Target-Anwendung

ConSALTing Machine: getrennt/Suche läuft…/verbunden

Konfiguration: aktualisiert/nicht aktuell

Spinner: blau färben

# CSensorData

Datentyp für die Übertragung der Sensordaten zwischen Target und Android

* float batteryState
* float saltAmount
* float pressure

Einstellungen: aktuell/Aktualisierung nötig

Abfrage der Bluetooth-Characteristic Machine Control von der App aus

# main

* Zusätzlich Status der Einstellungen
* Nach dem Starten der App: Versuch, zu verbinden
  + Wenn Verbindung besteht: Abfrage der Bluetooth-Characteristic Machine Control, dann TextViews Verbindungsstatus und Aktualisierungsstatus aktualisieren
  + Wenn Verbindung fehlgeschlagen: MessageBox mit Hinweis 🡺 Button erneut suchen, Abbrechen, TextViews aktualisieren: getrennt/nicht aktuell
* Variable mit aktueller Konfiguration der ConSALTingMachine
* Bei Änderung von Slider oder Spinner: prüfen, ob neue Konfiguration mit alter Konfiguration übereinstimmt 🡺 TextViews aktualisieren
* Salzfüllstand zurücksetzen: MessageBox anzeigen, dort Bestätigung nötig (Buttons Weiter oder Abbrechen)
* onPause: BLE-Verbindung schließen
* onResume: Versuch, BLE-Verbindung herzustellen (sofern alle Berechtigungen erteilt)

## Bluetooth-Zustände

* Anzeige der Zustände in der obersten TextBox

connecting:

disconnected:

connected:

# Sensordaten:

zu Beginn: alle TextViews auf ---

Button einfügen: „Sensordaten aktualisieren“

Android-Anwendung

# CSensorData

s.o.

# CMeal

* String mName
* float mSaltAmount

UI

// declare UI variables  
TextView mTVConnectionState;  
ImageButton mConnectionButton;  
TextView mTVTargetConfiguration;  
ImageButton mSynchronizeButton;  
Spinner mSpinner;  
TextView mTVSaltConcState;  
Slider mSlider;  
Button mBResetSaltAmount;

Funktionen

# Funktionen Target-Anwendung:

* Hardware-Ansteuerung: Motor für Salzausgabe, Status-RGB-LED für Kalibriervorgang abgeschlossen, Akku leer, Salz leer, Fehler, Verbunden …
* Sensoren auslesen, Kraftsensor, Knopf: Kalibrier- und Salzausgabefunktion
* Kommunikation
* Salzvorrat per Software zurückrechnen
* für Funktionen: verschiedene Drück-Kombinationen am (einzelnen) Knopf

# Funktionen App:

* Batteriezustandsausgabe
* Salzvorrat prozentual
* Button: Salzmenge wurde aufgefüllt <?>
* Auswahl Kartoffeln/Reis/Nudeln/… (Beschränkung hauptsächlich auf Nudeln für experimentelle Tests, für App trotzdem Auswahlmöglichkeit) (Die Gerichte kennt nur die App, an das Target werden nur die Salzmengen gesendet
* Feineinstellungsmöglichkeit für mehr/weniger Salz
* Ausgabe: Verbindungsaufbau/Verbunden/keine Verbindung/…
* Übertragung der Einstellungen an Target, direkt nach der Auswahl in der App

# Ideen:

* Salzmenge ermitteln: automatisch bei leerem Gerät?