

# Smartbit

Marius Cerwenetz

Institut für Softwaretechnik und Datenkommunikation

2022-07-12

# Agenda

Einführung

Anforderungen

Smartbit

Architektur

Evaluation

Fazit und Ausblick

# Schwierigkeiten beim Programmierenlernen

- ▶ Programmier-Neulinge
- ▶ Grundkonzepte
- ▶ Theoretische Übungsaufgaben

# Microcontroller als Alternative

- ▶ Interaktive Aufgaben
- ▶ Ausgabemöglichkeiten (LEDs, Piepser, Aktoren)
- ▶ Ausprobieren

# Vorteile Smartphones gegenüber Microcontroller-Schaltungen

- ▶ Hohe Verfügbarkeit
- ▶ Keine Verdrahtungsfehler
- ▶ Unabhängige Spannungsversorgung
- ▶ Zahlreiche Sensoren bereits integriert
- ▶ Drahtlose Verbindungstechnologien (WLAN, Bluetooth, UMTS/LTE)
- ▶ Zweckgebundene Ausgabemöglichkeiten

## Probleme

- ▶ Anbindung in Programmierumgebungen
- ▶ Smartphone-App

## Lösung

Lösung zur Kommunikation mit dem Smartphone

# Anforderungen

- ▶ UI-Elemente als Ausgabe
- ▶ Sensordatenübermittlung
- ▶ Geringe Latenzen
- ▶ Sicherheit

# Aufbau der Smartbit-Lösung



Bibliothek



Middleware



Android-App



# Schnittstelle zum Programmcode

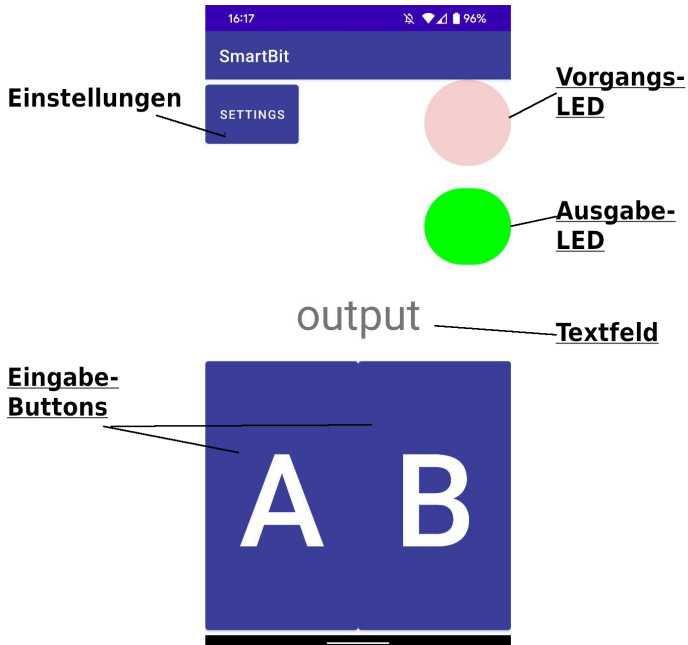
Funktionsaufrufe für Sensorwerte und Ausgaben.

Einbindbar in:

- ▶ C
- ▶ Java
- ▶ Python

```
1 from smartbit import Phone
2 p = Phone()
3 accel = p.get_x_accelo()
4 p.write_text("hallo")
5
```

# Smartphone-App



# Middleware

```
swt@pb22:~/thesis01/software$ python3 ./server.py  
INFO:MqttHandlerThread:trying to connect to mqtt server  
INFO:MqttHandlerThread:connected to server pma.inftech.hs-mannheim.de  
INFO:MqttHandlerThread:mqtt subscribed to topic: 22thesis01/test  
█
```





