

Projekt Gesundheitstechnik

Thermographie in der Medizintechnik

Achim Klenk, Falko Kristen, Arthur Schneider, Aaron Schultz
WiSe 2022/23



**Hochschule für Technik
und Wirtschaft Berlin**

University of Applied Sciences

Inhaltsverzeichnis

- Konzeptidee
- Einsatzgebiete
- Aufbau und Funktionsweise
 - Konstruktion
 - Elektronik
 - Software
- Demonstration
- Projektverlauf und Vision

Konzeptidee

- Thermographie – die Wärmebildkamera
- Handheld Device für
 - klinische Anwendungen
 - Großveranstaltungen
 - Eindämmung Corona-Pandemie

Marktanalyse

- Vergleichbare Marktprodukte



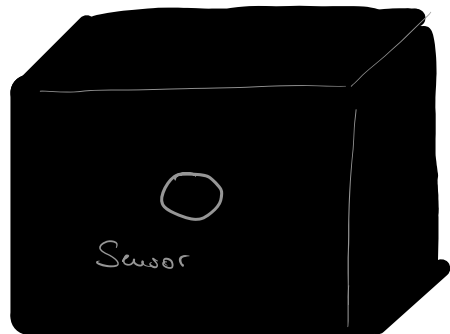
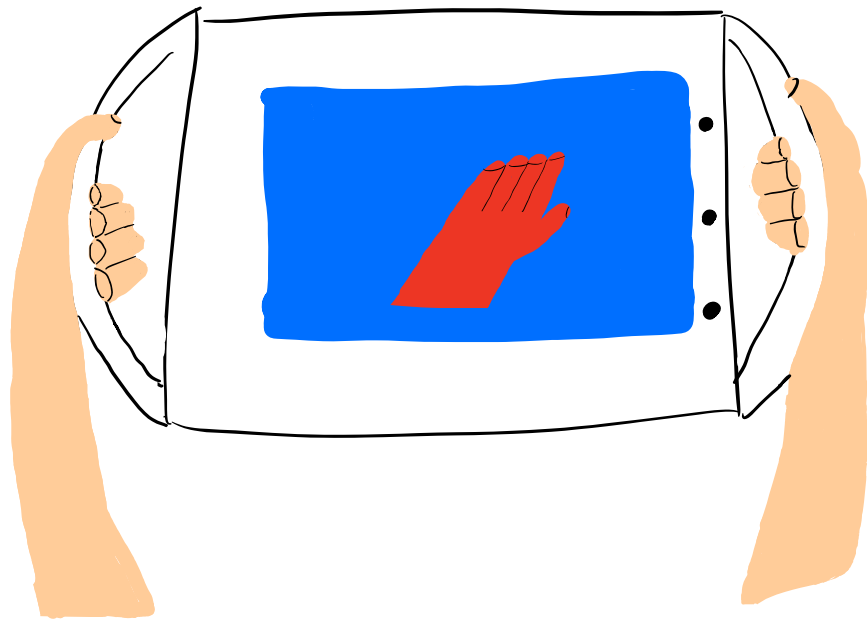
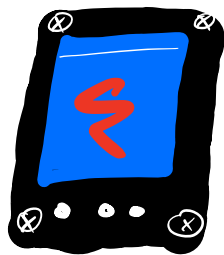
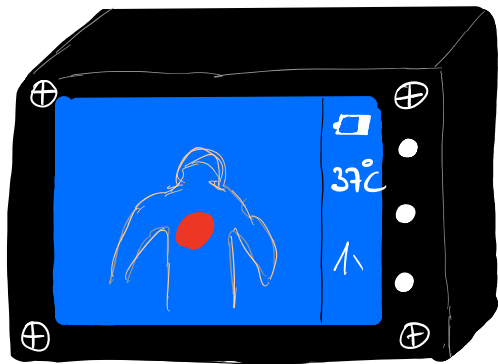
<https://ais.badische-zeitung.de/piece/0b/f1/bb/7e/200391550-h-720.jpg>



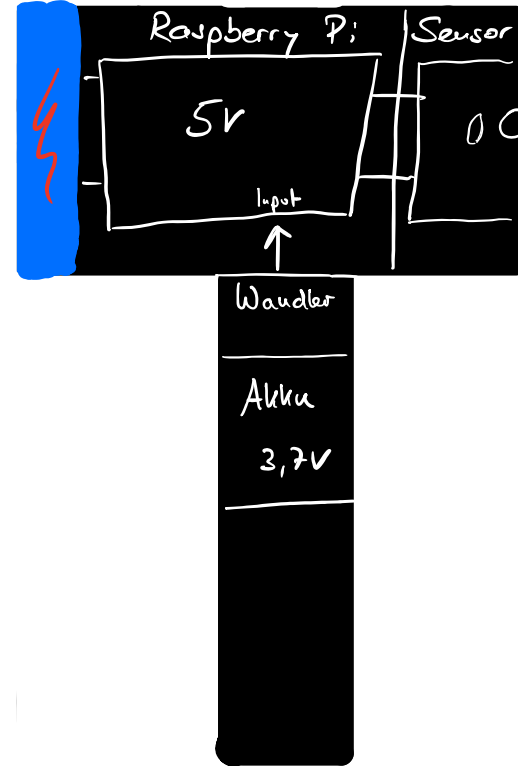
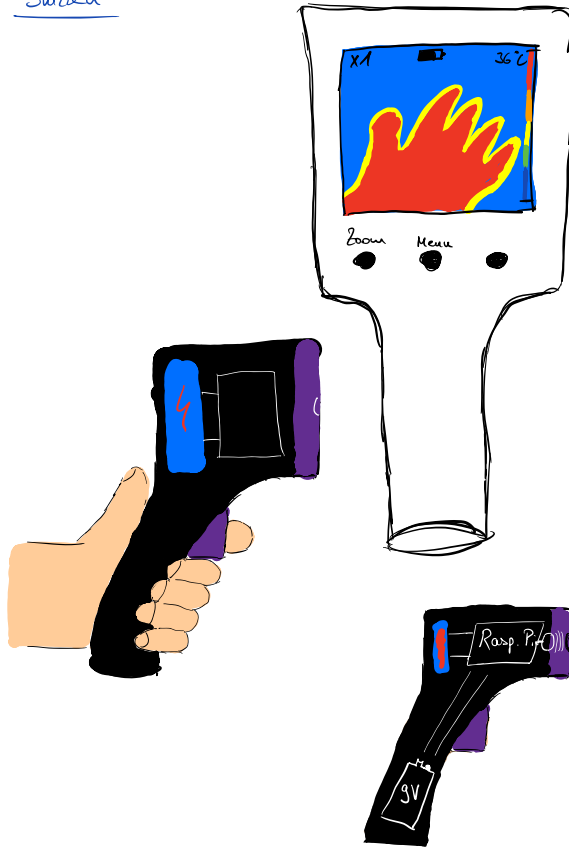
https://www.elektro.net/file/format/95413/textPictureFull/30e0d2/CA1900_W%C3%A4rmebildkamera.jpg



<https://cdn02.plentymarkets.com/jd5w7us67fek/item/images/6661/full/eak-5605-1614683616.jpg>



Screen

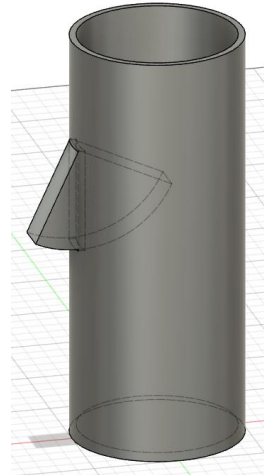
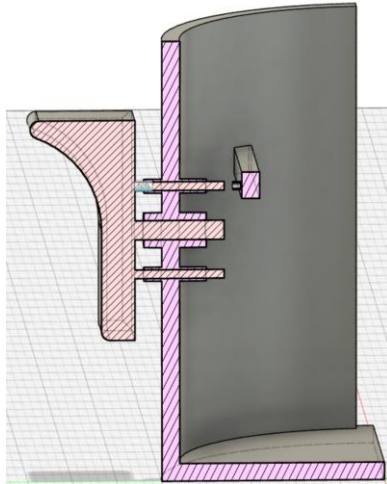


Aufbau und Funktionsweise - Konstruktion

- Anforderungen
 - Dimension eines Handhelds
 - Modulares, ergonomisches Design
 - Beide Sensoren sollen verbaut sein
 - Physischer Button

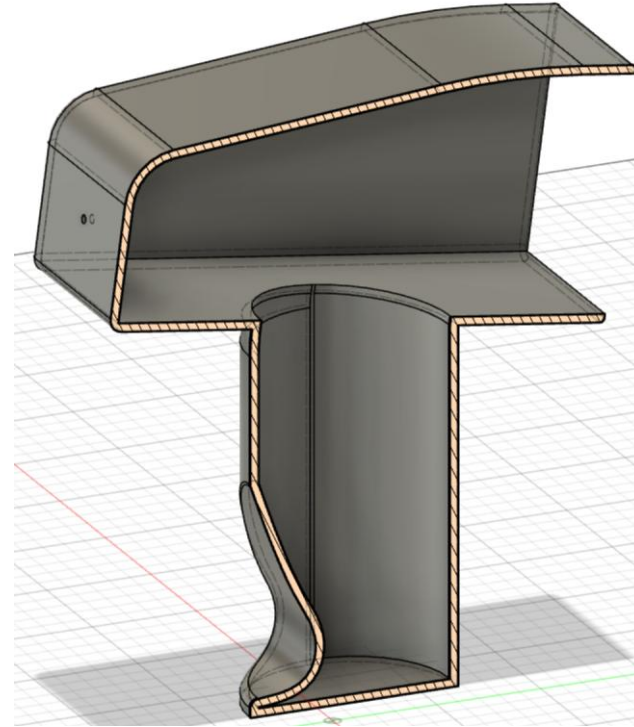
Vorbereitung für Konstruktion

- Bauteilrecherche (Größe der einzelnen Bauteile)
- Konzept über Platzierung der Bauteile im Gehäuse
- ungefähre Dimensionen des Gerätes festlegen
- Positionierung und Funktionsweise des Buttons bestimmen



Erste Konstruktionsidee v.0

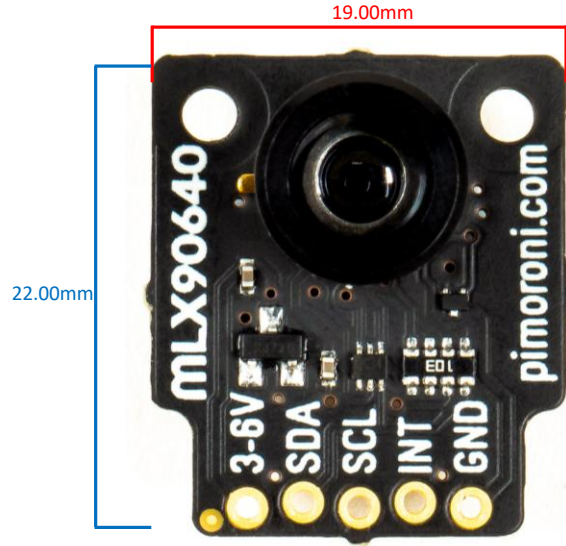
- Entstanden ohne jegliche Hilfe
- Zwei – Hälften – Design
- Ergonomie und Ästhetik im Vordergrund



Aufbau und Funktionsweise - Hardware

- Wahl der Sensoren
- Kabelloses Gerät
- Zwei Budget Modelle der Infrarot Sensoren

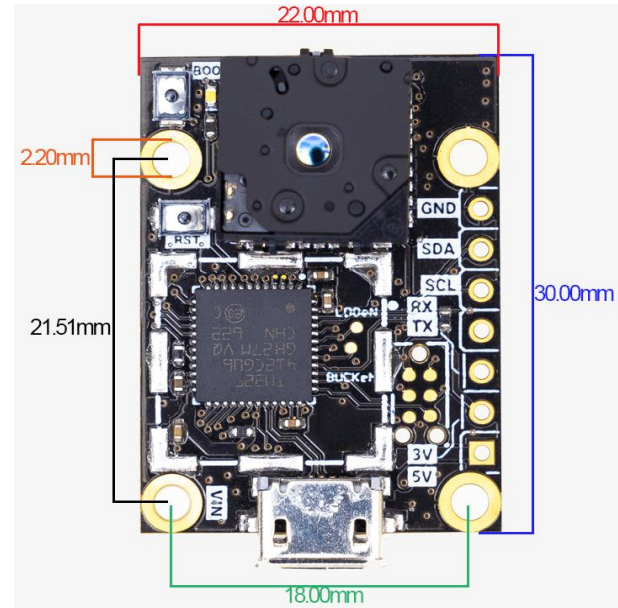
Low budget



MLX 90640 (32 x 24 Pixel)

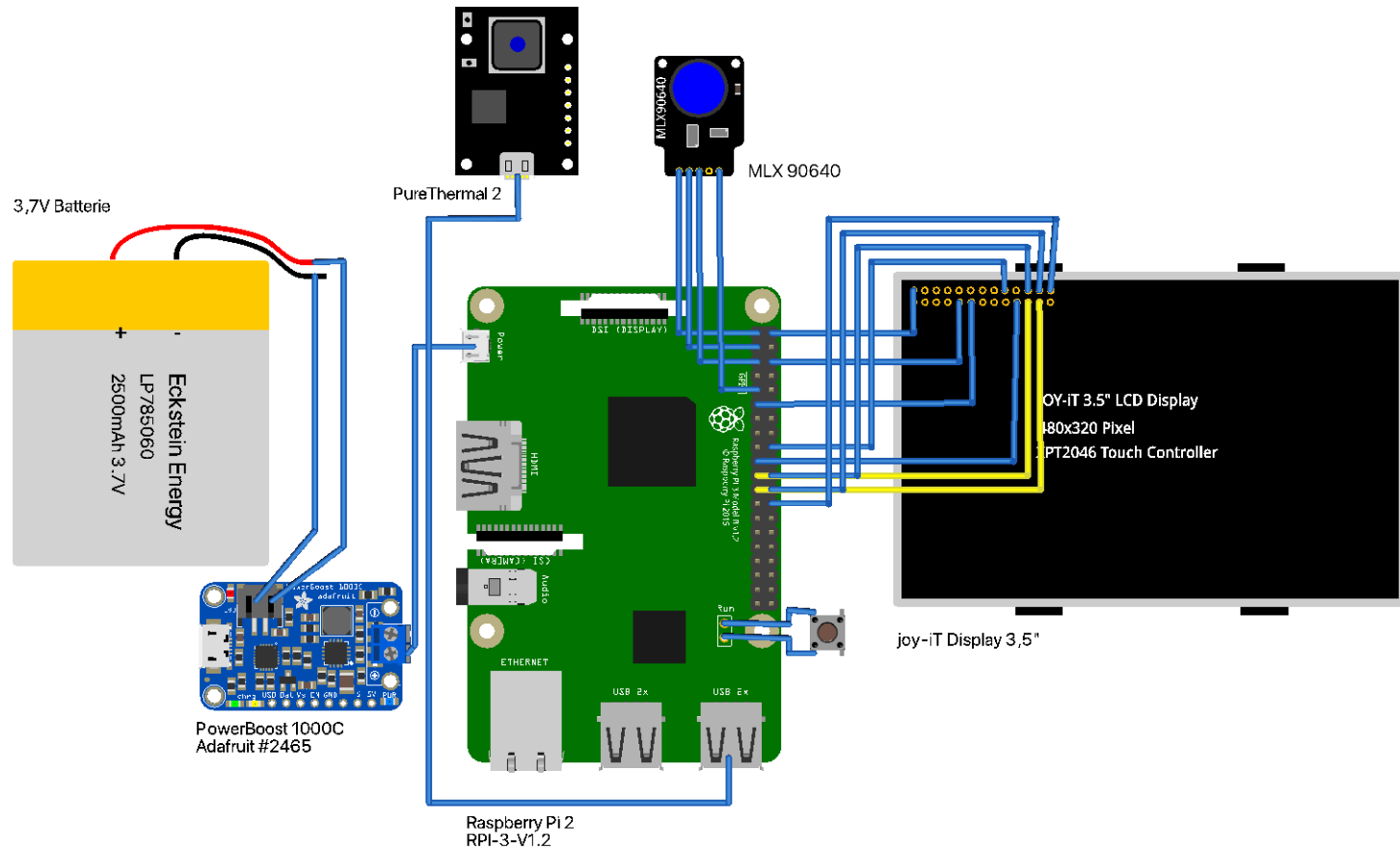
https://cdn.shopify.com/s/files/1/0174/1800/products/mlx90640-2_1500x1500.jpg?v=1656081430

High budget



FLIR Lepton 3.5 (160 x 120 Pixel)

<https://groupgets.com/manufacturers/getlab/products/purethermal-2-flir-lepton-smart-i-o-module>



GUI (Graphical User Interface)

Voraussetzungen

- Sprache: Python
- Ausführbar auf:
 - Raspberry Pi (Linux) mit FLIR Lepton 3.5
 - 160 x 120 Pixel
 - Raspberry Pi (Linux) mit MLX 90640
 - 32 x 24 Pixel
 - Windows mit Webcam
 - Mac OS mit Webcam

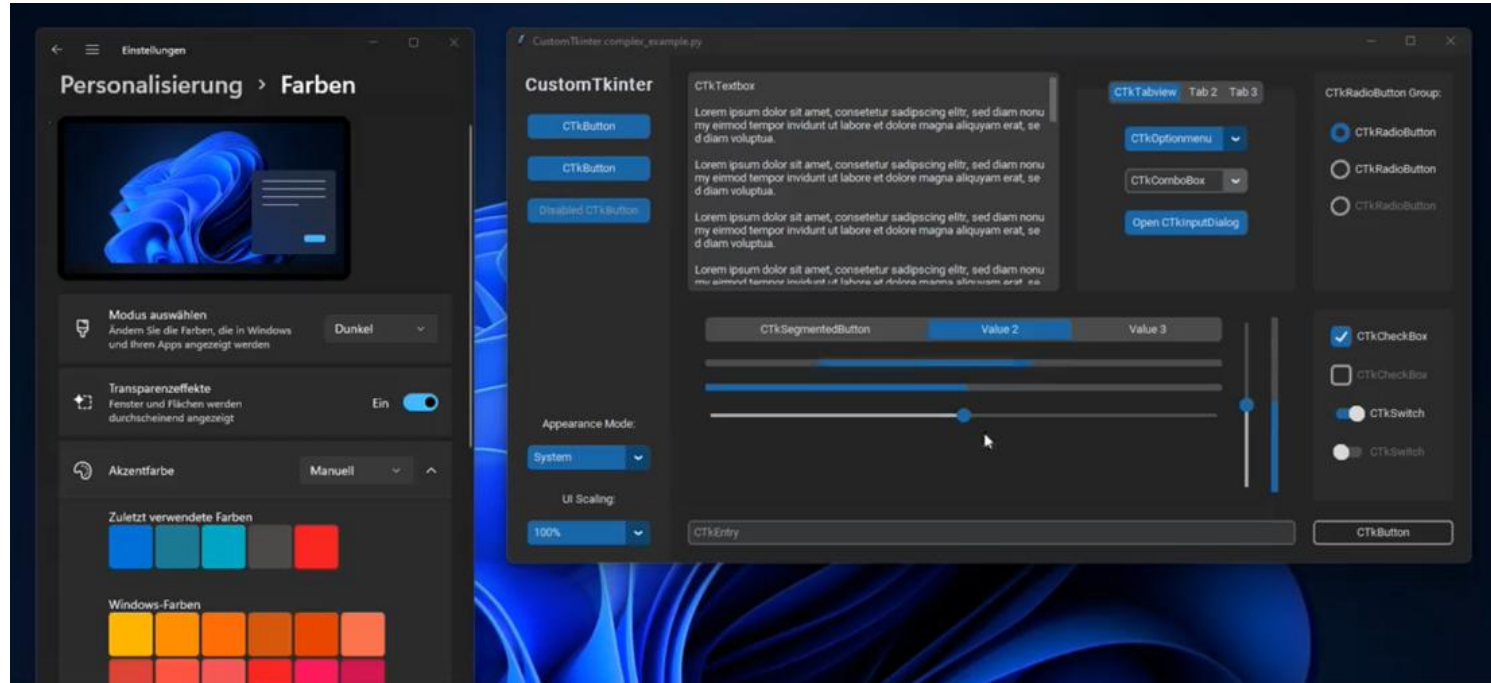
Voraussetzungen

- Sprache: Python
- Ausführbar auf:
 - Raspberry Pi (Linux) mit FLIR Lepton 3.5
 - 160 x 120 Pixel
 - Raspberry Pi (Linux) mit MLX 90640
 - 32 x 24 Pixel
 - Windows mit Webcam
 - Mac OS mit Webcam

Bibliotheken:

```
1  contourpy==1.0.6
2  customtkinter==4.6.3
3  cyclers==0.11.0
4  darkdetect==0.8.0
5  fonttools==4.38.0
6  kiwisolver==1.4.4
7  matplotlib==3.6.2
8  numpy==1.24.1
9  opencv-python==4.7.0.68
10 packaging==23.0
11 Pillow==9.4.0
12 pyparsing==3.0.9
13 python-dateutil==2.8.2
14 six==1.16.0
15 tk==0.1.0
```

CustomTKinter by Tom Schimansky



Funktionen & Features (GUI)

- Automatische Sensorerkennung
- Modi: Map Modus, Fieber Modus, Visier Modus
- Modus Info
- Screenshot
- Freeze Funktion
- 2x Zoom
- Live Min-, Mid-, Max-Temperatur mit Seitenleiste
- Colormaps (openCV) - Bone, Turbo, Plasma, Hot, Cool
- Fotogalerie - Ansicht, Scrollen, Löschen
- Abschaltfunktion

Überblick



VS Code Counter v3.2.0

Kentaro Ushiyama | 271,986 | ★★★★★ (38)

Count lines of code in many programming languages.

[Disable](#) [Uninstall](#) ⚙️

This extension is enabled globally.

Languages

language	files	code	comment	blank	total
Python	11	1,675	264	606	2,545

Directories

path	files	code	comment	blank	total
.	11	1,675	264	606	2,545

```
GESUNDHEITSTECHNIK_THERMOGRAPH
├── assets
├── Hardware
├── icons
├── other
├── Snapshots
├── Thermograph_OS
├── __pycache__
├── modules
│   ├── __pycache__
│   ├── design_states.py
│   ├── functions_main_window.py
│   ├── memory_states.py
│   ├── window_filter_info.py
│   ├── window_filter.py
│   ├── window_gallery.py
│   ├── window_main.py
│   ├── window_menu.py
│   ├── window_settings_cmap.py
│   ├── window_settings.py
│   └── window_shutdown.py
├── purethermal1_uvc_capture_master
├── .gitignore
├── main_os.py
├── start_client.sh
├── start_client.sh.save
├── start_counter.txt
├── README.md
└── requirements.txt
```

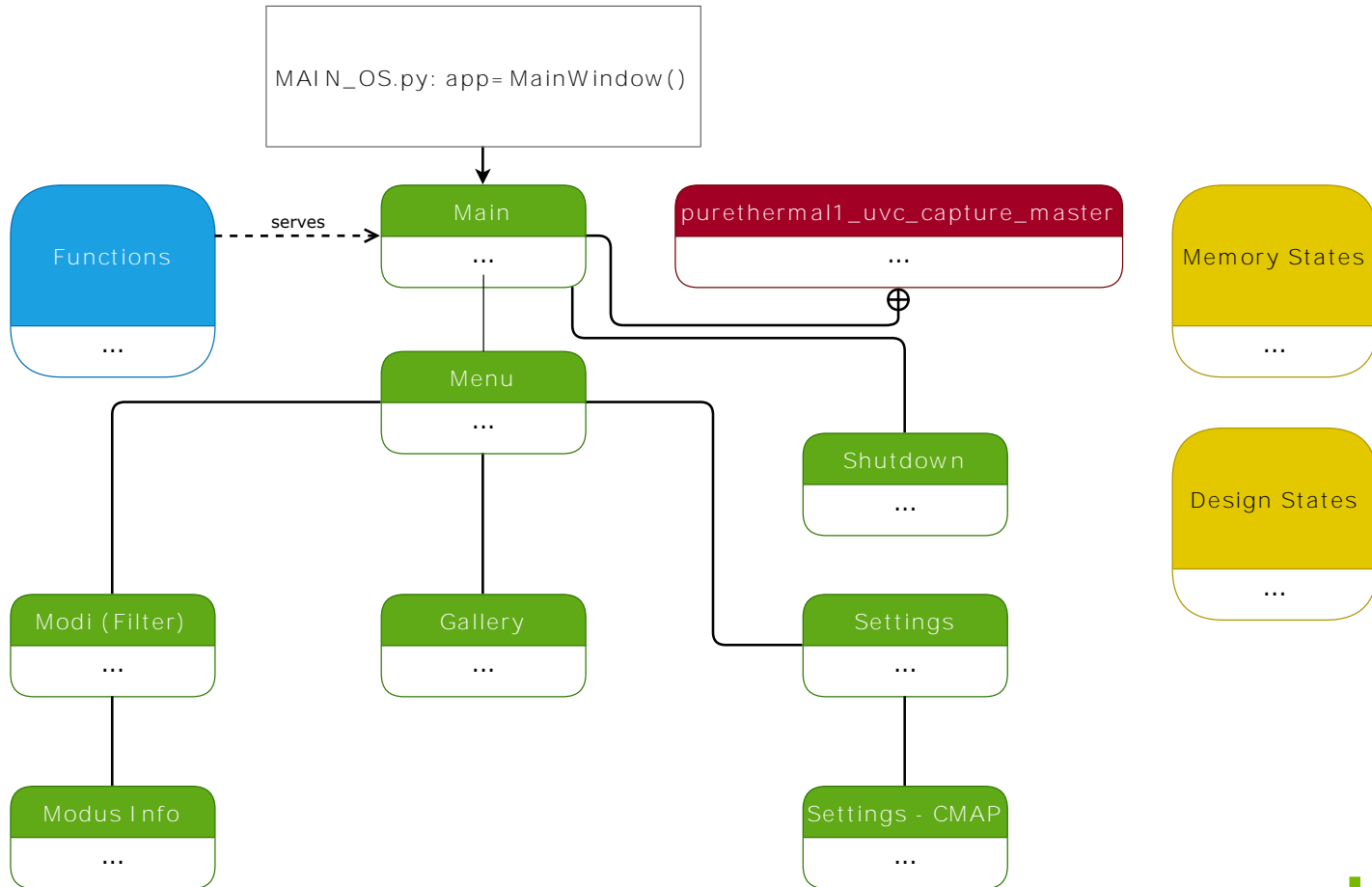
Aufbau

- Fenster:
 - Main
 - Shutdown
 - Menu
 - Modi (Filter)
 - Modus Info
 - Gallery
 - Settings
 - Heatmap Settings

- Speicherklassen:
 - Memory States (Setup)
 - Design States

- Funktionsklassen:
 - Functions





Aufbau von Main Fenster

```
##### SCRIPT IMPORTS #####
## extern library imports ##
## intern module imports ##

##### MAIN WINDOW #####
class MainWindow(customtkinter.CTk):
    def __init__(self):
        super().__init__()

        ##### WINDOW MAIN (self) CONFIGURATIONS #####
        ##### GRID LAYOUT #####
        ##### WIDGETS MAIN #####
        ## Camera Label ##
        ## Zoom Label ##
        ## Heatmap Canvas ##
        ## PlusButton ##
        ## Minus Button ##
        ## Screenshot Button ##
        ## Menu Button ##
        ## Exit Button ##

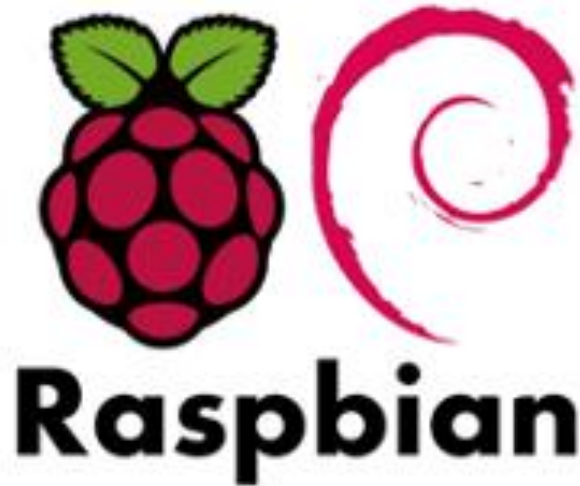
        ##### MAIN VIDEO STREAM NORMAL_CAMERA #####
        ##### MAIN VIDEO STREAM FLIR #####
        ##### MAIN VIDEO STREAM MLX #####

    ### FUNCTION TO INITIALIZE SECONDARY WINDOWS ###
    def openMenu(self):
        menu = MenuWindow(self)

    def openShutdown(self):
        shutdown = ShutdownWindow(self)
```

Betriebssystem

- Raspberry Pi
OS 32-Bit



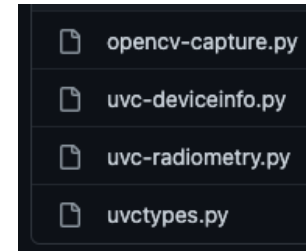
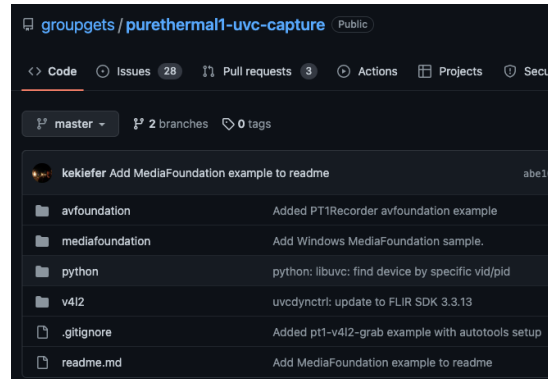
https://cdn.xingosoftware.com/elektor/images/fetch/dpr_1/https%3A%2F%2Fwww.elektormagazine.de%2Fassets%2Fupload%2Fimages%2F11%2F20180327115958_raspbian-logo.png

Sensorimplementierung

- Sensor MLX 90640
 - Schnittstelle: I2C
 - Bibliothek: adafruit_mlx90640
- Sensor Flir Pure Thermal 2
 - Schnittstelle: Micro-USB
 - Bibliothek: uvctypes, openCV

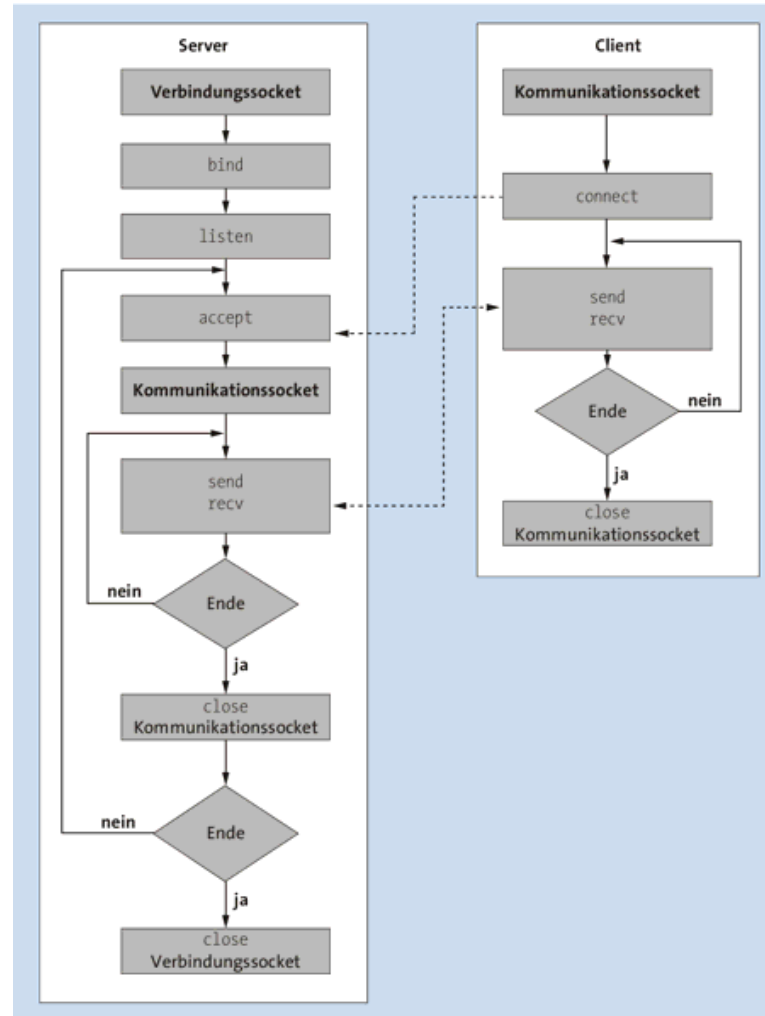
```
## MLX Setup ##
import board, busio
import adafruit_mlx90640
i2c = busio.I2C(board.SCL, board.SDA, frequency=800000)
mlx = adafruit_mlx90640.MLX90640(i2c)
mlx.refresh_rate = adafruit_mlx90640.RefreshRate.REFRESH_2_HZ # sets refresh rate
frame2d = np.zeros((24*32))

mlx.getFrame(frame2d)
img = np.reshape(frame2d, (24,32))
```



Socketstream

- Austausch von Daten zwischen mehreren Programmen
- Socket besteht aus Client (Sender) und Server (Empfänger)
- separate Terminals, um py-Dateien simultan auszuführen



BASH - Skript

- Textdatei mit Endung „.sh“
- Genutzt, um Befehle und Kommandos auszuführen
- Stream Rohdaten Flir Sensor
 - Zweitterminal Gnome



The screenshot shows a terminal window with the following content:

```
teamthermo@raspberrypi:~/Desktop/gesundheitstechnik_thermograph/Thermograph_OS $ sudo nano start_client.sh
```

Below the terminal window is a window titled "teamthermo@raspberrypi: ~/Desktop/gesundheitstechnik_thermograph/Thermograph_OS". It contains a menu bar with "Datei", "Bearbeiten", "Reiter", and "Hilfe". Below the menu bar is a status bar that says "GNU nano 5.4" and "start_client.sh". The main area of the window shows the command prompt "#!/bin/sh" and the command "gnome-terminal -x "/home/teamthermo/Desktop/gesundheitstechnik_thermograph/Thermograph_OS/purethermal1_uvc_capture_master/python/uvr_radiometry.py" &".

Autostart

- *.desktop* Datei
- Start der GUI bei Bootup des Raspberry Pi's

```
teamthermo@raspberrypi: ~  
Datei Bearbeiten Reiter Hilfe  
GNU nano 5.4 /etc/xdg/autostart/display.desktop  
[desktop Entry]  
Name=start_thermograph.sh  
Exec=/usr/bin/python3 /home/teamthermo/Desktop/gesundheitstechnik_thermograph/Thermograph_OS/main_os.py
```

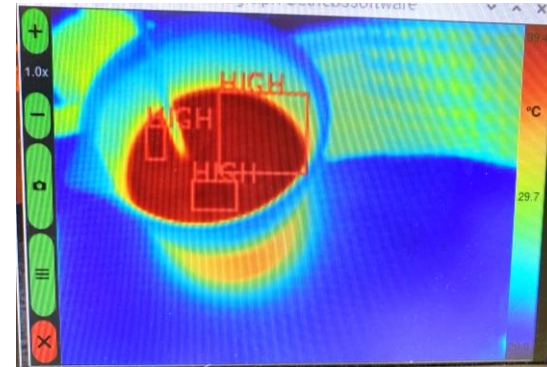
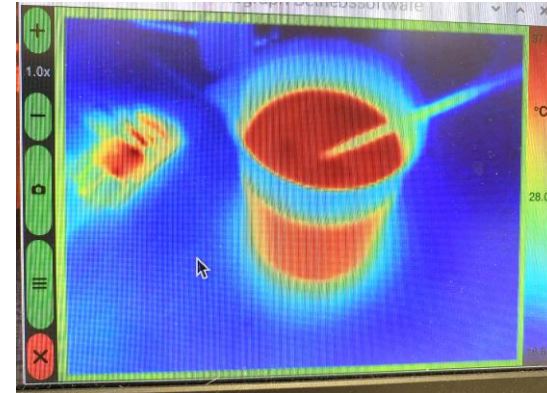
Funktion Heatmap

- Pseudokolorierung anhand der ausgelesenen Temperaturwerte

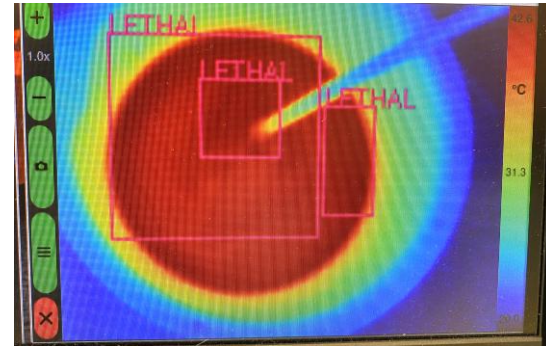
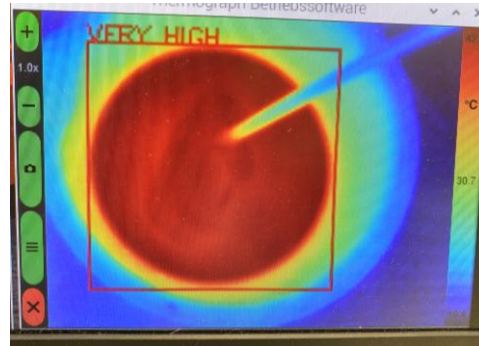
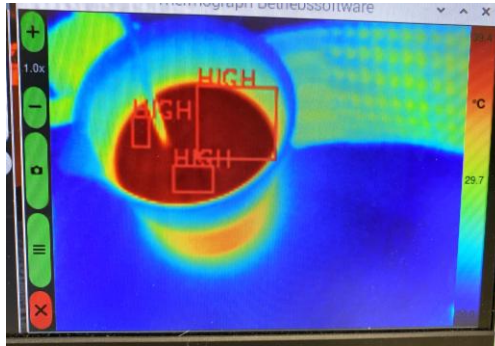
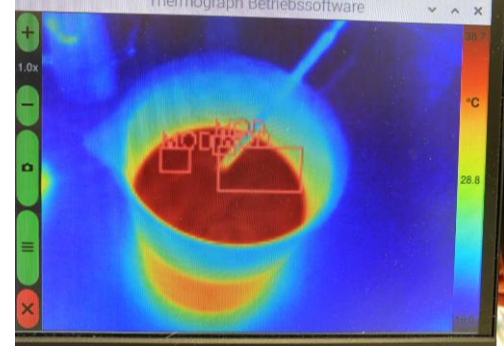
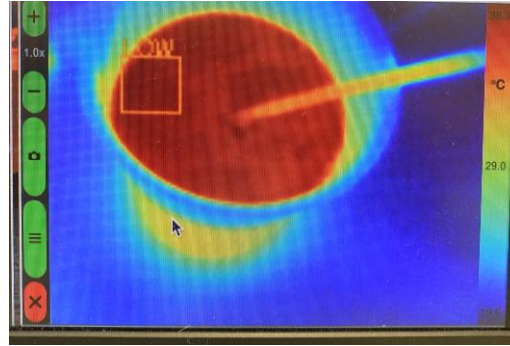
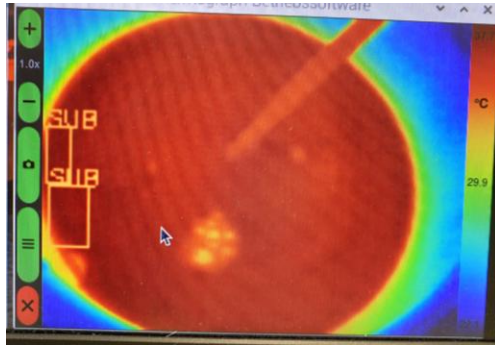


Funktion Fiebererkennung

36,5°C - 37,4°C:	Normal-Temperatur
37,5°C - 38,0°C:	Subfebrile Temperatur
38,1°C - 38,5°C:	Leichtes Fieber
38,6°C - 39,0°C:	Mäßiges Fieber
39,1°C - 39,9°C:	Hohes Fieber
40,0°C - 42,0°C:	Sehr hohes Fieber
> 42,0°C	Tödlich

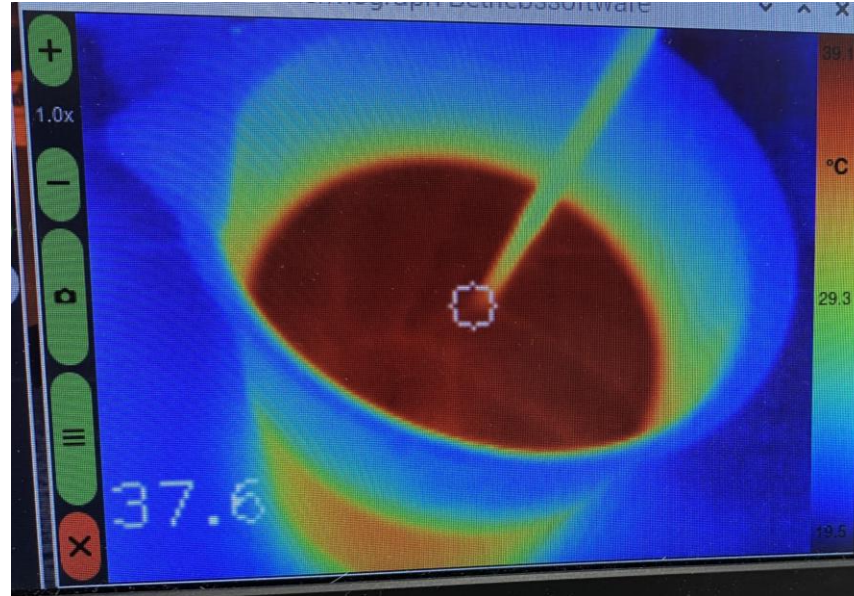


Fiebererkennung

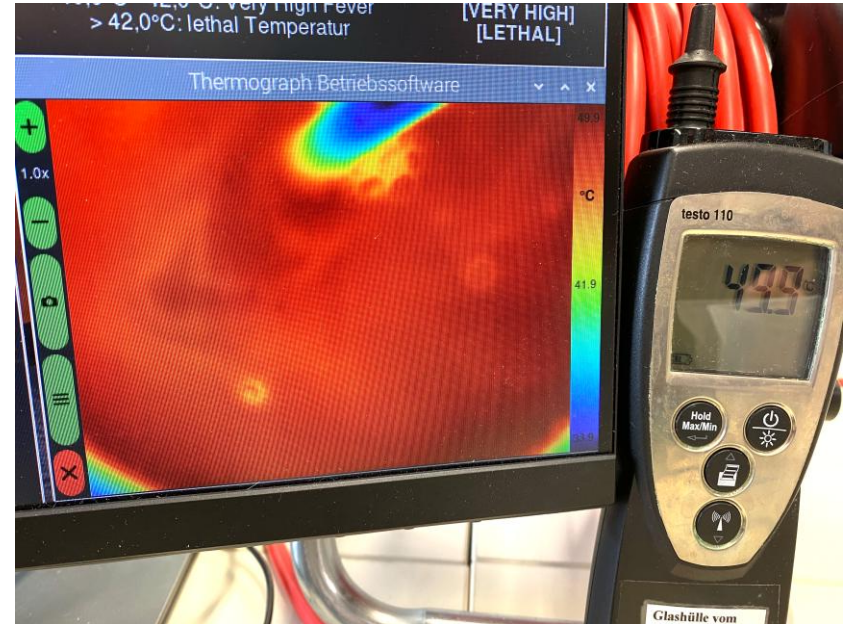
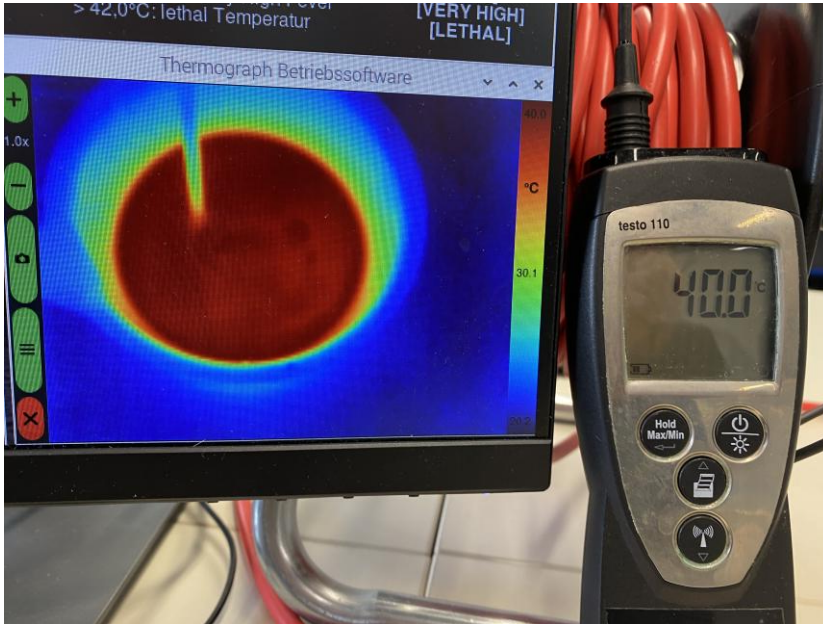


Funktion Target

- Mittelpunkt des Bildes ist Temperaturmesspunkt



Zuverlässigkeit Messungen



Demonstration

Projektverlauf

- Anfänge
 - Organisation
 - Aufgabenaufteilung
- Misserfolge
 - Sensoren
 - Rohdaten
 - Heatmap
 - Fehlendes Konstruktionswissen

Vision

- Display mit HDMI Anschluss
- Gehäuseverbesserung
- Programmierung in *C/C++*
- Usability
 - Akkustand
 - Spracheinstellung
 - Lautsprecher für akustische Signale
 - Sensorwechsel per Einstellungen
 - Farbeinstellungen

Quellenverzeichnis

- <https://ais.badische-zeitung.de/piece/0b/f1/bb/7e/200391550-h-720.jpg>
- https://www.elektro.net/file/format/95413/textPictureFull/30e0d2/CA1900_W%C3%A4rmebildkamera.jpg
- <https://cdn02.plentymarkets.com/jd5w7us67fek/item/images/6661/full/eak-5605-1614683616.jpg>
- https://cdn.shopify.com/s/files/1/0174/1800/products/mlx90640-2_1500x1500.jpg?v=1656081430
- <https://groupgets.com/manufacturers/getlab/products/purethermal-2-flir-lepton-smart-i-o-module>
- <https://openbook.rheinwerk-verlag.de/python/bilder/client-server.png>
- https://cdn.xingosoftware.com/elektor/images/fetch/dpr_1/https%3A%2F%2Fwww.elektormagazine.de%2Fassets%2Fupload%2Fimages%2F11%2F20180327115958_raspbian-logo.png
- <https://staging-assets.raspberrypi.com/static/md-bfd602be71b2c1099b91877aed3b41f0.png>
- <https://app.diagrams.net/>
- <https://online.visual-paradigm.com/de/>
- <https://www.ernstlx.com/linux90script.html>
- <https://www.makeuseof.com/how-to-run-a-raspberry-pi-program-script-at-startup/>
- <https://realpython.com/python-sockets/>
- <https://www.autodesk.de/products/fusion-360/overview?term=1-YEAR&tab=subscription>
- <https://www.internisten-im-netz.de/krankheiten/fieber/temperatur-regulation-im-koerper.html#:~:text=37%2C5%C2%B0C%20%2D%2038,9%C2%B0C%3A%20Hohes%20Fieber>
- <https://account.htw-berlin.de/htwfont/index>
- <https://corporatedesign.htw-berlin.de/musterdokumente/powerpoint-vorlagen/>



Vielen Dank.

www.htw-berlin.de



**Hochschule für Technik
und Wirtschaft Berlin**

University of Applied Sciences



**Hochschule für Technik
und Wirtschaft Berlin**

University of Applied Sciences

www.htw-berlin.de