|  |  |
| --- | --- |
| Quellbild anzeigen  Alarmsystem  Arbeitsprobe | Übersicht  Ein Alarmsystem, welches mit einem Arduino Bewegungen registriert. Eine Datenbank auf einem Raspberry, wo alle Bewegungen eingetragen werden und eine Verbindung mit dem selbst programmierten Chatbot, um die Daten abzufragen.  Simon Müller  Projektwoche September 2021 |

Inhalt

[1. Einleitung 2](#_Toc85969842)

[1.1 Projektumfeld 2](#_Toc85969843)

[1.2 Projektziel 2](#_Toc85969844)

[1.3 Projektbegründung 2](#_Toc85969845)

[2. Projektplanung 2](#_Toc85969846)

[2.1 Projektphasen 2](#_Toc85969847)

[3. Entwurfsphasen 3](#_Toc85969848)

[3.1 Datenmodell 3](#_Toc85969849)

[3.2 Arduino Code 4](#_Toc85969850)

[3.4 Klassendiagramm 4](#_Toc85969851)

[4. Implementierungsphase 5](#_Toc85969852)

[4.1 Implementierung der Datenstrukturen 5](#_Toc85969853)

[4.2 Implementierung der Benutzeroberfläche 6](#_Toc85969854)

[5. Fazit 6](#_Toc85969855)

[5.1 Soll-/Ist-Vergleich 6](#_Toc85969856)

[5.3 Ausblick 6](#_Toc85969857)

[6. Benutzerhandbuch (Mit Chatbot) 6](#_Toc85969858)

[Abbildungsverzeichnis 7](#_Toc85969859)

[Quellen 7](#_Toc85969860)

# 1. Einleitung

## 1.1 Projektumfeld

Wir sind eine Klasse mit 13 Schüler in der IBZ Schule in Aarau. In der Projektwoche vom September 2021 haben wir den Auftrag bekommen, eine Arbeitsprobe zu erstellen, die es unsere Chancen auf eine Praktikumsstelle für August 2022 erheblich verbessert. Die Arbeitsprobe mache ich allein, so arbeite ich zu Hause am Schullaptop oder am eigenen Computer.

Eigentlich wollte ich eine IOS-APP Programmieren. Da wir aber das Modul noch nicht hatten in dem wir das lernten, habe ich mich entschieden es mit meinem grössten C# Projekt zu verbinden: der Chatbot.

Das Projekt und alle Dateien dazu sind auf GitHub: [Appli4life/Alarmanlage (github.com)](https://github.com/Appli4life/Alarmanlage)

Das Chatbot-Projekt findet sich hier: [Appli4life/Chatbot (github.com)](https://github.com/Appli4life/Chatbot)

Der Chatbot konnte ich bisher noch nicht veröffentlichen. Das liegt daran, dass wir das Programmieren einer Setup.exe Datei noch nicht erlernt haben.

## 1.2 Projektziel

Als Arbeitsprobe realisiere ich ein Alarmsystem für mein Zimmer zuhause. Dafür verwende ich ein Arduino und ein Raspberry PI. Ich ergänze danach den Chatbot, sodass er Zugriff auf diese Datenbank hat und mit einem Schlüsselwort die Daten abfragt.

Auflistung der zwingenden Ziele:

* Arduino Bewegungssensor aufsetzen
* Raspberry Datenbankserver aufsetzten (Tabelle für alle Bewegungen)
* Arduino schreibt Datensätze in Datenbank
* Datenbankserver aus dem Internet erreichbar machen
* Datenbank Benutzer für Sicherheit erstellen
* Programm, welches auf die Datenbank zugreift

Auflistung der optionalen Ziele:

* Arduino spiel Ton ab bei Bewegung
* Arduino startet bei Bewegung eine Tonaufnahme
* Programm kann das Alarmsystem einschalten/ausschalten

Es gibt kein bestimmter Abgabetermin für die Arbeitsprobe. Dennoch habe ich beschlossen es bis spätestens Ende Oktober fertigzustellen.

## 1.3 Projektbegründung

Die Arbeitsprobe ist beim Bewerben um ein Praktikumsstelle, ein grosser Vorteil gegenüber anderen Kandidaten. Sie zeigt, welche Fähigkeiten der Bewerber hat und wie organisiert er arbeitet. Ich kann mit unterschiedlichen Geräten arbeiten und diese programmieren, was mir auch zeigt wie die Verbindung zwischen den Geräten funktioniert.

Ich habe somit eine herausfordernde Arbeit vor mir auf die ich mich freue sie zu Lösen.

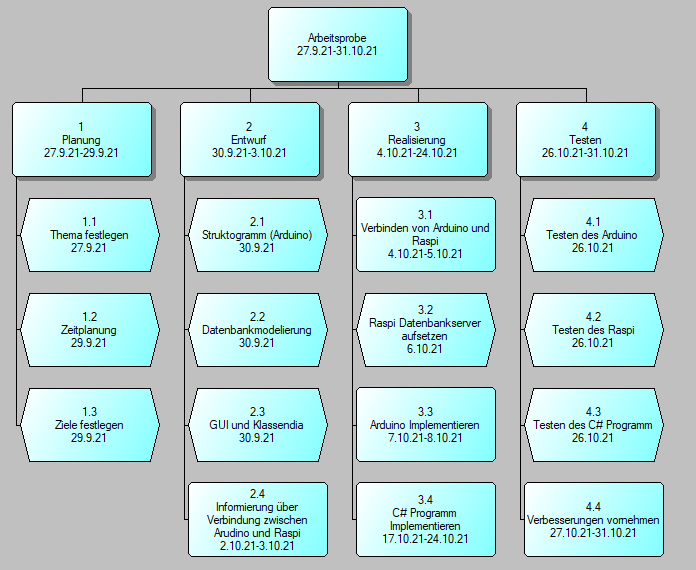
# 2. Projektplanung

## 2.1 Projektphasen

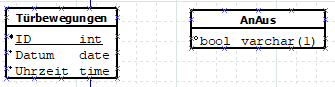
Die Arbeitsprobe sollte ich Ende Oktober fertigstellen können. In der Projektwoche 27.09.21 - 01.10.21 werde ich die ganze Planung abschliessen und Unklarheiten klären, die sich mir stellen. Anschliessend vom 04.10.21 - ca. 24.10.21 realisiere ich das Alarmsystem und nebenbei dokumentiere ich das Ganze. In der letzten Oktoberwoche wird das Alarmsystem von mir getestet, verbessert und ich werde allenfalls die optionalen Ziele (siehe 1.2 Projektziel) realisieren.

# 3. Entwurfsphasen

Phasenstrukturplan

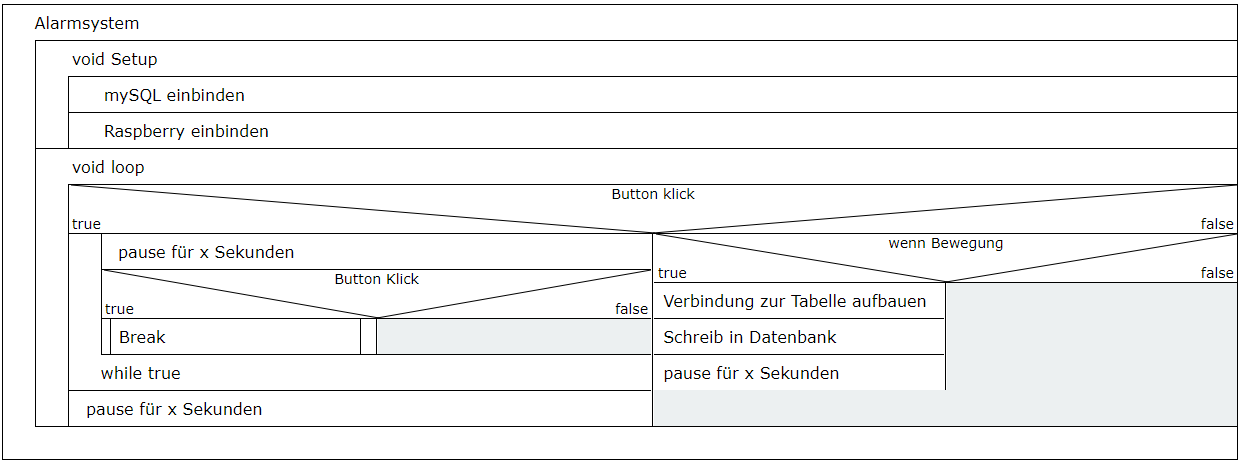


## 3.1 Datenmodell



2 Datenbankmodell

## 3.2 Arduino Code



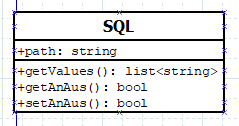
Struktogramm

Ich habe einen PIR Sensor Online gekauft, der Bewegungen registriert und sich leicht mit dem Arduino verbinden lässt.

Wie sich herausgestellt hat, wird zur Kommunikation zwischen Raspberry und Arduino ein Python Skript gebraucht. Dieses ähnelt sehr dem Arduino Code, darum gibt es auch keinen Entwurf.

Das Python Skript startet automatisch beim Einschalten des Raspberry.

## 3.4 Klassendiagramm



Ergänzte Klasse

Diese Klasse wird zum Chatbot ergänzt.

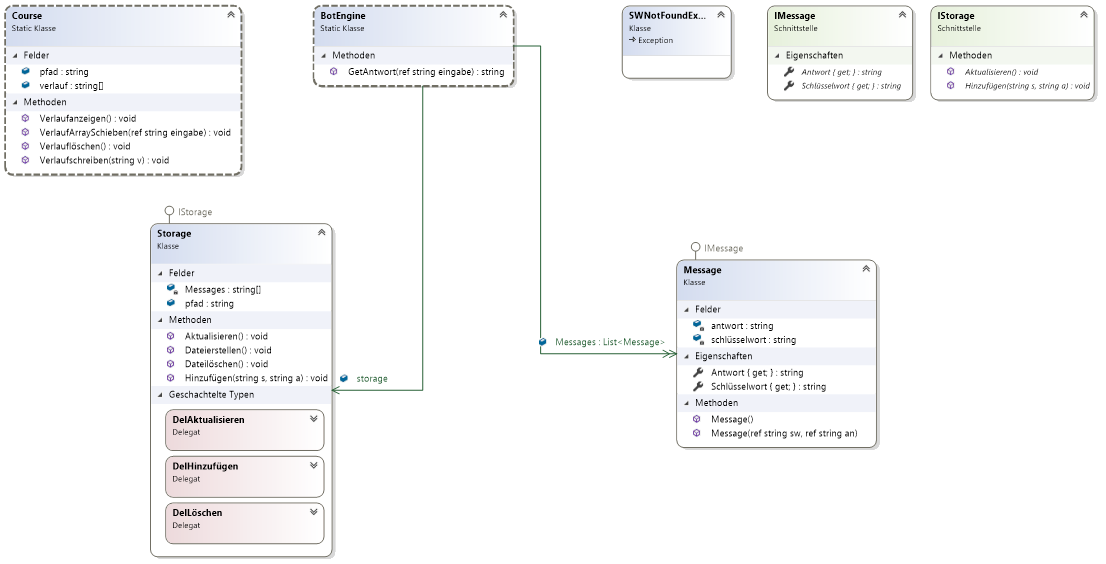
path: Stellt die IP-Adresse meiner Internetbox da, die eine Portweiterleitung zum Raspberry eingestellt hat.

getValues: Holt sich alle Einträge aus der Tabelle Türbewegungen.

getAnAus: Überprüft den Eintrag aus der AnAus Tabelle und gibt True/False zurück, wenn die Alarmanlage An oder Aus ist.

setAnAus: Schaltet die Alarmanlage An oder Aus. Liefert True/False zurück bei Erfolg.

Diese Klassen sind bereits im Chatbot vorhanden.

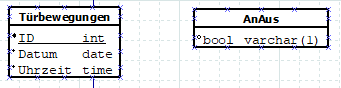


Chatbot Klassen

# 4. Implementierungsphase

## 4.1 Implementierung der Datenstrukturen

Datenbank: Alarmsystem



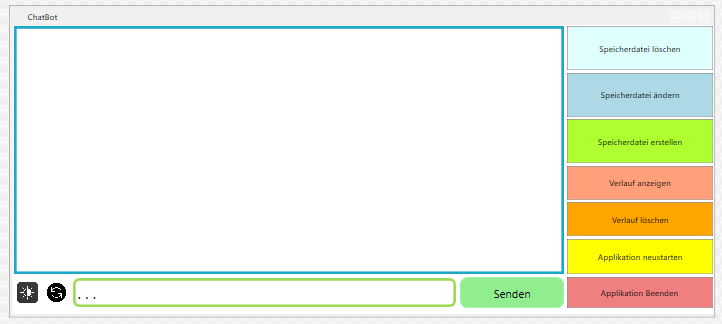
Datenstruktur

Alle Türbewegungen werden mit ID, Datum und Uhrzeit gespeichert. In der Tabelle AnAus, gibt es immer nur einen Eintrag, der festlegt, ob die Alarmanlage An- oder Ausgeschalten wird/ist.

Für das Eintragen von Daten in die Tabelle Türbewegungen und das Lesen von Daten aus der Tabelle AnAus (was das Python Skript können muss), wurde ein Datenbankbenutzer erstellt, der die Rechte INSERT und SELECT auf die jeweiligen Tabellen hat.

Für das Lesen von Daten aus der Tabelle Türbewegungen und das Aktualisieren von Daten in der Tabelle AnAus (was der Chatbot können muss), wurde ebenfalls ein Datenbankbenutzer erstellt, der die Rechte SELECT und UPDATE auf die jeweiligen Tabellen hat.

## 4.2 Implementierung der Benutzeroberfläche



Chatbot GUI

Wie das GUI angewendet wird, finden Sie [hier](#_6.__Benutzerhandbuch).

# 5. Fazit

## 5.1 Soll-/Ist-Vergleich

Als Arbeitsprobe war dies ein voller Erfolg. Ich musste einige Dinge während dem Realisieren ändern, trotzdem war es ein gutes Projekt. Änderungen:

* Anstelle einer IOS-App habe ich die Alarmanlage mit dem Chatbot verbunden. Grund dafür war, dass das Modul Mobile Apps Realisieren erst im Frühling in der Schule kommt. Ausserdem war der Chatbot bisher mein grösstes Projekt, und zeigt viele Fähigkeiten, die wir gelernt haben.
* Zum Verbinden des Arduinos mit dem Raspberry musst ich noch zusätzlich Zeit aufwenden. Dies war ungeplant, aber auch eine gute Herausforderung.

Von den [Zwingenden Zielen](#_1.2_Projektziel) konnte ich alle umsetzen. Hier eine Beschreibung wie die Alarmanlage funktioniert:

**Arduino:**

Das Arduino empfängt zuerst eine 1 oder 0 vom Python Skript. Bei einer 1 geht das Programm in eine Endlosschleife und wartet wieder auf eine 0 des Skripts. In der Endlosschleife ist die Alarmanlage aus.

Ein PIR Motion Sensor ist mit Steckbrückenkabel an das Arduino angeschlossen. Der Sensor sendet 3 Sekunden lang ein Signal, wenn er eine Bewegung im Umkreis von 7 Metern registriert hat. Auf dem Arduino läuft ein Programm, dass alle 0,5 Sekunden auf ein Signal des Sensors wartet. Wenn es ein Signal bekommt, sendet es an das Python Skript, welches auf dem Raspberry läuft und die Antwort des Arduinos überprüft, eine 1, ansonsten eine 0.

Wiederholung

**Python Skript:**

Das Skript starte beim Hochfahren des Raspberry automatisch. Es verbindet sich mit dem Arduino und mit der Datenbank auf dem Raspberry. Auf der Datenbank ist ein Benutzer für das Skript welcher bestimmte Rechte hat ([siehe hier](#_4.1_Implementierung_der)). Das Skript nimmt sich den Eintrag aus der Tabelle AnAus. Wenn dieser auf True ist, sendet es dem Arduino eine 1, ansonsten eine 0. Jetzt bekommt es eine 1 oder 0 des Arduinos zurück. Ist dies eine 1, schreibt er in die Tabelle Türbewegungen einen neuen Eintrag.

Falls der Eintrag in der Tabelle AnAus auf False ist, geht es in eine Endlosschlaufe und überprüft alle 2 Sekunden den Eintrag, ob er wieder auf True ist. Wenn dies so ist, sendet er dem Arduino wieder eine 1.

Wiederholung.

Ich konnte nur eines der drei optionalen Ziele erfüllen. Hier eine Übersicht.

* Die Alarmanlage lässt sich von überall aus dem Internet ab- und anschalten. Dies wurde mit einer Tabelle in der Datenbank und einem einzelnen Eintrag darin umgesetzt.
* Das Arduino Spielt keinen Ton ab bei einer Registrierten Bewegung.
* Das Arduino beginnt auch keine Tonaufnahme.

## 5.3 Ausblick

Nachdem ich das Modul für die Mobilen Apps gehabt habe, werde ich eine App programmieren, die mit dieser Alarmanlage verbunden ist.

Ich werde am Chatbot weiterprogrammieren, weil es mir Spass macht und ich die gelernten Fähigkeiten von den Modulen, die noch kommen direkt in ein Projekt einbauen kann.

An der Alarmanlage werde ich sicher weiterarbeiten, wenn ich wieder die Zeit finde. Ich denke dies hat viel Potenzial und ich kann noch viel weiters hinzufügen.

## 6. Benutzerhandbuch (Mit Chatbot)

# Abbildungsverzeichnis

[1 Phasenstrukturplan 3](https://ipsobildung-my.sharepoint.com/personal/simon_mueller_student_ipso_ch/Documents/Arbeitsprobe/Dokumentation/Simon%20Müller%20Arbeitsprobe%20Dokumentation.docx#_Toc86239647)

[2 Datenbankmodell 3](https://ipsobildung-my.sharepoint.com/personal/simon_mueller_student_ipso_ch/Documents/Arbeitsprobe/Dokumentation/Simon%20Müller%20Arbeitsprobe%20Dokumentation.docx#_Toc86239648)

[3 Struktogramm 4](https://ipsobildung-my.sharepoint.com/personal/simon_mueller_student_ipso_ch/Documents/Arbeitsprobe/Dokumentation/Simon%20Müller%20Arbeitsprobe%20Dokumentation.docx#_Toc86239649)

[4 Ergänzte Klasse 4](https://ipsobildung-my.sharepoint.com/personal/simon_mueller_student_ipso_ch/Documents/Arbeitsprobe/Dokumentation/Simon%20Müller%20Arbeitsprobe%20Dokumentation.docx#_Toc86239650)

[5 Chatbot Klassen 5](https://ipsobildung-my.sharepoint.com/personal/simon_mueller_student_ipso_ch/Documents/Arbeitsprobe/Dokumentation/Simon%20Müller%20Arbeitsprobe%20Dokumentation.docx#_Toc86239651)

[6 Datenstruktur 5](https://ipsobildung-my.sharepoint.com/personal/simon_mueller_student_ipso_ch/Documents/Arbeitsprobe/Dokumentation/Simon%20Müller%20Arbeitsprobe%20Dokumentation.docx#_Toc86239652)

[7 Chatbot GUI 6](https://ipsobildung-my.sharepoint.com/personal/simon_mueller_student_ipso_ch/Documents/Arbeitsprobe/Dokumentation/Simon%20Müller%20Arbeitsprobe%20Dokumentation.docx#_Toc86239653)

# Quellen

<https://codingworld.io/project/den-arduino-mit-dem-raspberry-pi-verbinden-seriell#:~:text=Daf%C3%BCr%20wird%20ein%20funktionst%C3%BCchtiger%20Raspberry%20Pi%20und%20ein,auch%20schon%20mit%20der%20Programmierung%20des%20Arduinos%20beginnen>.

<https://forum.arduino.cc/t/arduino-mit-raspberry-datenbank-per-python-verbinden/584762>

<https://tutorials-raspberrypi.de/lokale-mysql-datenbank-raspberry-pi-datenlogger/>