEURSWETENSCHAPPEN
*master in de industriële wetenschappen: elektronica-ICT*

# Masterproef
## Accelerating Health Tech

Promotor :
Prof. dr. ir. Ronald THOELEN

Bram Oosterbos
*Scriptie ingediend tot het behalen van de graad van master in de industriële wetenschappen: elektronica-ICT*

2014•2015

Faculteit Industriële ingenieurswetenschappen

*master in de industriële wetenschappen: elektronica-ICT*

# Masterproef

## Accelerating Health Tech

Promotor :
Prof. dr. ir. Ronald THOELEN

Bram Oosterbos

*Scriptie ingediend tot het behalen van de graad van master in de industriële wetenschappen: elektronica-ICT*

## Foreword

I would like to use the foreword to thank to the people who helped and guided me during this masters' thesis. Firstly my promoters Prof. R. Thoelen and *ir.* V. Claes for the tips and guidance they offered me throughout this year. Secondly, the institutions which made it possible for me to make this masters' thesis, namely the UHasselt, PXL and IMO-IMOMEC. Next, a special thanks to *ir.* Y. Philippaerts who gave me info about requesting patents while working with open source hardware. And also thanks to I-minds for providing me with some a blood pressure meter and extra info on making medical devices.

This masters' thesis in combination with the online portal that is made can be used as a guide to make a medical device. Not only on a technical level but also as a guide trough the important steps you need to take for making a medical device. If you are not making a medical device but a regular electronic device, this thesis is still useful because some of the steps are similar.

Bram Oosterbos

May 30th, 2015

# Table of contents

**FOREWORD** .......... **1**

**TABLE OF CONTENTS** .......... **3**

**LIST OF FIGURES** .......... **5**

**ABSTRACT** .......... **7**

**ABSTRACT** .......... **9**

**1 INTRODUCTION** .......... **11**

**2 PROBLEM DEFINITION** .......... **13**

**3 INNOVATIVE PORTAL** .......... **15**

3.1 OBJECTIVE .......... 15
3.2 IMPLEMENTATION .......... 15
*3.2.1 Search function* .......... *16*
*3.2.2 Publicly accessibly* .......... *17*
*3.2.3 Informative function* .......... *18*

**4 OPEN-SOURCE HARDWARE** .......... **21**

4.1 INTRODUCTION .......... 21
4.2 CURRENT MARKET SITUATION .......... 21
4.3 DIFFERENT TYPES OF BOARDS .......... 22
*4.3.1 Single board computers* .......... *22*
*4.3.2 Microcontroller boards* .......... *23*
4.4 OPEN SOURCE REGULATION .......... 24
4.5 FUTURE .......... 24
4.6 FINANCIAL ASPECT .......... 24

**5 MEDICAL DEVICE PROTOTYPING** .......... **25**

5.1 CE MARKING .......... 25
5.2 CLINICAL STUDY .......... 27

**6 PROJECT EXAMPLES** .......... **29**

6.1 POLAR BLUETOOTH HEARTRATE ANDROID APP .......... 29
*6.1.1 Software* .......... *29*
*6.1.2 Hardware* .......... *32*
6.2 OMRON BLUETOOTH BLOOD PRESSURE METER .......... 33

**7 CONCLUSION** .......... **35**

**8 APPENDIX** .......... **37**

8.1 RESEARCH DESIGN .......... 37
8.2 OPEN SOURCE HARDWARE AVAILABLE ON THE ONLINE PORTAL .......... 39
8.3 STEP BY STEP PROCESS STARTING FROM AN IDEA GOING TO A FINISHED PRODUCT .......... 40

## List of figures


3.1 Print screen of the portals open source hardware search tool. ... 16
3.2 Step by step process starting from an idea going to a finished product. ... 18
4.1 Raspberry Pi 2 ... 22
4.2 Arduino Uno ... 23
5.1 Steps to get your CE marking ... 25
5.2 Classification chart of medical devices ... 26
5.3 CE marking dimensions ... 26
6.1 Declaration of the variables used in the app. ... 30
6.2 Declare the actions to the buttons. ... 30
6.3 Transform data package to readable heart rate. ... 31
6.4 Validate heart rate and sending it to the database. ... 31
6.5 Polar Wearlink®+ transmitter with Bluetooth® ... 32
6.6 Package example coming from the heart rate monitor. ... 32
6.7 Omron 708-BT Bluetooth® blood pressure meter ... 33

## Abstract

Om een technologische stroomversnelling te creëren in de medische sector hebben we een innovatief portaal gemaakt. Op dit online platform kan je de nodige tools en info vinden die het gemakkelijker moeten maken om prototypes en afgewerkte medische producten te maken. Zo een product kan een draadloos communicerende weegschaal zijn of een geautomatiseerd infuussysteem. Op dit moment worden vele goede ideeën nooit omgezet naar een product omdat de weg veel te ingewikkeld is en het vaak veel geld kost. Deze masterproef concentreert zich vooral op het inkorten van de periode tussen het bedenken van het idee en de klinische testen.

Het innovatief portaal bevat een tool die je helpt bij het kiezen van de juiste open source hardware, want niet alle hardware is even geschikt voor elk project. Voor elk bordje is er een samenvatting te vinden van de licentie waar zij mee werken. Er zijn enkele voorbeelden toegevoegd om de mogelijkheden van elk bordje te illustreren. Daarnaast worden ook richtlijnen gegeven voor het aanvragen van een octrooi en het behalen van een CE markering van je product.

Dankzij dit portaal zal het dus eenvoudiger en goedkoper zijn om prototypes of afgewerkte producten te maken voor medische doeleinden. En op die manier willen we de technologische vooruitgang binnen de medische sector in een stroomversnelling brengen.

## Abstract

To create a technological acceleration within the medical sector we made an innovative portal. On this online platform you can find the necessary tools and information to make prototypes or medical products. Such a product could be a wireless communicating scale or even an automated infusion system. Nowadays a lot of good ideas are never to become a product because it is a difficult and expensive process. This masters' thesis concentrates on shortening the time period between the idea and the clinical tests.

The innovative portal holds a tool which helps you to choose the right open source hardware, because not all of the hardware is evenly equipped for every project. There is a summarized license for every hardware board so you can quickly see what is allowed. And there are some examples added to illustrate the capabilities of the open source hardware. Next, are the guidelines given to request a patent or to request a CE marking for your product.

Thanks to this portal it will be easier and less expensive to build prototypes or products for medical purposes. And this way we are trying to accelerate Health Tech.

# 1 Introduction

We live in a world filled with technology. Every day we come in contact with smartphones, computers, etc. Everything is getting automated and connected to the internet. In the morning your smartwatch measures your heart rate and uploads that to your online health diary. Next you way yourself on a Bluetooth scale that is also connected to that diary. During the day your steps get counted which are also uploaded to that same online diary. At the end of the week you have a nice overview of all these values.

All these perfect working devices on the consumer market to track your health and fitness, but none of them can be used for medical purposes. It takes about twice the time in order to get devices like these cleared for medical use because of more strict rules that need to be followed. While this does not seem like a problem it is in our own interest to develop these products in a shorter time period for the medical market as well. This way it gets easier to monitor patients, they do not have to be in the hospital, instead they can be at home or at work. That is not only more pleasant for the patient but also less expensive. So it is key to get that technology integrated into the medical world. In order to do so there are some obstacles we have to tackle and they are explained in the problem definition.

# 2 Problem definition

Nowadays we find a lot of electronic devices within de healthcare sector. But it evolves very slowly compared to the consumer technological development. This is a result of the complicated and expensive road you have to take when making electronic devices for healthcare purposes. First of all making a prototype can be an expensive and very time consuming. After every pilot, you have to make a new PCB and that costs a lot of time and money. For example: *"The development time up until clinical testing takes around 2-3 years and costs around 10 to 20 million dollars."* [1] When making a smart wearable for the consumer market, the time needed to develop this product is much shorter in comparison to making it for the medical market. This is due to the regulation with which you have to take into account.

Making a prototype or product for the medical sector takes up to much time and money. And that affects our health indirectly. If medical devices would evolve as fast as products on the consumer market there would be more and definitely better products to improve our healthcare. The slower development of medical devices is a consequence of all the extra regulations compared to consumer products. Summarized, developing products intended for medical use is too slow and too expensive.

# 3 Innovative Portal

At the center of this masters' thesis there is this innovative portal which helps you every step of the way when making a device intended for the medical sector. Before there wasn't a central point where you could find all the information needed in order to make such devices. The info was either incomplete or to complicated, but now whit this portal there is. It helps you to choose the right open source hardware for your project depending on your needs. It also provides information regarding patent request and CE markings.

On this portal there is a list of open source hardware boards that can be used to make prototypes. You can filter out the right board and find extra info about that board. Firstly, for every board you can find an example to get you started and show the capabilities of that board. Secondly, there is a summary of every boards open source license. This gives you an easy and direct view on what you may or may not do with the open source hardware.

Other info you can find on the portal is regarded to requesting a patent or making, CE markings or making a prototyping app in a fast and easy way. Requesting a patent for your product is not difficult though you have to take into account the open source licenses of the boards. When requesting your patent timing is crucial. There is also an example on how to make a prototype app to test mobile integration.

## 3.1 Objective

The main goal off this master's thesis is as the title sais "accelerating Health Tech", this portal does that by:

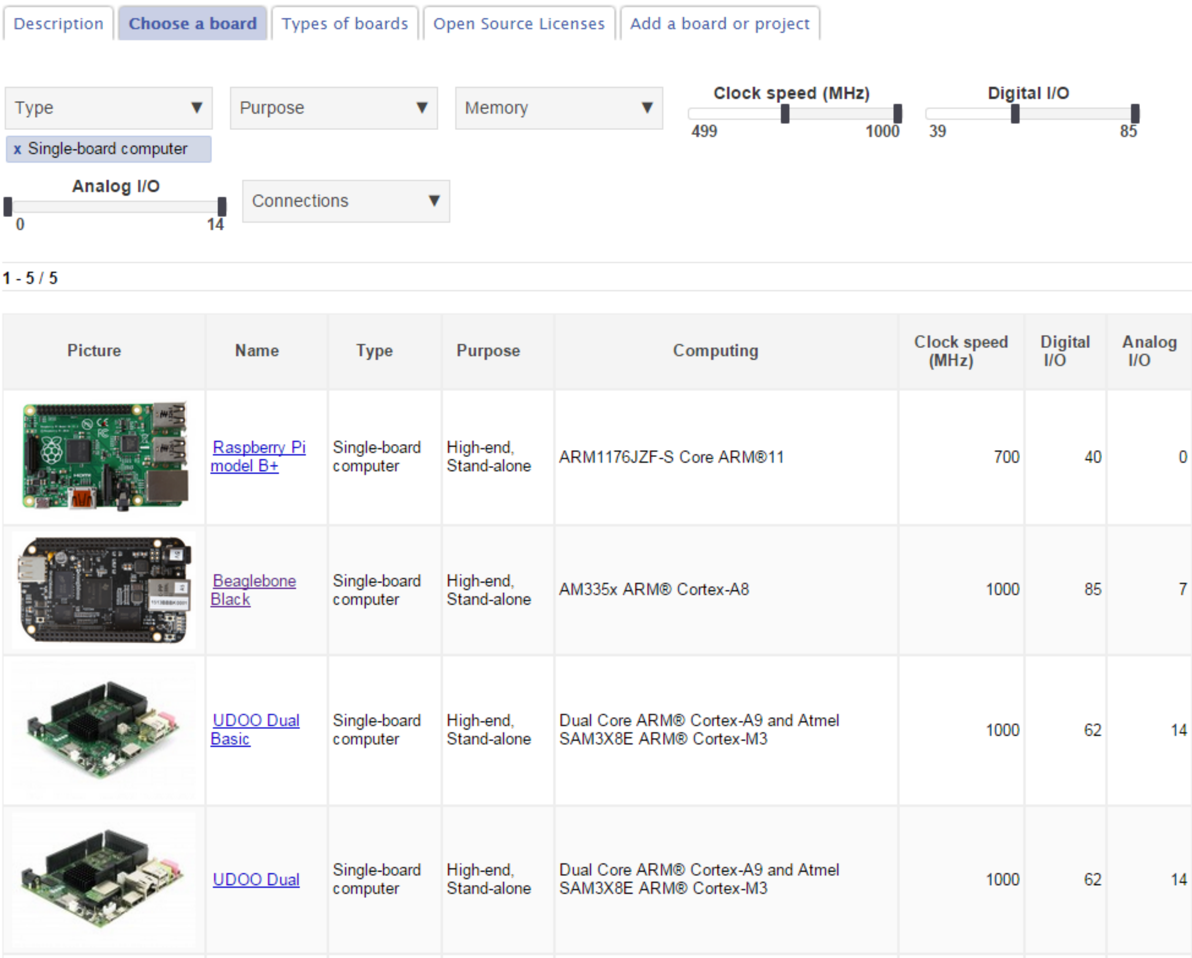
- Helping to find the right open source hardware for your prototypes;
- Giving simple and clear examples;
- Presenting the necessary legal information in one place;
- Explaining every important step of the process in an easy and clarifying way.
- To be a lead for future healthcare projects or startups.

## 3.2 Implementation

The website is made using google sites because of three main reasons. First of all the simplicity of google sites, it is more like making a document than a website. By clicking on the edit button you can change text and layout without making adjustments to HTML code. Everything is automatically adjusted to the right theme and gets aligned perfectly. Second, though it is very simple, there are a lot of possibilities when it comes to customization and add-ons. Integration of lists, links, images, graphs and many more are only one click away. And last, the sharing capabilities are a big advantage for a project like this. Because this portal is a work in progress, admin rights can easily be switched between people so someone else can further work on the website.

### 3.2.1 Search function

The search function of the portal is a good example of a simple but powerful option which google sites provides. It is a google gadget called *'Awesome Table'* which uses a google spreadsheet as a database. So every board which can be searched has been put in a google spreadsheet with all their characteristics. And thus you can filter out some boards depending on what you need specifically for your project. You can find an example in Figure 3.1 where some of the filters are activated.




| Picture | Name | Type | Purpose | Computing | Clock speed (MHz) | Digital I/O | Analog I/O |
|---|---|---|---|---|---|---|---|
| | Raspberry Pi model B+ | Single-board computer | High-end, Stand-alone | ARM1176JZF-S Core ARM®11 | 700 | 40 | 0 |
| | Beaglebone Black | Single-board computer | High-end, Stand-alone | AM335x ARM® Cortex-A8 | 1000 | 85 | 7 |
| | UDOO Dual Basic | Single-board computer | High-end, Stand-alone | Dual Core ARM® Cortex-A9 and Atmel SAM3X8E ARM® Cortex-M3 | 1000 | 62 | 14 |
| | UDOO Dual | Single-board computer | High-end, Stand-alone | Dual Core ARM® Cortex-A9 and Atmel SAM3X8E ARM® Cortex-M3 | 1000 | 62 | 14 |

3.1 Print screen of the portals open source hardware search tool.

It is not only a simple and fast way for setting up a filter tool. It gives multiple ways to filter on the characteristics of the open source hardware boards. The 'CategoryFilter' shows all the boards from the chosen category. If multiple categories are chosen, all the boards who are part of one of the chosen categories are displayed. This type of filter is used for the type of the board. The next filter is the 'csvFilter'. This one works almost the same as the 'CategoryFilter' with the only difference that if more requirements are chosen, only the boards who apply to all of the requirements are shown. This filter is used for the Purpose of the boards.

There is also the possibility to hide a filter. This way the values are not displayed in the list but you can still filter on them. This is done for the memory and connections of the boards. For the clock speed and memory of the boards no special filter is needed because 'Awesome Table' automatically inserts a filters when only number are written in the column.

Another very useful tool for this website is google forms. This tool gave us the opportunity to let visitors add boards to the website. The way this works is that a user fills in all the fields of the form like the name, website, number of inputs, etc. It will send and sort all the data automatically into to the spreadsheet which acts as a database for the search function.

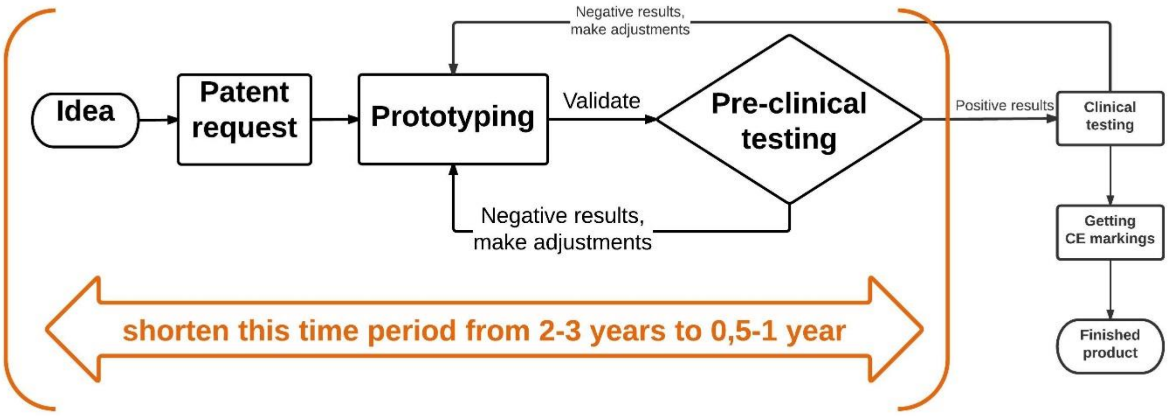
Initially the search function was planned as a smart search function. A search field where you could type the sentence *"WiFi heart rate and blood pressure monitor"*. Than the search result would give you all de boards with a build in WiFi shield and enough inputs and outputs to connect the heart rate and blood pressure sensor. Unfortunately it was not possible to insert such algorithms while using google sites.

### 3.2.2 Publicly accessibly

Another useful feature of google sites is the way you can control who can or cannot visit the website. There are three ways you can make the site accessible. First, "public to the internet", this way everyone can find and visit the website. This also means that the website can be found using google or any other search engine. Second, "Everyone with the link", only those people who are in possession of a link can access the website. There is no login needed to access the website when you have the link. And everyone who has that link can further distribute it. Last, "specific people", only the people you granted access can visit the website. They then have to login using their own credentials.

### 3.2.3 Informative function

As written earlier, the main goal is to shorten the time needed to develop a prototype for medical purposes. In Figure 3.2 is a schematic which illustrates the main steps starting from an idea ending with a finished product. A larger figure is provided in Appendix 8.3. Between the orange brackets you see the steps on which this masters' thesis focusses to shorten the development time of medical devices. But also the step on what to do to get a CE marking for your product are explained.




3.2 Step by step process starting from an idea going to a finished product.

The idea is something you probably have when visiting the portal, but it can still change when looking forward into the schematic. Some things you have not thought of could influence your initial idea. Requesting a patent is not a required step but one to put some thought into. Very important is that this step is done before you start prototyping. Because we are using open source hardware and you cannot request a patent when you have already released technical details about the product or prototype. If information was released before the date on which you requested the paten, it could be declared false. But most of the open source licenses mandate that you publish your product details. So you first have to request your patent to not conflict with the open source licenses. And from the moment the patent is requested your product is protected unless the patent does not get approved.

Getting a CE marking for your product proves that your product suffices to all the demands regarding safety, health and environmental protection. It is necessary to get that CE marking if you want to sell your product on the European market. [2] To get a CE marking affixed on you medical device you have to:

- Proof that your product is indeed a medical device;
- Check if the product meets all its specific requirements enlisted in the directive;
- Find out if your product needs to be checked by a '*Notified body*'[1];
- Assess the conformity of your product;
- Compile all the necessary technical documents.

When all of these steps are done and approved you can affix the CE marking to your product. All of these steps are explained in Section 5.1

[1] A notified body is an organization designated by an EU country to assess the conformity of certain products before being placed on the market. These bodies carry out tasks related to conformity assessment procedures set out in the applicable legislation, when a third party is required. The European Commission publishes a list of such notified bodies.

# 4 Open-source hardware

## 4.1 Introduction

*"Open source hardware is hardware whose design is made publicly available so that anyone can study, modify, distribute, make and sell the design or hardware based on that design."* [3]

So when you buy open source hardware, you can use it for your prototype, and when it does not meet your expectations you are able to modify it Of course taking in to account the open source license. When you do this license which every open source hardware board has, tells you what you can or cannot do with that board. For every board on the portal you will find the full open source license and a small summarization.

## 4.2 Current market situation

"In just under three years there were five million Raspberry Pi's sold worldwide. This makes them the fastest-selling British computer in history". [4] Knowing that it is only one of the many companies selling and producing open source boards you should get an idea on how big the market is at this moment. Another examples is that more and more Tech companies are selling products with an imbedded open source hardware board. Examples are a complete home automation kit with a Raspberry Pi as central component and a 3D printer with an Arduino that does all the calculations.

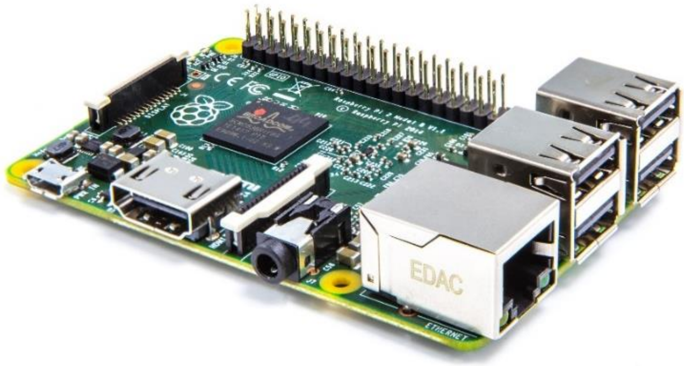
Unfortunately there are no exact numbers on how big the market is. It is just not possible to say how many percent of the market exists out of open source hardware. Simply because everyone can copy the design and distribute it under the same license. That makes it so difficult to know how many of these boards get sold every year. But as we see more and more derivatives come to market, it is safe to say that the popularity and thus the market is growing every year.

## 4.3 Different types of boards

For this masters' thesis, we split the boards up into two main categories being single board computers and microcontroller boards. The main difference is that single board computers are fully functional computers where microcontrollers still need to be link to another computer to be programmed.

### 4.3.1 Single board computers

As the name says, it is a computer but it is stripped down to its purest form, as you can see in Figure 4.1. With a single board computer you just connect a keyboard, mouse and monitor to have a working computer. All of the single board computers provided on the portal are open source and so they work with Linux or a derivative of Linux which often comes pre-installed. The combination of open source and Linux makes these boards a highly customizable product, thus perfect for prototyping. If the open source license allows it, you can even embed some of them directly into a finished product. Single board computers are available at a starting price around €35.



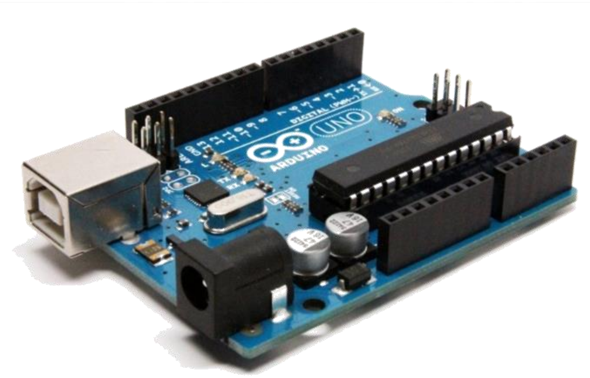

4.1 Raspberry Pi 2 [4]

Connecting sensors like a heart rate monitor to a single board computer is as easy as connecting the cables to the I/O pins. Something to take into account here is the number of sensors you want to connect. Make sure that the board you buy is future proof. This means enough inputs and outputs to possible expansion of the project and enough memory to get all the values calculated.

Most of these single board computers have Linux or a derivative of Linux as their operating system. For example the Raspberry Pi uses Raspbian, which is specially designed for it. But there are more operating systems that work along with the ARM processor like Arch, Pidora, OpenELEC, etc. Once the operating system is installed you can start writing little programs to interact with sensors which are coupled to the I/O-pins. The recommended language for the Raspberry is Python but any other language that can be compiled with the ARM can be used. A few examples are C, C++, Java, etc.

### 4.3.2 Microcontroller boards

Microcontroller boards differ to single board computers in a way that they need to be coupled to a computer to be programmed. Where you directly write a program on a single board computer, you have to make a program on a computer and then download it to the microcontroller. Most of these microcontroller boards have their own software in which to write a program. The most common is the Arduino Uno as shown in Figure 4.2.




4.2 Arduino Uno [6]

Why use a microcontroller board over a single board computer? It is safe to say that the single board computers have more capabilities than the microcontroller boards. The main advantage of a microcontroller board is the size. You can strip them down to a minimum of I/O's and one micro USB connection which makes them very compact compared to a single board computer. They still need multiple ports for a keyboard, monitor and mouse.

The success of microcontroller boards is the fact that they are really easy to use compared to a regular microcontroller. The board has all the connections needed to program the on board microcontroller, sensors and power. Whereas a regular microcontroller has none of these. These boards are off course more expensive but you get a complete package in a plug and play format, perfect for fast prototyping.

## 4.4 Open source regulation

As mentioned before, all the open source hardware have their own open source license, a set of rules you will need to follow in order to use them. Most of the open source hardware manufacturers use a license provided by Creative Commons. They have six different licenses of which the CC BY-SA is most used for open source hardware. CC BT-SA stand for Creative Commons Attribution-ShareAlike. Summarized this license means that you can adjust, change or tweak anything on the board for any reason as long as you distribute it under the same license and give the maker credit for it. [7]

## 4.5 Future

Making predictions about the future is very difficult. Though it is safe to say that the market of open source hardware is growing year by year and it will keep on doing so for a couple of years. Hopefully thanks to the portal more and more people will come in contact with open source hardware as a prototyping tool. And start developing more frequent medical devices on a smarter and faster way than before. Which ultimately ends with an acceleration of Health Tech and a better life for patients. Once adopted by a few more and more companies will start to see the benefits of open source hardware boards.

## 4.6 Financial aspect

Unfortunately we could not find a small business or company who would work along to calculate if it would be less expensive using open source hardware for prototyping. On the other hand we did find multiple examples online in which a cost calculation was performed to make an electronic product. But without credible references to these calculations it is not possible to use them as proof that it would be less expensive using open source hardware for prototyping. A fact is that none of the calculations came out below the price of any open source hardware board enlisted on the portal.

# 5 Medical device prototyping

Making prototypes for medical devices does not differ that much compared to making prototypes for the consumer market. Both processes are meant to develop a product, and solve issues along the way. As written earlier it takes twice the time to make a prototype for the medical market compared to the consumer market. Where do they differ? You have to suffice to different requirements regarding your CE marking and you will have to do a clinical study.

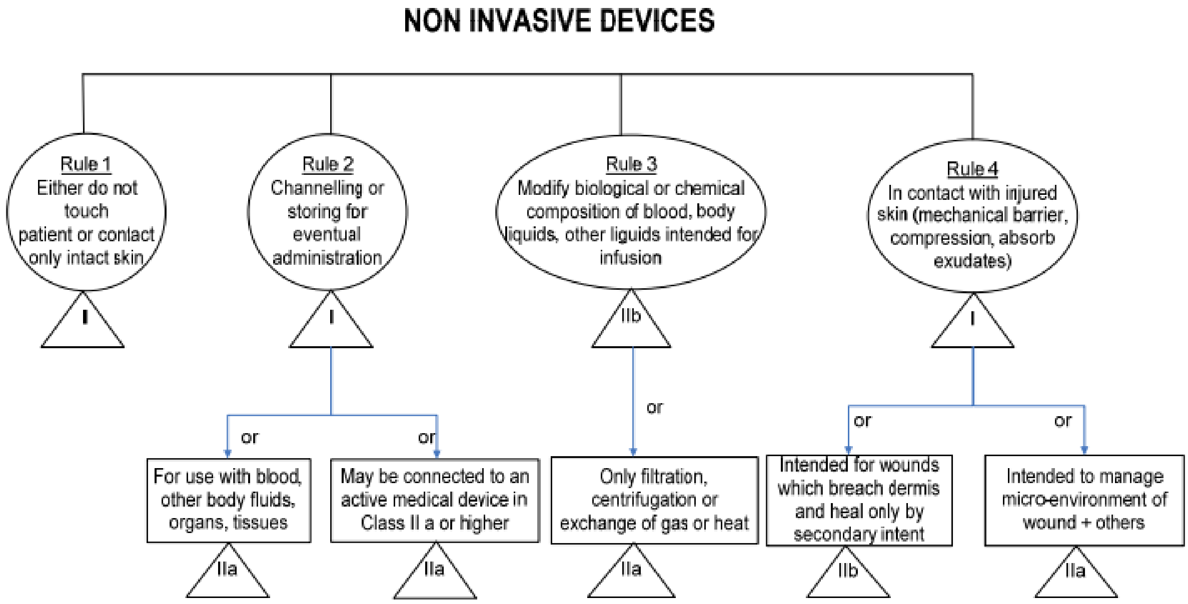
## 5.1 CE marking

As written in 3.2.3 it is necessary to get a CE marking if you would like to sell your product on the European market. Figure 5.1 shows the main steps you have to take to get your CE marking.


5.1 Steps to get your CE marking [7]

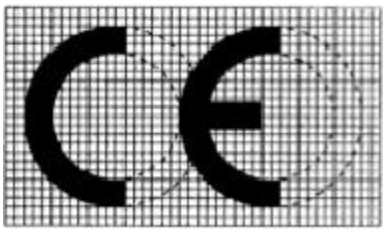
Step one are the medical device directives. This is where you have to proof that your product is a medical device, but that it is not an active implantable device or intended for in vitro diagnostics. In order to do this you need directive 93/42/EEC on medical devices (MDD). Step two is where you need to verify the requirements of your product. All of these essential requirements can be found in directive 93/42/EEC. The list of requirements consist out of six general requirements and eight requirements regarding design and construction. [8] A link to the directive is provided on the portal in the section about CE markings.

Step three is where you have to find out if your product needs to be assessed by a Notified Body or if you can do this yourself. In order to find out you need to know which classification your device falls under. Only if your device is a Class I, which means that there is a low risk, you do not need a Notified Body. If your product is a Class IIa or IIb which means it is a medium risk product or a Class III which means it is a high risk product, the assessment has to be done by a Notified Body. To know which class your device is in, use Figure 5.2. [7]




5.2 Classification chart of medical devices [9]

Step four, assess the conformity of your medical device. Depending on the classification, you have different options as how to assess and the stringency of the assessment. Exact details can be found in the MDD. After, you have to draft the Declaration of Conformity (DoC) to declare your responsibility. Step five is to draw up all the technical documentation. These documents are needed to assess the conformity depending on the requirements of the directive. Very important is that you keep copies of the documents for at least five years. At this point you need to submit all the documents and show you sufficed all the requirements to a Notified Body. Upon confirmation of the Notified Body you can go over to the sixth and final step which is to affix the CE marking to your product. Make sure that the dimensions are right and at least 5mm big. [7]



5.3 CE marking dimensions [10]

## 5.2 Clinical study

A clinical study is intended to gain medical knowledge by research on human volunteers. In case of medical devices, this means testing the features of that medical device on humans to discover or verify the effects it had on the human body. All the requirements needed to conduct a medical trial can be found in Directive 2001/20/EC. The reason why there are clinical trials is to create an environment in which medical devices or products could be tested. And when all requirements are met this environment ensures the highest standards of safety for the participants of the clinical study. [11] The Directive 2001/20/EC can also be found on the portal.

# 6 Project examples

In this masters' thesis is only one project example explained. To find more examples made by other students visit the portal at https://sites.google.com/site/prototypebordjes/ and click on the tab "example projects".

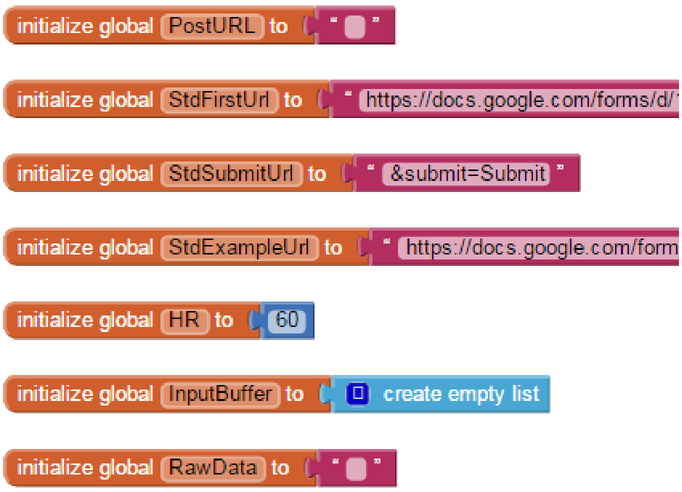
## 6.1 Polar Bluetooth heartrate Android app

A project example is the Android app which reads out data from a Polar Bluetooth heartrate monitor. Making an android app normally requires a fair amount of programming skills and development time. But there is another way of making Android apps, one which does not take that long and requires only a basic knowledge in any programming language.

### 6.1.1 Software

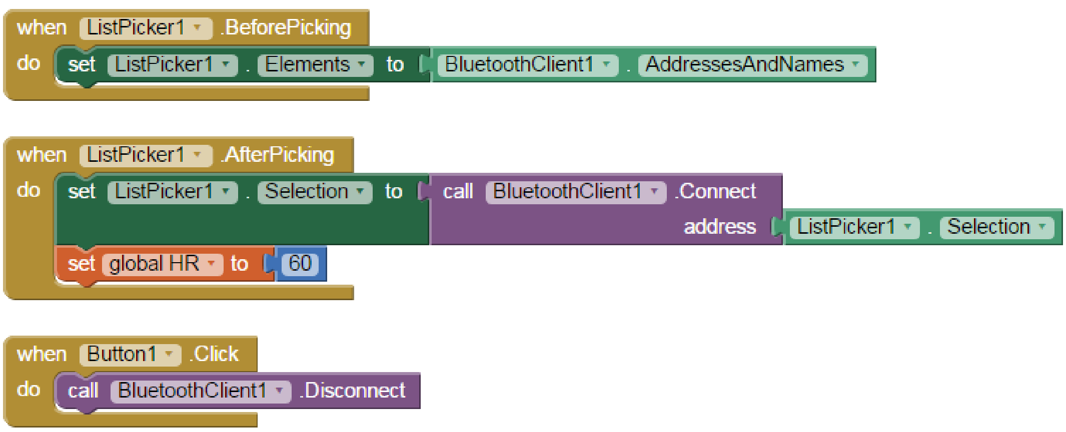
The tool used in this case is *'MIT App Inventor'*, but any other equivalent software development program would work too. The reason why it is so simple is thanks to the drag and drop system. You only need to understand the logic behind programming and not the programming language itself. Another big advantage of App Inventor is the wide range of tutorials. Starting with a tutorial on how to set everything up for making your first app, ending with tutorials on making advanced apps and games. The limits of App Inventor are very wide. To give an example, recently a group of students made a crowdsourced app to track busses in Nairobi. [12]

As in every software, you first need to declare the variables. Figure 6.1 shows all the variables used in the app. PostURL is an empty string that will be composed out of StdFirstUrl and StdSubmitUrl. StdFirstUrl holds the first standard part of the URL used to send the heart rate date to an online google spreadsheet. StdSubmitUrl is the ending of that URL, this is necessary to automatically send the data into the spreadsheet. StdExampleUrl has just been used for testing. HR stands for heart rate and is used to store the value of the heart rate. InputBuffer is used to store the separated data coming from the heart rate monitor and RawData is the non-adjusted data package received from the heart rate monitor.



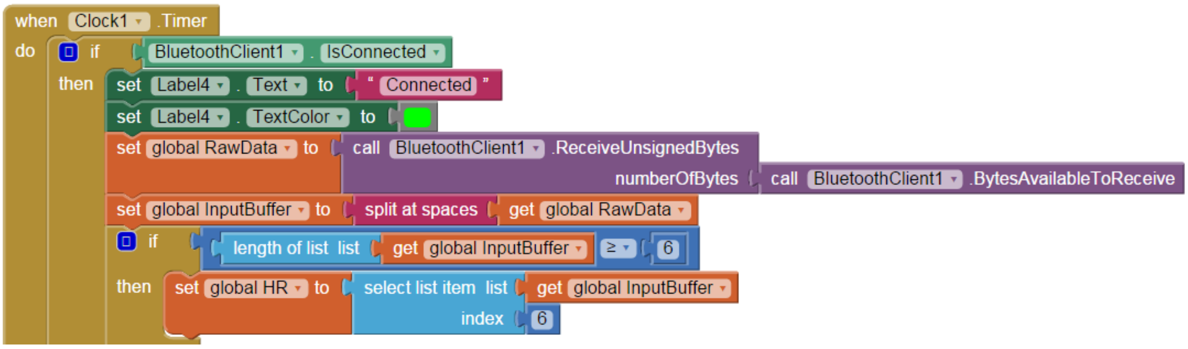

6.1 Declaration of the variables used in the app.

The next part of the 'code' is where the actions are coupled to the buttons as illustrated in Figure 6.2. The first block fills ListPicker1 with all the available Bluetooth devices. Before the devices are visible in this list, they need to be linked to the smartphone in its settings. The second block links the app to the selected Bluetooth device from the list. It also sets the value of the heartrate to 60 just to show that the app is receiving a heartrate. The third block disconnects the app from the Bluetooth device it was linked to when Button1 is pressed.



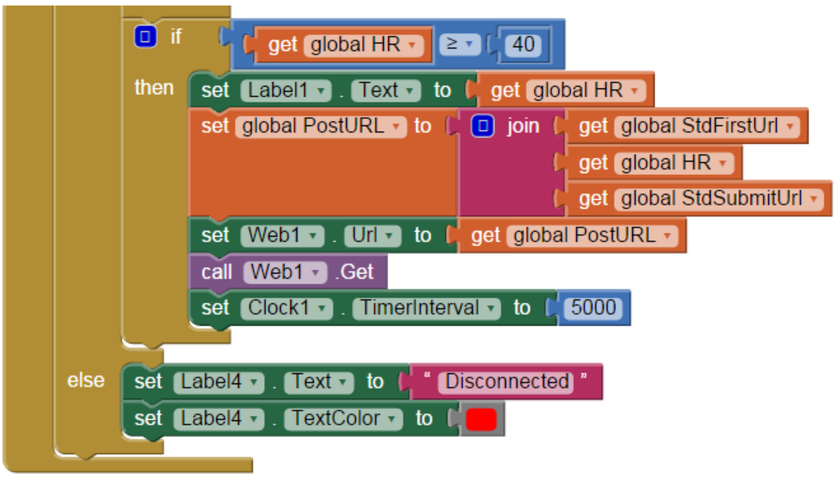

6.2 Declare the actions to the buttons.

The next block of the code is a loop. First part of this loop is where the received package from the Polar heart rate monitor is transformed to a readable heart rate as shown in Figure 6.3. First, if the phone is connected to the Bluetooth heart rate monitor, the text of Label4 is changed from a red "Disconnected" to a green "Connected". After, the data package received from the heart rate monitor is stored in the variable RawData. Next, the data package RawData is split into separate values which are stored in the list InputBuffer. After, if the list InputBuffer consists out of more than six items, the sixth item is stored in HR. It is necessary to check if the list is bigger than six items, cause when the monitor was not able to measure a valid heart rate, it only sends data about the battery.




6.3 Transform data package to readable heart rate.

The next part of the loop is where we test on a valid heart rate and send the value to a database as shown in Figure 6.4. First is tested if the heart rate is bigger than 40 beats per minute. If not, there is something wrong with the measurement or possibly with your heart. If it is bigger than 40bpm, the text of Label1 is changed to the new received heart rate. Next, the URL is constructed to send the heart rate to a google spreadsheet which acts as a database. After it is constructed, that URL is send to deliver the heart into the database. And last a timer is activated to wait five seconds. After these five seconds the loop can be run again.




6.4 Validate heart rate and sending it to the database.

At the very bottom, if the connection is broken up or if it is lost, the text of Label4 is changed again to a red “Disconnected”. The next step is to add the capabilities to read out a temperature sensor using Near Field Communication (NFC).

### 6.1.2 Hardware

The heartrate monitor used for this app is a Polar WearLink®+ transmitter with Bluetooth® as shown in Figure 6.5. It is a chest belt heartrate monitor which detects an electric current using two electrodes at the back of the belt. At the moment the belt detects a heartrate it automatically starts sending out data packages. Possible wrong measurements can occur if the two electrodes at the back of the belt are not moist enough. In that case a corrupted package is send, but as written earlier, those packages are filtered out.


6.5 Polar Wearlink®+ transmitter with Bluetooth® [12]

In figure 6.6 is an example shown of a package send by the heart rate monitor. Hdr has always a value of 254 (0xFF), Chk equals 255 – Len. Seq is a value in the range from 0 to 15 and status gives info about battery voltage and heart beat detection. And HeartRate is of course the measured heart rate. When no heart rate is detected, the sensor stops sending data automatically to safe its battery.

| Hdr | Len | Chk | Seq | status | HeartRate | RRInterval_16-bits |
|---|---|---|---|---|---|---|
| FE | 08 | F7 | 06 | F1 | 63 | 03 64 |
| FE | 0A | F5 | 06 | F1 | 61 | 03 64 |

6.6 Package example coming from the heart rate monitor.

## 6.2 Omron Bluetooth blood pressure meter

Another example should have been an app that reads out the Omron blood pressure meter shown in Figure 6.7. What this does is, it measures your systolic (upper), diastolic (lower) and the heart rate during the measurement. The systolic is the pressure (mm Hg) in the arteries at the moment your heart muscles contract, the diastolic is the pressure (mm Hg) in the arteries between the heart muscle contractions. After the measurement you should hold the ‘upload’ button down for a few seconds until a connection is made with the smartphone on which it should send the measured data to the phone. With the Omron health app you can than keep a blood pressure diary to have a more accurate view on your blood pressure compared to one measurement at the doctors cabinet.


6.7 Omron 708-BT Bluetooth® blood pressure meter [13]

Only problem is that the blood pressure meter was not able to send the data to the diary app. And so the assignment was to build a new app that worked together with this meter. After setting everything up, making a connection with the meter was no problem. Once the blood pressure measurement is done, the upload button was pushed but no data was received. To debug this problem as Bluetooth scanner was used. The scanner displays every data send by a selected Bluetooth device. At the moment the upload button was pressed after a measurement, no data was received.

The conclusion is that either the upload button does not work, or a transmitting problem occurs. Either way, it was not possible to solve the problem. To compensate, another app was made to read out the data from a Polar heart rate monitor as explained in Section 6.1.

# 7 Conclusion

A conclusion is to evaluate if the objectives of this masters' thesis are met. The main goal is to accelerate Health Tech but it is not possible yet to know if I succeeded. It will take some time for people to start developing medical devices with the help of the online portal. If so, the sooner these products come to market, the better it worked.

It is safe to say that all the necessary items are there to accelerate Health Tech. There is an online portal which acts as a guide for all the important steps in making a medical device. It involves a tool which makes it easy to select the right open source hardware for the prototyping phase. For all the open source hardware is the license summarized which gives an instant idea on what can or cannot be done with the hardware. As well as the info on how and when to request a patent for your product, timing is crucial when working with open source hardware. And the main steps on how to get a CE marking for your product are explained. To be complete, multiple examples where uploaded to the portal in order to get users started on prototyping with open source.

Future expansions for this masters' thesis are the addition of Near Field Communication capabilities for the heart rate app along with widening the range of compatible heart rate monitors. Also, over time more examples and boards should be added to have a more complete range of open source hardware and its capabilities. And last, the online portal could be transformed from a google site to an independent website.

# 8 Appendix

## 8.1 Research design

**Situating**

There are loads of products with healthcare capabilities on the consumer market. Problem is that they are not used to their full potential, actually being used in healthcare applications. Why is that, and why are these product only developed and produced for the consumer market? It is due to extra and stricter requirements which are put on for medical devices which is not a bad thing. But now it takes just too much time to develop these product compared to the consumer market. Could it be possible to shorten the time needed to develop these devices for medical use and create a health environment at the same technological level as our daily life. We all would profit from that, as they can make these devices faster, more will come to market and a better health environment would be created.

So to round it up, if we could develop medical devices in a shorter time period, we would accelerate Health Tech and indirectly create a healthier environment.

**Definition of the problem/ research question**

How can we shorten the time needed to develop medical devices and by doing so accelerate Health Tech?

**Objectives**

Build an online platform with all the tools and info needed to accelerate Health Tech. Illustrate the steps you have to take starting from an idea going to a finished product. Explain for all these steps what they mean and what to look out for.

Add a tool to the platform which allows users to find the right open source hardware. Include a filter function so the user can isolate the boards that accommodate best to his project. For every open source board in the search tool there has to be a clarifying summarization of the licenses they work with.

Create an example on how users can make a prototype app in an easy and fast way. Include this example to the portal and let users add their own examples. Add info on how and when they should request a patent keeping in mind they are working with open source hardware. And include the steps on how to get a Ce marking for their medical device.

All these objectives should be accomplished keeping in mind that the time needed to develop a medical device needs to be reduced. So make sure the portal is logically organized and all the info is short but clear.

**Methods and materials**

To build the portal I will be using google sites. There are other similar tools do make a website like Weebly, WordPress, etc. They all work with the same principle, make a website fast and easy without losing any time on writing code like HTML or PHP. The advantage of a google site compared to its competitors are the wide range of apps and tools the visualize info on the website. The compatibility with google docs and easy changeable administrator rights are also better than its competitors.

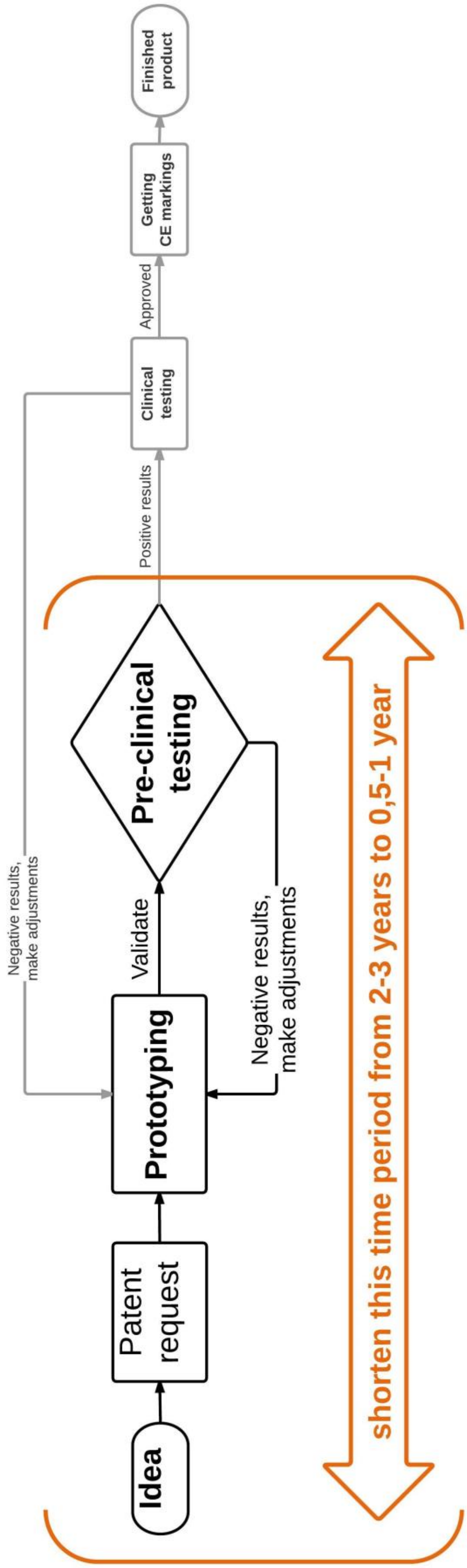
To make the example android app I will be using MIT App Inventor. There are even more competitors out there than for google sites. A few examples are Android Studio, AppMakr, Pocket Code, etc. I could not test all of these tools and software in order to decide with one to use so I followed the suggestion of my promotor on using App Inventor. The big advantages are the drag and drop system and easy testing on a Android device without having to install the app time after time.

## 8.2 Open source hardware available on the online portal

These are the most common open source hardware boards on the market. In this table not all of the characteristics are shown, for the full list of connections go to the portal using this link: https://sites.google.com/site/prototypebordjes/choose-a-board

| Name | Type | Purpose | Computing | Memory | Clock speed (MHz) |
|---|---|---|---|---|---|
| Raspberry Pi model B+ | Single-board computer | High-end, Stand-alone | ARM1176JZF-S Core ARM®11 | 512 MB SDRAM | 700 |
| Raspberry Pi model B | Single-board computer | High-end, Stand-alone | ARM1176JZF-S Core ARM®11 | 512 MB SDRAM | 700 |
| Beaglebone Black | Single-board computer | High-end, Stand-alone | AM335x ARM® Cortex-A8 | 512 MB DDR3 | 1000 |
| Banana Pi | Single-board computer | High-end, Stand-alone | Dual Core ARM® Cortex-A7 | 1 GB DDR3 | 1000 |
| Arduino Uno | microcontroller board | Stand-alone | ATmega328 | | 16 |
| Arduino Nano | microcontroller board | Stand-alone, Portable | ATmega168 or ATmega328 | | 16 |
| LillyPad Arduino USB | microcontroller board | Portable | ATmega32u4 | | 8 |
| Arduino Yún | Single-board computer | Stand-alone | ATmega32u4 and Atheros AR9331 | | 400 |
| UDOO Dual Basic | Single-board computer | High-end, Stand-alone | Dual Core ARM® Cortex-A9 and Atmel SAM3X8E ARM® Cortex-M3 | 1 GB DDR3 | 1000 |
| UDOO Dual | Single-board computer | High-end, Stand-alone | Dual Core ARM® Cortex-A9 and Atmel SAM3X8E ARM® Cortex-M3 | 1 GB DDR3 | 1000 |
| UDOO Quad | Single-board computer | High-end, Stand-alone | Quad Core ARM® Cortex-A9 and Atmel SAM3X8E ARM® Cortex-M3 | 1 GB DDR3 | 1000 |
| PandaBoard | Single-board computer | High-end, Stand-alone | Dual-core ARM® Cortex-A9 MPCore | 1 GB low power DDR2 | 1000 |
| WandBoard Solo | Single-board computer | Stand-alone | Freescale i.MX6 Solo Cortex-A9 Single core | 512 MB DDR3 | 1000 |

## 8.3 Step by step process starting from an idea going to a finished product



References


[1] S. T. Ham, "Mapping the Medical Device Development Process," California Polytechnic State University, 4 june 2010. [Online]. Available: http://digitalcommons.calpoly.edu/itsp/11/. [Accessed 14 April 2015].

[2] Europa, „Product requirements CE marking," Europa, May 2015. [Online]. Available: http://europa.eu/youreurope/business/product/ce-mark/index_en.htm. [Geopend May 2015].

[3] "Open source hardware association," Open source hardware association, 14 May 2015. [Online]. Available: http://www.oshwa.org/definition/. [Accessed 14 May 2015].

[4] "raspberrypi.org," Raspberry Pi foundation, 18 February 2015. [Online]. Available: https://www.raspberrypi.org/five-million-sold/. [Accessed 17 May 2015].

[5] Arduino, „Arduino," Arduino, 2015. [Online]. Available: http://www.arduino.cc/en/Main/ArduinoBoardUno. [Geopend 2015].

[6] „Creative Commons licenses," Creative Commons, 2015. [Online]. Available: http://creativecommons.org/licenses/. [Geopend 2015].

[7] Europa, „CE marking - manufacturers," European Comission, 2015. [Online]. Available: http://ec.europa.eu/growth/single-market/ce-marking/manufacturers/notified-body/index_en.htm?filter=15. [Geopend March 2015].

[8] „European comission," 14 July 1993. [Online]. Available: http://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:31993L0042&from=EN. [Geopend 2015].

[9] „European comission," June 2010. [Online]. Available: http://ec.europa.eu/health/medical-devices/files/meddev/2_4_1_rev_9_classification_en.pdf. [Geopend 2015].

[10] European comission, 27 May 2015. [Online]. Available: http://ec.europa.eu/growth/sectors/medical-devices/regulatory-framework/revision/index_en.htm. [Geopend May 2015].

[11] „clinical trials," European comission, May 2015. [Online]. Available: http://ec.europa.eu/health/human-use/clinical-trials/index_en.htm. [Geopend May 2015].

[12] Felicia, MIT App Inventor, 18 May 2015. [Online]. Available: http://appinventor.mit.edu/explore/. [Geopend May 2015].

[13] Coolblue, „Coolblue," Polar, 2015. [Online]. Available: http://www.hartslagmetercenter.be/product/138682/polar-wearlink-bluetooth-hartslagsensor-m-xxl.html?_ga=1.188062112.963990995.1430394309. [Geopend 2015].

[14] „Bloeddrukmeter webshop," Omron, 2015. [Online]. Available: http://www.bloeddrukmeterwebshop.be/omron-708-bt. [Geopend 2015].

[15] European comission, 06 May 2015. [Online]. Available: http://ec.europa.eu/growth/single-market/goods/building-blocks/notified-bodies/index_en.htm. [Geopend May 2015].

[16] „Clinical Trials," U.S. National Institutians of Health, December 2014. [Online]. Available: https://clinicaltrials.gov/ct2/about-studies/learn. [Geopend April 2015].

# Auteursrechtelijke overeenkomst

Ik/wij verlenen het wereldwijde auteursrecht voor de ingediende eindverhandeling:
**Accelerating Health Tech**

Richting: **master in de industriële wetenschappen: elektronica-ICT**
Jaar: **2015**

in alle mogelijke mediaformaten, - bestaande en in de toekomst te ontwikkelen - , aan de Universiteit Hasselt.

Niet tegenstaand deze toekenning van het auteursrecht aan de Universiteit Hasselt behoud ik als auteur het recht om de eindverhandeling, - in zijn geheel of gedeeltelijk -, vrij te reproduceren, (her)publiceren of distribueren zonder de toelating te moeten verkrijgen van de Universiteit Hasselt.

Ik bevestig dat de eindverhandeling mijn origineel werk is, en dat ik het recht heb om de rechten te verlenen die in deze overeenkomst worden beschreven. Ik verklaar tevens dat de eindverhandeling, naar mijn weten, het auteursrecht van anderen niet overtreedt.

Ik verklaar tevens dat ik voor het materiaal in de eindverhandeling dat beschermd wordt door het auteursrecht, de nodige toelatingen heb verkregen zodat ik deze ook aan de Universiteit Hasselt kan overdragen en dat dit duidelijk in de tekst en inhoud van de eindverhandeling werd genotificeerd.

Universiteit Hasselt zal mij als auteur(s) van de eindverhandeling identificeren en zal geen wijzigingen aanbrengen aan de eindverhandeling, uitgezonderd deze toegelaten door deze overeenkomst.

Voor akkoord,

**Oosterbos, Bram**

Datum: **1/06/2015**