



## DOKUMENTATION

M242 LB3 «WINDOW MANAGER» | KAUFMANN, WETTSTEIN, CANZANO | AP18B

## INHALTSVERZEICHNIS

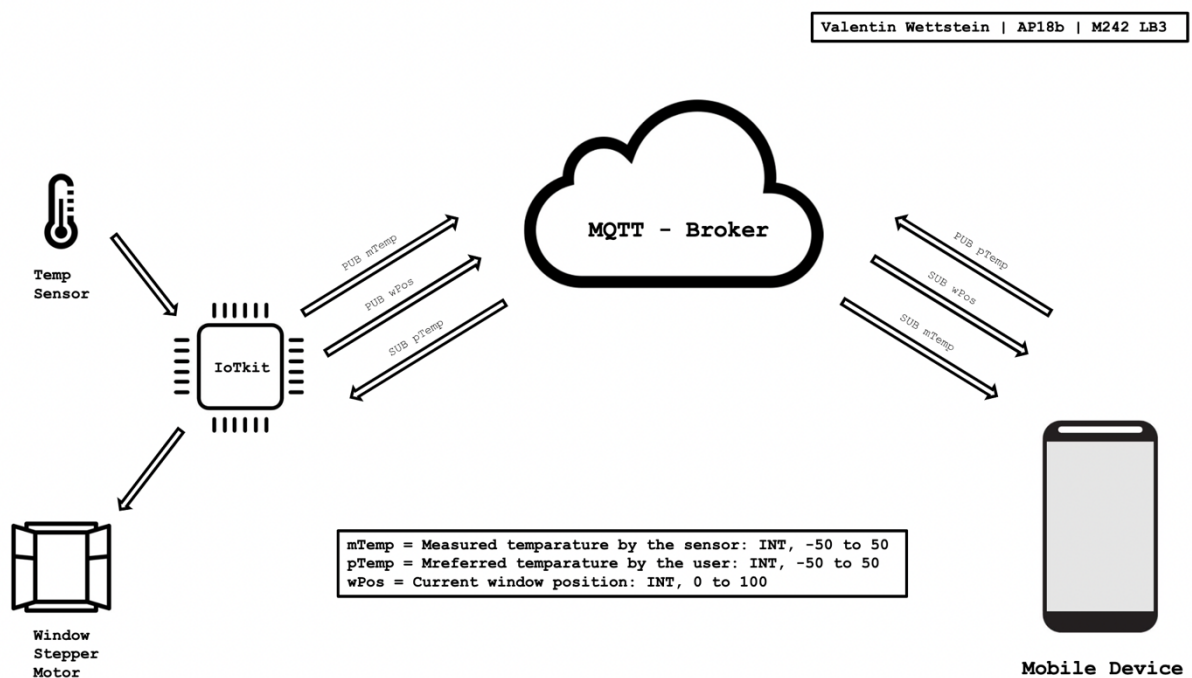
Vorwort .....	<b>Error! Bookmark not defined.</b>
Projektidee .....	3
Sensoren.....	<b>Error! Bookmark not defined.</b>
Wichtigste Lernschritte .....	4
Benutzeranleitung.....	4
Schlusswort .....	<b>Error! Bookmark not defined.</b>

## PROJEKTIDEE

Wer kennt das Problem nicht: Man öffnet das Fenster vor dem Einschlafen und wenn man am Morgen aufwacht, ist es eiskalt, da die Aussentemperatur im Verlauf der Nacht stark gesunken ist.

Unser Ziel als Team für diese LB3 war es, dieses Problem mit der Hilfe von Mikroprozessoren ein für Alle Male zu lösen. Dies wollten wir erreichen, indem wir die Temperatur messen und je nach bevorzugter Temperatur des Endnutzers wird das Fenster leicht geöffnet oder geschlossen, so dass die gewünschte Innentemperatur aufrecht erhalten werden kann.

## MQTT - ANWENDUNG



In dieser Ausführung wurde zur Datenübermittlung das MQTT – Protokoll verwendet. Hierbei sendet und Empfängt sowohl das IOTKIT, als auch ein MQTT – Client (Hier ein Mobile Device) Werte via einen MQTT – Broker an einander (Hier cloud.tbz.ch). Das IOTKIT sendet hierbei in einem definierten Zeitintervall wiederholt sowohl die gemessene Temperatur als auch die aktuelle Fensterposition. Diese Werte empfängt der MQTT – Client. Auf dem Client kann man die gewünschte Temperatur an das IOTKIT senden. Dieser Wert wird abgeglichen und so wird die Fensterposition bestimmt, damit die Temperatur den Wünschen der temperatur fühlenden Person entspricht.

## WICHTIGSTE LERNSCHRITTE

- Testen vom MQTT – Broker (via Mosquitto)
- Aufsetzen MQTT – Infrastruktur (Mobile Phone + IOTKIT)
- Logik von LB2 in das MQTT – Programm auf dem IOTKIT einbauen

## BENUTZERANLEITUNG

Siehe Readme auf [GitHub](#)

## SELBSTBEURTEILUNG ERFÜLLUNG DER BEWERTUNGSKRITERIEN

Kriterium	Gütestufe 3 (=3P)	Gütestufe 2 (= 2P.)	Gütestufe 1 (= 1P.)	Gütestufe 0 (= 0P.)
K1	Umgebung auf eigenem Notebook eingerichtet und voll funktionsfähig <ul style="list-style-type: none"> <li>• Account auf os.mbed.com erstellt</li> <li>• Serial Driver installiert</li> <li>• Terminal Programm installiert</li> <li>• <a href="#">Mbed Studio</a> installiert und Programme können lokal compiliert werden</li> </ul>	Es sind drei Kriterien erfüllt	Es sind zwei Kriterien erfüllt	Nur ein oder kein Kriterium erfüllt
K2	Eigene Lernumgebung ist eingerichtet <ul style="list-style-type: none"> <li>• Dokumentation vorhanden</li> <li>• Persönlicher Wissenstand in Bezug auf die wichtigsten Themen ist dokumentiert (IoT, Sensoren, Aktoren, Service)</li> <li>• Wichtige Lernschritte sind dokumentiert</li> <li>• Anhand der Dokumentation können Dritte das Projekt nachbauen</li> </ul>	Es sind drei Kriterien erfüllt	Es sind zwei Kriterien erfüllt	Nur ein oder kein Kriterium erfüllt
K3	IoTKit <ul style="list-style-type: none"> <li>• Beispiel Programm verwendet</li> <li>• Beispiel Programm geringfügig abgeändert, z.B. nur URL</li> <li>• Beispiel Programm erweitert, z.B. mehr Sensordaten senden oder andere Daten</li> <li>• Kommunikation erfolgt verschlüsselt, z.B. mittels <a href="#">HTTPS</a>.</li> </ul>	Es sind drei Kriterien erfüllt	Es sind zwei Kriterien erfüllt	Nur ein oder kein Kriterium erfüllt
K4	(Cloud) Backend Applikation <ul style="list-style-type: none"> <li>• (Cloud) Dienst aus den Beispielen verwendet</li> <li>• (Cloud) Dienst, welcher nicht in den Beispielen vorkommt, verwendet</li> <li>• Eigene (Cloud) Backend Applikation implementiert (= zwei Punkte)</li> <li>• Kommunikation erfolgt verschlüsselt, z.B. mittels <a href="#">HTTPS</a>.</li> </ul>	Es sind drei Kriterien erfüllt	Es sind zwei Kriterien erfüllt	Nur ein oder kein Kriterium erfüllt
K5	Zusätzliche Bewertungspunkte <ul style="list-style-type: none"> <li>• Allgemein (Kreativität, Komplexität, Umfang)</li> <li>• Umsetzung eigener Ideen</li> <li>• Persönlicher Lernentwicklung (Vergleich Vorwissen – Wissenszuwachs)</li> <li>• Reflexion</li> </ul>	Es sind drei Kriterien erfüllt	Es sind zwei Kriterien erfüllt	Nur ein oder kein Kriterium erfüllt

K1 (2P):

- Account wurde erstellt
- Terminal funktioniert
- Programme können kompiliert und ausgeführt werden

K2 (2P)

- Dokumentation vorhanden
- Wichtigste Lernschritte und Teile der Programme sind dokumentiert
- Dritte können das Projekt nachmachen

K3 (2P)

- Mehrere Beispielprogramme getestet, erweitert und schlussendlich fusioniert und erneut um Einiges erweitert
- Sensordaten werden ausgelesen und HTTP – Request wird gesendet sowie Response empfangen

K4 (2P+)

- Komplettes eigenes Front – und Backend ohne Vorlage und Hilfe aus dem Internet implementiert; voll funktionsfähig sowie livegeschaltet

K5 (2P+)

- Idee wurde umgesetzt und funktioniert, hat einen realen Nutzen
- Mechanische Teile zur einwandfreien Funktion wurden extern aufgetrieben und zusammengesetzt (Schrittmotor, Achse, Befestigungen, etc...)
- Enorme Lernentwicklung nicht nur in C++, sondern auch Elektrotechnik (Funktion Schrittmotor) sowie selbstverständlich allgemein in Mikroprozessoranwendungen insb. Im MBED – Studio.
- Video, welches sowohl zeigt, dass die Anwendung funktioniert als auch dass ich mehr schlafen soll wurde bereitgestellt

## INFO HTTPS / VERGLEICH HTTP

Das «S» in HTTPS steht für Secure. Es stellt eine Transportverschlüsselung dar. Syntaktisch ist HTTPS identisch mit dem Schema für HTTP, die zusätzliche Verschlüsselung der Daten geschieht mit SSL oder TLS. Unter anderem gibt es bei HTTPS eine geschützte Identifikation und Authentifizierung der Kommunikationspartner. Dies geschieht mit einer asymmetrischen Verschlüsselung, wobei ein gemeinsamer symmetrischer Sitzungsschlüssel ausgetauscht wird. Der Standard-Port für HTTPS-Verbindung ist 443.