

Entwicklung von vernetzten Benutzungsschnittstellen für die Heimautomation

von

Nicolas Mehlei

als

Independent Coursework Projektarbeit

betreut von

Prof. Dr.-Ing. Johann Habakuk Israel
Hochschule für Technik und Wirtschaft Berlin

Berlin, 08. April 2017

Inhaltsverzeichnis

1	Einleitung	2
2	Situation	3
2.1	Existierende Systeme	3
2.1.1	Klassisch-elektrisch	3
2.1.2	KNX	3
2.1.3	Funk-Systeme	3
2.1.4	Philips Hue	3
2.1.5	Osram Lightify	4
2.2	Problematik	4
2.3	Lösungsansatz	4
3	Analyse	5
3.1	Benutzergruppen	5
3.2	Funktionale Anforderungen	5
3.3	Nicht-funktionale Anforderungen	6
3.3.1	Intuitivität	6
3.3.2	Feedback	6
3.4	Zielgruppe	7
4	Entwurf	8
4.1	Schnittstellen-Entwürfe	8
4.1.1	Variante 1	8
4.1.2	Variante 2	9
4.1.3	Gegenüberstellung	10
4.2	System-Bestandteile	10
4.2.1	Mikrocontroller	10
4.3	Gateway-Service	11
4.3.1	Referenz-Implementation	11
4.4	Kommunikation	12
4.4.1	Nachrichten	12
4.4.2	Auto-Konfiguration	12
4.5	Verhalten	13
5	Implementierung	14
5.1	Veranschaulichung	14
5.2	Feedback-Verhalten	15
5.3	Firmware-Update	16
6	Test	17
6.1	Funktionalität	17
6.2	User-Studie	17
6.2.1	Erwartungen	17
6.2.2	Ergebnisse	17
6.2.3	Beobachtungen	17
6.2.4	Auswertung	18

<i>Inhaltsverzeichnis</i>	1
7 Zusammenfassung	20
8 Ausblick	21
8.1 Optik	21
8.2 Stromversorgung	21
8.3 Barrierefreiheit	21
Literaturverzeichnis	22
Glossar	22

1 Einleitung

Für viele Menschen ist ein Lichtschalter etwas sehr Triviales. Die „komplexeste“ Aufgabe, die hiermit erfüllt werden kann, ist das gezielte Dimmen von Licht.

Modernere Lichtsysteme bieten heutzutage jedoch weit mehr Möglichkeiten als nur An, Aus sowie potenziell „gedimmtes“ An. Hierbei kann es jedoch kompliziert sein, die eigene Wohnung oder das eigene Haus mit einer durchgehenden Versorgung gleichartiger und benutzerfreundlicher Lichtschalter zu versorgen, welche auch diese moderneren Lichtsysteme steuern können.

Diese Arbeit soll potenzielle Probleme mit den Steuerelementen der vorhandenen Lichtsysteme aufzeigen und den Entwurf und die Implementation einer prototypischen alternativen Lösung darstellen.

2 Situation

Nachfolgend soll ein grober Überblick über die aktuelle Marktlage gegeben sowie erläutert werden, welcher Aspekt hiervon als problematisch angesehen werden könnte. Basierend darauf wird ein möglicher Lösungsansatz skizziert.

2.1 Existierende Systeme

Zunächst soll eine Übersicht über vorhandene Lichtsteuerungssysteme, deren Inter-Kompatibilität und deren Möglichkeiten der Steuerung gegeben werden.

2.1.1 Klassisch-elektrisch

Als klassisch-elektrische Systeme sind im Rahmen dieser Arbeit Lichtsteuerungssysteme gemeint, welche über die konventionelle Schaltung von elektrischen Leitungen funktionieren. Dies ist der Quasi-Standard, mit dem ein Großteil der heutigen Wohnungen und Häuser ausgestattet sind.

Die hier üblichen Eingabegeräte sind entweder Kippschalter oder Schalter mit integriertem Drehregler zum Dimmen des Lichts.

2.1.2 KNX

KNX ist der am meisten verbreitete Standard für Hausautomatisierung und somit auch als Lichtsteuerungssystem. Er wird jedoch nur in den seltensten Fällen von Privatpersonen installiert. Da hierfür in der gesamten Wohnung bzw. das gesamte Haus vieradrige Kabelleitungen (für den sogenannten Bus) verlegt werden müssen, ist dies eher eine Lösung für industrielle Gebiete oder als Erweiterung bei einem Hausbau. [2]

Da diese Arbeit sich primär mit den Möglichkeiten für Privatpersonen (Consumer-Bereich) beschäftigt ist KNX als eher unwichtig einzustufen.

2.1.3 Funk-Systeme

Eine Alternative für klassisch-elektrische Lichtsteuersysteme sind sogenannte Funk-Systeme. Diese bestehen meist aus Schaltern, Dimmern, Tastern und Fernbedienungen und kommunizieren über ein Frequenzband von 433 oder 868 MHz.

Die Funktionalität der Lichtquellen sowie der Eingabegeräte ist hierbei beinahe identisch mit klassisch-elektrischen Systemen, bis auf dass die Kommunikation via Funk erfolgt. Ein Mischbetrieb ist nicht möglich, ergo muss ein Funk-Schalter auch mit einem Funk-Taster verwendet werden. [2]

2.1.4 Philips Hue

Philips Hue ist ein 2012 erschienenes Produkt, welches aus Lichtquellen und Bedienelementen besteht. Hue war die erste direkt Consumer-taugliche Möglichkeit, farblich-dynamische Lampen in einer Wohnung zu verteilen, welche komfortabel gesteuert werden konnten.

Hue setzt stark auf ein *Szenen* genanntes System. Hierbei wird eine Farbe für einen Raum definiert, wodurch sich alle Lampen des jeweiligen Raums auf eben diese Farbe schalten.

Hue kann ausschließlich über die eigene App sowie über einen kabellosen Lichtschalter bedient werden. [2]

2.1.5 Osram Lightify

Lightify, von der Firma Osram, ist ein sehr ähnliches System zu Philips Hue. Es bedient sich in vielen Bereichen derselben Technik. Es kann - im Gegensatz zu Hue - an konventionelle Lichtschalter angebunden werden, jedoch ist es hierüber nicht möglich, die erweiterten Funktionen von Lightify zu nutzen, wie z.B. das Ändern der Lichtfarbe.

2.2 Problematik

Alle vorgestellten Systeme haben gemeinsam, dass deren Steuerelemente ausschließlich für das jeweils eigene System anwendbar sind. Da es nicht immer möglich ist, alle Lichtquellen einer Wohnung oder eines Hauses mit demselben Lichtsteuersystem zu versorgen, kommt es leicht zu Situationen, wo ein Mischbetrieb mehrerer Steuerelemente stattfinden muss. Diese fehlende Einheitlichkeit ist weder sonderlich benutzerfreundlich noch trägt es zu der Ästhetik der Wohnung bei.

2.3 Lösungsansatz

Ein Lösungsansatz ist hierbei die Erschaffung einer Alternative zu den proprietären system-gebundenen Bedienungselementen. Diese Alternative muss sich an die Schnittstellen der vorhandenen Lichtsteuersysteme koppeln und hierüber die Schaltung des Lichts erfolgen lassen können.

Um mehr Funktionalität bieten zu können als dies ein konventioneller Lichtschalter kann, soll hierbei auch die von Hue angebotene Szenen-Wahl ermöglicht werden, vorausgesetzt das zugrundeliegende Lichtsteuerungssystem erlaubt dies.

Als Referenzplattform wird für dieses Projekt Philips Hue verwendet, da dieses System den größten Funktionsumfang (inkl. der Möglichkeit farblichen Lichts) und nur hierfür die (sehr kostspielige) Hardware zur Verfügung stand. Es wird jedoch darauf geachtet, dass keine feste Bindung an spezifische Konzepte und Paradigmen von Philips Hue eingegangen wird, um eine möglichst abstrakte Schnittstelle zu erhalten und sich möglichst leicht an verschiedene Systeme knüpfen zu können.

3 Analyse

In diesem Kapitel sollen die Anforderungen des in Abschnitt 2.3 skizzierten Lösungsansatzes analysiert werden. Hierzu wird zunächst erörtert, wie die späteren Benutzer unterteilt werden könnten sowie welche Funktionen ein solches System zur Verfügung stellen muss.

3.1 Benutzergruppen

Die Benutzer lassen sich in zwei Kategorien einteilen: den *Gast-Nutzer* sowie der *Haupt-Nutzer*.

Der *Gast-Nutzer* ist, wie der Begriff bereits andeutet, der Gast, der eine Wohnung betritt. Ihm kann keine Erfahrung vorausgesetzt werden und es darf keine Einweisung notwendig sein. Der Gast-Nutzer muss ausschließlich in der Lage sein, das Licht an- und auszuschalten. Jede weitere Funktionalität ist optional und kann potenziell verwirren.

Der *Haupt-Nutzer* ist ein Bewohner der betreffenden Wohnung. Er wird schnell Erfahrung sammeln und es ist möglich, ihn in sehr geringem Maße in das System einzuführen. Dies kann beispielsweise über eine kleine Anleitung oder einen kleinen Aufkleber auf dem späteren Produkt erfolgen. Er ist der Nutzer, für den am ehesten erweiterte Funktionen geeignet wären.

3.2 Funktionale Anforderungen

Grundlegend benötigt das in Abschnitt 2.3 skizzierte Steuerelement zwei Funktionen: „Toggle“, also das von konventionellen Lichtschaltern gewohnte An- sowie Ausschalten des Lichts sowie die Funktion zum Ändern des derzeitigen Presets.

Toggle-Funktion

Die primäre Funktion eines Lichtschalters bedarf nicht vieler Worte. Er soll das Licht des jeweiligen Raums umschalten, also bei deaktiviertem Licht eben dieses aktivieren und entsprechend umgekehrt. Diese Funktionalität darf in keinem Falle anders als gewohnt funktionieren oder irgendwelche weiteren Aktionen auslösen, da dies nicht dem mentalen Modell des Benutzers entsprechen würde, der die Funktionsweise klassischer Lichtschalter seit Kindestagen an kennt und dieser entsprechend verwirrt werden würde und somit nicht intuitiv wäre.

Preset-Aktivierung

Die sekundäre Funktion ist das Aktivieren eines *Presets*. Das Konzept eines Presets ist angelehnt an das einer *Szene* seitens Philips Hue. Es definiert ein Farbschema, nach dem - bei Aktivierung - der Raum farblich beleuchtet wird.

Um dem Benutzer eine Vorschau über die anwählbaren Presets zu geben, soll jedes Preset über eine *Vorschau-Farbe* verfügen. Diese Farbe sowie deren Intensität soll exemplarisch veranschaulichen können, welche Lichtsituation dieses Preset für den jeweiligen Raum mit sich bringen würde.

3.3 Nicht-funktionale Anforderungen

Zusätzlich zu den funktionalen Anforderungen gibt es eine Reihe an nicht-funktionalen Anforderungen, welche für ein solches Gerät beachtet werden müssen.

3.3.1 Intuitivität

Da zumindest für den Gast-Nutzer keinerlei Einweisung vorausgesetzt werden darf, muss das konzipierte Gerät intuitiv genug sein, um sofort nutzbar zu sein. Um dies zu gewährleisten soll versucht werden, die eigene Lösung so nah wie möglich an das vom Nutzer gewohnte Gerät (der klassische Lichtschalter) zu halten, zumindest so weit dies möglich ist durch die Hinzufügung erweiterter Funktionen.

3.3.2 Feedback

Für ein Bedienelement ist ein Feedback wichtig, damit der User erkennen kann, ob sein Task erfolgreich war.

Die Möglichkeiten des Feedbacks lassen sich in die Kategorien der 5 menschlichen Basissinne unterteilen. Ignoriert man hierbei den Geruchs- sowie Geschmackssinn, welche bei technischen Schnittstellen eher unüblich sind, so verbleiben der Hörsinn, der Sehsinn sowie der Tastsinn.

Hörsinn (Auditives Feedback)

Ein auditives Feedback kann u.a. über einen kleinen Lautsprecher erfolgen, der z.B. einen leisen Piep-Ton bei Erkennung eines Kommandos abspielt. Dies kann jedoch unpraktisch sein, denn ein leiser Ton wird bei entsprechender Umgebungslautstärke nicht gehört werden, ein zu lauter Ton kann jedoch für den Nutzer selbst oder für sich in der Nähe befindliche (potenziell schlafende) andere Nutzer als störend empfunden werden.

Sehsinn (Visuelles Feedback)

Ein visuelles Feedback wird es automatisch geben, da in den meisten Fällen der Benutzer selbst das Dazukommen oder Wegfallen der gesteuerten Lichtquelle sehen wird. Dies allein ist jedoch nicht ausreichend, da es Fälle gibt, wo dies nicht erkennbar ist. Dies ist z.B. der Fall, wenn der Benutzer in einer schnellen Bewegung den Raum verlässt und parallel hierzu das Licht deaktiviert. Da sein Blick nach vorne gerichtet ist, wird er den Unterschied ggf. nicht bemerken können.

Des Weiteren ist es mitunter möglich, dass die jeweiligen Lichtquellen sich nicht sofort an- bzw. ausschalten, sondern dies z.B. mittels eines Verlaufs über einen Zeitraum einer Drittelsekunde erfolgt. Dies würde das sofortige Feedback erschweren bzw. verzögern.

Ist das Bedienelement mit einer eigenen Lichtquelle, wie z.B. einer oder mehrerer LEDs ausgestattet, so kann auch hierüber ein visuelles Feedback erfolgen. Auch dies setzt voraus, dass der Blick hierbei auf das Gerät gerichtet ist, weswegen dies eher eine sekundäre Feedback-Möglichkeit darstellen sollte.

Tastsinn (Taktils Feedback)

Das taktile Feedback ist die Feedback-Möglichkeit, welche am ehesten mit einem Lichtschalter assoziiert wird, immerhin sind diese üblicherweise nicht viel mehr als ein großer mechanischer Schalter. Im Sinne des erwarteten Verhaltens ergibt es somit Sinn auch weiterhin auf ein taktils Feedback zu sitzen. Wird weiterhin ein mechanischer Schalter verwendet, so dürfte dies ohnehin gewährleistet sein. Wird stattdessen auf alternative Kontaktflächen (wie z.B. Touch-Flächen) gesetzt, kann das taktile Feedback mittels Vibrationsmotoren künstlich erzeugt werden.

3.4 Zielgruppe

Als Zielgruppe gilt der einfache Privathaushalt (Consumer), da kommerzielle und industrielle Zwecke komplett andere Anforderungen stellen würden und diese ohnehin bereits von anderen Lösungen abgedeckt wären.

Der Einfachheit wird nachfolgend nur noch von Mietern und Wohnungen gesprochen, jedoch stehen einem Hausinhaber selbstverständlich dieselben Möglichkeiten zur Verfügung.

4 Entwurf

Das Kapitel Entwurf befasst sich mit der Umsetzung der in der *Analyse* herausgefundenen Anforderungen. Fortan wird die in diesem Dokument aufgezeigte Lösung als *RoomControlInterface* (kurz: RCI) bezeichnet, um sich von dem üblichen und sonst doppelt verwendeten Begriff des Lichtschalters zu differenzieren. Wird nachfolgend von einem *Lichtschalter* gesprochen, so ist hiermit der klassische Lichtschalter gemeint.

4.1 Schnittstellen-Entwürfe

In Abschnitt 3.2 wurden zwei Funktionen definiert, welche vom RCI ausgelöst werden können. Nun wird eine Benutzeroberfläche für das Steuerelement benötigt, welche es den Benutzer ermöglicht diese Aktionen auszulösen

Als rahmengebend für die verwendbare Oberfläche wurde eine quadratische Fläche von jeweils 8 cm Kantenlänge angenommen. Diese Größe ist für Lichtschalter (und verwandte Elemente wie Steckdosen) sehr üblich und soll einerseits dafür sorgen, dass eine bauliche Kompatibilität gewährleistet ist (also in vorhandene Wohnungen und Häuser eingebaut werden kann) als auch einen gewissen Wiedererkennungswert ermöglichen und somit den Einstieg vereinfachen soll.

Als Eingabemethode würden sich sowohl physische Taster als auch Touchflächen anbieten. Ein wichtiges Kriterium für die spätere einfache Verwendbarkeit ist die Möglichkeit, den Schalter bedienen zu können ohne den Blick darauf zu richten, da es sich hierbei um einen sehr üblichen Use Case handelt.

Während der Entwurfsphase wurden zwei verschiedene Varianten einer möglichen Bedienoberfläche erdacht. Diese werden nun kurz vorgestellt und gegenübergestellt.

4.1.1 Variante 1

Die erste Oberflächen-Variante besteht aus einer großen Touch-Fläche sowie vier darunter angeordneten physischen Tasten. Die Touch-Fläche löst die Toggle-Funktionalität aus, wohingegen die vier Tasten jeweils ein Preset aktivieren. Für jeden Preset gibt es darüber hinaus eine LED, welche als Farbvorschau für die über das Preset anwählbare Lichtfarbe und -Intensität fungiert.

Diese Farb-Vorschau aktiviert sich nur, wenn sich die Hand dem Schalter nähert, sodass diese bei Nichtverwendung deaktiviert und somit Personen, welche sich im Raum befinden, nicht durch leuchtende LEDs stören.

Die Touch-Fläche belegt absichtlich die meiste der zur Verfügung stehenden Fläche, damit bei einer blinden Verwendung eine hohe Chance besteht, die Touch-Fläche zu treffen.

Da die Verwendung der Touch-Fläche kein inhärentes taktilen Feedback besitzt, wird dies hier via eines Vibrationsmotors künstlich erzeugt.

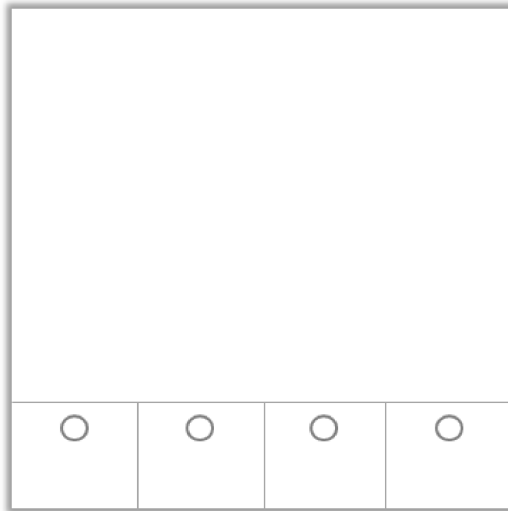


Abbildung 4.1: Front-Oberfläche von Variante 1

4.1.2 Variante 2

Während Variante 1 ein Hybrid zwischen Touch- und physischen Schaltflächen ist, ist Variante 2 hingegen komplett Touch-orientiert. Die komplette Front ist eine plane Fläche und bietet keine direkt ersichtlichen anderen Komponenten oder Aufteilungen. Dies ermöglicht es, dass eine Touch-Geste an einer beliebigen Stelle der gesamten Front die Toggle-Funktion auslösen kann und somit in diesem Aspekt mehr einem klassischen Lichtschalter ähnelt.

Die erweiterte Funktionalität - das Ändern des aktuell aktiven Presets - ist somit absichtlich verborgen. Dies wird ermöglicht, indem eine „Gedrückthalten“-Touch-Geste ausgeführt wird. Hierbei ändert sich die Touch-Funktionalität, indem in diesem Zustand die verfügbare Touch-Fläche der Schalter-Front in vier Quadranten unterteilt werden. Zur Veranschaulichung sind diese Quadranten in Abbildung 4.2 mit Rahmen unterteilt, diese sind jedoch nur zur Veranschaulichung und sind nicht sichtbar.

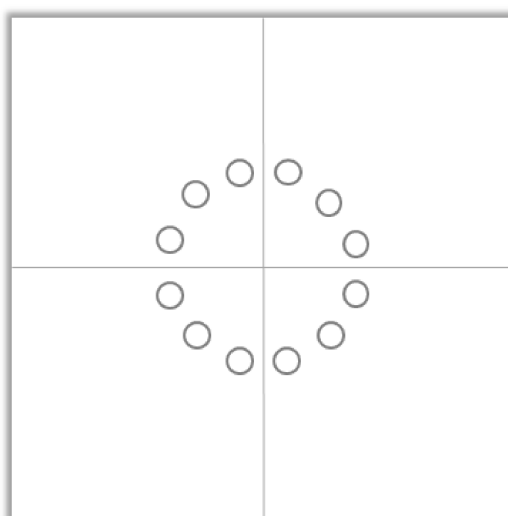


Abbildung 4.2: Front-Oberfläche von Variante 2

Da Variante 2 ausschließlich auf Touch-Flächen setzt, wird auch hier das taktile Feedback mittels

eines eingebauten Vibrationsmotors künstlich erzeugt.

In der Mitte befindet sich ein Ring aus einzelnen LEDs. Diese sind, wie auch die Quadranten, im Ruhezustand unsichtbar. Wird in den Preset-Auswahl-Modus gewechselt, so leuchten die LEDs für ihren jeweiligen Quadranten in der zum Preset passenden Vorschaufarbe.

Wird ein Preset aktiviert, so leuchtet für einen kurzen Moment der komplette LED-Ring in der Vorschau-Farbe, um so zusätzlich zum taktilen Feedback auch ein visuelles Feedback zu geben.

4.1.3 Gegenüberstellung

Beide vorgestellten Varianten haben ihre eigenen Vor- und Nachteile. Nachfolgend kann jedoch nur eine Variante weiter auskonzipiert werden, weswegen eine Entscheidung erfolgen muss.

Variante 1 profitiert von der konventionellen „Schalter-Haptik“ für die Preset-Schaltflächen und hat ausschließlich direkt ersichtliche Eingabemöglichkeiten. Die Preset-Schaltflächen sind jedoch sehr klein und somit potenziell schwer zu bedienen. Des Weiteren kann die Farbvorschau-Funktion den Nutzer verwirren, da zumindest ein Gast-Nutzer nicht verstehen würde warum in einem Lichtschalter farbliche LEDs aktiv werden.

Variante 2 hat eine „aufgeräumtere“ Oberfläche mit großen Schaltflächen und versteckt die potenziell für Gast-Nutzer verwirrenden Funktionen. Jedoch müssen genau diese versteckten Funktionen für den Haupt-Nutzer erlernt werden, ehe er diese finden (und somit nutzen) kann. Des Weiteren ist die Bedienung für den Haupt-Nutzer minimal langsamer, da bei der „Gedrückthalten“ Touch-Geste der Nutzer einen kleinen Moment warten muss.

Da in Variante 2 die Intuitivität für den Gast-Nutzer stärker vorhanden sein dürfte und die Nachteile von Variante 1 schwerer wiegen als von 2 wird nachfolgend weiter mit dem Ansatz von Variante 2 verfahren.

4.2 System-Bestandteile

Das Gesamtsystem besteht aus einer flexiblen Anzahl an RCI-Geräten sowie mindestens einem Gateway-Service. Ein RCI-Gerät besteht aus einer Platine (dem „Kern-Modul“) und einem Gehäuse.

4.2.1 Mikrocontroller

Kern des RCI muss ein Mikrocontroller sein, der in der Lage ist, die Touch-Signale zu registrieren und die entsprechenden Befehle an eine weitere Komponente weiterzureichen. Als primären Mikrocontroller bietet sich bspw. ein Arduino an. Arduinos gelten als Quasi-Standard für Einsteigerfreundliche Mikrocontroller-Entwicklung.[1]

Da jedoch eine Kommunikationsschnittstelle zwingend vonnöten ist, um den erkannten Benutzer-Befehl weiterzugeben und hierbei auf Wireless LAN gesetzt werden sollte (für die leichtere Kommunikation mit einer Server-Anwendung), reduzierte sich die Auswahl verwendbarer Arduinos ausschließlich auf den MKR1000. Diese sind jedoch sehr kostspielig (verglichen mit anderen Arduino-Produkten) und hätten die Anzahl an herstellbarer Prototypen auf ein einziges Exemplar limitiert.

Der ESP8266 ist weniger bekannt als ein Arduino und die Plattform ist nicht so stark verbreitet und dokumentiert, hat jedoch mehr Leistungsreserven und integrierte Wireless LAN Konnektivität. Mit Stückpreisen von einem Bruchteil des MKR1000 war auch die Herstellung mehrerer Prototypen realisierbar und daher wurde entschieden diesen als Plattform zu nutzen.[3]

Die Touch-Ansteuerung kann hierbei ein MPR121 genannter Mikrocontroller übernehmen. Dieser ist jedoch selbst nicht programmierbar und ist somit nur als Zusatzkomponente verwendbar.

Stromversorgung

Ein potenzieller Nachteil wird der Zwang für eine konstante Stromversorgung sein. Laut Spezifikationen dauert der Verbindungsaufbau über Wireless LAN mehrere Sekunden, womit zwingend die Latenz bis zur Ausführung des User-Befehls (hier: „Licht anschalten“) in untragbare Länge gezogen werden würde. Daher darf der Verbindungsaufbau nicht erst bei Tastendruck erfolgen. Da eine Batterie-basierte Stromversorgung jedoch - damit sinnvolle Laufzeiten erreicht werden können - voraussetzt, dass das Gerät für lange Zeiten (im Bereich von Stunden) in einen Stromsparmodus versetzt wird, in dem Wireless LAN abgeschaltet werden würde, schließt sich dies von selbst aus.

Die konstante Stromversorgung kann sowohl intern (Unterputz) als auch extern erfolgen. Da es sich hierbei um einen Prototypen handelt, wird der Weg der externen Stromversorgung gegangen. Eine mögliche Optimierung dieser Thematik wird im Kapitel *Ausblick* erläutert.

4.3 Gateway-Service

Es wird angenommen dass in jeder Wohnung, in der das RCI verwendet wird, mindestens ein Lichtsteuerungssystem verwendet wird. Ein Lichtsteuerungssystem beschreibt bspw. eine Basisstation von Philips Hue, Osram Lightify oder via Funk/Netzwerk ansteuerbare Relais-Komponenten. Damit die RCIs, die sich in der Wohnung befinden, entgegengenommene User-Befehle an eben diese Systeme weitergeben können, bedarf es eines sogenannten Gateway-Services.

Der *Gateway-Service* ist eine sich im jeweiligen Netzwerk befindliche Serveranwendung, welche in der Programmiersprache C# als .NET-Anwendung entwickelt wird. Es soll jedoch darauf geachtet werden dass diese Anwendung Mono-kompatibel bleibt. Mono ist eine freie Implementation des .NET Frameworks und eine Mono-Kompatibilität ermöglicht es, die Anwendung bspw. auf einem Raspberry Pi zu betreiben. Dies ist wichtig, da es ansonsten nicht-triviale Anforderungen an die Infrastruktur der jeweiligen Wohnung stellen würde. Ein Raspberry Pi lässt sich jedoch leicht für den dauerhaften Betrieb versteckt verbauen.

Es kann pro Wohnung mehrere Gateway-Services geben (z.B. um unterschiedliche Lichtsteuerungssysteme anzubinden), jedoch ist jeder Schalter an einen Gateway-Service gebunden. Dies ermöglicht es bspw., für das Wohnzimmer auf Philips Hue und im Badezimmer auf Funkschalter zu setzen, nicht jedoch eine Mischung daraus.

Das RCI ist hierbei funktionell vollkommen entkoppelt vom Gateway und der jeweiligen Gateway-Implementation, solange das (überschaubare) Kommunikationsprotokoll eingehalten wird. Dies ermöglicht es, mehrere Implementationen für unterschiedliche Lichtsteuerungssysteme vorzuhalten, ohne dass die Firmware- oder Konfiguration der RCIs angepasst werden muss.

4.3.1 Referenz-Implementation

Wie in Abschnitt 2.3 erwähnt, soll die Referenz-Implementation eines solchen Gateway-Services für das Philips Hue Lichtsteuerungssystem entwickelt werden.

Die Philips Hue Basisstation bietet hierfür eine REST-basierte API, welche vom internen Netzwerk aus angesprochen werden kann und über dieses sowohl der aktuelle Stand abgefragt werden kann als auch Kommandos abgesetzt werden können.

Die komplette Schnittstelle des Gateway-Services sowie die Firmware des RCI bleibt jedoch abstrakt und ist somit nicht fest an diese Implementation gekoppelt und kann somit ohne weitere Veränderung auch mit einer weiteren/alternativen Implementation verwendet werden.

4.4 Kommunikation

Die Kommunikation zwischen RCI und Gateway-Service erfolgt via MQTT. MQTT ist ein leichtgewichtiges Kommunikationsprotokoll, welches spezifisch für die Verwendung von „Internet of Things“-Anwendungen entwickelt wurde und auf dem Publish-Subscribe Messaging Pattern basiert. Dadurch eignet es sich hervorragend für die Anbindung der RCIs-Geräte zu ihren jeweiligen Gateways. [4]

Diese Kommunikation erfolgt über verschiedenartige Nachrichten, welche - je nach Typ - von entweder dem Gateway oder dem RCI versendet werden können.

4.4.1 Nachrichten

Das RCI-Protokoll umfasst vier verschiedene Nachrichten-Arten. Diese Nachrichten werden als JSON-formatierten Text serialisiert übertragen.

ToggleLightCommand

Funktion: Derzeitigen Lichtstatus umschalten. Dementsprechend entweder das Licht an- oder ausschalten.

Parameter: Raum-ID

ActivatePresetCommand

Funktion: Ein gegebenes Preset aktivieren.

Parameter: Raum-ID, Preset-Nummer

RequestZoneControlInterfaceInitializationCommand

Funktion: Initialisierungs-Daten anfordern.

Parameter: RCI-Seriennummer

InitializeZoneControlInterfaceCommand

Funktion: Ein spezifisches RCI-Gerät initialisieren. Dies kann auch ohne eine Anforderung und nach bereits erfolgter Initialisierung geschehen, z.B. wenn die Preset-Daten geändert wurden und das RCI darüber benachrichtigt werden muss.

Parameter: RCI-Seriennummer, Raum-ID, Menge an Preset-Vorschaufarben

4.4.2 Auto-Konfiguration

Während dem Boot-Prozess, nachdem die Kommunikation mit dem MQTT-Server hergestellt wurde, wird ein für die RCI-Seriennummer spezifisches Topic abonniert. Wurde das Abonnement bestätigt, so wird mittels des *RequestZoneControlInterfaceInitializationCommand* die Initialisierung angefordert und gewartet bis auf dem Topic die Antwort in Form einer *InitializeZoneControlInterfaceCommand*-Nachricht empfangen wurde.

Mitsamt der darin befindlichen Initialisierungsdaten wird die Initialisierung des RCI abgeschlossen und in den normalen Operationsmodus überführt.

Dieses Verhalten lehnt sich teilweise an das Verhalten von DHCP an. DHCP-Clients übermitteln beim Nachfragen nach einer IP-Adresse ihre eigene MAC-Adresse. DHCP-Server können so konfiguriert werden, dass sie für spezifische MAC-Adressen vordefinierte Werte (bspw. eine reservierte IP-Adresse) verwenden. Was bei DHCP somit MAC-Adressen und Netzwerkkonfigurationen sind, sind bei der RCI-Auto-Konfiguration Seriennummern und Licht-Konfigurationswerte.

4.5 Verhalten

Das RCI ist eine Zustandsmaschine. Es besitzt insgesamt fünf Zustände:

- Pre-Initialization
- Idle
- Pressing
- Preset-Selection
- Error

Direkt nach dem Boot-Vorgang befindet sich das RCI im Pre-Initialization Zustand. Hierbei wird, wie in Abschnitt 4.4.2 erläutert, auf eine Initialisierung vonseiten des Gateway-Services gewartet. Erfolgt solch eine Initialisierung, so geht das RCI in den Idle Zustand über.

Befindet sich das RCI im Idle Zustand und wird eine Touch-Berührung erkannt, so befindet es sich ab diesem Zeitpunkt im Pressing Zustand. Endet diese Berührung innerhalb von weniger als 500 Millisekunden, so wird die Toggle-Funktionalität ausgeführt und das Gerät geht wieder in den Idle Zustand über. Dauert die Berührung länger, so aktiviert sich stattdessen der Preset-Selection Zustand und erlaubt die Auswahl eines Presets. Wird dies innerhalb von wenigen Sekunden getan, so wird das Preset ausgewählt und wieder in den Idle Zustand gewechselt. Dauert dies länger, so wird die Aktion abgebrochen und ebenfalls wieder in den Idle Zustand übergegangen.

Sollte wirklich ein Fehlerfall eintreten, so tritt das Gerät in den Fehler-Zustand ein und kann einfach neugestartet werden. Da die RCI-Geräte vollkommen statuslos konzipiert wurden, besteht hierbei keine Gefahr für nachteilige Effekte oder Datenverlust.

5 Implementierung

Im Zuge dieses Kapitels sollen einige ausgewählte Themen, welche sich während der Implementierung der RCIs ergaben, näher betrachtet werden.

5.1 Veranschaulichung

Der Kern eines jeden RCI besteht aus einer Platine mitsamt verschiedener darauf befindlichen Komponenten. Abbildung 5.1 zeigt eine solche Platine.

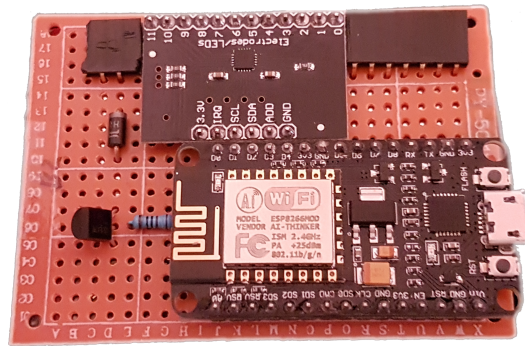


Abbildung 5.1: Kern-Modul

Der Gehäuse-Aufbau unterlag während der Prototyping-Phase einer Vielzahl von Iterationen, während erlernt wurde welche Möglichkeiten sich durch verschiedene Konstruktionsarten ergeben.

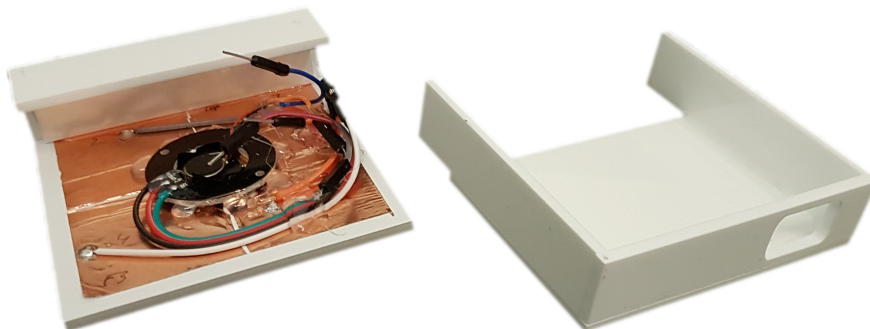


Abbildung 5.2: Gehäuse-Segmente

Der finale Stand zum Zeitpunkt der Erstellung dieses Dokuments ist eine Kombination aus 8 Acrylglas-Teilen, welche miteinander verklebt werden. Das verwendete Acrylglas-Material unterscheidet sich hierbei. Während die Frontplatte eine lichtdurchlässigere Milchglas-Optik besitzt, um das Licht der sich dahinter befindlichen LEDs durchscheinen zu lassen, sind die 7 restlichen Teile

aus wenig lichtdurchlässigem weißen Acrylglas.

Der untere Teil des Gehäuses hat hierbei eine Aussparung um ein Stromkabel (derzeit ein Micro USB Kabel, angebunden an ein Netzteil) an den internen Mikrocontroller zu führen.

Kombiniert werden die Einzelteile zu insgesamt zwei Segmenten, welche ineinander gesteckt werden können.

Abbildung 5.3 zeigt den RCI an einer Wand angebracht.



Abbildung 5.3: RCI an Wand angebracht

Hierbei sei angemerkt, dass der RCI hier nur auf das Gehäuse des konventionellen Lichtschalters geklebt wurde. Ein kompletter Abbau der vorhandenen Lichtinfrastruktur der Demo-Wohnung war - zumindest in der Prototyp-Phase des Projekts - keine adäquate Möglichkeit, weswegen hier auf solch einen Kompromiss gesetzt wurde. Das RCI-Gehäuse hat eine Tiefe von 2,4 cm, der darunterliegende Lichtschalter von 1,2 cm, wodurch sich insgesamt eine effektive Tiefe von 150% der geplanten Tiefe ergab.

5.2 Feedback-Verhalten

Das vorgesehene taktile Feedback kann entweder beim ersten Berühren ("KeyDown"), beim Loslassen ("KeyPress") oder nach einem spezifischen zeitlichen Intervall nach einem dieser Events erfolgen.

Die erste beinahe-finale Firmware-Iteration sah vor, dass erst beim Loslassen die Vibration erfolgte. Da sich die beiden verfügbaren Funktionen durch die Länge des „Gedrückthaltens“ unterschieden, erschien dies sinnvoll, da sich erst beim Loslassen die gedrückte Dauer ermitteln ließ. Dies stellte sich jedoch als sehr unpraktisch heraus, da durch das Wegnehmen des Fingers die Vibration nur für wenige Millisekunden gespürt werden konnte, wodurch nur das (eigentlich ungeplante) akus-

tische Feedback, welches durch die Vibration des Vibrationsmotors entstand, zur Wahrnehmung übrigblieb.

Daraus lernend wurde die nächste Iteration dementsprechend umkonzipiert, sodass beim ersten Berühren bereits die erste Vibration erfolgte und bei einem Gedrückthalten was lange genug währt um die zweite Aktivität zu initiieren eine erneute Vibration erfolgt.

Eigene Tests waren vielversprechend, da durch diese Veränderung selbst bei sehr flüchtigen und schnellen Bewegungen das Feedback sehr deutlich spürbar war, jedoch wird erst der Test mit weiteren (objektiven) Testern zeigen ob dies ausreicht.

5.3 Firmware-Update

Der RCI erlaubt es, im laufenden Betrieb über Wireless LAN die eigene Firmware zu aktualisieren. Hierfür lauscht der RCI nach einem erfolgreichen Boot-Prozess auf einem spezifizierten Port auf eingehende Update-Kommandos.

Dies erleichtert die Wartung, da bei einem Problem leicht eine aktualisierte Firmware eingespielt werden kann. Ausschließlich bei Problemen, die einen problemlosen Boot-Prozess verhindern, muss ein direkter physischer Zugriff erfolgen.

In einem Produktiv-Szenario würde diese Schnittstelle selbstverständlich Passwort-gesichert sein, um Sicherheit vor unautorisiertem Zugriff zu gewährleisten.

6 Test

In diesem Abschnitt soll nun gezeigt werden, wie die in Abschnitt 3 definierten funktionalen und nicht-funktionalen Anforderungen auf deren korrekte und ausreichende Umsetzung getestet wurden.

6.1 Funktionalität

Der RCI wurde über einen Zeitraum von einer Woche durchgehend produktiv verwendet. Hierbei ergaben sich keinerlei Probleme und zu jedem Zeitpunkt war die Funktionalität vollständig und latenzfrei einsetzbar.

Die funktionellen Aspekte des RCI können somit als ausreichend getestet angesehen werden können.

6.2 User-Studie

Für die Überprüfung der nicht-funktionalen Anforderungen wurde eine User-Studie mit drei Teilnehmern durchgeführt. Hierfür wurde der Fragebogen „QUESI - Questionnaire for Intuitive Use“ verwendet, welcher spezifisch auf die Fragestellungen der Intuitiven Nutzung von Geräten eingeht.

6.2.1 Erwartungen

Wenn alle in den vergangenen Kapiteln aufgestellten Thesen korrekt waren, so sollten alle Test-Kandidaten sofort in der Lage sein mit dem System umzugehen zu können (zumindest im Rahmen der Benutzer-Gruppe „Gast-Benutzer“).

Dass ein Benutzer von selbst den Preset-Aktivierungs-Modus findet ist eher unwahrscheinlich, jedoch ist dies ohnehin nicht erwünscht.

6.2.2 Ergebnisse

Der Fragebogen erlaubt Antworten auf einer Skala von „trifft gar nicht zu“ bis „trifft völlig zu“. Um auf diesen Werten mathematisch agieren zu können werden diese Werte nachfolgend mit einer Skala von 1 bis 5 verwendet, wobei 5 „trifft völlig zu“ entspricht.

Es wurde darauf geachtet, den Test-Kandidaten vor der Test-Nutzung keinerlei Hinweise oder Tips zu geben, um das Ergebnis nicht zu verfälschen. Die einzige Anmerkung war, so zu handhaben wie man es halt von einem Lichtschalter gewöhnt ist.

Die Ergebnisse-Werte der Fragebögen sind in Tabelle 6.2.2 gelistet.

6.2.3 Beobachtungen

Kandidaten A und B kamen auf Anhieb mit dem Gerät klar und identifizierten es sofort als Lichtschalter, da es an einer für Lichtschalter üblichen Position angebracht war und dessen übliche Form und Größe hatte.

#	Frage	A	B	C	Ø
1.	Es gelang mir, das System ohne Nachdenken zu benutzen	5	5	4	4.66
2.	Ich habe erreicht, was ich mit dem System erreichen wollte	5	5	5	5.00
3.	Mir war sofort klar, wie das System funktioniert	4	4	3	3.66
4.	Der Umgang mit dem System erschien mir vertraut	5	4	2	3.66
5.	Bei der Benutzung des Systems sind keine Probleme aufgetreten	4	5	5	4.66
6.	Die Systembenutzung war unkompliziert	5	5	4	4.66
7.	Es gelang mir, meine Ziele so zu erreichen, wie ich es mir vorgestellt habe	5	3	5	4.33
8.	Es fiel mir von Anfang an leicht, das System zu benutzen	5	5	4	4.66
9.	Mir war immer klar, was ich tun musste, um das System zu benutzen	5	5	3	4.33
10.	Die Benutzung des Systems verlief reibungslos	4	5	5	4.66
11.	Ich musste mich kaum auf die Benutzung des Systems konzentrieren	4	5	3	4.00
12.	Das System hat mich dabei unterstützt, meine Ziele vollständig zu erreichen	5	5	5	5.00
13.	Die Benutzung des Systems war mir auf Anhieb klar	4	5	3	4.00
14.	Ich tat immer automatisch das Richtige, um mein Ziel zu erreichen	4	3	4	3.66

Tabelle 6.1: Ergebnisse des QUESI Fragebogens

Kandidatin C hingegen war hier zögerlicher, da sie verwirrt über das Stromkabel und der fehlenden sichtbaren Kipp-Möglichkeit war. Sobald sie jedoch den Schalter berührte und durch das taktile Feedback verstand wie der Lichtschalter funktionierte, kam sie auch sehr gut zurecht.

Test-Kandidat A war direkt begeistert und schien sofort sehr gut damit klarzukommen. Da er bemerkte dass das Gerät auf Touch reagiert, versuche er sogar auf Anhieb via Touch-Geste ein dimmen durchzuführen. Diese Funktionalität wurde zwar (da dies via der Presets gelöst werden soll) absichtlich nicht implementiert, war jedoch interessant dass dies versucht wurde.

Interessanterweise hatte der Vibrationsmotor in manchen Fällen einen sekundären Nutzen. In den prototypisch-gebauten RCIs wurden (ungeplant ausgelöst durch unzureichende Lagerbestände) Vibrationsmotoren mit unterschiedlicher Lautheit verbaut. Ein Kandidat merkte an, dass der „lauteste“ RCI ihm gefiel, da das hierdurch zusätzliche auditive Feedback praktisch wäre.

6.2.4 Auswertung

Da die funktionalen Anforderungen bereits in Abschnitt 6.1 als getestet anerkannt wurde, geht es hier primär um die nicht-funktionalen Anforderungen. Diese wurden in der Analyse aufgestellt und bestehen aus Intuitivität und Feedback.

Das taktile *Feedback* hat - laut den Aussagen der Tester - sehr gut funktioniert und es war zu jedem Zeitpunkt gut erfüllbar gewesen, ob der Schalter den Druck registriert hat. Dies galt auch für Fälle, in denen der Tester bspw. schnell einen Raum verlassen hat während er das Licht deaktiviert hat und so das RCI nur kurz berührt hat und darüber hinaus das visuelle Feedback des sich deaktivierenden Lichts nicht sehen konnte.

Generell ist anhand der Werte erkennbar, dass das Gerät auch ohne Einweisung gut verwendet werden konnte. Einschnitte, was die *Intuitivität* angeht, sind jedoch erkennbar, bspw. anhand der Frage 3 und 4. Der Grund hierfür ist verbunden mit der Anmerkung der Test-Kandidatin C, dass die Optik des Gehäuses noch zu wenig nach einem Lichtschalter aussieht. Diese Problematik wird im Ausblick in Abschnitt 8.1 angesprochen.

Keiner der drei Studienteilnehmer kam von selbst auf die Idee, die „Gedrückt halten“ Geste auszuführen und somit auf den Preset-Wechsel-Modus zu kommen. Dies erfordert zwar - wie geplant -

einen Hinweis (sei es nun durch eine Anleitung oder einen auf dem späteren Produkt befindlichen Aufkleber), jedoch hatte es den gewünschten Effekt dass kein Nutzer über plötzlich unerwartetes visuelles Feedback (durch das Aufleuchten der LEDs) verwirrt werden konnte.

7 Zusammenfassung

Das Ziel dieser Arbeit war die Entwicklung eines alternativen Bedienelements für verschiedene Lichtsteuersysteme.

Zum Anfang dieser Arbeit wurde hierfür die Situation aktueller Lichtsteuersysteme und deren Bedienelementen sondiert und ein Lösungsansatz für eine vorhandene Problematik skizziert.

Während der Analyse des Projektziels stellten sich die Anforderungen heraus, welche erfüllt werden müssten damit das RoomControlInterface-getaufte Gerät (kurz: RCI) als Alternative zu einem konventionellen Lichtschalter angewendet werden kann.

Im Zuge des Entwurfs wurde die technische Grundlage für ein solches RCI-Gerät konzipiert und ein Gesamtsystem basierend aus Bedienelementen und einer Server-Komponente geplant.

Im Laufe der Implementierung wurden sekundäre Komponenten wie bspw. ein Gehäuse gebaut sowie Aspekte des RCI optimiert, wie z.B. ein sinnvolleres Feedback-Verhalten sowie die Möglichkeit, kabellos ein Firmware-Update durchführen zu können.

Im Rahmen des Prototypen-Testings wurde eine Studie mit drei Teilnehmern durchgeführt. Diese ergab größtenteils positives Feedback, was die Bedienung und die Intuitivität angeht.

Insgesamt wurden alle Anforderungen an das Projekt hinreichend erfüllt, sodass es als Erfolg angesehen werden kann. Es gibt jedoch Aspekte, welche weiter optimiert werden können, u.a. zur Steigerung des Wiedererkennungswerts in Hinsicht auf traditionelle Lichtschalter-Lösungen. Es lässt sich aber sagen, dass die in dieser Arbeit erarbeitete Lösung das Potenzial hat, sich gut in das vorhandene Gesamtthema integriert und produktiv angewendet kann.

8 Ausblick

Während der Entwurf- und Implementationsphase ergaben sich viele Themengebiete und Ideen, welche der Funktionalität zuträglich gewesen wäre, jedoch nicht im Rahmen dieser Arbeit genauer betrachtet werden konnten.

Innerhalb dieses Kapitels werden nun ausgewählte Thematiken angeschnitten.

8.1 Optik

Basierend auf den Aussagen aus Abschnitt 6.2 lassen sich zwei Optimierungsideen herauslesen: Die Gehäusetiefe weiter verringern sowie die Front mehr „drückbar“ aussehen lassen.

Damit ersteres so umgesetzt werden kann, müsste die Platine jedoch mittels Ätzverfahren hergestellt werden anstatt wie hier mittels manuellem Löten. Dies verkompliziert den Herstellungsprozess zwar stark, erlaubt jedoch eine geringere Bautiefe und somit auch ein flacheres Gehäuse.

Um die „drückbare“ Optik herzustellen, wäre es möglich, in die Front einen abgesetzten Bereich einzulassen, welcher zwar ebenfalls mechanisch gesehen nicht beweglich wäre, jedoch optisch so aussehen würde und somit denselben Zweck erfüllt. Der Effekt des „Drückens“ wird dann mittels des taktilen Feedbacks simuliert.

Im Zuge dieser Änderungen wäre es ratsam, einen Designer miteinzubeziehen, um die angestrebten Optimierungen aus Expertensicht verifizieren zu lassen.

8.2 Stromversorgung

Eine Versorgung mit 230 Volt Netzspannung mittels eines integrierten Netzteils hätte den Rahmen dieser Arbeit überschritten, insbesondere da durch die fehlende formale Elektrotechnik-Ausbildung bei solch einem Vorhaben leicht Lebensgefahr bestanden hätte. Dies ist jedoch ein Aspekt, der zu einem späteren Zeitpunkt erneut evaluiert wird. Dies wurde auch durch den Kommentar von Testperson C bekräftigt, nach deren Aussage das RCI nicht als Lichtschalter wiedererkannt wurde, da ein Kabel reingeführt ist und dies somit eher nach etwas anderem aussieht.

8.3 Barrierefreiheit

Im Rahmen dieser Arbeit wurde bewusst das Thema Barrierefreiheit ausgegrenzt, da es für den Prototypen nicht relevant war, sowie - durch die inhärente Komplexität die mit der Beachtung der vielen Regeln der Barrierefreiheit einhergehen würde - schlicht und einfach den Rahmen dieser Arbeit gesprengt hätte. Soll der Prototyp jedoch in ein marktreifes und flächendeckend genutztes Produkt ohne Ausgrenzung von spezifischen Benutzergruppen verwendet werden können, so müsste dies selbstverständlich nachgeholt werden.

Literaturverzeichnis

- [1] ARDUINO: *Arduino Products*. – Homepage - Zugriffen am: 08.04.2017 - Verfügbar unter <https://www.arduino.cc/en/Main/Products>
 - [2] ENGELHARDT, Erich F.: *Hausautomation mit Raspberry Pi*. Franzis Verlag, 2016. – ISBN 978-3-645-60313-3
 - [3] ESPRESSIF SYSTEMS: *ESP8266EX Overview — Espressif Systems*. – Homepage - Zugriffen am: 08.04.2017 - Verfügbar unter <https://espressif.com/en/products/hardware/esp8266ex/overview>
 - [4] MQTT: *MQTT*. – Homepage - Zugriffen am: 08.04.2017 - Verfügbar unter <http://mqtt.org/>
-