





Pflichtenheft

Projekt: *LED-Cube*

Version 1.1

Carlucci Ramon, Guntli Michael, Kuhn Thomas, Weber Remo

Name	Datum	Unterschrift
Carlucci Ramon	03.10.2019	
Guntli Michael	03.10.2019	
Kuhn Thomas	03.10.2019	
Weber Remo	03.10.2019	

Änderungsgeschichte

<u>Datum</u>	<u>Version</u>	<u>Autor</u>	<u>Beschreibung</u>
2019-09-19	1.0	R. Carlucci	Dokument erstellt
2019-10-03	1.1	T. Kuhn	Pflichtenheft dem Projekt angepasst, mit Team besprochen, überarbeitet, von Team abgesegnet & unterschrieben

Inhaltsverzeichnis

1	EINLEITUNG.....	6
1.1	Zweck	6
1.2	Produktüberblick	6
1.3	Definitionen, Akronyme und Abkürzungen.....	6
1.4	Referenzen	6
2	ALLGEMEINE BESCHREIBUNG.....	7
2.1	Systemübersicht.....	7
2.2	Produktfunktionen	7
2.3	Benutzereigenschaften.....	8
2.4	Einschränkungen	8
2.5	Annahmen und Abhängigkeiten.....	8
2.6	Priorisierung der Anforderungen	8
3	EXTERNE SCHNITTSTELLEN	9
3.1	Hardwareschnittstellen	9
3.2	Benutzerschnittstellen	9
3.3	Softwareschnittstellen.....	9
3.4	Kommunikationsschnittstellen.....	9
4	FUNKTIONALE ANFORDERUNGEN.....	10
4.1	Überblick über die Systemfunktionen.....	10
4.2	Actors	10
4.3	Kurzbeschreibung der Use Cases.....	10
4.4	Use Case "Kommunikation starten"	11
4.5	Use Case "Programm auswählen"	12
4.6	Use Case "Programm selbst definieren"	13
5	SONSTIGE ANFORDERUNGEN	14
A	REFERENZEN	15

B	SCHEMA HARDWARE LED-CUBE	16
C	SCHEMA HARDWARE ARDUINO MEGA V3	17

Abbildungsverzeichnis

Abbildung 1: erste grobe Systemübersicht des LED-Cubes.....	6
Abbildung 2: Kontextdiagramm (Festlegung der Systemgrenze).....	7
Abbildung 3: Hardwareschnittstellen	9

Tabellenverzeichnis

Tabelle 1: Actors-Liste.....	10
Tabelle 2: Use Cases.....	10

Kapitel 1 Einleitung

1 Einleitung

1.1 Zweck

Im vorliegenden Dokument sind die Anforderungen definiert, welche im Projekt *LED-Cube* umgesetzt werden müssen. Es beschreibt den Auftrag zwischen Auftraggeber und Auftragnehmer. Der Ausdruck *Pflichtenheft* ist hier im Sinne der *IEEE Recommended Practice for Software Requirements Specification, ANSI/IEEE Std 830-1998* verwendet. Die dort definierte *Requirements Specification* beinhaltet sowohl die Benutzeranforderungen (*Lastenheft* gemäss DIN 69901-5) als auch Realisierungsvorgaben an die Entwicklungsgruppe (*Pflichtenheft* gemäss DIN 69901-5).

1.2 Produktüberblick

Im Rahmen des Projekts *LED-Cube* entsteht ein GUI zu einem bestehenden LED-Cube. Der Nutzer soll über das GUI den LED-Cube steuern können. Die Abbildung 1 gibt eine grobe Übersicht über das ganze System.

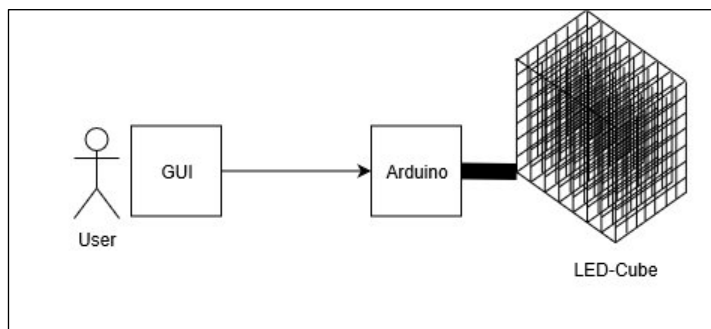


Abbildung 1: erste grobe Systemübersicht des LED-Cubes

Der zu entwickelnde Teil ist die Software, welche teils auf dem Rechner und teils auf dem Arduino ausgeführt wird.

1.3 Definitionen, Akronyme und Abkürzungen

- GUI graphical user interface (grafische Benutzeroberfläche)
- LED-Cube Würfel gitterstrukturartig aufgebaut mit den Dimensionen 8x8x8 Leds
oder
Name des Projekts
(es ist aus dem Zusammenhang klar, ob vom Projekt oder vom physischen Würfel die Rede ist)
- Programm im Zusammenhang mit dem LED-Cube
z.B. *es wird ein bestimmtes Programm auf dem Würfel abgespielt*
Das meint auf dem LED-Cube (physisch) werden verschiedene Zustände der Leds sequenziell nacheinander abgespielt.

1.4 Referenzen

siehe Anhang A auf Seite 15 dieses Dokumentes.

Kapitel 2 Allgemeine Beschreibung

2 Allgemeine Beschreibung

Der bestehende LED-Cube wird momentan über einen Arduino angesteuert, das bedeutet es kann ein fixes Programm auf dem Würfel abgespielt werden. Nun soll ein GUI entwickelt werden, über welches verschiedene Programme auf dem Würfel abgespielt werden können. Der Arduino wird weiterhin Bestandteil des Systems bleiben, da dieser die IOs zum Ansteuern des Würfels besitzt.

2.1 Systemübersicht

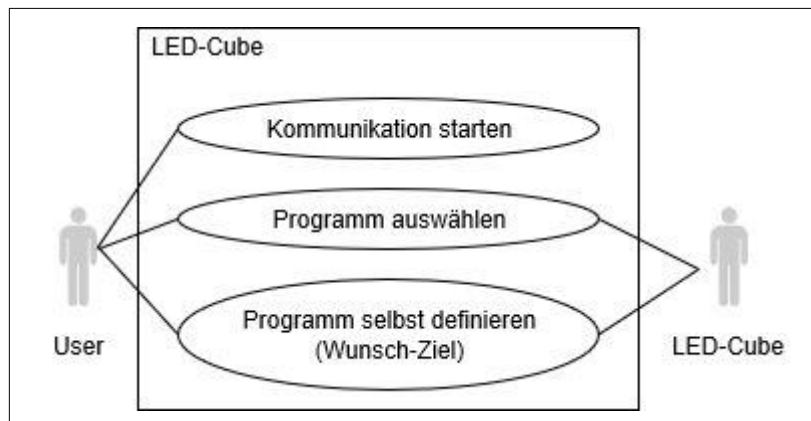


Abbildung 2: Kontextdiagramm (Festlegung der Systemgrenze)

Der LED-Cube ist die Steuerung über ein GUI des LED-Cubes. Der User kann auf dem GUI ein Programm auswählen, welches anschließend auf dem LED-CUBE wiedergegeben wird. Zum Projekt gehört das Schreiben der Software zu diesem System.

2.1.1 Software

Zu den Softwareaufgaben gehören folgende Punkte: entwerfen eines GUI, Implementation der USB-Schnittstelle vom Arduino, Handler, welcher Benutzereingaben verarbeitet und weiter zur Ausgabe an den Würfel gibt und das Anpassen der vorhandenen Software, damit diese implementiert werden kann.

2.1.2 Arduino Mega V3

2.1.2.1 Schema

Schema des Arduino Mega V3 im Anhang C Schema Hardware Arduino Mega V3.

2.1.2.2 Software Bibliothek UART-Library

Information zur UART-Library vom Arduino Mega V3: <https://www.arduino.cc/en/Reference/SoftwareSerial>

2.1.3 Hardware

Die Hardware ist vorhanden und wird nicht angepasst/geändert.
Details zur Hardware im Anhang B Schema Hardware LED-Cube.

2.2 Produktfunktionen

- Plug & Play
- vordefinierte Programme abspielen
- Programme selbst definieren und abspielen

Kapitel 2 Allgemeine Beschreibung

2.3 Benutzereigenschaften

Jeder Benutzer mit wenig technischem Hintergrund soll das Gerät verwenden können.

2.4 Einschränkungen

Einschränkungen sind gegeben durch den Clock-Speed 16 MHz und die maximale Baudrate von 115200 des Arduino Mega V3.0 und der Rechenleistung des verwendeten PCs.

2.5 Annahmen und Abhängigkeiten

Es wird angenommen, dass die vorhandene Hardware einwandfrei läuft.

2.6 Priorisierung der Anforderungen

2.6.1 Muss-Anforderungen

- Programm auswählen, welches anschliessend auf dem LED-Cube wiedergegeben wird

2.6.2 Soll-Anforderungen

- Kommunikation starten (COM auswählen, Kontrollanzeige dass eine Verbindung besteht)

2.6.3 Wunsch-Anforderungen

- Programm selbst definieren

Kapitel 3 Externe Schnittstellen

3 Externe Schnittstellen

3.1 *Hardwareschnittstellen*

3.1.1 zu LED-Cube

Die Schnittstelle zwischen Arduino und LED-Cube ist bereits bestehend und detailliert ersichtlich im Anhang B Schema Hardware LED-Cube. Zum besseren Verständnis ist unten noch eine Abbildung gezeigt.

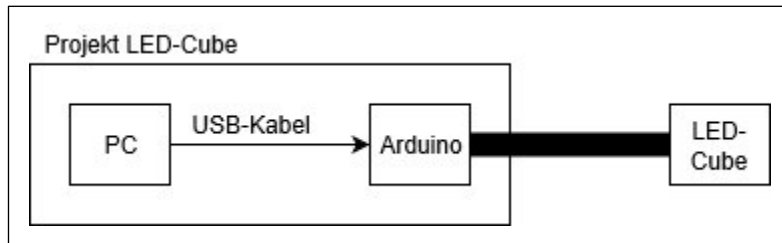


Abbildung 3: Hardwarechnittstellen

3.2 *Benutzerschnittstellen*

Dem User steht ein GUI auf dem PC zur Verfügung, welches einfach zu bedienen ist.

3.3 *Softwareschnittstellen*

keine Softwareschnittstellen vorhanden

3.4 *Kommunikationsschnittstellen*

keine Kommunikationsschnittstellen vorhanden

Kapitel 4 Funktionale Anforderungen

4 Funktionale Anforderungen

4.1 Überblick über die Systemfunktionen

Die folgenden Use Cases beziehen sich auf die Abbildung 2 auf der Seite 7.

4.2 Actors

Tabelle 1: Actors-Liste

<i>Actor</i>	<i>Beschreibung</i>
User	wählt gewünschtes Programm per Klick aus oder konfiguriert eigenes Programm.

4.3 Kurzbeschreibung der Use Cases

Tabelle 2: Use Cases

<i>Use Case</i>	<i>Beschreibung</i>
Kommunikation starten (Soll-Anforderung)	Das Programm erkennt, dass der LED-Cube angeschlossen wurde und verbindet sich automatisch. Falls keine automatische Verbindung nicht möglich ist, kann der COM manuell ausgewählt werden. Eine Anzeige signalisiert, ob der LED-Cube verbunden ist oder nicht.
Programm auswählen (Muss-Anforderung)	Das GUI besitzt für jedes mögliche abzuspielende Programm eine Taste, wird diese gedrückt, wird das entsprechende Programm auf dem LED-Cube abgespielt.
Programm selbst definieren (Wunsch-Anforderung)	<p>Auf dem GUI befindet sich ein 8x8 Feld aus Tasten. Der User kann ein Muster mit den Tasten zeichnen und mit dem Knopf Start wird das Muster auf die vorderste Ebene des Würfels geschrieben und zyklisch eine Ebene nach hinten geschoben. Ist das Muster an der hintersten Ebene angekommen, wird es wieder auf der vordersten Ebene angezeigt und wieder zyklisch nach hinten geschoben.</p> <p>Auf dem GUI befindet sich ein Textfeld. Der User kann ein Text eingeben und mit dem Knopf Start wird der Text buchstabenweise auf dem Würfel abgespielt. Das heisst der erste Buchstabe wird auf der vordersten Ebene dargestellt und zyklisch bis zur hintersten Ebene durchgeschoben. Danach wird dieser Vorgang für die alle anderen Buchstaben wiederholt. Wenn alle Buchstaben durch sind beginnt der Text wieder von vorne.</p>

Kapitel 4 Funktionale Anforderungen

4.4 Use Case "Kommunikation starten"

4.4.1 Vorbedingungen

Das USB-Kabel zum Arduino des LED-Cubes ist noch nicht eingesteckt.

4.4.2 Nachbedingungen

Die Kommunikation mit dem Arduino des LED-Cubes wurde aufgenommen. Anzeige «Verbunden» ist grün.

4.4.3 Nicht-funktionale Anforderungen

keine

4.4.4 Hauptszenario

Programm mit dem GUI wird gestartet. Dann wird nach dem Arduino des LED-Cubes gesucht und falls vorhanden verbunden.

4.4.5 Unterszenario

Programm mit dem GUI läuft bereits. Ein USB-Gerät wird eingesteckt. Das Programm prüft, ob es sich um das Arduino des LED-Cubes handelt und falls dies der Fall verbindet sich das Programm damit.

4.4.6 Fehlerszenarien

4.4.6.1 Arduino des LED-Cubes wird nicht erkannt

Falls das Arduino des LED-Cubes nicht erkannt wird, kann der User den COM manuell auswählen.

4.4.6.2 mehrere Arduinos sind am PC angehängt

Falls vorhanden bestimmte Nummer vom Arduino auslesen und anhand von dieser Nummer mit dem richtigen Arduino verbinden. Wenn dies nicht möglich ist, ist der User gezwungen den COM manuell auszuwählen.

4.4.7 Regeln

keine

4.4.8 Anmerkungen

keine

4.4.9 Beispiele

keine

Kapitel 4 Funktionale Anforderungen

4.5 Use Case "Programm auswählen"

4.5.1 Vorbedingungen

Der LED-Cube ist mit dem PC verbunden und die Kommunikation zwischen Programm und Arduino steht.

4.5.2 Nachbedingungen

Programm wird wiederholt auf dem Würfel abgespielt, bis ein anderes Programm ausgewählt wird.

4.5.3 Nicht-funktionale Anforderungen

Die Anzahl Programme ist noch nicht bekannt, so programmieren damit es einfach erweiterbar ist.
In der ersten Version der Software wird mit 6 Programmen gearbeitet.

4.5.4 Hauptszenario

Es wird die Taste für das ein bestimmtes Programm gedrückt.

4.5.5 Unterszenario

Es wird mehrmals die Taste für dasselbe Programm gedrückt. In diesem Fall wird das Programm nicht neu gestartet, sondern wird weiterhin abgespielt.

4.5.6 Fehlerszenarien

4.5.6.1 Verbindung zum Arduino verloren

Aus irgendeinem Grund werden die gesendeten Informationen vom Arduino nicht empfangen. Folge davon ist es wird nichts auf dem Würfel abgespielt.

4.5.6.2 gesendete Information kann nicht interpretiert werden

Es wird eine Information dem Communication-Handler gesendet, doch dieser kann die Information nicht interpretieren.

4.5.7 Regeln

keine

4.5.8 Anmerkungen

keine

4.5.9 Beispiele

keine

Kapitel 4 Funktionale Anforderungen

4.6 Use Case "Programm selbst definieren"

4.6.1 Vorbedingungen

Der LED-Cube ist mit dem PC verbunden und die Kommunikation zwischen Programm und Arduino steht.

4.6.2 Nachbedingungen

4.6.2.1 bei Eingabe von Muster

Muster wird auf die vorderste Ebene des Würfels geschrieben und zyklisch eine Ebene nach hinten geschoben. Ist das Muster an der hintersten Ebene angekommen, wird es wieder auf der vordersten Ebene angezeigt und wieder zyklisch nach hinten geschoben.

4.6.2.2 bei Eingabe von Text

Der erste Buchstabe wird auf der vordersten Ebene dargestellt und zyklisch bis zur hintersten Ebene durchgeschoben. Danach wird dieser Vorgang für die alle anderen Buchstaben wiederholt. Wenn alle Buchstaben durch sind beginnt der Text wieder von vorne.

4.6.3 Nicht-funktionale Anforderungen

keine

4.6.4 Hauptszenario

Es wird ein Muster auf dem 8x8 Tastenfeld erstellt oder ein Text eingegeben. Es wird die Taste Start gedrückt.

4.6.5 Fehlerszenarien

4.6.5.1 Verbindung zum Arduino verloren

Aus irgendeinem Grund werden die gesendeten Informationen vom Arduino nicht empfangen. Folge davon ist es wird nichts auf dem Würfel abgespielt.

4.6.5.2 gesendete Information kann nicht interpretiert werden

Es wird eine Information dem Communication-Handler gesendet, doch dieser kann die Information nicht interpretieren.

4.6.6 Regeln

keine

4.6.7 Anmerkungen

keine

4.6.8 Beispiele

keine

Kapitel 5 Sonstige Anforderungen

5 Sonstige Anforderungen

keine

Anhang A – Referenzen

A Referenzen

- [1] IEEE Recommended Practice for Software Requirements Specification. ANSI/IEEE Std 830-1998

C Schema Hardware Arduino Mega V3



Anhang C – Schema Hardware Arduino Mega V3

