



---

Bachelorarbeit

**Entwicklung eines Frameworks  
zur Darstellung von  
Smartphone-Sensordaten für die  
didaktische Unterstützung von  
Programmiervorlesungen**

---

Marius Cerwenetz

Abschlussarbeit

zur Erlangung des akademischen Grades

Bachelor of Science

Vorgelegt von	Marius Cerwenetz
am	XX. Juli 2022
Referent	Prof. Dr. Peter Barth
Korreferent	Prof. Dr. Jens-Matthias Bohli

## **Schriftliche Versicherung laut Studien- und Prüfungsordnung**

Hiermit erkläre ich, dass ich die vorliegende Arbeit selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe.

Mannheim, XX. Juli 2022

---

Marius Cerwenetz

## **Zusammenfassung**

# Inhaltsverzeichnis

<b>1</b>	<b>Einführung</b>	<b>2</b>
<b>2</b>	<b>Grundlagen</b>	<b>3</b>
<b>3</b>	<b>Entwicklung</b>	<b>8</b>
<b>4</b>	<b>Diskussion und Fazit</b>	<b>11</b>
<b>5</b>	<b>Ausblick</b>	<b>12</b>
	<b>Literaturverzeichnis und Online-Quellen</b>	<b>12</b>

# Kapitel 1

## Einführung

MINT-Berufe leiden hierzulande unter einem akuten Fachkräftemangel. Das Institut der deutschen Wirtschaft ermittelte für April 2021 ein Unterangebot von 145.100 Personen in 36 MINT-Berufskategorien. [1] Digitalisierungs-Projekte geraten dadurch ins Stocken.

Nicht zuletzt er auch ein Kräftemangel in der Softwareentwicklung. Es fehlen Programmiererinnen und Programmierer. Softwareentwicklung ist gerade in der Lernphase nicht trivial und abstrakt. Unlebendige Übungsaufgaben die beispielsweise Konsolenein- und ausgaben realisieren schrecken zukünftige Programmierinnen und Programmierer eher ab als sie zu ermutigen.

Microcontroller sind bereits eine große Hilfe, da hier spielerisch kleine Projekte realisiert werden können. So können schon früh in Schulen Kinder an die Programmierung herangerführt werden. Sie lernen spielerisch kleine Programme zu entwickeln und verstehen die ihnen beigebrachten Abläufe durch schnelle Anwendung. Microcontroller sind jedoch auch mit Anschaffungskosten verbunden und für kleine Anwendungen, welche Sensoren verwenden, wird viel zusätzliches Material wie zum Beispiel Breadboards, Verbindungskabel und Erweiterungsboards benötigt. Moderne Smartphones bieten hier Abhilfe da Sie meistens mit verschiedenen Sensoren bespickt sind, wie zum Beispiel: Kompass, GPS, Microphon oder Kamera. Viele Kinder besitzen bereits mit 10 Jahren ein Smartphone. [2]

Im Rahmen dieser Arbeit soll ein Framework entwickelt werden, dass das Smartphone von Anwendern einbindet um Sie beim Programmierenlernen zu unterstützen.

## Kapitel 2

# Grundlagen

Das Smartphone soll für das Austauschen von Daten verwendet werden. Hier wird klar unterschieden in Anfragen die eine Benutzerinteraktion verlangen und welche, die passiv Daten auslesen. Die Übertragung wird nach dem Request-Response-Prinzip realisiert. Das Framework besteht aus drei großen Teilen:

1. Einer Anwendung auf einem Android-Smartphone
2. Einer Middleware auf dem PC oder Laptop des Anwenders
3. Einer Bibliothek in Java, C++ und Python ebenfalls auf PC oder Laptops des Anwenders

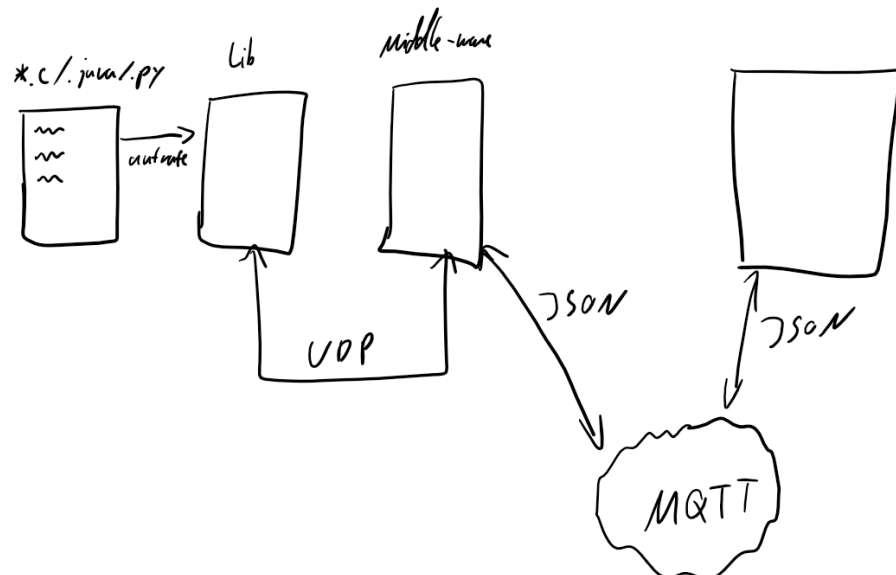


Abbildung 2.1: Framework-Ablauf

Auf die Funktionsweise und den Ablauf wird im folgenden genauer eingegangen.

## Bibliothek und API

Zur Ansteuerung des Smartphones werden vier Softwarebibliotheken jeweils in den Programmiersprachen Python, C++, C und Java zur Anbindung realisiert. Die Bibliotheken in den Programmiersprachen C++, Java und Python dienen zur Erstellung von Smartphone-Objekten über die Methodenaufrufe realisiert werden können. Die Bibliothek in C ist prozedural realisiert und bietet den Aufruf von globalen Methoden ohne Objektbindung.

Die Aufrufe sind logisch in zwei Kategorien geordnet: Dem Aufruf zur Übermittlung von Sensordaten und dem Aufruf zum Starten von Activities zwecks einer Nutzerbedienung mit anschließendem Rückübermitteln der Nutzerauswahl.

Die Anfragen werden in der jeweiligen Programmiersprache angestoßen und werden im weiteren Verlauf von der Middleware über das UDP-Protokoll auf dem gleichen PC entgegengenommen.

## Middleware

Die Middleware dient als Vermittlungsschicht zwischen Bibliothek und Smartphone. Sie nimmt per UDP Methodenaufrufe entgegen und sendet diese als Nachricht für das Smartphone weiter. Die Nachricht wird dabei über einen MQTT-Server übertragen.

## MQTT

Die Nachrichten werden von der Middleware per MQTT übertragen. MQTT beziehungsweise Message Queuing Telemetry Transport, ist ein Netzwerkprotokoll. Es dient der Übertragung von Messdaten. Sensor-Geräte senden dabei Nachrichten an einen Broker in ein Topic. Ein Broker ist eine MQTT-Serverkomponente die Nachrichten annimmt. Broker verwalten die gesendeten Daten hierarchisch in Themenbereichen, den Topics. Andere Geräte können am Broker Topics abonnieren (subscribe). Publiziert nun der Sensor-Client einen neuen Messwert auf einem Topic das ein Empfänger abonniert hat wird ihm die Nachricht zugestellt.

Die Nachrichten werden dabei mit einem Quality of Service-Flag übertragen. Dies beschreibt die Sicherungsgarantie mit der eine Nachricht übermittelt wird. Sie ist aufgeteilt in drei Stufen

1. at most once

2. at least once
3. exactly once

In der ersten Stufe werden Nachrichten zwar übermittelt, jedoch kann es vorkommen, dass bei Verbindungsabbrüchen Nachrichten nicht weitergeleitet werden.

In der zweiten Stufe werden Nachrichten so lange versendet, bis Clienten die das jeweilige Topic abonniert haben eine Empfangsbestätigung senden. Es wird jedoch nicht sichergestellt, ob die Nachricht nicht mehrmals versandt wurde.

Dies wird in der dritten Stufe, *exactly once*, sichergestellt. Nachrichten werden nur so lange gesendet bis die Übertragung einmal bestätigt wurde.

## Nachrichtenformate

Die Nachrichten werden im JSON-Format übertragen. JSON (JavaScript Object Notation) ist ein Datenformat mit dem Daten in Textform kommuniziert werden können. Die Notation besteht aus Schlüssel-Werte-Paaren denen eine Semantik zugeordnet ist. Für die Anwendung steht folgender Ausschnitt beispielsweise für das Starten einer bestimmten Activity.

```
1  {  
2    "issue_id" : 0,  
3    "type" : activity_request,  
4    "value" : "boolean_activity",  
5    "arguments" : []  
6  }
```

Das Feld `issue_id` benennt eine aktuelles Thema, eine aktuelle Session. Eine Session besteht aus Anfragen (requests) und Antworten (responses). Sessions haben eine eindeutige Nummerierung zur Identifizierung, die `issue_id`.

Das Type-Feld beschreibt, ob es sich bei der Nachricht um ein request oder um eine response handelt. Dieses Feld kann die folgenden Werte annehmen:

- `sensor_request`
- `sensor_request_burst`
- `activity_request`
- `sensor_response`
- `activity_response`



Bei einem `sensor_request` handelt es sich um die Eien Abfrage einer Übermittlung eines einzelnen Sensorwertes. Ähnlich dazu verhält sich das `Burst-Request`, dass für einen angegebenen Zeitraum Werte übermittelt.

Ein `Activity-Request` fordert das Starten einer Activity mit der der Nutzer interagieren muss. Je nach Request-Art werden entweder `Sensor-Responses` oder `Activity-Responses` zurückgesendet. Die Antwort auf `Burst-Anfragen` erfolgt mit einzelnen `sensor_responses`.

## Android Anwendung

Die Android-Anwendung läuft auf dem Smartphone des Anwenders. Sie hat ein sehr schlichtes äußeres Erscheinungsbild und reagiert nur auf Anfragen der Middleware. Sie besteht aus einer Haupt-Activity und mehreren Unter-Activites die gegebenenfalls aufgerufen werden.

In der Haupt-Activity wird ein Service gestartet und gebunden. Der Service baut eine Verbindung zu einem MQTT Topic auf einem MQTT-Server auf, reagiert auf Nachrichten und versendet gegebenenfalls selbst Nachrichten. Geht eine Nachricht ein die eine User-Interaktion verlangt wird die jeweilige Activity gestartet. Der User interagiert mit der Activity und bestätigt seine Eingabe. Die Activity bindet daraufhin den MQTT-Service und versendet die Antwort an die Middleware, wo Sie an die Bibliothek und somit an die Endanwendung auf dem PC des Users gesendet wird.

Geht eine Nachricht ein, die die Übermittlung von Sensordaten verlangt wird keine eigene Aktivität gestartet. Der Service startet in diesem Fall selbst den Ausleseprozess und übermittelt dann, je nach Anfrage, einen oder mehrere Werte. Die Verfügbarkeit der jeweiligen Sensoren wird beim Start der Anwendung ermittelt. Falls ein Sensor angefragt wird der auf dem Smartphone nicht existiert wird eine Nachricht mit einer Fehlermeldung zurückgegeben.

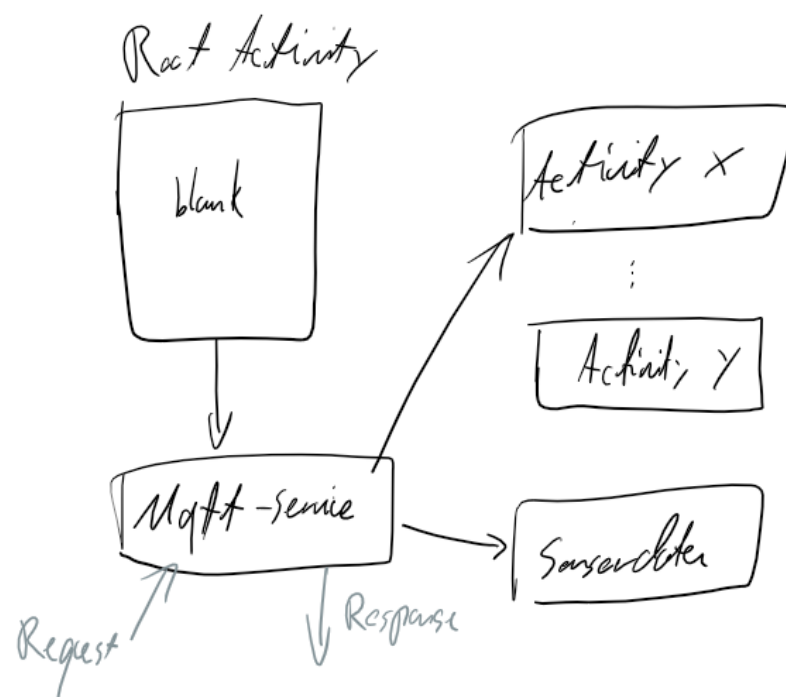


Abbildung 2.2: Android-App-Aufbau

## Kapitel 3

# Entwicklung

### Anwendungsfälle

In diesem Kapitel werden die konkreten Anwendungsfälle der Software beschrieben.

#### Sensorauslesung

Diese Anwendungsfälle betreffen das Übertragen von Sensor-Rohdaten.

#### Benutzereingaben

Diese Anwendungsfälle starten eine Activity in der Androidanwendung die eine Nutzerinteraktion zum Fortfahren fordert.

#### Boolean-Switcher

Ein Nutzer möchte ein Objekt vom Datentyp boolean auf sein Smartphone übertragen, dort negieren und anschließend wieder zurücksenden. In der Anwendung wird dafür ein Schalter dargestellt den er von an auf aus und umgekehrt stellen kann. Mit einem Klick auf einen ok-button wird der Zustand übertragen.

#### String-Manipulation

Ein Nutzer möchte einen String an das Smartphone übertragen. Dort soll dieser String verändert und zurückgeschickt werden können. Dafür erscheint ein Eingabefeld und

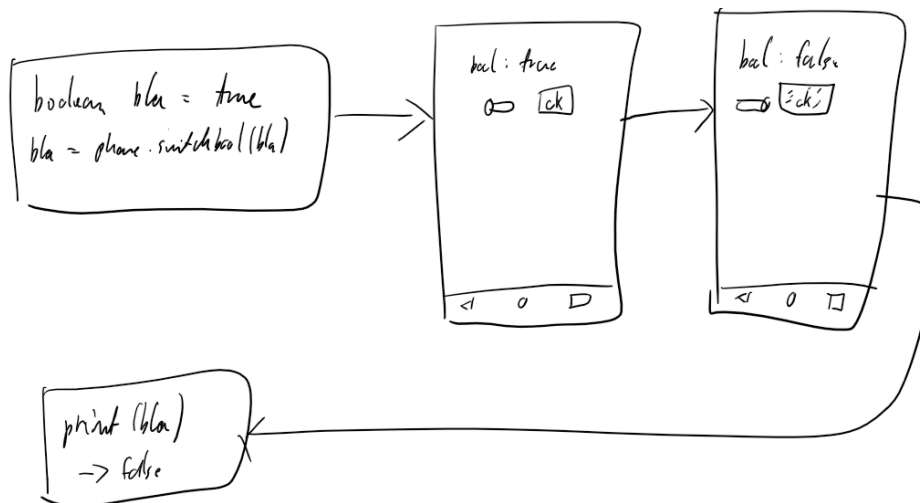


Abbildung 3.1: Zeichnung-Boolean-Switcher

ein OK-Button zur Übertragung. Im Eingabefeld steht der alte Wert des Strings. Die Änderung soll unmittelbar nach der Übertragung sichtbar werden.

### Vibrationsalarm

Ein Nutzer möchte das Smartphone auf Anfrage vibrieren lassen. Der Aufruf einer Methode lässt das Smartphone einmal vibrieren.

### Annäherungssperre

Ein Nutzer möchte dass ein Programm so lange wartet bis sich etwas seinem Smartphone annähert. Erst danach soll es weiterlaufen.

### Sonstige

### Klatsch-Zähler

Der Anwender möchte wissen wie oft in einem bestimmten Zeitraum geklatscht wurde. Ein Methodenaufruf startet in der Androidanwendung eine Activity, die innerhalb des im Argument genannten Zeitraums die Anzahl der maximalen Lautstärkeamplituden des Mikrophons misst und die Anzahl zurückgibt.

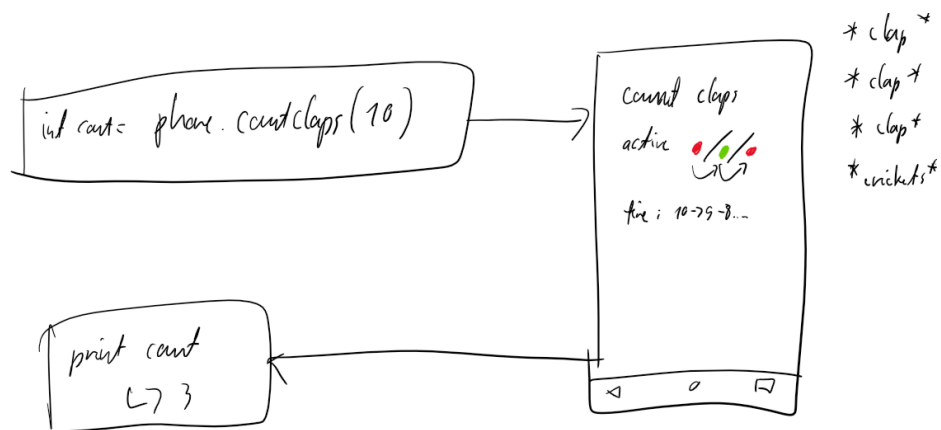


Abbildung 3.2: Klatsch-Zähler

### Dreh-Zähler

Ein Nutzer möchte in einem bestimmten Zeitraum zählen wie oft das Smartphone gedreht wurde.

### Softwarelösungen

## **Kapitel 4**

# **Diskussion und Fazit**

# Kapitel 5

## Ausblick

### Literaturverzeichnis

- [1] Koppel Oliver Anger Christina, Kohlisch Enno. Mint-frühjahrsreport 2021. mint-engpässe und corona-pandemie: von den konjunkturellen zu den strukturellen herausforderungen, gutachten für bda, bdi, mint zukunft schaffen und gesamtmittel, köln. <https://www.iwkoeln.de/studien/christina-anger-enno-kohlisch-oliver-koppel-axel-pluennecke-mint-engpaesse-und-corona-pandemie-von-den-konjunkturellen-zu-den-strukturellen-herausforderungen.html>, 2021.

### Online-Quellen

- [2] bitkom. <https://www.bitkom.org/Presse/Presseinformation/Mit-10-Jahren-haben-die-meisten-Kinder-ein-eigenes-Smartphone>. [letzter Zugriff: 07. April. 2022].