

Pflichtenheft

# **Visuelle Programmiersprache für den Physikunterricht zur Datenerfassung auf einem Raspberry Pi**

**Version 0.1.0**

David Gawron      Stefan Geretschläger      Leon Huck  
Jan Küblbeck      Linus Ruhnke

24. Mai 2019

# Inhaltsverzeichnis

1 Produktübersicht . . . . .	2
2 Zielbestimmung . . . . .	2
2.1 Musskriterien . . . . .	3
2.2 Sollkriterien . . . . .	4
2.3 Wunschkriterien . . . . .	5
2.4 Abgrenzungskriterien . . . . .	5
3 Produkteinsatz . . . . .	5
4 Produktumgebung . . . . .	5
4.1 Software . . . . .	5
4.2 Hardware . . . . .	6
4.3 PhyPiDAQ . . . . .	6
5 Funktionale Anforderungen . . . . .	6
5.1 GUI . . . . .	6
5.1.1 Menüfeld . . . . .	6
5.1.2 Optional: Zusätzliche Funktionen im Menüfeld . . . . .	7
5.1.3 Konfigurationsfeld . . . . .	7
5.1.4 Darstellungsfenster . . . . .	7
5.2 Konfigurationserstellung . . . . .	7
5.3 Messablauf . . . . .	8
5.4 Fehlermeldungen . . . . .	9
5.5 Bedienungshilfen . . . . .	9
5.6 Sprache . . . . .	10
5.6.1 Optional: Internationalisierung . . . . .	10
5.7 Sonstiges . . . . .	10
5.7.1 Optional: Sonstiges . . . . .	10
5.8 Schnittstelle . . . . .	10
6 Produktdaten . . . . .	11
7 Nichtfunktionale Anforderungen . . . . .	11
7.1 Produktleistungen . . . . .	11
7.2 Qualitätsanforderungen . . . . .	11
7.3 Sonstige . . . . .	11
8 Globale Testfälle und Testszenarien . . . . .	12
9 Systemmodelle . . . . .	19
10 Benutzungsoberfläche . . . . .	19
11 Spezielle Anforderungen an die Entwicklungsumgebung . . . . .	19
12 Zeit- und Ressourcenplanung . . . . .	19
13 Ergänzungen . . . . .	19
14 Glossar . . . . .	19

# 1 Produktübersicht

## 2 Zielbestimmung

Die Anwendung soll es Lehrern ermöglichen, Schülern ab der siebten Klasse Grundkenntnisse der digitalen Messwerterfassung in einer für Schüler interessante und motivierenden Weise näher zu bringen. Dabei werden technische Details, wie Konfigurationsdateien, vor dem Schüler verborgen. Damit wird der Schüler nicht überfordert, sondern soll ermutigt werden, selbstständig mit der Software umzugehen. Dabei wird ihm die Grundkenntnisse von Ursache und Wirkung durch spaßige Messungen demonstriert.

Es wird eine graphische Oberfläche angeboten, die es dem Schüler ermöglicht allein per Drag and Drop eine Messanordnung zu erstellen. Die Details die dahinter stecken sind vor dem Schüler verborgen. Die Anwendung motiviert den Schüler dazu mit Sensoren, Transformationen und Darstellungen zu spielen und Dinge auszuprobieren. Dabei wird ihm durch eine intuitive Status- und Fehleranzeige gezeigt, ob seine Konstruktion funktioniert. Wenn nein, dann zeigt sie ihm an, wo das Problem liegt und warum es nicht funktioniert. Eine lästige Fehlersuche bleibt dem Schüler dann erspart.

Außerdem liefert die Anwendung dem Schüler zu den vorhandenen Bausteinen und zu der Anwendung allgemein die nötigen Informationen die er für die Nutzung braucht. Dabei wird auf ein ausführliches Tutorial am Anfang verzichtet. Die Anwendung liefert die Informationen häppchenweise durch Informationsanzeigen an den jeweilig relevanten Stellen. Damit findet der Schüler die Hilfe die er sucht an der Stelle an der er sie braucht.

Die Anwendung ermöglicht es dem Lehrer vor dem Unterricht eine Reihe von Messversuchen teilweise oder ganz zu konfigurieren und zu speichern. Dabei kann er genau bestimmen was er den Schülern zeigen will. Diese Konfigurationen kann er dann schnell und einfach im Unterricht zum Einsatz bringen.

Weiter ermöglicht es die Anwendung, dass Schüler aus höheren Klassen mit mehr Details der digitalen Messwerterfassung zusammen gebracht werden. Diese nutzen keine oder nur eine kleine vorgefertigte Konfiguration. Sie können dann auch selbst einen Sensor konfigurieren. Trotz des höheren Detailgrads bleibt die Anwendung übersichtlich und strukturiert. Damit bleiben auch komplexere Messversuche für den Schüler motivierend.

Die Anwendung ist als Schnittstelle zwischen PhyPiDAQ und dem Nutzer zu verstehen. Sie ermöglicht dem Nutzer über eine übersichtliche, strukturierte und intuitive GUI sowie über eine einfache Bedienung die Nutzung eines Raspberry Pi mit PhyPiDAQ. Dadurch soll dem Schüler digitale Messwerterfassung, Physik und auch Informatik in einer motivierenden Weise näher gebracht werden. Womöglich kann die Anwendung den Schüler auch für diese Themen begeistern.

## 2.1 Musskriterien

- Datenhandhabung
  - Der Benutzer kann Ergebnisse aus einer Messung speichern und laden.
  - Der Benutzer kann gleichzeitig Daten aus einer Messung darstellen und diese Daten auch weiter verwenden.
  - Der Benutzer kann eine Messkonfiguration speichern und laden.
  - Der Benutzer kann die Anwendung nutzen, ohne persönliche Daten preiszugeben.
- Benutzbarkeit der GUI
  - Die Anwendung reagiert auf Veränderungen so schnell, dass das Prinzip von Ursache und Wirkung intuitiv erfahren werden kann.
  - Der Benutzer kann Komponenten einer Messkonfiguration per Drag and Drop hinzufügen.
  - Der Benutzer kann Komponenten intuitiv mit einander verbinden.
  - Die Anwendung lässt sich mit dem Wissen eines Schülers aus der siebten Klasse bedienen.
  - Die Anwendung lässt sich auch mit einer rot-grün-Schwäche komplett nutzen.
- Funktionen
  - Die Anwendung erhält Daten durch den RaspberryPi mit PhyPiDAQ oder aus einer Datei und kann diese dann transformieren und darstellen.
  - Die Anwendung enthält vorgefertigte Standardelemente, wie z.B. to do
  - Die Anwendung enthält vorgefertigte Standardkonfigurationen.
- Information und Rückmeldung
  - Ist ein verwendeter Sensor nicht angeschlossen oder fehlerhaft, meldet die Anwendung dies über eine Pop-Up-Nachricht.

- Die Anwendung enthält Erklärungen und Informationen zu den einzelnen Komponenten, d.h. zu den Sensoren, Transformationen und Darstellungen.
- Die Anwendung enthält Erklärungen und Informationen zu den GUI-Bereiche, d.h. zu dem Auswahlbereich der Komponenten, zu dem Messversuchsbereich und zu dem Anzeigebereich.
- Falls nicht alle Kanäle verbunden sind, meldet die Anwendung dies dem Benutzer mit einer Fehlermeldung, falls dieser versucht die Messung zu starten.
- Sonstiges
  - Die Anwendung kommuniziert mit den Nutzer auf deutsch.
  - Die Anwendung soll auf die Unterstützung neuer Sensoren erweiterbar sein.
  - Die Anwendung hält die Datenschutzrichtlinie „to do“ ein.
  - Die Anwendung hält die Schulrichtlinie „to do“ ein.

## 2.2 Sollkriterien

- Die Anwendung soll auf die Erstellung eigener Transformationen erweiterbar sein.
- Die Anwendung soll auf andere Sprachen erweiterbar sein.
- Die Anwendung soll allein auf dem Raspberry Pi laufen können.
- Farbkodierung der GUI-Elemente
- Einfache Erweiterbarkeit

Einheitliches Interface für Sensoren

## 2.3 Wunschkriterien

- Übertragung der Messdaten über Netzwerkschnittstelle
- Spiele, „nachmachen“ von vorgegebenen Messergebnissen

- Ausführliche Beschreibung der physikalische Hintergründe

## **2.4 Abgrenzungskriterien**

- Unterschiedliche Benutzerkonten sind nicht zu implementieren
- Personenbezogene Daten werden nicht gespeichert

## **3 Produkteinsatz**

- Anwendungsbereich: Die Software wird durch Schulen in Deutschland auf schulei-gener Hardware eingesetzt.
- Zielgruppe: Lehrer und Schüler ab der 7. Klasse.
- Betriebsbedingungen: TODO

## **4 Produktumgebung**

Die Anwendung läuft auf einem Computer, die Messdaten werden über Messsensoren an einem Raspberry Pi erfasst.

### **4.1 Software**

Die Anwendung läuft auf Computern mit den Betriebssystemen Linux ab Kernel 7 und Microsoft Windows ab Windows 7 oder neuer. Die Anwendung muss auf dem Computer vollständig installiert sein.

### **4.2 Hardware**

Die Anwendung läuft auf gewöhnlichen Computern. Auf dem per USB-Schnittstelle verbundenen Raspberry Pi muss PhyPi DAQ installiert und verwendungsfähig sein. Die an das Raspberry Pi angeschlossenen Messsensoren müssen richtig und sinnvoll angeschlossen sein.

### 4.3 PhyPiDAQ

Bei PhyPiDAQ<sup>1</sup> handelt es sich um eine Anwendung zur Datenerfassung und Analyse mit einem Raspberry Pi. Diese ist nicht Bestandteil des Produktes, wird jedoch zur Datenerfassung und Datenverarbeitung und somit zur Funktionalität der Anwendung benötigt. Das Programm, welches in der Programmiersprache python3 geschrieben ist bietet einfache und einheitliche Schnittstellen zur Verwendung der Messsensoren.

## 5 Funktionale Anforderungen

### 5.1 GUI

F10 Die Benutzer erreichen nach Öffnung der Anwendung direkt die GUI.

F20 Der Benutzer öffnet durch den Einstellungen-Button die Einstellungen.

F30 Der Benutzer kann durch den Datei-Button die Dateiverwaltung öffnen.

F40 Der Benutzer kann durch den Hilfe-Button das Hilfe Fenster öffnen.

#### 5.1.1 Menüfeld

F50 Der Benutzer öffnet durch den Sensoren-Button die Auswahl an Sensoren.

F60 Der Benutzer öffnet durch den Verbindungen-Button die Auswahl an Verbindungen.

F70 Der Benutzer öffnet durch den Darstellungen-Button die Auswahl an Darstellungen.

F80 Der Benutzer erhält durch verschiedene Farben der Konfigurationsbausteine eine visuelle Repräsentationen ihrer Komplexität.

---

<sup>1</sup><https://github.com/GuenterQuast/PhyPiDAQ>

### **5.1.2 Optional: Zusätzliche Funktionen im Menüfeld**

- F90 Der Benutzer kann durch den Sensoren hinzufügen-Button weitere Sensoren hinzufügen.
- F100 Der Benutzer kann durch den Transformationen hinzufügen-Button weitere Transformationen hinzufügen.
- F110 Die in Transformationen in F100 sollen in eigenen Python-Skripten geschrieben werden.

### **5.1.3 Konfigurationsfeld**

- F120 Durch Drag und Drop kann der Benutzer Konfigurationsbausteine im Konfigurationsfeld platzieren.
- F130 Durch Betätigen des Messung starten-Button startet der Nutzer eine Messung.

### **5.1.4 Darstellungsfenster**

- F140 Das Darstellungsfenster ist bei der Initialisierung der Anwendung leer.
- F150 Durch die Benutzerkonfiguration im Konfigurationsfeld wird durch F90 automatisch die Darstellungsart im Darstellungsfenster geöffnet.

## **5.2 Konfigurationserstellung**

- F160 Über F30 kann der Benutzer gespeicherte Standardkonfigurationen öffnen oder alte Messwerte verwenden.
- F170 Durch F160 geladene Konfigurationen und Messdaten werden automatisch nach Format überprüft.
- F180 Aus F50 kann der Benutzer Sensoren durch Drag und Drop in das Konfigurationsfeld ziehen.
- F190 Die Anwendung sollte automatisch überprüfen, ob der ausgewählte Sensor richtig angeschlossen ist.



- F200 Die Anwendung gibt durch F190 automatisch eine visuelle Rückmeldung über die Farbe des Sensors.
- F210 Aus F60 kann der Benutzer eine oder mehrere Verbindungen durch Drag und Drop in das Konfigurationsfeld ziehen.
- F220 Die ausgewählten Verknüpfungen kann der Benutzer mit einem ausgewählten Sensor verknüpfen.
- F230 Die Anwendung sollte automatisch überprüfen, ob die Verbindungen eine geeignete Kombination mit dem Sensor darstellt.
- F240 Aus F70 kann der Benutzer eine Darstellungsart per Drag und Drop in das Konfigurationsfeld zu ziehen.
- F250 Die Ausgewählte Darstellung kann der Benutzer mit einer Verbindung verknüpfen.
- F260 Die Anwendung sollte automatisch überprüfen, ob die gewählte Darstellungsart eine sinnvolle Repräsentation der Messdaten darstellt.
- F270 Über F30 kann der Benutzer seine eigene Konfiguration über einen Konfiguration speichern- Knopf speichern.
- F280 Bausteine die ein Benutzer nicht mehr in dem Konfigurationsfeld haben möchte können durch Drag und Drop in die Menüleiste wieder versteckt werden.

### **5.3 Messablauf**

- F290 Die Anwendung sollte dem Benutzer erlauben den Wertebereich und die Dauer der Messung festzulegen.
- F300 Der Nutzer kann durch F90 die Messung nach der Benutzerkonfiguration starten.
- F310 Durch F300 wird automatisch mit der visuellen Darstellung der Messdaten begonnen.
- F320 Der Nutzer kann durch den Messung Messung löschen-Button eine durchgeführte Messung pausieren und die visuelle Darstellung auf den Ausgangszustand bringen.
- F330 Durch F320 wird nicht die Benutzerkonfiguration gelöscht.
- F340 Durch F320 muss der Benutzer erst die Messung starten um weiter zu messen.

- F350 Der Benutzer kann durch den Messung pausieren-Button die Messung pausieren.
- F360 Der Benutzer kann durch den Messung-fortsetzen-Button die Messung fortsetzen.
- F370 Der Benutzer kann eine Messung durch F310 nur fortsetzen, wenn sie zuvor durch F300 pausiert wurde.
- F380 Der Benutzer kann durch den Messdaten speichern-Button die gemessenen Daten speichern.
- F390 Der Benutzer kann durch den Graph speichern-Button den durch die Messung erzeugten Graphen speichern.
- F400 Bei F370 und F380 öffnet sich automatisch das Verzeichnis und der Nutzer muss einen eigenen Dateinamen eingeben und speichern.

## **5.4 Fehlermeldungen**

- F410 Durch F130 geladene Konfigurationen und Messdaten werden automatisch nach Format überprüft und eine aussagekräftige Fehlermeldung zurückgegeben.
- F420 Durch F190 und F220 wird bei einer ungültigen Kombination von zwei Konfigurationsbausteinen eine aussagekräftige Fehlermeldung zurückgegeben
- F430 Die Anwendung sollte dem Benutzer automatisch durch Pop-Up Benachrichtigung darauf hinweisen, dass bei Betätigen des Messung löschen-Button oder bei Schließen der Anwendung die Messdaten ohne vorheriges Betätigen des Messdaten speichern-Button verloren gehen.

## **5.5 Bedienungshilfen**

- F440 Bei enthaltenen Konfigurationsbausteine sollen einen Information-Button besitzen, über welche der Benutzer Informationen und Hilfestellung bereitgestellt bekommt.
- F450 Über den Hilfe-Button erhält der Nutzer eine kurze Beschreibung zur Funktionalität der Anwendung.

## **5.6 Sprache**

F450 Die Anwendung ist in deutscher Sprache.

### **5.6.1 Optional: Internationalisierung**

F460 Die Anwendung stellt dem Benutzer weitere Sprachpakete für die GUI zur Verfügung.

F470 Durch F20 ist die Sprache für den Nutzer änderbar.

## **5.7 Sonstiges**

F480 Die in der Anwendung enthaltenen Farben sollten mit Rücksicht auf Benutzer mit Rot-Grün Schwäche oder Farbenblindheit ein barrierefreies Verwendung der Anwendung ermöglichen.

### **5.7.1 Optional: Sonstiges**

F490 Durch F20 sollte der Benutzer die in der Anwendung verwendeten Farben ändern können.

F500 Durch F20 sollte der Benutzer die in der Anwendung verwenden Buchstabengröße verändern können.

## **5.8 Schnittstelle**

- just a test

## **6 Produktdaten**

Zu speichern sind ausschließlich:

- Konfigurationsdaten

- Physikalische Messdaten

Es sollen keine benutzer- oder personenbezogenen Daten gespeichert werden.

## **7 Nichtfunktionale Anforderungen**

### **7.1 Produktleistungen**

NF010 Performance beim Auslesen von Daten abhängig von PhyPiDAQ.

NF020 Verarbeitung ausgelesener Daten innerhalb von 10 ms.

### **7.2 Qualitätsanforderungen**

NF030 Die Benutzeroberfläche ist für Siebtklässler verständlich.

NF080 Programmfehler werden dem Benutzer gegenüber klar kommuniziert.

### **7.3 Sonstige**

NF110 Die Software ist vollständig quelloffen und frei.<sup>2</sup>

NF120 Die Datenschutzgrundverordnung wird nicht verletzt.

NF130 Schriftgröße von Text kann verändert werden.

NF140 Farbschema von farbigen UI-Elementen ist veränderbar.

## **8 Globale Testfälle und Testszenarien**

T010 to do Benutzung des Startbildschirms

**Stand**

---

<sup>2</sup>Definition: <https://www.gnu.org/philosophy/free-sw.de.html>

**Aktion**

**Reaktion**

T020 drag and drop Sensor

**Stand** Offenes Fenster der Hauptanwendung.

**Aktion** Der Benutzer öffnet das gewünschte Tab und zieht einen Sensor in die Konstruktionsfläche in die Mitte und lässt es dort los.

**Reaktion** Das Icon des Sensors bleibt als Kopie auf der Konstruktionsfläche erhalten.

T025 to do Überprüfung T020

**Stand**

**Aktion**

**Reaktion**

T030 drag and drop Transformation und Überprüfung

**Stand** Offenes Fenster der Hauptanwendung.

**Aktion** Der Benutzer öffnet das Tab der Transformationen und zieht eine Transformation in die Konstruktionsfläche in die Mitte und lässt sie dort los.

**Reaktion** Das Icon der Transformation bleibt als Kopie auf der Konstruktionsfläche erhalten.

T035 to do Überprüfung T030 automatisch

T040 drag and drop einer Darstellung und deren Überprüfung

**Stand** Offenes Fenster der Hauptanwendung.

**Aktion** Der Benutzer öffnet das Tab der Darstellung und zieht eine Darstellung in die Konstruktionsfläche in die Mitte und lässt sie dort los.

**Reaktion** Das Icon der Darstellung bleibt als Kopie auf der Konstruktionsfläche erhalten.

T045 to do Überprüfung T040 automatisch

T050 Einstellen von Messbereich und Dauer einer Messung

1. **Stand** Offenes Fenster der Hauptanwendung.

**Aktion** Der Benutzer drückt auf den „Anpassen“ Knopf.

**Reaktion** Das Fenster zum Anpassen der Messung öffnet sich.

2. **Stand** Offenes Fenster zum Anpassen der Messung.

**Aktion** Der Benutzer verändert die Einstellungen und drückt dann auf den „OK“ Knopf.

**Reaktion** Die Einstellungen werden übernommen. Das Fenster zum Anpassen der Messung schließt sich.

T060 Aufrufen von Informationen

**Stand** Offenes Fenster der Anwendung.

**Aktion** Der Benutzer drückt auf einen „I“ Knopf.

**Reaktion** Ein Informationsfenster mit den zugehörigen Informationen öffnet sich.

T070 Lade Datei als Quelle und Überprüfung

1. **Stand** Offenes Fenster der Hauptanwendung.

**Aktion** Der Benutzer drückt auf den „lade von Datei“ Knopf.

**Reaktion** Ein Fenster zum Auswählen eines Pfades öffnet sich.

2. **Stand** Offenes Fenster Auswählen eines Pfades.

**Aktion** Der Benutzer wählt einen Pfad zu einer gültigen Datei als Quelle aus. Danach drückt er auf den „Laden“ Knopf.

**Reaktion** Die Daten werden als Sensor Icon zu den verfügbaren Sensoren hinzugefügt.

T090 Laden einer Messung

1. **Stand** Offenes Fenster der Hauptanwendung.

**Aktion** Der Benutzer drückt auf den „Laden“ Knopf.

**Reaktion** Das Fenster zum Laden eines Versuchs öffnet sich.

2. **Stand** Offenes Fenster zum Laden.

**Aktion** Der Benutzer gibt den Speicherort einer gültigen Versuchsdatei an. Dann drückt der Benutzer auf den „Laden“ Knopf.

**Reaktion** Der Versuch wird geladen. Das Fenster zum laden schließt sich. In der Konstruktionsfläche wird der Versuch dargestellt.

#### T100 Starten einer Messung

**Stand** Offenes Fenster der Hauptanwendung mit einem gültigen Aufbau in der Konstruktionsfläche.

**Aktion** Der Benutzer drückt auf den „Start“ Knopf.

**Reaktion** Das Konstruktionsfenster wird vor Veränderung geschützt. Die Anzeigefläche rechts stellt die gewünschten Darstellungen dar. Falls ein Sensor nur endlich viele Werte eingeben soll, pausiert sich das Programm automatisch danach.

#### T110 to do

**Stand**

**Aktion**

**Reaktion**

#### T115 Zurücksetzen einer Messung

**Stand** Offenes Fenster der Hauptanwendung mit einer gültigen Messung die pausiert ist.

**Aktion** Der Benutzer drückt auf den „starte neu“ Knopf.

**Reaktion** Die Messdaten werden verworfen. Die Darstellungen werden zurückgesetzt. Die Dauer der Messung wird zurückgesetzt.

#### T120 Pausieren der Messung

**Stand** Offenes Fenster der Hauptanwendung mit einer laufenden Messung.

**Aktion** Der Benutzer drückt auf den „Pause“ Knopf.

**Reaktion** Die Messung stoppt. Die Darstellungen bleiben in ihrem jetzigen Zustand.

#### T130 Speichern von Messdaten

1. **Stand** Offenes Fenster der Hauptanwendung mit einem pausiertem Versuch, der schon gelaufen ist.

**Aktion** Der Benutzer drückt auf den „speichere Daten“ Knopf.

**Reaktion** Ein Fenster zum Auswählen eines Speicherorts öffnet sich.

2. **Stand** Offenes Fenster zum Auswählen eines Speicherorts.

**Aktion** Der Benutzer wählt einen Speicherort aus und drückt auf den „speichern“ Knopf.

**Reaktion** Eine Datei wird am Speicherort erstellt. Die Datei enthält die Messdaten. Das Fenster zum Auswählen eines Speicherorts schließt sich.

#### T140 Wiederaufnahme einer Messung

**Stand** Offenes Fenster der Hauptanwendung mit einem pausiertem Versuch der schon gelaufen ist.

**Aktion** Der Benutzer drückt auf den „wiederaufnehmen“ Knopf.

**Reaktion** Die Anwendung fährt an der Stelle fort, an der die Anwendung pausiert wurde.

#### T150 Speichern einer Konfiguration

1. **Stand** Offenes Fenster der Hauptanwendung und der Versuch läuft nicht.

**Aktion** Der Benutzer drückt auf den „Speichern“ Knopf.

**Reaktion** Das Fenster zum Speichern eines Versuchs öffnet sich.



2. **Stand** Offenes Fenster zum Speichern.

**Aktion** Der Benutzer ändert optional den Pfad des Speicherorts. Dann drückt der Benutzer auf den „Speichern“ Knopf.

**Reaktion** Der Versuch wird am Zielort als Datei gespeichert. Das Fenster zum speichern schließt sich.

T160 Verwerfe aktuelle Konfiguration und erstelle neue Konfiguration

1. **Stand** Offenes Fenster einer Messung.

**Aktion** Der Benutzer drückt auf den „neue Messung“ Knopf.

**Reaktion** Ein Fenster öffnet sich mit der Frage, ob man die aktuelle Konfiguration verwerfen will.

2. **Stand** Offenes Fenster mit der Frage, ob man die Konfiguration verwerfen will.

**Aktion** Der Benutzer drückt auf den „Bestätigen“ Knopf.

**Reaktion** Das Fenster schließt sich. Die aktuelle Konfiguration wird verworfen.

Txxx Sensor hinzufügen

1. **Stand** Offenes Fenster der Hauptanwendung .

**Aktion** Der Benutzer drückt auf den Knopf „füge Sensor hinzu“.

**Reaktion** Das Untermenü zur Auswahl des Sensors öffnet sich.

2. **Stand** Offenes Fenster zur Auswahl des Sensors.

**Aktion** Der Benutzer drückt auf den Button „Abbrechen“.

**Reaktion** Das Untermenü zur Auswahl des Sensors schließt sich.

3. **Stand** Offenes Fenster zur Auswahl des Sensors.

**Aktion** Der Benutzer wählt aus der Liste einen Sensor aus und drückt auf den „Weiter“ Knopf.

**Reaktion** Falls der Sensor angeschlossen ist, öffnet wechselt die Anwendung zum Fenster um diesen zu konfigurieren.

4. **Stand** Offenes Fenster zur Konfiguration des Sensors.

**Aktion** Der Benutzer drückt auf den „Zurück“ Knopf.

**Reaktion** Die Anwendung wechselt zu dem Fenster Auswahl des Sensors zurück.

5. **Stand** Offenes Fenster zur Konfiguration des Sensors.

**Aktion** Der Benutzer verändert optional die voreingestellten Werte in dem jeweiligen gültigen Wertebereich. Der Benutzer drückt auf den „Fertig“ Knopf.

**Reaktion** Das Untermenü zur Konfiguration schließt sich. Im Fenster der Hauptanwendung wird der konfigurierte Sensor in die Liste der verwendbaren Sensoren hinzugefügt.

#### Txxx Konstruktion eines Versuchs

1. **Stand** Offenes Fenster der Hauptanwendung mit beliebigem offenen Tab links oben.

**Aktion** Der Benutzer drückt auf den gewünschten geschlossenen Tab.

**Reaktion** Der aktuelle Tab wechselt zu dem gewünschten Tab.

2. **Stand** Offenes Fenster der Hauptanwendung.

**Aktion** Der Benutzer öffnet das gewünschte Tab und zieht ein Element in die Konstruktionsfläche in die Mitte und lässt es dort los.

**Reaktion** Das Icon des Elements bleibt als Kopie auf der Konstruktionsfläche erhalten.

3. **Stand** Offenes Fenster der Hauptanwendung mit mindestens zwei Elementen in der Konstruktionsfläche und mindestens zwei offenen Kanälen.

**Aktion** Der Benutzer drückt auf einen offenen Kanal und danach auf einen anderen offenen Kanal.

**Reaktion** Eine Verbindung zwischen den zwei Kanälen wird erstellt.

4. **Stand** Offenes Fenster der Hauptanwendung mit mindestens einem Elementen in der Konstruktionsfläche.

**Aktion** Der Benutzer wählt das Element aus und drückt die „Entf“ Taste der Tastatur oder auf den „Löschen“Knopf.

**Reaktion** Das ausgewählte Element verschwindet aus der Konstruktionsfläche. Alle Verbindungen des Elements werden auch gelöscht.

5. **Stand** Offenes Fenster der Hauptanwendung mit mindestens einer Verbindung in der Konstruktionsfläche.

**Aktion** Der Benutzer drückt auf ein Verbindungsende und danach auf der andere.

**Reaktion** Die Verbindung wird gelöscht.

6. **Stand** Offenes Fenster der Hauptanwendung mit mindestens einer Verbindung und einem offenen Kanal in der Konstruktionsfläche.

**Aktion** Der Benutzer drückt auf ein Verbindungsende und danach auf einen offenen Kanal.

**Reaktion** Die Verbindung wird zu dem gewählten Kanal umgestellt.

## 9 Systemmodelle

## 10 Benutzungsoberfläche

## 11 Spezielle Anforderungen an die Entwicklungsumgebung

## 12 Zeit- und Ressourcenplanung

Phase	Verantwortlicher	Zeitraum
Pflichtenheft	Jan Küblbeck	KW 20–22
Entwurf	Leon Huck	KW 23–26
Implementierung	Stefan Geretschläger	KW 27–29, 31
Klausurenphase	—	KW 30, 32
Qualitätssicherung	David Gawron	KW 33–35
Abnahme	—	KW 36
Abschlussprüfung	Linus Ruhnke	KW 37/38

## 13 Ergänzungen

## 14 Glossar