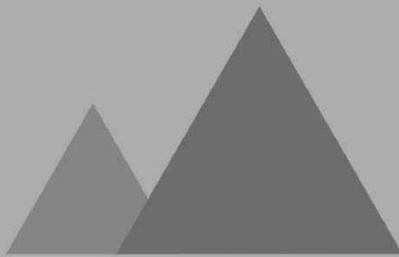


Please delete placeholder  
and replace with own  
picture



## Pflichtenheft LED-Mapping

Dieses Dokument definiert die Ziele der Bachelorthesis  
„LED-Mapping“.

Bachelorthesis: Pflichtenheft  
Patrik Aebischer, Elia Bösiger  
Biel, 06. März 2018

# Inhaltsverzeichnis

Abkürzungs- und Begriffsverzeichnis	1
Abbildungsverzeichnis	1
Literaturverzeichnis	1
Änderungsübersicht	2
1 Beschrieb	3
1.1 Offizielle Aufgabenstellung	3
1.2 Produkteinsatz	3
1.3 Ablauf	3
1.4 Detaillierte Beschreibung	4
2 Vorgehen	5
3 Stakeholder	5
4 Systemabgrenzung	6
4.1 Use-Case-Diagramm	6
4.2 Was ist nicht Teil des Projektes	6
5 Anforderungen	7
5.1 Muss-Kriterien	7
5.2 Kann-Kriterien	8
6 Test-Cases	8
6.1 Muss-Kriterien	8
6.2 Kann-Kriterien	9

## Abkürzungs- und Begriffsverzeichnis

Abkürzung/Begriff	Beschreibung
LED	Light Emitting Diode (Deutsch: lichtemittierende Diode)
App	Application Software (Deutsch: Anwendungssoftware)
Leuchtquelle	Zum Beispiel ein Pixel, ein LED oder eine ähnliche Leuchtquelle, die unabhängig von anderen angesteuert werden kann.
User Interface (UI)	Über das User Interface (Benutzerschnittstelle) werden den Services Befehle erteilt, sowie wichtige Daten visualisiert.
Agent	Der Agent ist das Herzstück des Systems. Er führt mehrere Services so aus, dass ein funktionstüchtiges Programm entsteht.
Service	Ein Service hat nur einen bestimmten Zweck (zum Beispiel Leuchtquelle ein-/ausschalten) und interessiert sich nicht, was um ihn sonst noch passiert.
IoT	Internet of Things (Geräte, die über das Internet miteinander kommunizieren).
MQTT	Message Queuing Telemetry Transport. Publish / Subscribing-Protokoll, das vorallem im IoT Verwendung findet.
Tinkerforge	Einfache Elektronikbauteile für diverse Anwendungszwecke. Zum Beispiel messen von Temperaturen oder steuern von LEDs.

## Abbildungsverzeichnis

Abbildung 1 Komponentendiagramm	4
Abbildung 2 Use-Case-Diagramm	6

## Literaturverzeichnis

Koenig, R. (2018). *Bachelorthesis-Aufgabe*. Biel.

## Änderungsübersicht

Datum	Autor	Beschreibung der Änderung	Betroffene Kapitel
06.02.18	Patrik Aebischer Elia Bösiger	Erste Version	Alle Kapitel

# 1 Beschrieb

## 1.1 Offizielle Aufgabenstellung

*Diese Arbeit beschäftigt sich mit der Ansteuerung von komplexen Lichtszenen (künstlicher Sternenhimmel / beleuchtete Krippenspiele / optische Werbung / ...), bei denen die räumliche Verteilung der einzelnen adressierbaren Leuchtmittel grundsätzlich unbekannt ist.*

*Dazu wird einem optischen Verfahren ad Hoc automatisch der relative Standort der einzelnen Leuchtmittel in Bezug zu einem Referenzpunkt ermittelt, sodass mit Hilfe der entstandenen Mapping Lichtszenen für den Referenzpunkt korrekt erstellt werden können.*

*Das Verfahren muss automatisch, mit Hilfe von handelsüblicher gut verfügbarer Hardware erfolgen, und muss in der Handhabung benutzerfreundlich und möglichst einfach zu bedienen sein.*

(Koenig, 2018)

## 1.2 Produkteinsatz

In herkömmlichen Bildschirmen ist jedes Pixel genau einem „geographischen“ Punkt zugeordnet. Ein Programm kann nun ganz einfach jedes einzelne Pixel ansteuern und ein Muster erzeugen. Bei vielen Anwendungen im täglichen Gebrauch ist dies jedoch nicht der Fall. Leuchtquellen wie LEDs von Lichterketten, Strassenlaternen oder Innenbeleuchtungen können zufällig angeordnet sein. Und jedes Mal, wenn etwas geändert wird, sind die Leuchtquellen wieder an anderen Positionen und müssen mühsam von Hand neu konfiguriert werden. Unsere Bachelorthesis soll diesen Mappingvorgang automatisieren. Beispielsweise will jemand eine Party mit dem Thema Italien organisieren. Er besitzt viele LED-Lichterketten. Die einzelnen LEDs sind RGBW-Wiedergabefähig. Das bedeutet, sie können die Farben Rot, Grün, Blau und weiss annehmen. Je nach Mischverhältnis können die meisten für das menschliche Auge sichtbaren Farben dargestellt werden. Mit ihnen will er jetzt nur für einen Abend das Wohnzimmer dekorieren. Verlegt und angeschlossen ist bereits alles, jetzt muss man nur noch wissen, welche LED sich wo befindet. Hat man viele LEDs, gestaltet sich das sehr schwierig, da man jede einzelne ansteuern und sich deren Nummer merken muss. Mit der mobilen App unserer Thesis können die einzelnen LEDs ganz einfach lokalisiert und anschliessend angesteuert werden. Dazu richtet man einfach eine Kamera so auf das Szenario, dass alle LEDs in ihrem Sichtfeld sind und klickt auf „mappen“. Nach dem automatischen Mapping können die LEDs ganz einfach, zum Beispiel in den Farben Italiens (grün, weiss, rot), zum Leuchten gebracht werden.

Wichtig zu beachten ist allerdings, dass das Ganze nur in einer Ebene funktioniert. Die Leuchtquelle müssen zwar nicht zwingend alle in der gleichen Ebene angeordnet sein, das Bild erscheint aber immer nur „zweidimensional“ und nur von einem einzigen Punkt aus als korrekt (Parallaxenfehler können auftreten). Es ist also schlussendlich eine zweidimensionale Projektion der tatsächlichen Anordnung der Leuchtquellen (wie ein Foto).

Zielgruppen sind alle Personen, die ein Smartphone bedienen können und Freude an Leuchtdekorations haben. Alle die nicht immer die gleiche Dekoration haben, sich aber keine Zeit für ein mühsames manuelles Einstellen nehmen wollen, können auf unsere Services zurückgreifen.

## 1.3 Ablauf

Nach dem Starten der mobilen App befindet sich der Benutzer in einem Menu. Er kann hier entscheiden, ob er zum Beispiel eine alte Konfiguration laden oder ein neues Mappen starten will. Sofern alle benötigten Services vorhanden sind, kann der Benutzer das Mappen starten. Dazu richtet er eine Kamera (die vom Handy, oder eine externe) auf den Bereich wo sich die Leuchtquellen befinden. Wichtig ist, dass die Kamera sehr ruhig gehalten wird. Der Benutzer betätigt einen Button auf dem Handy und das Mappen wird gestartet. Nacheinander (später vielleicht aus Zeitgründen auch mehrere zusammen) leuchten die Leuchtquellen (zum Beispiel LEDs) auf und werden gemappt. Nachdem der Mappingvorgang erfolgreich abgeschlossen wurde, kann der Benutzer über das Menu zu verschiedenen Aktionen gelangen. Zum Beispiel kann er jeder Leuchtquelle eine Farbe zuweisen und ihr sagen, ob sie leuchten soll. Weitere Aktionen sind im Kapitel „Funktionale Anforderungen“ weiter unten in diesem Dokument zu finden.

## 1.4 Detaillierte Beschreibung

Das Ziel des Projektes ist, verschiedene Services zur Verfügung zu stellen, die miteinander so kommunizieren, dass zufällig angeordnete Leuchtquellen über ein User Interface gemappt und anschliessend angesteuert werden können. Das gesamte Programm funktioniert nur genau dann, wenn alle erforderlichen Services zur Verfügung stehen. Ein einzelner Service alleine ist verhältnismässig nutzlos. Der Vorteil dieser Struktur ist jedoch, dass ein einzelner Service von jemand anderem problemlos für ein anderes Gerät geschrieben werden kann, und wenn dieser die Schnittstellen richtig implementiert hat, kann der einzelne Service ganz einfach ausgetauscht werden. Im Zentrum dieser Struktur steht ein Agent. Er nimmt Daten von den verschiedenen Services entgegen und leitet entsprechende Reaktionen ein.

Mit Hilfe eines Kamera-Services sollen alle Leuchtquellen auf einmal erfasst werden. Der Benutzer stellt eine Kamera (in unserem Fall eine Handykamera) auf ein Stativ und richtet sie auf den Bereich, den er erfassen will. Nun kann er über ein User-Interface-Service (in unserem Falle eine App), den Befehl zum Mappen geben. Der Agent nimmt ihn entgegen und meldet dem Kamera-Service, dass er ein erstes Bild schiessen soll. Sobald dieses Bild beim Agent eingetroffen ist, merkt er sich dieses und sagt dem Leuchtquellen-Service, dass er die erste Leuchtquelle, zum Beispiel eine einzelnes LED, einschalten soll. Der Leuchtquellen-Service meldet dem Agent das Einschalten. Der Agent wiederum verlangt nun vom Kamera-Service ein nächstes Bild. Sobald er dieses erhalten hat, leitet er die beiden Bilder an den Datenverarbeitungs-Service weiter. Dieser vergleicht die beiden Bilder und stellt fest, an welcher Position sich die Helligkeit drastisch verändert hat. Dort befindet sich nun die erste Leuchtquelle, von welcher der Service sich jetzt die Position und die ID merkt. Dieses Verfahren geht nun so lange weiter, bis alle Leuchtquellen erfasst wurden. Da jeder weitere Schritt immer vom Vorangegangenen abhängig ist und die Programmierung entsprechend erfolgt, werden Race-Conditions gleich von Vornherein verhindert. Der Agent zum Beispiel kann erst weiter arbeiten, wenn er vom Datenverarbeitungs-Service gemeldet bekommen hat, dass der vorhergehende Schritt abgeschlossen wurde.

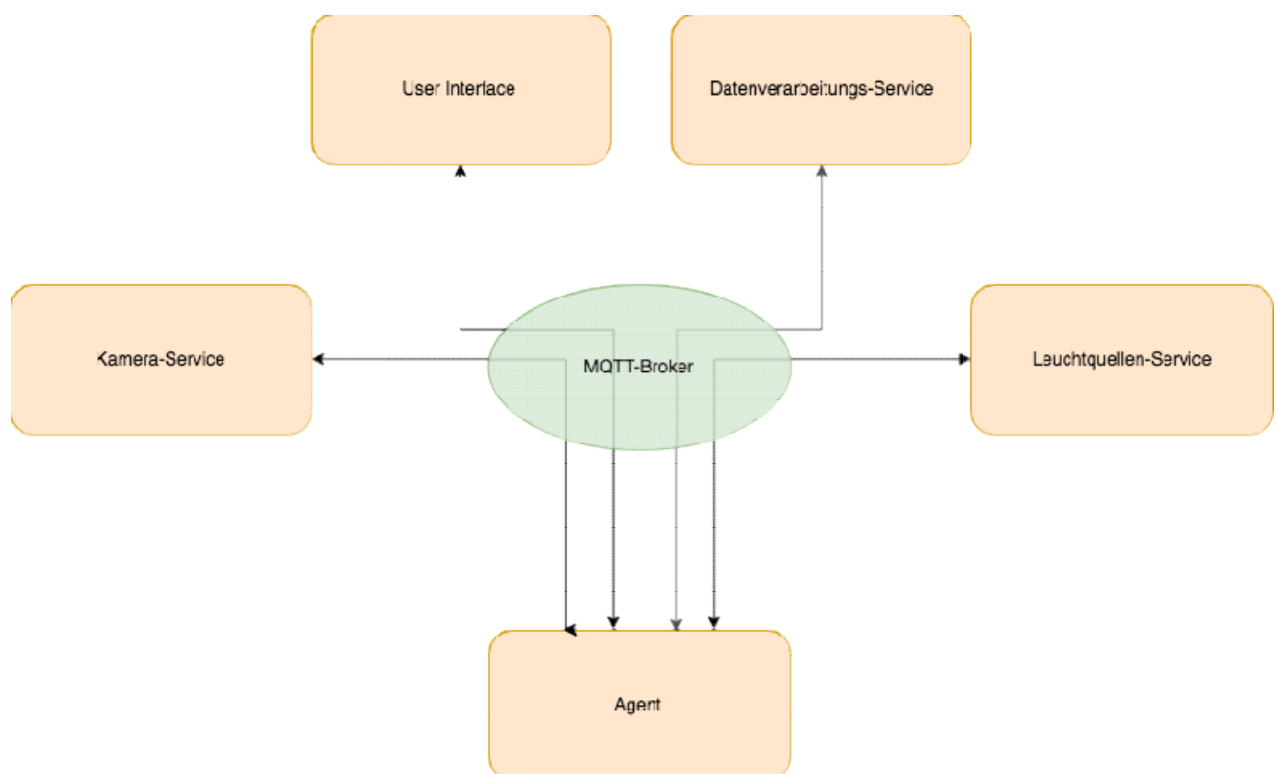


Abbildung 1 Komponentendiagramm

## 2 Vorgehen

Das Endprodukt soll aus verschiedenen Services bestehen. Das bedeutet, die einzelnen Komponenten Kamera, User Interface und Leuchtquellensteuerung funktionieren in sich geschlossen und stellen ausschliesslich benötigte Daten einer Schnittstelle zur Verfügung. Dies ermöglicht ein einfaches Austauschen der Komponenten, sofern sie die Anforderungen an die Schnittstelle erfüllen. Das „Managen“ der Schnittstelle übernimmt ein Agent. Er bestimmt, abhängig von Benutzereingaben, wann welcher Service was zu tun hat. Die Kommunikation läuft über das IoT-Protokoll MQTT, welches ebenfalls ein eigener Service darstellt.

Als erstes werden ein Agent, sowie die Services für die Kamera, die Kommunikation und die Mobile-App erstellt. Als Leuchtquelle wird in einem ersten Schritt ein Bildschirm dienen. Auf ihm wird ein grobes Raster dargestellt, welches verschiedene Farben und Helligkeitsstufen annehmen kann. Damit wird eine einfache Leuchtquelle simuliert und erste Erfahrungen mit der Bilderfassung gesammelt. Sobald alle Tests dafür erfolgreich abgeschlossen worden sind, wird ein eigener Service für die LEDs bereitgestellt. Ebenfalls wird die erste Version der Mobile-App nur grundsätzliche Funktionen wie das Erfassen der Pixel und deren Ein- und Ausschalten enthalten.

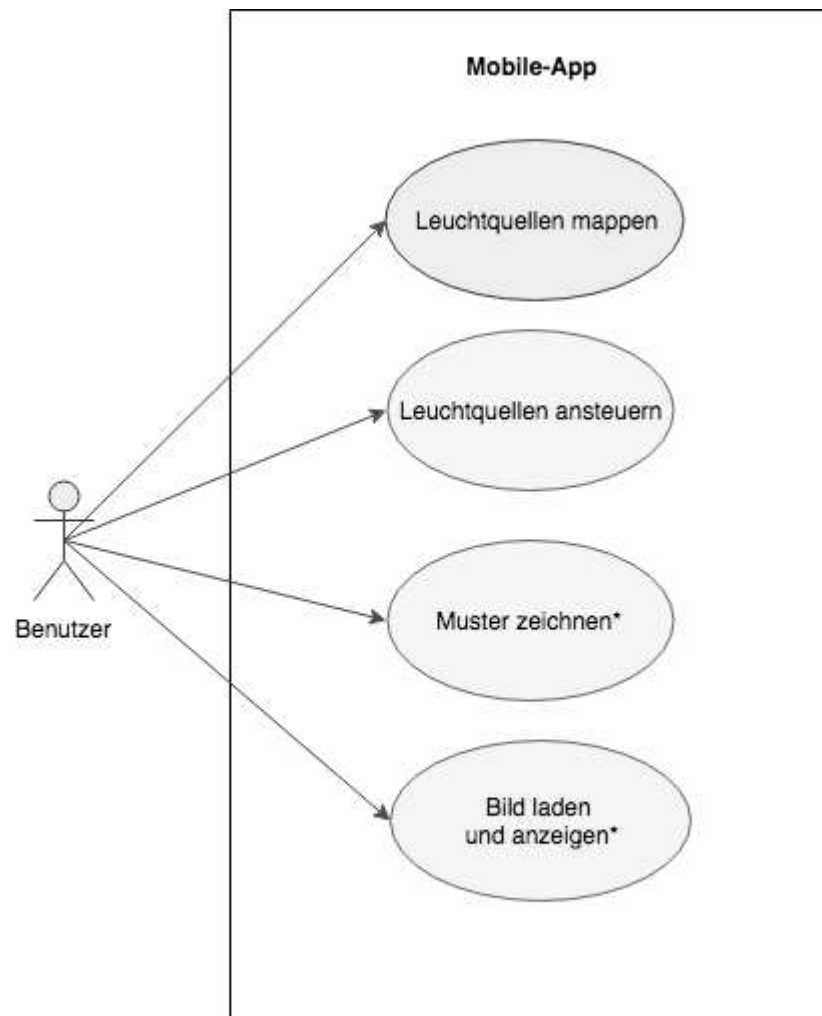
In weiteren Schritten wird der Bildschirm durch LEDs ersetzt, sowie die Mobile-App erweitert.

## 3 Stakeholder

<b>Zielpublikum</b>	Leute, die Freude an Lichtinstallationen haben und sich nicht mit technischen Details abgeben wollen.
<b>Projektleiter</b>	Elia Bösiger
<b>Entwicklungsteam</b>	Patrik Aebischer Elia Bösiger
<b>Betreuer</b>	Reto Koenig
<b>Experte</b>	Igor Metz

## 4 Systemabgrenzung

### 4.1 Use-Case-Diagramm



\* Optionale Anforderungen

Abbildung 2 Use-Case-Diagramm

### 4.2 Was ist nicht Teil des Projektes

Modul	Beschreibung
Wartung	Die Wartung der Applikation ist nicht Teil des Projektes.
Mehrsprachigkeit	Die Applikation wird lediglich in Deutsch realisiert.
Kompatibilität der Mobile-App	Die Mobile-App wird lediglich für Android-Geräte entwickelt.
3D	Die Motive, die von diesem Projekt schlussendlich dargestellt werden sind ausschliesslich zweidimensional (wie ein Foto). Die 3. Dimension ist kein Teil dieses Projektes.



## 5 Anforderungen

### 5.1 Muss-Kriterien

Modul	Beschreibung
Leuchtquellen mappen	Der Benutzer kann eine Kamera auf eine oder mehrere Leuchtquellen richten und diese automatisch mappen lassen. Mappen bedeutet, die Leuchtquellen werden einzeln durchgeschaltet und deren Positionen gespeichert. Die Lokalisierung der Leuchtquellen funktioniert nur zweidimensional. Der Benutzer ist dafür verantwortlich, dass der Bereich ganz erfasst wird, er sich aber nicht so weit davon entfernt hält, dass einzelne Quellen nicht mehr unterschieden werden können. Hat er den Bereich gewählt, startet er durch Knopfdruck das Mapping. Ist dieser Vorgang abgeschlossen, gibt die Mobile-App dem Benutzer eine entsprechende Rückmeldung. Während des Mapping-Vorgangs ist der Benutzer dafür verantwortlich, dass die Kamera ruhig gehalten wird und sich nicht zu sehr bewegt.
Leuchtquellen ansteuern	Der Benutzer kann die Leuchtquellen einzeln ansteuern und ihnen zum Beispiel eine fixe Farbe oder Helligkeit zuweisen. Auch kann er ihnen hier sagen, ob sie ein oder ausgeschaltet sind.
Mobile-App	Die Mobile-App wird für Android Geräte entwickelt. Sie enthält grundsätzlich nur das Userinterface, welches mobilefreundlich gestaltet wird. Es kann via Android -Tablets und -Smartphones auf die Mobile-App zugegriffen werden.
Kamera-Service	Der Kamera-Service ist für die Kamera zuständig. Er schiesst Fotos und stellt diese dem Agent zur Verfügung.
Bildschirm-Service	Der Bildschirm-Service ist ein Leuchtquellen-Service. Er ist für das korrekte Anzeigen des Rasters (welches in der ersten Testphase anstelle von anderen Leuchtquellen benutzt wird) zuständig.
LED-Strip-Service	Der LED-Strip-Service ist in einer zweiten Phase zuständig für das Schalten der LED-Strips von Tinkerforge.
Datenverarbeitungs-Service	Der Datenverarbeitungs-Service vergleicht die verschiedenen Bilder miteinander und definiert schlussendlich die Position der einzelnen Leuchtquellen.
Agent	Der Agent verwaltet alle Services und ist für das Funktionieren der gesamten Anwendung verantwortlich. Er nimmt Befehle von der Mobile-App entgegen und ist zuständig, dass diese korrekt ausgeführt werden. Wurde ein solcher Befehl richtig ausgeführt, meldet er dies an die Mobile-App und somit an den Benutzer zurück.
Kommunikation	Die Kommunikation läuft über einen MQTT-Broker, welcher sich im gleichen Netzwerk wie alle Services befinden muss. MQTT ist die Schnittstelle zu sämtlichen Services. Hier ist definiert in welcher Form die Befehle und Daten abgelegt werden müssen, damit die anderen Services verstehen, was sie zu tun haben.

## 5.2 Kann-Kriterien

Modul	Beschreibung
Muster zeichnen	Der Benutzer kann, zum Beispiel mit dem Finger, einfache Muster auf den Bildschirm zeichnen, welche dann von den Leuchtquellen abgebildet werden.
Bild laden und anzeigen	Der Benutzer kann ein vorhandenes Bild laden (zum Beispiel ein Schweizerkreuz) und dies von den Leuchtquellen abbilden lassen. Dabei ist zu beachten, dass die "Auflösung", welche von den Leuchtquellen dargestellt werden kann, begrenzt ist.

## 6 Test-Cases

### 6.1 Muss-Kriterien

Modul	Beschreibung
Leuchtquellen mappen	Die zweidimensionalen Leuchtquellen werden erkannt und korrekt gemappt. Die Tests finden unter „Laborverhältnissen“ statt. Das bedeutet, wir können einen optimalen Kontrast von der Oberfläche, auf welcher die Leuchtquelle liegt, zur Leuchtquelle selber bilden. Auch das Lichtverhältnis können wir bestimmen (nicht zu hell und nicht zu dunkel).
Leuchtquellen ansteuern	Die Leuchtquellen werden nach erfolgreichem Mapping auch korrekt angesteuert.
Mobile-App	Die Handhabung der Mobile-App ist intuitiv.
Kamera-Service	Der Kamera-Service liefert auf Anfrage die Bilder.
Bildschirm-Service	Der Bildschirm-Service schaltet eine bestimmte Kachel ein und aus. Auch die Farbe stimmt mit der gewünschten überein.
LED-Strip-Service	Der LED-Strip-Service schaltet eine bestimmte LED ein und aus. Auch die Farbe stimmt mit der gewünschten überein.
Datenverarbeitungs-Service	Der Datenverarbeitungs-Service erkennt die eingeschalteten Leuchtquellen korrekt und speichert die Korrekten Koordinaten und IDs ab.
Agent	Der Agent nimmt die Befehle korrekt entgegen und leitet sie dem entsprechenden Service weiter.
Kommunikation	Der MQTT-Broker nimmt die Messages entgegen und veröffentlicht sie im entsprechenden Topic.
Race-Conditions	Während allen Tests wird darauf geachtet, dass keine Race-Conditions auftreten. Das bedeutet, bei der Kommunikation zwischen den einzelnen Services wird sichergestellt, dass die Nachrichten auch angekommen sind. Der Empfänger der Nachricht bestätigt dies dem Sender. Dadurch wird verhindert, dass zum Beispiel Bilder des Kamera-Services nicht in der richtigen Reihenfolge beim Datenverarbeitungs-Service ankommen und dieser dann falsche Schlüsse zieht.

## 6.2 Kann-Kriterien

Modul	Beschreibung
Muster zeichnen	Über die Mobile-App können mit dem Finger Muster gezeichnet werden.
Bild laden und anzeigen	Ein Bild (Dateityp ist noch nicht definiert) kann mit der Mobile-App geladen werden und wird korrekt auf den Leuchtquellen angezeigt.