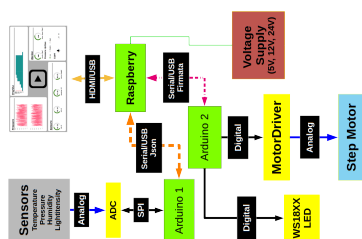


# Medical Hardware GUI



## Contents

Contents	1
1 Frontend	2
2 Backend	4
3 Electronics	6
4 Technical and Budget Planing	7
4.1 Time Requirements . . . . .	7
4.2 Software Requirements . . . . .	7
4.3 Electronic Requirements . . . . .	8

# 1 Frontend

- Angular  $\Rightarrow$  AngularClient, npm, ...
- Style  $\Rightarrow$  Material.IO, GraphJs, W3, SCSS
- TouchApi  $\Rightarrow$  HammerJs
- CommunicationApi  $\Rightarrow$  Socket.IO
- KeyBoard  $\Rightarrow$  npm-keyboard
- Deployment  $\Rightarrow$  Electron packager
- Integration  $\Rightarrow$  xTerminal
- Home Screen  $\Rightarrow$  NAVigation to Access to Software components
- Create Profile Screen
  - 1.Screen: Pop Up Window to name the Profile
  - 2.Screen: create single Elements and add them to Profile
  - 3.Screen: Add Element or save and run your Profile
- Machine Activation Screen
  - Pressure section: realtime Values 2 sensors
  - Profile section: progress with vertical red bar.
  - Device section: motor speed, cardio, frequency, pressure option, the light on/off
  - Sensor section: 2x temperature, humidity, delta pressure
  - Play/Pause section: start/stop the motor

SELECT  
CARDIO  
PROFILE



NEW CARDIO  
PROFILE



MANUAL  
CONTROL

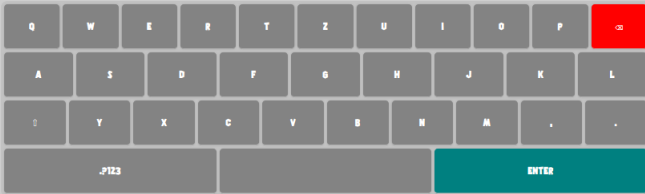


PREVIOUS  
EXPERIMENTS

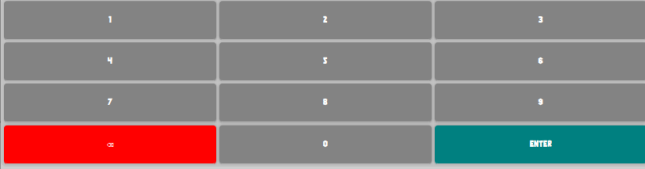
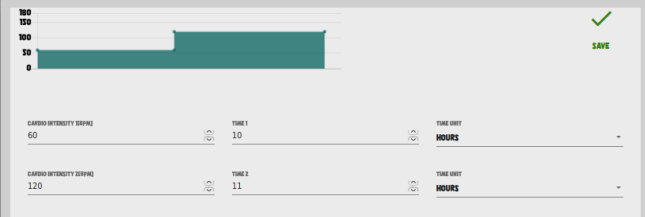


ADD MODULE

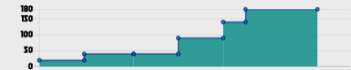
NEW NAME



ADD MODULE

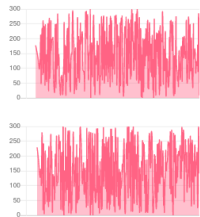


ADD MODULE



3

Pressure



Profile



Device



Pressure Option

☒ Low ☐ Middle ☐ High

Light



Sensors



## 2 Backend

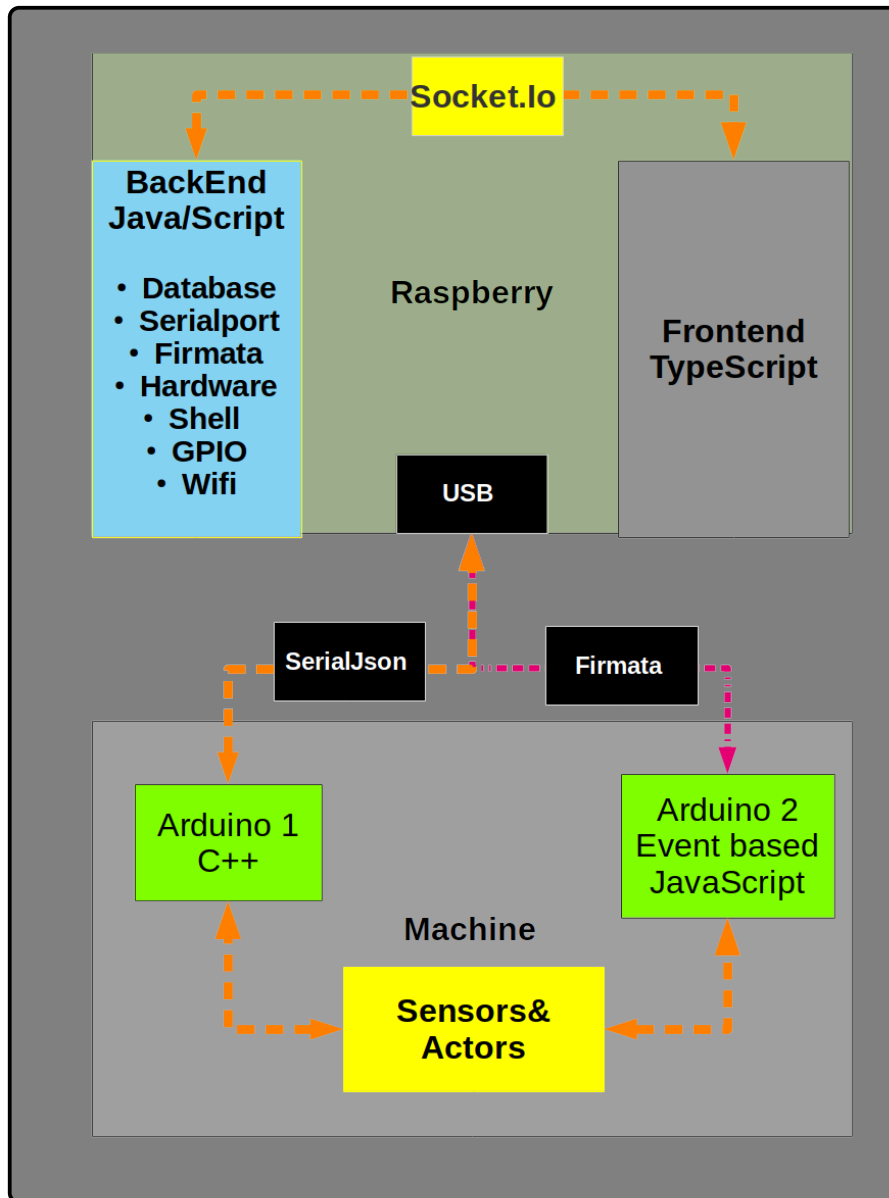
- Server  $\Rightarrow$  nodeJs
- Communication  $\Rightarrow$  Socket.IO
- DB  $\Rightarrow$  postgresql
- Email  $\Rightarrow$  nodeMailer
- Arduino 1  $\Rightarrow$  firmata slave code using javascript to communicate
- Arduino 2  $\Rightarrow$  C++ code sharing jsonData(Sensor values) over Serialport
- Integration  $\Rightarrow$  Linux Service

### Connections

- Raspberry  $\Rightarrow$  Arduino 1  $\Rightarrow$  Firmata over USB (johnny-five)
- Raspberry  $\Rightarrow$  Arduino 2  $\Rightarrow$  Serial over USB Baudrate: 51800
- Arduino 2(C++)  $\Rightarrow$  ADC  $\Rightarrow$  SPI (spi.h)

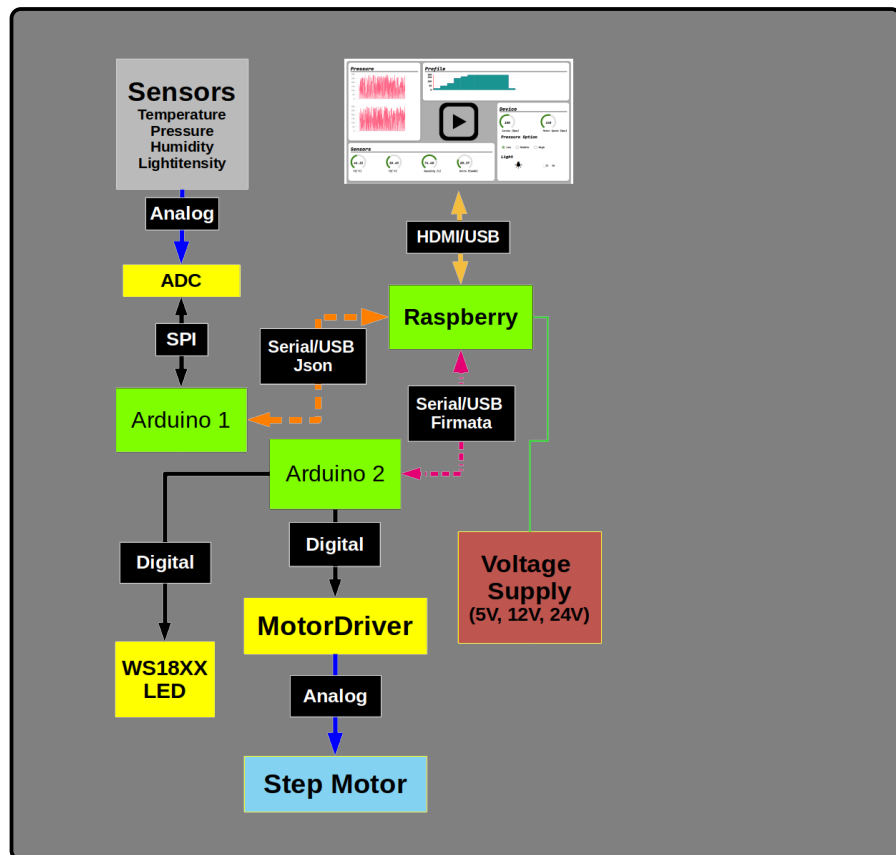
### Hardware:

- Interface: 10" Touchdisplay
- Peripheral: 8 channel 32 bit ADC, 7.2 A Stepper Driver
- Sensors: 2 Temperature, 2 absolut Pressure, Humidity
- Actors: WS28XXX Led stripe, Step Motor



### 3 Electronics

- Electrical Supply (12V/60W, 5V/10W )
- Electromagnetic Protection of TouchScreen: high Current of stepper motor
- grounded Supply



## 4 Technical and Budget Planing

### 4.1 Time Requirements

2 monthes for Software and Hardware Development 01.08-01.10.2020 were planed. At the end it took me 3 and a half monthes to finish and test it.

### 4.2 Software Requirements

- Realtime Communication Hardware Touchdisplay
- Database of Profiles
- Database of Logdata
- Deploy Software to bin file
- Linux run on boot
- GUI

Company logo

display sensor values in Realtime

create intensity Profile and save it to database

read Profile from Database

run machine with time sensitive Profile

run machine with manual control

send logfiles to a specified email adress

## 4.3 Electronic Requirements

- Realtime SensorValues
- robust
- elegant
- compact
- silent motor control