**Firmware (Feather / Circuitpython):**

MainLoop

Manager

Bluetooth

Magnetic\_Sensor

Light\_Sensor

SCD30\_Sensor

BatteryVoltage

* Auf BLE Verbindung warten
* BLE Message lesen
* BLE Message senden
* Messung starten / Resultate erhalten
* Init
* lesen
* Init
* lesen
* Init
* lesen
* Init
* Lesen
* Schreiben
* Auf Verbindung warten
* Init
* lesen
* busio.I2C lib
* adafruit\_scd30.SCD30 lib
* Analogio lib
* Digitalio lib

* Adafruit\_ble lib

* Analogio lib

logger

* Schreiben (global)

watchdog

* Init

Starte advertizement

Warte auf eine Verbindung vom Master

Stoppe advertizement

Lese Befehl von Master

Lese Daten von Sensoren aus

Sende Sensordaten an Master

Gehe in light sleep für 20s

**Master (RaspberryPI / Python):**

logger

Schreiben (global)

MainLoop

Bluetooth

* Adafruit\_ble lib

* Messungen auslösen / Ergebnisse lesen

Cache

* Lesen
* Schreiben
* löschen

SSL

* Messungen senden

Scanne nach advertizements

Verbinde auf nächsten adverzizender slave

Sende Lese-Befehl

Lese & speichere Antwort (Messung)

Schliesse Verbindung

Noch Advertizements vorhanden

Lese noch vorhandene Messungen vom Cache

Bereits auf alle

Advert. verbunden

Verbinde zum Server

Cache neue Messung

Fehlgeschlagen

Sende Daten & Empfange confirmation

Erfolgr.

Fehlgeschlagen

oder fehlerhafte confirmation

Cache löschen

Erfolgr.

Warte 30s

**Server (RaspberryPI / Python)**

MainLoop

SSL

ErrorCheck

Database

Email

* Listen to port
* Accept client
* Read from client
* Write to client
* Init
* Read buffered measurements
* Init
* Check measurements
* Get Errors
* Init
* Send status
* Send master timeout
* Send master reconnect
* Init
* store measurement

Listen & accept client (master)

Read data (measurements)

Check & Store data in Data storage

Send positive confirmation

Send negative confirmation

Erfolgr.

Fehlgeschl.

Close connection

SSL (Separate Thread)

Data storage

Wait until data in Data storage

Read & delete data from Data storage

Check data from storage & store errors

Send correct data to database (if sending not successful -> buffer Data )

Try send not already sent Data to DB & send status email containing errors every 60 min & send email if no data received from master for more than 10 min

While no data received