CatchCalculateCommunicate

Software zur Temperaturüberwachung

# Hintergrund

In Rechenzentren spielt die Temperatur eine große Rolle. Die Temperaturen dürfen einen bestimmten Wert nicht übersteigen und auch nicht unterschreiten, da sonst die Gefahr besteht, dass sich in den Systemen Kondenswasser bildet. Natürlich besitzt jeder Server einen eingebauten Temperatursensor. Aber was passiert wenn dieser ausfällt? Um für diesen Fall eine Backuplösung zu konstruieren, nutzen wir einen RaspberryPi welcher mit mehreren Sensoren ausgestattet ist um weiterhin die Umgebungstemperaturen zuverlässig aufzuzeichnen.

# Voraussetzungen an die Software

Ich wurde nun beauftragt eine Software zu Designen welche die aufgenommenen Daten sammelt, verarbeitet und darstellt. Es ist wichtig, dass die Software zuverlässig arbeitet und mit genauen Datenwerten umgehen kann. Zusätzlich sollen die Grenzwerte automatisch erkannt werden und im Falle einer Überschreitung ein passendes Feedback ausgegeben werden. Um es Personen möglich zu machen jederzeit die Daten einsehen zu können, ist das Ziel über das Netzwerk die Daten abzurufen und in einer Art Diagramm mit Verlauf darzustellen. Um mögliche Temperaturunterschiede vorauszusehen.

## Schwerpunkte in der Übersicht

* Aufnahme der Daten (zuverlässig & genau)
* Darstellung der Daten für den Benutzer
* Überwachung der Grenzwerte

# Durchführung

Da das Gesamtprojekt noch in der frühen Anfangsphase steht, ist es mir nicht möglich mit einem Prototyp zu arbeiten. Dadurch bin ich gezwungen mit dem Black-Box Verfahren zu arbeiten.

## Datenspeicherung

Die Daten welche aus der „Black Box“ stammen werden zunächst in einer Datenbank erfasst um z.B. einen Verlauf darstellen zu können. Die Manipulation der Datenbank könnte mittels eines Python Skripts erfolgen.

## Darstellung

Da die Anforderung besteht von Überall auf die Daten zugreifen zu können, wäre eine Möglichkeit die Daten mittels einer Webseite zugänglich zu machen. Falls die Webseite aus dem WWW erreichbar ist, muss auch sehr stark auf die Sicherheit und Benutzerkontenführung geachtet werden.

Der Benutzer soll die Daten in Form von Diagrammen einsehen können. Zusätzlich sollen noch weitere Benutzereingaben wie zum Beispiel löschen des Verlaufs möglich sein.

## Überwachung

Die Software soll auch ohne dem Zutun eines Benutzers Gefahren erkennen und frühestmöglich entsprechendes Feedback ausgeben (Z.B. mittels einer Email).

# Zeitplanung

1. Meilenstein
   1. Anlegen einer Datenbank
   2. Auslesen & Speichern der Daten
2. Meilenstein
   1. Weboberfläche mit Anzeige
   2. Benutzeraktionen
3. Meilenstein
   1. Autonome Überwachung
   2. Meldungssystem