

# **Emil-Fischer-Gymnasium Schwarzheide**

Eigener Geocache mit einem Arduino

Facharbeit

Informatik

vorgelegt von  
Lange, Justin

Klasse 9c

im Schuljahr 2021/2022

## **Inhaltsverzeichnis**

1. Vorwort
  2. Einleitung
  3. Theoretischer Hauptteil
    - 3.1. Grobe Erklärungen
      - 3.1.1. Was ist ein Geocache?
      - 3.1.2. Was ist ein Arduino?
    - 3.2. Erklärung des Aufbaus des Projektes
    - 3.3. Erklärung der Verkabelung
    - 3.4. Erklärung der einzelnen Bauteile und deren Funktionsweisen
  4. Praktischer Hauptteil
    - 4.1. Bau des Projektes
    - 4.2. Erneute Erklärung der Funktionsweise anhand des Projektes selber
    - 4.3. Der Code
  5. Zusammenfassung
  6. Danksagungen
- Anhang, Bilder/Medien
- Literaturverzeichnis

## **1. Vorwort**

In dieser Facharbeit soll es darum gehen, wie man seinen eigenen Geocache mit einem Arduino baut. Die Idee dazu hatte ich dadurch, dass ich schon seit längerer Zeit mit meiner Familie geocachen gehe und auch seit kurzen einen eigenen Account habe. Zudem habe ich in meiner Werkstatt ein Arduino Uno übrig, für welchen ich weder einen Verwendungszweck noch einen ordentlichen Lagerplatz habe. Ich arbeite schon seit längerem mit Arduinos und habe schon einige Projekte, wie zum Beispiel mein eigenes Werkstatt- Automationssystem oder mit einem Wlan-fähigen Arduino (auch ESP genannt) und Open Source Software (frei verfügbare Software) mein Zimmer mit RGB ausgestattet. Zudem wollte ich schon länger mal ein eigenes kleines Rätsel machen und da bietet sich dies natürlich an. Ich nutze diese Facharbeit um mein ersten eigenen Geocache zu veröffentlichen. Dieser wird später nach Genehmigung unter dem Namen Cache 'o Phone in Guteborn zu finden sein.

## **2. Einleitung**

In diesem Projekt/ in dieser Facharbeit soll ein kleines Rätsel mit Hilfe eines Arduinos gebaut werden. Dieses wird dann später als Geocache veröffentlicht und kann somit auch von Geocachern gefunden und gelöst werden. Das Rätsel ist folgenden Maßen aufgebaut: Als erstes soll, wie bei einem echten Telefon auch, ein TAE Stecker in die dafür angebrachte Dose gesteckt werden. Dann kann man ein Telefonhörer, welcher aus einem alten Telefon stammt, abgenommen werden. Daraufhin spielt dieser eine kleine Geschichte ab, in welcher Zahlen versteckt sind. Diese soll man dann auf einem Keypad eingeben. Tut man dies in der richtigen Reihenfolge, lässt sich ein kleines Türchen, hinter dem sich das Logbuch befindet, öffnen.

## **3. Theoretischer Hauptteil**

### **3.1. Grobe Erklärungen**

#### **3.1.1. Was ist ein Geocache**

Zuerst einmal, was ist überhaupt ein Geocache?

Ein Geocache kann eine kleine Box, eine Filmdose, eine große Kiste oder vieles andere sein. In dieser befindet sich dann ein sogenanntes Logbuch. In dieses schreibt man seinen Username, welchen man bekommt, wenn man sich in der Geocaching App registriert. Diesen kann man sich selber aussuchen. Den Cache selber kann man in der App finden d.h. er wird dort angezeigt und man kann auch sehen wo dieser ungefähr ist. Man kann nicht genau sehen, wo der Cache versteckt ist, da das Ziel ist, dass man diesen selber sucht und findet. Nach dem Fund muss man sich in das Logbuch des Caches eintragen. Zusätzlich muss man sich jedoch in das digitale Logbuch der App, oder der Internetseite eintragen. Nur so ist es ein gültiger Fund. Wie das ganze genau funktioniert, kann man nochmal auf [geocaching.com](http://geocaching.com)<sup>1</sup> erfahren. Dort gibt es ein kurzes Video, welches das genauer erklärt.

#### **3.1.2. Was ist ein Arduino?**

Ein Arduino ist ein sogenannter Microcontroller. Das sind eine Art kleine Computer, welche aber den Vorteil haben, das Dinge wie Arbeits- und Programmspeicher direkt im selben Chip (Computer-Chip nicht die zum Essen) befinden. Jedoch sind diese teilweise nicht so leistungsfähig wie ein richtiger PC, aber manchmal leichter zu programmieren.

Den Arduino kann man mit der Programmiersprache C bzw. C++ programmiert. Diese ist relativ einfach zu verstehen und somit ist die Nutzung eines Arduinos auch sehr einfach. An den Seiten gibt es jeweils vom USB-Port aus gesehen links 14 so genannte I/O Pins. I/O steht für Input/Output was bedeutet, dass man durch diese Pins sowohl etwas Bestimmtes ein- oder ausgeben kann. Auf der rechten Seite befinden sich 6 Analoge Pins. Diese haben im Gegensatz zu den I/O Pins die Besonderheit, dass man auf diesen die eingehende Spannung messen kann. Auf der rechten Seite gibt es noch andere Pins, welche zur Spannungsversorgung genutzt werden können. Programmiert wird in der Arduino IDE oder einer anderen sogenannten Entwicklungsumgebung, welche das Arduino unterstützt. Zum Hochladen des Codes muss das Arduino mittels eines USB-Typ A Kabels, an den PC angeschlossen und z.B. in der Arduino IDE der Port ausgewählt werden.

### **3.2. Erklärung des Aufbaus des Projektes**

Der Geocache selber soll in einer etwas größeren Holzbox gebaut werden. In der unteren Hälfte (vorzugsweise in der unteren linken Ecke) befindet sich eine kleine Tür, hinter welcher sich das Logbuch befindet. Diese wird jedoch bis zur richtigen Eingabe des Codes von einem elektromagnetischen Schloss geschlossen gehalten. Oben soll noch ein kleines Fach für eine Powerbank mit einem USB-Typ A Anschluss für die Stromversorgung sein.

Aber zuerst einmal zum Aufbau. In der Holzkiste soll nach den öffnen verschiedene Bauteile zu sehen sein. Darunter ein Telefon, ein sogenanntes Keypad (eine Art Tastatur mit den Zahlen 0-9, A-D, # und \*) eine TAE- Dose (Telekommunikations Anschluss Einheit) und die oben genannte Tür. Nachdem in das oben gelegene Fach eine Powerbank eingelegt und über das USB-Kabel mit dem Geocache verbunden wurde soll man wie auch bei einem echten Telefon ein Kabel in die entsprechende TAE- Dose gesteckt werden. Sobald dies getan ist, wechselt eine kleine LED, welche neben der Dose platziert ist, die Farbe von Rot auf Grün. Als nächstes soll das Telefon von der Halterung abgenommen und ans Ohr gehalten werden. Nach kurzer Zeit soll dann über das Soundmodul, welches im inneren der Box versteckt ist, eine kleine Geschichte erzählt werden, in welcher eine Zahlenkombination versteckt ist. Diese soll auf dem Keypad in der richtigen Reihenfolge eingegeben werden. Trifft der Fall ein, dass die Zahlenkombination richtig eingegeben wird, soll sich das elektromagnetische

Schloss deaktivieren und die Tür geöffnet werden. Wird sie jedoch falsch eingegeben, wird die Nachricht erneut abgespielt und es muss erneut versucht werden.

### **3.3. Erklärung der Verkabelung**

Wie schon in der Erklärung des Aufbaus beschrieben, startet alles im Powerbankfach. Dort gibt es ein USB-Typ A Kabel, welches an die Powerbank angeschlossen werden soll. Von diesem Kabel aus werden ein rotes und ein schwarzes Kabel abgehen. Das ist das Plus und Minus bzw. positive und negative Kabel. Das negative Kabel wird auf eine Schraubklemme gelegt, welche auf das ganze System verteilt wird. Das positive Kabel wird zuerst über die TAE-Dose gelegt, sodass das ganze System erst Strom bekommt sobald das TAE-Kabel eingesteckt ist. Dieses Kabel ist im Prinzip nur eine Schlaufe, welche den Eingang der Dose mit dem Ausgang verbindet. Der Ausgang wird dann, wie bei der negativen Leitung, auf eine Schraubklemme gelegt und auf das ganze System verteilt. Als erstes geht der Strom zum Arduino, was das Soundmodul, die Schlösser und das Keypad steuert und sozusagen das Herz von allem ist. Das Keypad wird über einen VGA-Verbinder, welches dazu dienen soll, das Keypad im Falle eines Versagens einen leichten Austausch ermöglichen, an die Pins D2 bis D9 des Arduinos angeschlossen.

Als nächstes komme ich zu den elektromagnetischen Schlössern. Da diese so viel Strom ziehen, dass es das Arduino durchbrennen lassen würde, kann ich diese nicht direkt daran anschließen. Die Lösung dafür? Ein sogenanntes Relay, welches im Prinzip ein elektrisch betätigter Schalter ist. Das Gute daran ist, dass die Seite, die das Relay schaltet und die andere, die geschaltet wird, aufgrund des Aufbaus elektrisch voneinander getrennt sind. Für einen erleichterten Aufbau nutze ich hierfür ein vorgefertigtes Relay Modul von der Firma AZ-Delivery<sup>®</sup>, welches alle benötigten Bauteile und das Relay bereits auf der Platine hat. Das Soundmodul wird sowohl mit dem Arduino als auch mit dem Telefon, welches man von außen benutzen kann, verbunden. Mit dem Arduino wird das Ganze über die Pins D10 und D11 verbunden und kommuniziert dann mit dem Arduino bzw. bekommt von diesem den Befehl, die auf einer SD-Karte abgespeicherte Audiodatei abzuspielen. Damit man auch etwas hören kann, wird der Lautsprecher des Telefons an die Audio-Ausgangspins des Soundmoduls angeschlossen. Damit die Audiodatei nicht sofort nach dem Anstecken der Powerbank, bzw. nach dem Hochfahren des Arduinos abgespielt wird, gibt es einen kleinen

Knopf, welcher gedrückt werden soll, um das Abspielen zu starten. Ein genaueres Verkabelungsdiagramm kann in Bild 1 auf der Seite Bilder/Medien gefunden werden.

### **3.4. Erklärung der einzelnen Bauteile und deren Funktionsweise**

#### **Das elektromagnetische Schloss**

Das elektromagnetische Schloss besteht, wie der Name es schon verrät, aus einem Elektromagneten, welcher in unserem Fall mit 5 Volt unter Spannung gesetzt wird, einem keilförmigen Schlossteil, welches federgelagert ist, damit z.B. beim Schließen der Tür der Keil von alleine zurück geht und nicht gedrückt werden muss, und einem Gehäuse in dem das Ganze verbaut ist. Sobald an den abführenden Kabeln die passende Spannung, welche aber von Schloss zu Schloss variieren kann, angelegt wird (hier 5V), fließt durch den Elektromagneten Strom. Dieser erzeugt nun ein Magnetfeld, welches den keilförmigen Schlossteil zurück und in das Gehäuse zieht. Sobald keine Spannung mehr anliegt, deaktiviert sich der Elektromagnet, das Magnetfeld bricht zusammen und der Keil wird durch die Feder wieder in seine Ursprungsposition zurückgedrückt.

#### **Das Keypad/Nummernpad**

Das Keypad ist in einer 4x4-Matrix aufgebaut, welche in dem ersten 3x3-Feld von oben links aus gesehen den Zahlen 1-9 besteht. In der vierten Spalte befinden sich die Buchstaben A-D und in der vierten Zeile sind das \*- Zeichen, eine 0 und das #- Zeichen. Ein Bild des Keypads kann nochmal in Bild 2 gefunden werden. Das Keypad an sich besteht aus einer Platine (auch PCB [Printed Circuit Board] genannt), einer Membranschicht und den Tasten, welche aus Plastik bestehen und einem Deckel, der Ausfräsungen an den Stellen hat, wo die Tasten sind. Wenn man eine Taste betätigt, drückt man gleichzeitig auch eine Membran auf die Platine. Diese stellt dann durch eine kleine leitende Scheibe die auf der Innenseite befestigt ist, den Kontakt zwischen zwei Pins/Ausgängen des Keypads her. Damit zwischen der Membran und der PCB kein Dauerkontakt besteht, ist an der Stelle, wo die Membran die Platine berührt eine Ausfräsung, die nur von einem, innerhalb der Membran befestigten Plättchen überbrückt werden kann. Zusätzlich kehrt die Membran nach loslassen des Knopfes in die Ursprungsposition zurück

## **Das Relay**

Das Relay ist ein interessantes Bauteil. Es ist eine Art elektrischer Schalter, bei dem die Seite, die schaltet und die die geschalten wird, elektrisch voneinander getrennt sind. Dies trägt unter anderem dazu bei, dass das Relay mit z.B. 0,01 Ampere bei 5V geschalten werden kann und die geschaltene Seite mit Spannungen mit z.B. 230V, welches der Netzspannung in Deutschland entspricht. In meinem Fall schalte ich mit dem Relay nur das 5V Schloss, da dies wie schon gesagt, zu viel Strom zieht, um es direkt an das Arduino anzuschließen. Bei einem Relay wird die Trennung von schaltend und geschalten dadurch erreicht, dass die schaltende Seite, einen Elektromagneten beinhaltet, welcher dann über einen kleinen Hebel einen Plastikschieber bewegt. Dieser betätigt dann ein oder zwei Paar Kontakte. Dieser Plastikschieber dient zur besagten Isolierung/ Trennung der schaltenden- und geschalteten Seite.

## **Das DFPlayer Mini Sound Modul**

Der DFPlayer Mini ist ein Soundmodul, das in Kombination mit einem Arduino zur Ausgabe von Tonmaterial, das auf einer Mikro SD-Karte oder einem angeschlossenen USB-Stick gespeichert sind, genutzt werden kann. Alternativ kann dies auch als „stand alone“ Modul genutzt werden. In dem Fall wird allein das Modul und Bauteile, wie Knöpfe und Batterien zur Steuerung benötigt und nicht noch ein Arduino oder andere, ähnliche Bauteile. Zur Kommunikation mit dem Arduino, verwendet der DFPlayer Mini eine serielle Schnittstelle, welche über RX (Receiver bzw. Empfänger) und TX (Transmitter bzw. Sender) läuft. Verkauft wird das Modul von der Firma [DFRobot](#)<sup>4</sup>, kann aber auch bei anderen Händlern wie Conrad<sup>®</sup>, Amazon<sup>®</sup>, Reichelt<sup>®</sup> und anderen gekauft werden. Die SD-Karte wird bei dem Modul oben in die dafür vorgesehene Halterung eingeschoben und wird somit vom Modul erkannt. Einen USB-Port gibt es jedoch nicht und somit muss dieser selbst angebracht/verlötet werden. Dazu stehen vom Modul die Datenleitungen für die Kommunikation mit dem USB-Stick bereit. Eine genauere Beschreibung kann nochmal auf der Wiki- Seite des [DFPlayerMini's](#)<sup>5</sup> gefunden werden.

## 4. Praktischer Hauptteil

### 4.1. Bau des Projektes

Nun komme ich zum Bau des Projektes.

Angefangen habe ich damit, die Bauteile nach dem Kauf zu testen. Das heißt, ich habe auf den Arduino einen Beispielcode aufgespielt und damit alle weiteren Bauteile getestet. Als nächstes habe ich ein Flachbandkabel und einen VGA-Verbinder an das Keypad gelötet und diese mit Schrumpfschläuchen gegen Kurzschlüsse und andere Schäden zu schützen. Danach habe ich eine Platine angefertigt, welche genau auf das Arduino zugeschnitten ist. Später habe ich den DFPlayerMini verlötet und alles miteinander verbunden und Stecker angebracht, damit ich das Arduino austauschen kann, wenn es defekt ist. Zum Testen habe ich den DFPlayerMini in ein sogenanntes Steckbrett gesteckt, da dies im Gegensatz zu löten nicht permanent befestigt ist und somit Fehler leicht zu beheben sind. Danach habe ich die SD-Karte eingelegt und das Modul laut Verkabelungsdiagramm des Herstellers verbunden. Als nächstes habe ich den Code auf den Arduino hochgeladen und hörte, dass das Modul perfekt und fehlerfrei funktionierte. Danach habe ich es wie geplant, auf die aufsteckbare Platine gelötet und als Anschluss für den Lautsprecher eine Schraubklemme genommen. Damit ich nicht zu viel Lötzinn verschwende, habe ich Kupferdraht und kleine Kabel verwendet. Da im Verkabelungsdiagramm des Herstellers ein 1 KOhm Widerstand eingezeichnet ist, habe ich diesen gleich als kleine Brücke zwischen dem Pin und dem DFPlayerMini verwendet. Im Anschluss habe ich die Platine aufgesteckt und das Arduino mit Strom versorgt. Leider bekam ich eine Fehlermeldung, da eine Verbindung falsch verlötet war. Nachdem der entdeckte Fehler behoben war, konnte ich wie geplant weiterarbeiten.

Als nächstes habe ich mich um den Schlossmechanismus gekümmert. Ich habe die Schlösser mit Hilfe einer Crimpzange mit neuen Steckern versehen. Diese habe ich dann zusammen mit dem Relay so verlötet, dass am Ende nur Strom und einen Signalpin zum Relay führen muss. Jedoch hatte ich einige Zeit das Problem, als ich das Relay und somit die Schlösser einschalten wollte, die Spannung eingebrochen ist. Das bedeutet, dass die Spannung von den eigentlichen 5V auf ca. 2-3V gesunken ist. Aus diesem Grund habe ich als erstes versucht, die Spannung mit Hilfe eines sogenannten Boost Converters hochzusetzen. Als dies aber mehr Probleme als Lösungen gebracht hat, habe ich mich entschieden, eines der zwei Schlösser zu entfernen und am Ende das eine Schloss direkt an die Bossbars anzuschließen. Das verbleibende Schloss, das Relay, sowie das Arduino habe ich auf eine kleine Holzplatte und

damit auch an der Facharbeit festgeschraubt. Unter die Holzplatte habe ich doppelseitiges Schaumstoffklebeband geklebt. Als nächstes habe ich das Keypad mit passenden Schrauben befestigt. Danach habe ich alle Drähte laut Kabeldiagramm<sup>1</sup> verlötet und die zwei größere Schraubklemmen/Bossbars, ein Stück Metall mit mehreren Befestigungsmöglichkeiten für Kabel, mit dem Klebeband befestigt und alle Kabel angeschraubt. Da ich die Spannung bei meinem Labornetzteil zu hoch eingestellt hatte, ist der DFPlayer Mini leider durchgebrannt. Zum Glück hatte ich noch einen zweiten als Ersatz bestellt. Dadurch konnte ich das defekte Soundmodul schnell austauschen und habe zur Sicherheit noch einen 7805 Spannungsregler eingebaut. Als nächstes habe ich die Holzwand in die Holzbox, welche ich als Gehäuse nutze eingebaut. Damit diese auch auf der richtigen Höhe ist, habe ich kleine Holzleisten in die 3 Ecken eingeschraubt. Die vierte Ecke musste ich freilassen, da dort die Tür vorgesehen ist. Jedoch bemerkte ich schnell, dass die Holzwand sehr instabil war, schraubte ich deswegen an beide Seiten jeweils in die Mitte noch zwei Holzleisten an. Für die Ablage der Powerbank habe ich zusätzlich noch ein kleines Fach mit einem Loch für das USB-Kabel gebaut. Sowohl die Kiste, als auch die Holzwand habe ich zur Wetterbeständigkeit lasiert. Den oberen Griff haben ich mit einem kleinen lasiertem Holzstück verschlossen, damit ich an dieser Stelle einen Magneten für die Lampenhalterung befestigen konnte. Als alles, bis auf das Türchen, hinter dem das Logbuch sein soll, fertig war, wurde es Zeit den Code zu schreiben. Bevor ich jedoch den Code erkläre, beschreibe ich nochmal wie das Projekt aufgebaut ist, indem ich es mit dem Projekt selber demonstriere.

#### **4.2. Erklärung der Funktionsweise anhand des Projektes selber**

Kommen wir zuerst dazu, wie das Projekt jetzt schlussendlich funktioniert und aufgebaut ist. Das Gehäuse besteht aus einer lasierten Holzkiste. An diese wurde von außen ein Zahlenschloss angebracht, damit nur Geocacher mit dem Zahlencode den Cache öffnen können. Den Zahlencode zum Schloss habe ich in die Beschreibung des Geocaches in der App zum Schluss des Rätsels gestellt. Da ich auch vorhabe, diese Facharbeit nach Fertigstellung auf meiner Website ([justricity.de](http://justricity.de)) hochzuladen, werde ich den späteren Zahlencode hier nicht bekannt geben. Auf Bild 3 kann man die Box im geschlossenen Zustand von vorne bzw. wenn sie montiert wäre von der Seite sehen. Auf dem vierten Bild sieht man den eigentlichen Teil des Caches. Dort befinden sich ganz oben angebracht eine Lampe, zum Nutzen des Caches im Dunkeln, links darunter das Fach mit dem USB-Kabel

zum Anschließen und ablegen einer Powerbank. Rechts daneben befindet sich das Telefon mit seiner Halterung, durch welches später die Nachricht abgespielt werden soll. Links daneben und unter dem Powerbank Fach befindet sich das Keypad zur Eingabe. Hinter das große leere Fach kommt das Logbuch, vor welches sich eine Tür befindet, welches ich zur besseren Veranschaulichung abmontiert habe. Als letztes befindet sich in der unteren rechten Ecke die TAE-Dose mit den LEDs und dem passenden Kabel.

Als nächstes habe ich die Holzplatte abgenommen und es ist die Rückseite auf dem fünften Bild zu sehen. Dort befinden sich alle Bauteile, welche dieses Projekt zum Laufen bringen. Direkt unter dem Powerbankfach befinden sich die Bossbars. Eine für + und eine für -. Links neben diesen befindet sich das Arduino inklusive aufgesteckter Platine mit Soundmodul. Darunter befindet sich das Relay Modul mit dem rechts daneben befindlichem Elektromagnetschloss. Zwischen Arduino und Relay habe ich die ganzen Kabel verlegt und auf Ordnungsgemäße Verarbeitung geachtet. Hierzu haben ich die Kabel mit schwarzem Isolierband umwickelt und mit Kabelbindern zusammengebunden. Das spiralförmige Kabel des Telefons wurde direkt an die blaue Schraubklemme auf der Platine eingeschraubt. Unter den Bossbars und über dem Relay befindet sich das Keypad, welches über den VGA-Verbinder und das regenbogenfarbene Kabel an die Platine auf dem Arduino angeschlossen ist. Das kleine Holzplättchen unten ist von dem alten, zweiten Schloss. Dies kann ich aber leider aufgrund seiner Befestigung nicht entfernen ohne evtl. etwas zu beschädigen. Zwischen den beiden Bossbars habe ich noch einen Kondensator hinzugefügt, welcher die Spannung ein wenig stabilisieren soll. Ein Kondensator ist wie eine Art Batterie, nur dass dieser seine Ladung nicht sehr lange hält, sich dafür aber schnell wieder aufladen kann.

### **4.3. Der Code**

Da das Einfügen des Codes über vier Seiten lang ist und damit den vorgegebenen Rahmen der Facharbeit sprengen würde, werde ich diesen auf meiner Website [justricity.de](http://justricity.de) unter der Kategorie Cache o' Phone posten, sowie ihn als extra Datei auf dem USB-Stick beilegen. (Datei: Cache-o-phone.ino)

## **5. Zusammenfassung**

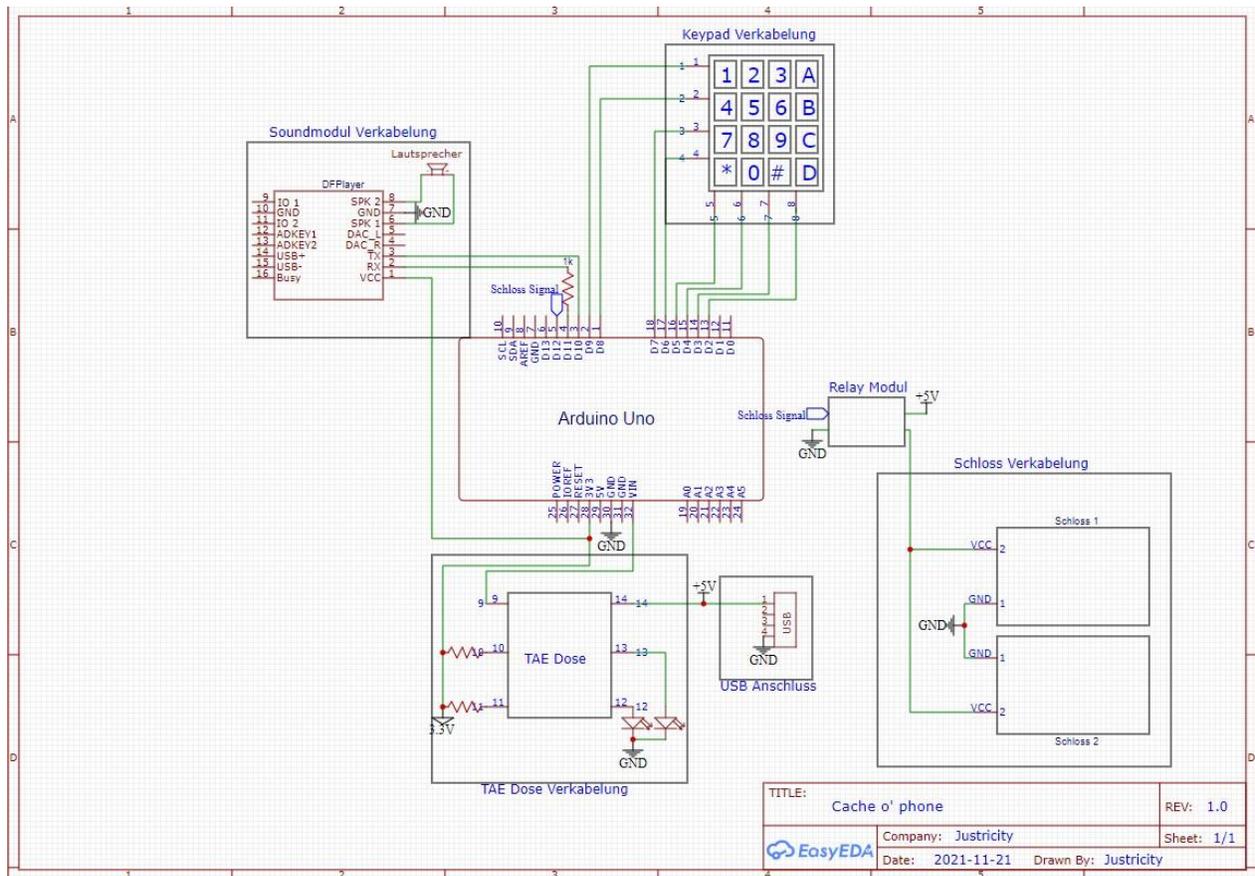
Das Erstellen dieser Facharbeit und das Bauen des Projektes war eine recht anstrengende und schwierige, aber dafür auch sehr interessante und lehrreiche Aufgabe, die mir sehr viel Spaß

gemacht hat. Ich konnte vieles neues lernen und seit langer Zeit etwas bauen, dass diesmal sogar anderen Leuten sinnvoll im Rahmen eines Caches zur Verfügung steht.

## 6. **Danksagungen**

Vielen Dank nochmal an meine Mutti, die einen Großteil der Kosten für die Teile übernommen hat, meine Freunde und Familie, die mich motiviert haben und mir Tipps gegeben haben und auch an Karsten, der mir geholfen hat, das Projekt schlussendlich bei Geocaching zu veröffentlichen, da dieser damit auch mein erster eigener Cach ist.

## Bilder/Medien



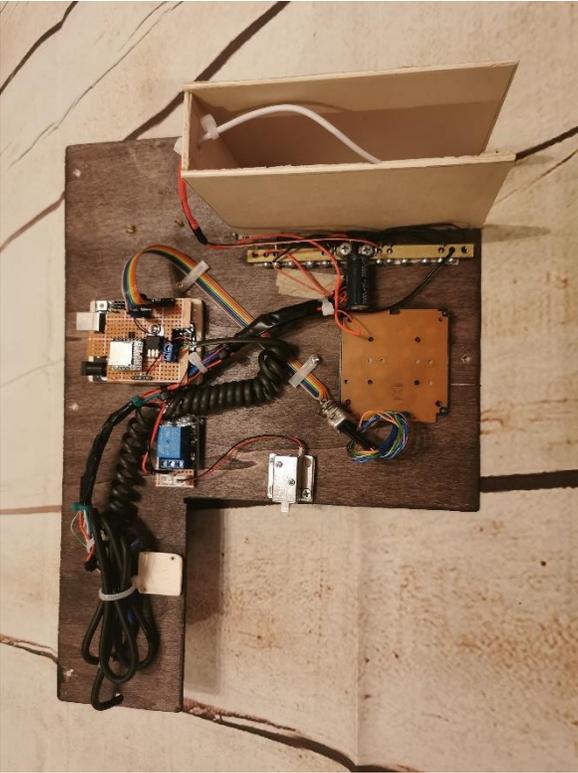
1: Verkabelungsdiagramm (Erstellt mit EasyEDA)



1: Bild des Keypads (Quelle 3)



3: Die geschlossene Box



5: Die Elektronik



4: Die geöffnete Box

## Literaturverzeichnis

---

<sup>1</sup> <https://www.geocaching.com/play> letzter Zugriff am 16.02.2022 um 16:00 Uhr

<sup>2</sup> <https://www.mikrocontroller-elektronik.de/wtv020-sound-oder-sprache-ausgeben/> letzter Zugriff am 06.12.2021 um 14:40 Uhr

<sup>3</sup> [https://www.google.com/url?sa=i&url=https%3A%2F%2Fwww.az-delivery.de%2Fproducts%2F4x4-matrix-numpad&psig=AOvVaw2eng\\_T\\_MFj9VJEQ2WTWTdv&ust=1638998698315000&source=images&cd=vfe&ved=0CAsQjRxgFwoTCODvIvQ0vQCFQAAAAAdAAAAABAJ](https://www.google.com/url?sa=i&url=https%3A%2F%2Fwww.az-delivery.de%2Fproducts%2F4x4-matrix-numpad&psig=AOvVaw2eng_T_MFj9VJEQ2WTWTdv&ust=1638998698315000&source=images&cd=vfe&ved=0CAsQjRxgFwoTCODvIvQ0vQCFQAAAAAdAAAAABAJ) letzter Zugriff am 07.12.2021 um 10:30 Uhr

<sup>4</sup> [dfrobot.com](https://dfrobot.com) letzter Zugriff am 26.12.2021 um 13:29 Uhr

<sup>5</sup> [https://wiki.dfrobot.com/DFPlayer\\_Mini\\_SKU\\_DFR0299](https://wiki.dfrobot.com/DFPlayer_Mini_SKU_DFR0299) letzter Zugriff am 26.12.2021 um 13:37 Uhr

---

## Erklärung

Hiermit erkläre ich, dass ich die vorliegende Arbeit selbstständig und ohne fremde Hilfe verfasst und keine anderen als die im Literaturverzeichnis angegebenen Hilfsmittel verwendet habe.

Insbesondere versichere ich, dass ich alle wörtlichen und sinngemäßen Übernahmen aus anderen Werken als solche kenntlich gemacht habe.

Verwendete Informationen aus dem Internet sind dem/der Lehrer(in) vollständig im Ausdruck zur Verfügung gestellt worden.

Name in Druckschrift:

Ort, Datum, Unterschrift

---

## Bewertungsbogen

Schuljahr:

Name:

Klasse:

Fach:

Thema:

Abgabetermin:

Fachlehrer(in):

Bewertung: