

DIPLOMARBEIT

Gesamtprojekt

Chess4U

Projektleitung, Marketing, Produktvideo und Finanzmanagement

Gregor Sedlak

5BHITM

Betreuer: Mag. DI Richard Wurzer

Stv. Projektleitung, Webseite, Raspberry PI Konfiguration, GPIO Ansteuerung

Denis Drekovic

5BHITM

Betreuer: Mag. DI Richard Wurzer

Programmierung Server und Client, Datenbank

Raphael Lewitsch

5BHITM

Betreuer: Mag. DI Richard Wurzer

PCB Layout, Multiplexer

Theodor Kirchknopf

5BHITM

Betreuer: Mag. DI Richard Wurzer

Ausgeführt im Schuljahr 2016/17

Abgabevermerk:

Datum: 04.04.2017

übernommen von:

KURZFASSUNG

Die Diplomarbeit „Chess4U“ soll die altbekannte Methode der Spielzugaufzeichnung auf Papier ersetzen. Dieses Projekt soll das Problem einer versäumten oder falsch notierten Spielzug-Aufzeichnung in der Schachwelt beheben.

Ein Schachbrett zeichnet mittels Sensoren an den Figuren und im Schachbrett alle Spielzüge auf. Mit diesem neuartigen Schachbrett müssen professionelle Schachspieler ihre Spielzüge nicht mehr per Hand auf einem Zettel notieren, da diese durch Soft- und Hardware erfasst werden und auf einer grafischen, benutzerfreundlichen Oberfläche in Echtzeit wiedergegeben werden.

Diese Umsetzung soll günstiger und somit auch für den Massenmarkt besser geeignet sein als bereits bestehende Systeme.

Darüber hinaus wird jede handwerklich- und technisch begabte Person dieses Projekt nachbauen können, da die Dokumentation inkl. dem Sourcecode unter anderem auch auf der Webseite von Chess4U¹ kostenlos erhältlich sein wird.

¹ www.chess4u.at, zul. geladen am 09.01.2017

ABSTRACT

The final year project 'Chess4U' should replace the renowned movement recording on paper. This project should solve the problem, which occurs by a missed or wrong recorded movement at a chess game.

A chessboard and pawns will be equipped with sensors which should record the chess movements. Certain chess movements might be displayed on request. Prospectively professional chess players will not have to record their movements manually on paper, because the movements will be recorded by a self-provided soft- and hardware. Furthermore, the movements will be displayed on a user-friendly graphical interface or rather they will be streamed in real-time.

This realization should be more favourable and thus more suitable for the mass market than already existing systems. In addition to this every technically skilled person should be able to recreate this project, because the documentation including the source code will be offered for free at the Chess4U¹ website.

¹ www.chess4u.at, last loaded on 2017-01-09

PROBLEMSTELLUNG

Bei professionellen Schachturnieren besteht die Pflicht, dass jeder Spielzug aufgezeichnet werden muss. Dies geschah bisher per Hand, mit einem Stift auf einem Zettel.

Die schriftliche Spielzugaufzeichnung ist sehr fehleranfällig. Es können immer wieder Spielzüge nicht aufgezeichnet oder falsch notiert werden.

Darüber hinaus wird ab einer bestimmten Spielzeit, abhängig von der gewählten Spielart (Schnellschach, Blitzschach), die Mitschreibepflicht außer Kraft gesetzt. So muss man beispielsweise bei Schnellturnieren die Züge nicht mehr mitschreiben, da dies zu viel Zeit in Anspruch nimmt, und das Risiko, dass Fehler entstehen, enorm ist (Schnellschach, Wikipedia, 2017) .

Vor allem, weil sich bei diesen Spielarten die Positionen der Schachfiguren in wenigen Sekunden ändern und es somit auch für Dritte sehr schwer bis unmöglich ist, diese Spielzüge aufzufassen. Hier wirkt diese Diplomarbeit entgegen und soll dieses Verfahren der mühsamen Spielzugaufzeichnung ersetzen.

EHRENWÖRTLICHE ERKLÄRUNG

Ich versichere,

- dass ich meinen Anteil an dieser Diplomarbeit selbstständig verfasst habe,
- dass ich keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe
- und mich auch sonst keiner unerlaubten Hilfe bzw. Hilfsmittel bedient habe.

Wien, am 04.04.2017

Gregor Sedlak

Denis Drekovic

Raphael Lewitsch

Theodor Kirchknopf

INHALTSVERZEICHNIS

KURZFASSUNG	II
ABSTRACT	III
PROBLEMSTELLUNG.....	IV
EHRENWÖRTLICHE ERKLÄRUNG	V
VORWORT.....	X
VERZEICHNIS DER TABELLEN.....	XI
VERZEICHNIS DER ABBILDUNGEN	XII
1 KONZEPT	16
1.1 RFID-TECHNOLOGIEN	17
1.1.2 FUNKTION DER RFID-TECHNOLOGIE.....	18
1.1.3 ARTEN VON TRANSPONDERN	19
1.1.4 FREQUENZBEREICHE	20
1.1.4.1 LF - LOW FREQUENCY	21
1.1.4.2 HF - HIGH FREQUENCY	21
1.1.4.3 UHF - ULTRA HIGH FREQUENCY	22
1.1.4.4 SUHF - ULTRA HIGH FREQUENCY.....	22
1.1.5 NFC.....	23
2 HARDWAREPLATTFORM	24
2.1 RASPBERRY PI.....	25
2.1.1 RASPBERRY SOFTWARE	26
2.1.1 RASPBERRY GRUNDKONFIGURATION	27
2.1.2 EXPLORE NFC ERWEITERUNGSMODUL(READER)	30
2.1.2.1 KONFIGURATION DER NFC-ERWEITERUNGSPLATINE.....	31
2.1.2.2 NXPPY PYTHON-WRAPPER.....	33

2.1.2.3 ERWEITERUNGSMODUL BEARBEITEN	36
2.1.2.3.1 ANTENNE(N) ERSTELLEN	40
2.2 MULTIPLEXER	42
2.2.1 EINLEITUNG MULTIPLEXER	42
2.2.2 ALLGEMEINE ERKLÄRUNG	42
2.2.3 SPEZIFISCHE ERKLÄRUNG	43
2.3 LEITERPLATTE	49
2.3.1 EINLEITUNG LEITERPLATTE (PCB)	49
2.3.2 ALLGEMEINE ERKLÄRUNG	49
2.3.3 SPEZIFISCHE ERKLÄRUNG	50
2.3.4 EAGLE 7.7.0 (SOFTWARE).....	51
3 LÖTEN AM PCB-LAYOUT	53
3.1.1 PCB MIT READER UND ANTENNEN VERBINDEN	54
4 SOFTWARE	58
4.1 ANTENNEN SWITCH	58
4.2 WEBSOCKETS.....	60
4.3 WEBSOCKET CHESS4U SERVER.....	63
4.4 WEBSOCKET CHESS4U CLIENT	65
4.5 DIE BENUTZEROBERFLÄCHE	71
5 DATENBANK.....	78
5.1 ALLGEMEIN	78
5.2 RELATIONALE DATENBANK	78
5.3 DATENBANKMANAGEMENTSYSTEM.....	79
5.4 RELATIONALES DATENBANKMANAGEMENTSYSTEM	80
5.5 MYSQL	80

5.6 ENTITY RELATIONSHIP DIAGRAM	81
5.7 TABELLEN - CHESS4U.....	83
6 WETTBEWERBE	86
6.1 ALLGEMEIN	86
6.2 NETIDEE.....	86
6.2.1 ABLAUF	87
6.2.2 VERANSTALTUNGEN:.....	87
6.3 JUGEND INNOVATIV.....	88
6.4 LIZENZEN	88
7 MARKETING	89
7.1 ALLGEMEIN	89
7.2 SOCIAL MEDIA	90
7.3 WEBSEITE	90
7.4 WERBEVIDEO	90
7.4.1 SCRIPT	91
7.4.2 STORYBOARD	92
7.4.3 IDEE	93
7.4.4 BEARBEITUNG DES FOOTAGE	93
7.4.4.1 AFTER EFFECTS	93
7.4.4.1.1 STABILISIERUNG DER AUFNAHMEN	93
7.4.4.1.2 INTRO & OUTRO	94
7.4.4.1.3 SPIELZUG MIT GLOW-EFFEKT.....	98
QUELLENVERZEICHNIS.....	104
ANHANG.....	108
ARBEITSAUFTEILUNG	108

AUSZUG AUS DEM DIPLOMARBEITSANTRAG 111

VORWORT

Während unserer Diplomarbeit haben uns viele Personen begleitet und unterstützt. Hiermit möchten wir uns bei allen Personen bedanken, die uns bei der Diplomarbeit mit zahlreichen Ideen und Vorschlägen geholfen haben.

Ein ganz besonderes Dankeschön widmen wir Mag. DI(FH) Richard Wurzer für die Betreuung unserer Diplomarbeit, für die ausgezeichnete Hilfsbereitschaft und für die Geduld, welche er uns aufgebracht hat.

VERZEICHNIS DER TABELLEN

<i>Tabelle 1 Mögliche Auswahl eines Multiplexer</i>	43
<i>Tabelle 2 Mögliche Auswahl einer Antenne</i>	45

VERZEICHNIS DER ABBILDUNGEN

<i>Abbildung 1 Konzept Chess4U</i>	<i>16</i>
<i>Abbildung 2 RFID-Tag, welcher am magnetischen Feld vom Reader gekoppelt ist. Entnommen aus (STMicroelectronics, 2016).</i>	<i>18</i>
<i>Abbildung 3 Frequenzbereiche der RFID-Technologie. Entnommen aus (Elektronik-Kompendium.de, 2017).</i>	<i>20</i>
<i>Abbildung 4 Ein geöffneter 13,56 MHz Tag.</i>	<i>21</i>
<i>Abbildung 5 Ein Transponder eines Tags.</i>	<i>21</i>
<i>Abbildung 6 Die Vorderseite eines geöffneten RFID-USB-Readers.</i>	<i>24</i>
<i>Abbildung 7 Die Rückseite inkl. Antenne eines geöffneten RFID-USB-Readers. ...</i>	<i>24</i>
<i>Abbildung 8 Einplatinencomputer „Raspberry Pi“ inkl. SD-Karte</i>	<i>25</i>
<i>Abbildung 9 Programm "Win32 Disk Imager" mit dem ausgewählten Image</i>	<i>26</i>
<i>Abbildung 10 Menüpunkt "Advanced Options"</i>	<i>27</i>
<i>Abbildung 11 Menüpunkt "SSH"</i>	<i>28</i>
<i>Abbildung 12 Informationen eines Routers über den Raspberry an.</i>	<i>29</i>
<i>Abbildung 13 Die eingegebene IP-Adresse im PuTTY-Client.....</i>	<i>29</i>
<i>Abbildung 14 Ein Raspberry PI inkl. einem Explore-NFC Erweiterungsmodul. Entnommen aus (Kiwi Electronics B.V., 2017).</i>	<i>31</i>
<i>Abbildung 15 Unterpunkt "SPI"</i>	<i>32</i>
<i>Abbildung 16 Ausgabe der UIDs, bei zwei erfassten RFID-Tags</i>	<i>33</i>
<i>Abbildung 17 Beispielcode für den Wrapper</i>	<i>35</i>
<i>Abbildung 18 Auslesen von RFID-Tags mit dem Python-Wrapper</i>	<i>35</i>
<i>Abbildung 19 Zeigt den Schneideverlauf(rot) auf dem Erweiterungsmodul an. (reichelt elektronik GmbH & Co. KG, 2017)</i>	<i>36</i>
<i>Abbildung 20 Das bearbeitete Erweiterungsmodul</i>	<i>37</i>

<i>Abbildung 21 Eine 2-Pin-Stecker-Klemme in frontaler und seitlicher Ansicht ...</i>	<i>37</i>
<i>Abbildung 22 Eine mögliche Positionierung der 2-Pin Stecker-Klemme</i>	<i>38</i>
<i>Abbildung 23 Die Erweiterungsplatine mit positionierter 2-Pin-Stecker-Klemme</i>	<i>39</i>
<i>Abbildung 24 2-Pin-Stecker-Klemmen an die Kontakte des Erweiterungsmoduls mit einem Kupferlackdraht angelötet</i>	<i>40</i>
<i>Abbildung 25 Erweiterungsmodul mit selbsterstellter Antenne</i>	<i>41</i>
<i>Abbildung 26 Kombination von "000 000" am Mastermultiplexer</i>	<i>44</i>
<i>Abbildung 27 Pin-Belegung eines Multiplexer. Entnommen aus (Instruments, 2016).....</i>	<i>46</i>
<i>Abbildung 28 Schematische Darstellung eines Multiplexer. Entnommen aus (Instruments, 2016).....</i>	<i>47</i>
<i>Abbildung 29 Beispiel eines Multiplexer mit Sensor und Strom. Entnommen aus (Instruments, 2016).....</i>	<i>48</i>
<i>Abbildung 30 Schaltplan Editor Ansicht EAGLE</i>	<i>51</i>
<i>Abbildung 31 Layout Editor Ansicht EAGLE</i>	<i>52</i>
<i>Abbildung 32 Leiterplatte samt Sockel für Befestigung der Multiplexer</i>	<i>53</i>
<i>Abbildung 33 Zeigt ein Erweiterungsboard für einen Raspberry PI an.</i>	<i>54</i>
<i>Abbildung 34 Zeigt die Anschlüsse des PCBs an</i>	<i>55</i>
<i>Abbildung 35 Zeigt freie GPIOs an. Entnommen von (Matt, 2012).....</i>	<i>56</i>
<i>Abbildung 36 Zeigt das PCB verbunden mit dem Raspberry PI an.</i>	<i>57</i>
<i>Abbildung 37 Script zum Auslesen von Tags und umschalten von GPIOs.....</i>	<i>58</i>
<i>Abbildung 38 Entstehung einer Websocket Verbindung. Entnommen aus (Boden, 2013).....</i>	<i>61</i>
<i>Abbildung 39 Library Struktur</i>	<i>62</i>

Abbildung 40 PHP Server Funktion onMessage	63
Abbildung 41 Abschnitt des Python Scripts	64
Abbildung 42 Verbindung Python Script -> PHP Server	64
Abbildung 43 login Funktion der PHP User Klasse	66
Abbildung 44 logout und isLoggedIn Funktion der PHP User Klasse	67
Abbildung 45 Anmelde-Formular der Seite index.php	68
Abbildung 46 Ausführung der PHP User Klasse	68
Abbildung 47 Implementierung des PHP Clients	70
Abbildung 48 die Seite index.php	71
Abbildung 49 Die Seite index.php mit dem Login-Formular	72
Abbildung 50 Die Seite login.php	73
Abbildung 51 die Seite login2.php	74
Abbildung 52 Die Tabelle zur Implementierung des Schachbrettes.....	75
Abbildung 53 Die Seite login2.php mit dem logout-Button	76
Abbildung 54 Die Seite logout.php.....	76
Abbildung 55 Das Spiel.....	77
Abbildung 56 ER-Diagramm Chess4U.....	82
Abbildung 57 Tabelle Schachspiel	83
Abbildung 58 Tabelle Spieler	83
Abbildung 59 Tabelle Spielerfassung	84
Abbildung 60 Tabelle Schachfigur.....	84
Abbildung 61 Tabelle Spielerfassung	85
Abbildung 62 Rollup-Netidee.....	89
Abbildung 63 Plakat- Chess4U	89

Abbildung 64 Script, Werbevideo.....	91
Abbildung 65 Storyboard, Werbevideo	92
Abbildung 66 Stabilisierung, After Effects	94
Abbildung 67 Chess4U-Logo freigestellt	95
Abbildung 68 Gaußscher Weichzeichner (After Effects)	96
Abbildung 69 Keyframes (After Effects)	97
Abbildung 70 fertiges Intro bzw. Outro	97
Abbildung 71 Spielzug-Aufnahme	98
Abbildung 72 RFID- Schleife freigestellt	99
Abbildung 73 Figuren freigestellt	100
Abbildung 74 Figuren über RFID Schleifen.....	100
Abbildung 75 Effekte	101
Abbildung 76 RFID - Schleife über Figur.....	102
Abbildung 77 Spielzugfigur freigestellt	102

1 KONZEPT

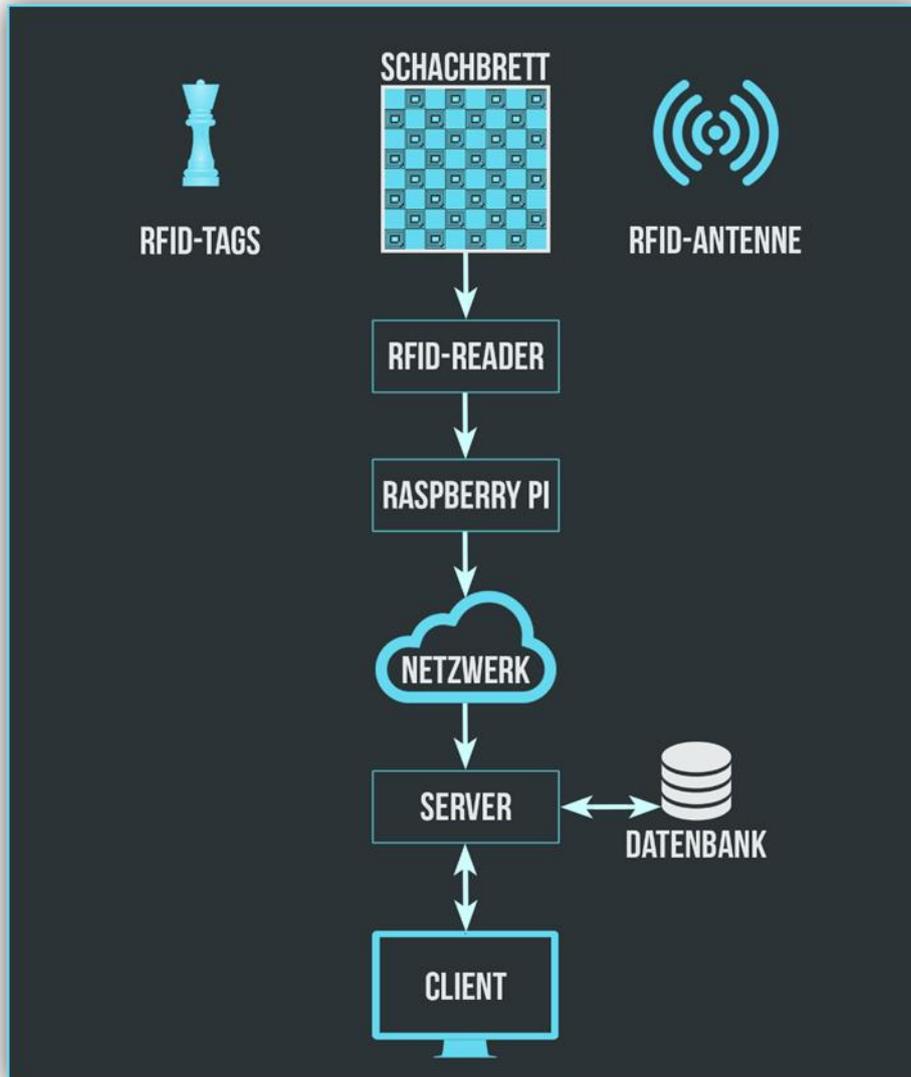


Abbildung 1 Konzept Chess4U

Anhand Abbildung 1 kann man das Konzept von Chess4U erkennen. Das Schachbrett scannt die RFID-Tags mit dem RFID Reader und den RFID Antennen und sendet die empfangenen Daten über ein Python Script auf dem Raspberry Pi über das Netzwerk an den Server. Der Server speichert die Daten in einer Datenbank und sendet sie an die Clients, auf welchen die Daten in einem schönen GUI ausgegeben werden.

1.1 RFID-TECHNOLOGIEN

RFID steht für “Radio Frequency Identification” und ist ein Verfahren zur kontaktlosen Übertragung und Identifizierung von Gegenständen (TAGnology RFID GmbH, 2016). Die Gegenstände können mit einer Seriennummer, in diesem Fall einem UID-Code (Unique Identification Code), eindeutig identifiziert werden. Zudem ist dieser UID-Code unveränderbar eingebrannt und unterscheidet sich somit von jedem anderen Tag (i-keys.de, 2016).

Der Kern der RFID-Technologie sind RFID-Chips, welche auch Transponder genannt werden und zum Senden und Empfangen von Daten dienen. Ein Transponder ist ein Kommunikationsgerät, welches das eingehende Signal, beziehungsweise die eingehenden Signale, aufnimmt, automatisch beantwortet oder auch weiterleitet (tec-trends.de, 2016). Transponder bestehen aus einer Antennenspule, einem Schaltkreis und einem Hauptspeicher, welcher je nach Chip-Typ nur einmal oder mehrmals beschrieben werden kann (Transponder, Wikipedia, 2017). Der Speicher eines beschreibbaren RFID-Chips beginnt bei wenigen Bits und erhöht sich bis in den Kilobyte-Bereich (RFID#Speicherkapazität, Wikipedia, 2017).

1.1.2 FUNKTION DER RFID-TECHNOLOGIE

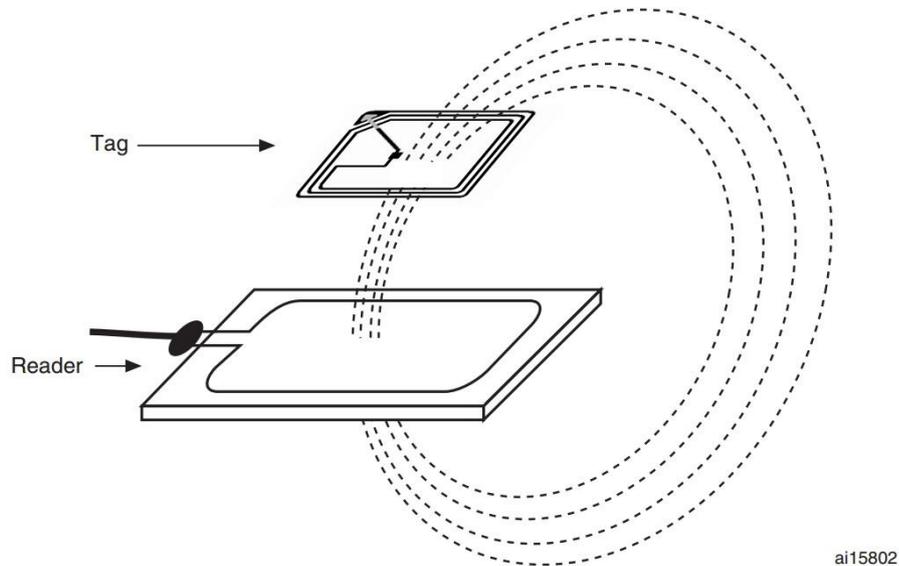


Abbildung 2 RFID-Tag, welcher am magnetischen Feld vom Reader gekoppelt ist. Entnommen aus (STMicroelectronics, 2016).

Die Technologie funktioniert wie folgt: Ein Signal, beziehungsweise schwache elektromagnetische Wellen, werden per Funk von einem Lesegerät (RFID-Reader) an den RFID-Chip zur Aktivierung gesendet und mittels weiteren Befehlen vom Reader werden die dazugehörigen Informationen vom RFID-Chip an den Reader wieder zurückgesendet. Es geht also um die Reichweite zwischen dem Reader und dem RFID-Chip, wenn der RFID-Chip in der Nähe der Antenne vom Reader ist, wird kontaktlos der Austausch von Informationen vom Speicher des Transponders ermöglicht. Des Weiteren besteht auch die Möglichkeit, dass Daten auf den Transponder geschrieben werden (TAGnology RFID GmbH, 2016).

Es wird zwischen zwei Arten von Transpondertechnik, *aktiven* und *passiven* Transpondern, unterschieden.

1.1.3 ARTEN VON TRANSPONDERN

Aktive Transponder haben eine integrierte Spannungsversorgung, einen Akku oder eine Batterie, die auch viel größere Lesereichweiten ermöglicht. Dadurch, dass aktive Transponder über eine integrierte Spannungsversorgung verfügen, sind diese nicht wartungsfrei und somit ist die Lebensdauer begrenzt. Dazu kommt noch, dass aktive Transponder teurer sind, da die Produktionskosten deutlich höher sind (TAGnology RFID GmbH, 2016).

Passive Transponder besitzen im Gegensatz zu aktiven Transpondern keine eigene Spannungsversorgung, somit beziehen passive Transponder ihre Energie direkt aus dem Energiefeld des Readers. Aufgrund der nicht integrierten Spannungsversorgung sind passive Transponder wartungsfrei, jedoch ist die Lesereichweite von vielen Faktoren, wie der Größe des Transponders, der Antenne und der Frequenz abhängig (TAGnology RFID GmbH, 2016).

1.1.4 FREQUENZBEREICHE

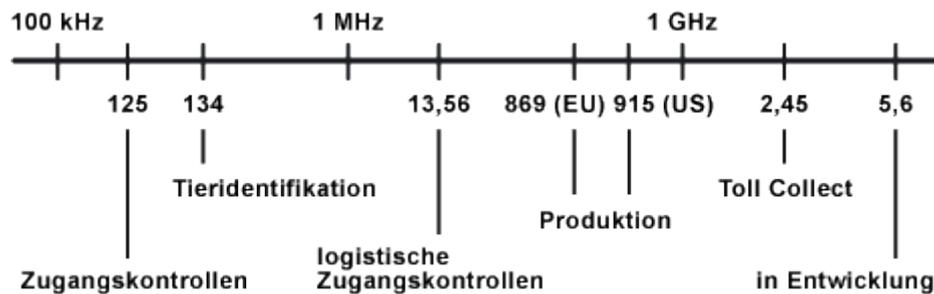


Abbildung 3 Frequenzbereiche der RFID-Technologie. Entnommen aus (Elektronik-Kompodium.de, 2017).

Es gibt keinen einheitlichen RFID-Standard, der bestimmte Frequenzbereiche definiert, jedoch haben sich verschiedene internationale als auch nationale Lösungen entwickelt. Diese Frequenzbänder sind von Land zu Land unterschiedlich. Darüber hinaus haben verschiedene Frequenzbereiche verschiedene Vorteile in Hinsicht auf Reichweite zum Reader, die Lesegeschwindigkeit, dem Preis und die Einsatzfähigkeit. Aus diesem Grund werden in verschiedenen Anwendungsbereichen unterschiedliche Transponder mit unterschiedlichen Frequenzen verwendet. Die RFID-Technologie wird derzeit in vier Hauptkategorien unterteilt. Jeder Frequenzbereich hat eigene Leistungseigenschaften, die in den unterschiedlichsten Gebieten zum Einsatz kommen, somit bestehen gewisse Vor- und Nachteile (Elektronik-Kompodium.de, 2017).

1.1.4.1 LF - LOW FREQUENCY

Dieser Standard arbeitet im Frequenzbereich von 119 kHz bis 135 kHz. Die *Vorteile* von diesem Frequenzbereich liegen bei der Zuverlässigkeit und der hohen Toleranz gegenüber folgenden Objekten: Textilien, Holz, Flüssigkeiten und Aluminium. Die *Nachteile* dieser Frequenz sind die geringen Lesereichweiten, der kleine Speicher im Tag und die langsame Datenübertragung. Dieser Frequenzbereich kommt beispielsweise bei folgenden Gebieten zum Einsatz: Zugangskontrolle, Tiererkennung und der Logistik (AGILLOX GmbH, 2017).

1.1.4.2 HF - HIGH FREQUENCY

Dieser Standard arbeitet mit 13,56 MHz. Die *Vorteile* von diesem Frequenzbereich liegen bei der hohen Speicherkapazität und der schnellen Datenübertragung. Die *Nachteile* dieser Frequenz liegen bei der mittleren Lesereichweite. Dieser Frequenzbereich kommt beispielsweise bei Lieferdiensten, Büchereien, Qualitätssicherungen und bei Gepäckkontrollen zum Einsatz (AGILLOX GmbH, 2017).



Abbildung 4 Ein geöffneter 13,56 MHz Tag.

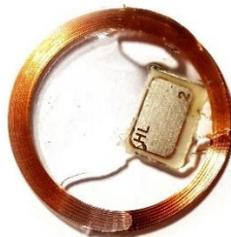


Abbildung 5 Ein Transponder eines Tags.

1.1.4.3 UHF - ULTRA HIGH FREQUENCY

Dieser Standard erstreckt sich in Europa von 865 - 870 MHz, in den USA arbeitet diese Frequenz mit 915 MHz. Die *Vorteile* von diesem Frequenzbereich liegen bei den hohen bis sehr hohen Reichweiten, der hohen Speicherkapazität und der hohen Übertragungsgeschwindigkeit. Die *Nachteile* dieser Frequenz liegen bei der mittleren Lesereichweite. Dieser Frequenzbereich kommt beispielsweise bei folgenden Gebieten zum Einsatz: Warenlogistik, Automation, Produktion (AGILLOX GmbH, 2017).

1.1.4.4 SUHF - ULTRA HIGH FREQUENCY

Dieser Standard arbeitet mit 2,4 GHz und kommt hauptsächlich in den USA zum Einsatz. Die *Vorteile* dieser Frequenz liegen bei der extrem hohen Reichweite, der hohen Speicherkapazität und der hohen Übertragungsgeschwindigkeit. Der *Nachteil* dieser Frequenz ist, dass es kaum praxistaugliche Lösungen in Europa dafür gibt. Jedoch kommt dieser Frequenzbereich in den USA in folgenden Gebieten zum Einsatz: Zufahrtskontrollen, Mautsystemen und Container-Identifikationen (AGILLOX GmbH, 2017).

1.1.5 NFC

NFC steht für Near-Field-Communication. Near-Field-Communication (auf Deutsch: Nahfeldkopplung) basiert auf der RFID-Technologie (Radio-Frequency Identification). Die Besonderheit von NFC ist, dass sich diese Technologie durch ein spezielles Kopplungsverfahren auszeichnet. Dieses Verfahren wurde auch im Standard ISO 14443, 18092 und 21481 genormt. Darüber hinaus wurde neben dem speziellen Kopplungsverfahren der Frequenzbereich festgelegt.

Dieser liegt im Bereich 135 kHz und 13,56 MHz. NFC wird meistens nur für sichere Kommunikationen verwendet. Deshalb ist auch die Reichweite von NFC auf eine kurze Distanz genormt worden, nämlich max. 10 cm. Man kann also sagen, dass NFC eine genormte Spezialisierung der RFID-Technik ist, die für kurze Distanzen und eine sichere Datenübertragung ausgelegt wurde (Fakir, 2017).

2 HARDWAREPLATTFORM

Für dieses Projekt ist ein RFID-Reader(Lesegerät) notwendig, damit die Spielfiguren(RFID-Tags) erfasst werden können.



Abbildung 6 Die Vorderseite eines geöffneten RFID-USB-Readers.

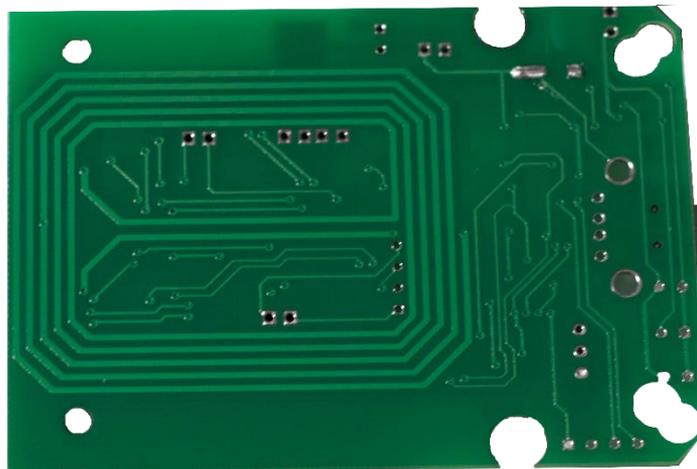


Abbildung 7 Die Rückseite inkl. Antenne eines geöffneten RFID-USB-Readers.

Erste Tests wurden mit einem USB-RFID-Reader durchgeführt.

2.1 RASPBERRY PI

Der Raspberry Pi ist ein Einplatinencomputer, der sowohl für alltägliche Aufgaben als auch für Projekte mit verschiedenen Einsatzgebieten geeignet ist. Ein Vorteil ist, dass ein Raspberry Pi wenig Strom verbraucht (Söldner, 2013). Für dieses Projekt ist der Raspberry Pi am ehesten geeignet, **da dieser über eine externe Spannungsversorgung verfügt und verschiedene Pins zur Verfügung stellt, die nachträglich für die Spannungsanlegung benötigt werden.** Darüber hinaus verfügt der Raspberry über eine Netzwerkschnittstelle.

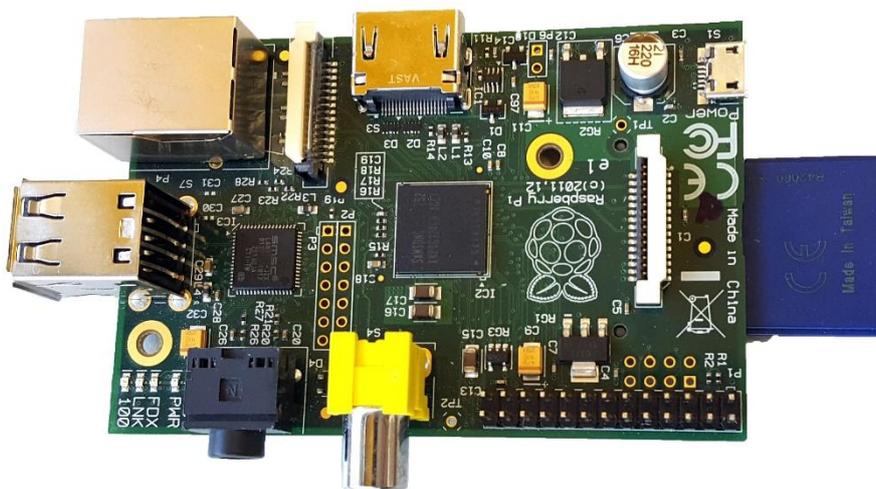


Abbildung 8 Einplatinencomputer „Raspberry Pi“ inkl. SD-Karte

2.1.1 RASPBERRY SOFTWARE

Als nächstes benötigt der Raspberry PI ein Image, welches auf die SD-Karte installiert werden muss, damit der Raspberry PI in Betrieb genommen werden kann. Das Image kann auf der Raspberry PI Webseite³ heruntergeladen und installiert werden. Nachdem das Image heruntergeladen wurde, benötigt man ein weiteres Programm, welches das Image auf ein Laufwerk (hier eine SD-Karte) installiert. Bei diesem Projekt wurde die kostenlose Software „Win32 Disk Imager“ verwendet. Die Software gibt es kostenlos auf SourceForge⁴. Nachdem man die Software installiert hat, startet man diese und wählt das zuvor heruntergeladene Image aus.

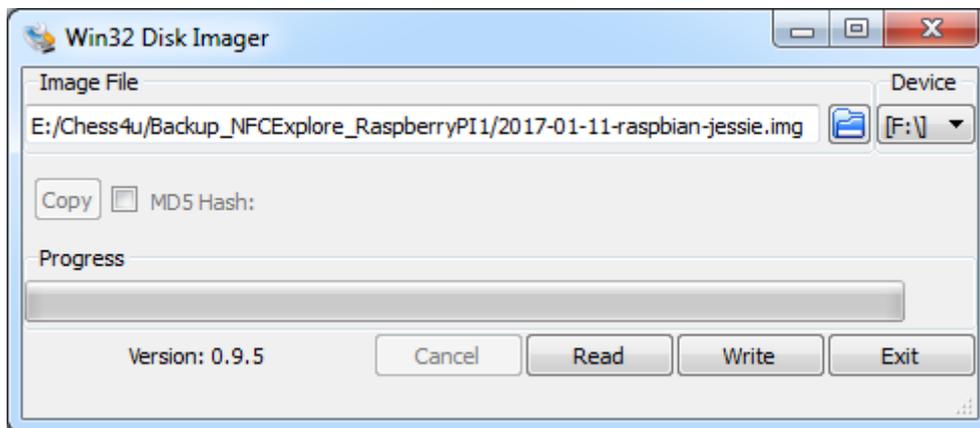


Abbildung 9 Programm "Win32 Disk Imager" mit dem ausgewählten Image

Unter Device wählt man den Laufwerksbuchstaben des Ziellaufwerks aus (in diesem Fall ist der Laufwerksbuchstabe „F:“ die SD-Karte). **Achtung:** Man muss sich versichern, dass das richtige Laufwerk ausgewählt ist, denn sonst wird ein anderes Laufwerk mit dem Image überschrieben, und es kommt zu Datenverlust. Danach muss man auf „Write“ klicken und das Programm beginnt den Schreibvorgang auf die SD-Karte. Dieser Vorgang kann einige Minuten in Anspruch nehmen.

³ www.raspberrypi.org zul. geladen am 02.01.2017

⁴ www.sourceforge.net/projects/win32diskimager zul. geladen am 03.01.2017

2.1.1 RASPBERRY GRUNDKONFIGURATION

Nachdem der Raspberry Pi die nötige Software zum Hochfahren geflasht bekommen hat, ist der nächste Schritt nun die Konfiguration. Es muss dafür gesorgt werden, dass der Raspberry Pi die SD-Karte erhält und mit Strom versorgt wird. Als nächstes muss der Raspberry mit einem Monitor/Fernseher (oder einem Gerät, welches über einen HDMI-Eingang verfügt), und einer Tastatur/Maus verbunden werden, damit man SSH (Secure Shell) aktivieren kann, um sich später nur über das Netzwerk verbinden zu können.

Nachdem der Raspberry Pi mit einem Monitor verbunden und hochgefahren ist, muss man als nächstes auf der graphischen Benutzeroberfläche die Konsole starten. Man muss den Befehl `sudo raspi-config` in die Konsole eintippen, um das Konfigurationstool des Raspberry zu starten. Dieser Befehl erfordert erhöhte Rechte, deshalb muss er mit `sudo` (Superuser root) ausgeführt werden. Nachdem man den Befehl erfolgreich ausgeführt hat, muss man sich im Menü bis zum vorletzten Punkt „Advanced Options“ durchklicken.

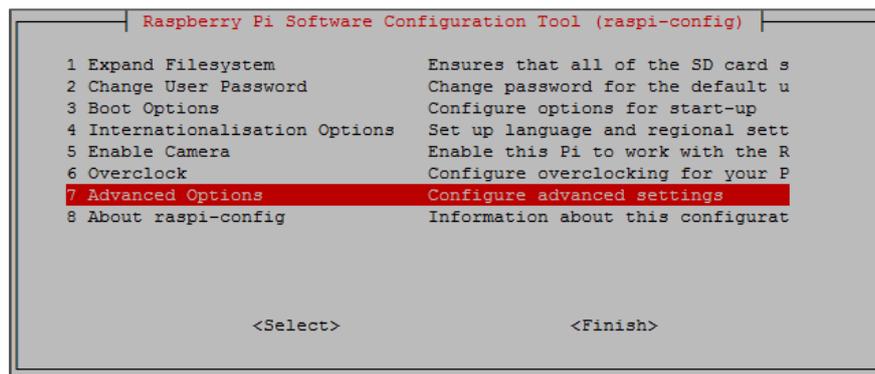


Abbildung 10 Menüpunkt "Advanced Options"

Danach wählt man diesen mit der Eingabe-Taste aus und gelangt in die erweiterten Optionen.

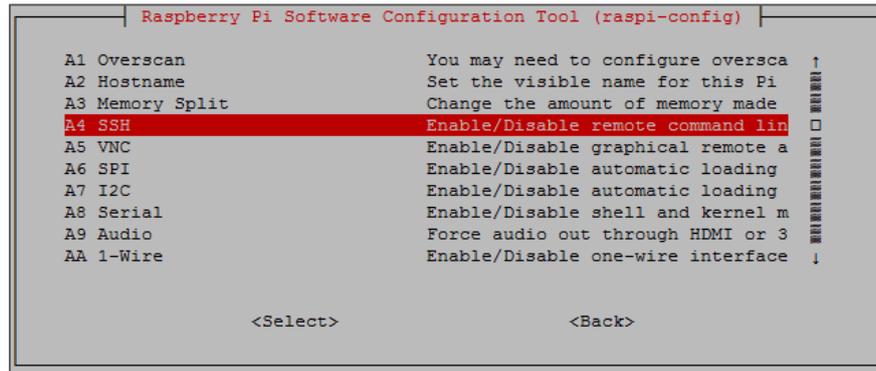


Abbildung 11 Menüpunkt „SSH“

Hier wählt man den 4. Menüpunkt „SSH“ aus und aktiviert schlussendlich den SSH-Server.

Wenn man den SSH-Server aktiviert hat, schließt man den Raspberry Pi per Netzwerkkabel am Router an, ermittelt die IP-Adresse und verbindet sich mit einem SSH Client. Man muss nun einen SSH Client installieren. In diesem Fall fiel die Entscheidung auf den kostenlosen SSH Client namens PuTTY. Diesen Client kann man auf der Herstellerwebseite⁵ kostenlos herunterladen. Nachdem man den PuTTY Client erfolgreich heruntergeladen und installiert hat, muss man nun die IP-Adresse des Raspberry Pi ermitteln. Der Rechner muss im selben Subnetz sein, um sich mit dem Raspberry Pi verbinden zu können. Man muss nun das Standardgateway (default gateway) des Routers in einen Browser eingeben, um in das Router-Menü zu gelangen. Dann kann man die IP-Adresse des Raspberry PI's herausfinden. Dieser Vorgang kann von Hersteller zu Hersteller beziehungsweise von der Benutzeroberfläche des Router-Menüs abweichen.

⁵ www.putty.org zul. geladen am 15.03.2017



Abbildung 12 Informationen eines Routers über den Raspberry an.

In diesem Fall hat der Raspberry Pi vom Router eine interne IP mit der Adresse 10.0.0.12 bekommen.

Als nächstes muss man diese IP-Adresse in den PuTTY-Client eingeben und eine Sitzung(Session) mit dem Button „Open“ starten.

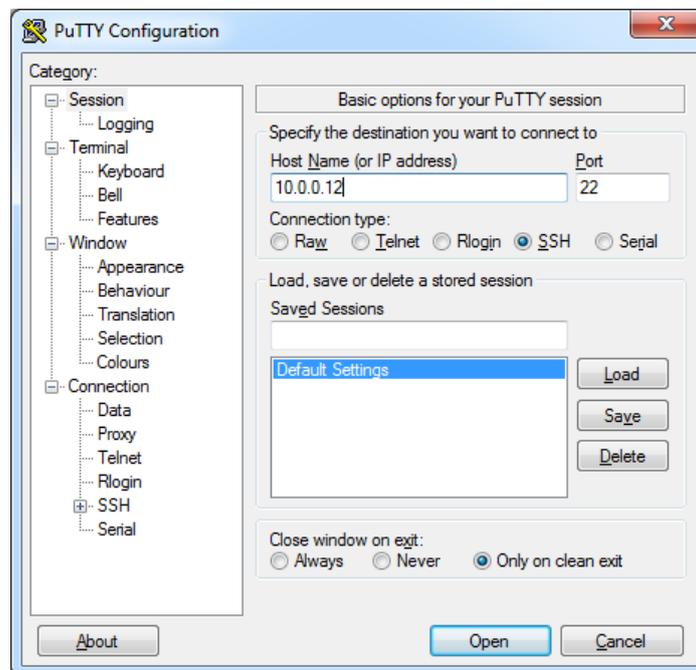


Abbildung 13 Die eingegebene IP-Adresse im PuTTY-Client

Nun öffnet sich eine Konsole (Terminal), welche auf die Eingabe des Benutzernamens wartet. Beim Raspberry Pi ist der Standard-Benutzername *pi* und das Standard-Passwort ist *raspberrypi*. Nun ist man erfolgreich eingeloggt.

Als erstes sollte man den Raspberry Pi, beziehungsweise alle Paketquellen, auf den aktuellsten Stand mit einem Update bringen. Dies geschieht mit dem Befehl `sudo apt-get update`. Dieser Vorgang kann je nach Internetverbindung und Geschwindigkeit der SD-Karte einige Minuten in Anspruch nehmen. Nachdem die Paketquellen auf den aktuellsten Stand gebracht wurden, muss man die installierten Pakete auch upgraden. Dies geschieht mit dem Befehl `sudo apt-get upgrade`. Auch dieser Vorgang kann einige Minuten in Anspruch nehmen. Wenn dieser Vorgang abgeschlossen ist, sind alle Grundkonfigurationsschritte am Raspberry Pi erfolgreich durchgeführt worden.

2.1.2 EXPLORE NFC ERWEITERUNGSMODUL(READER)

Für den verwendeten Raspberry Pi wird ein Reader, beziehungsweise eine NFC-kompatible Erweiterungsplatine (auch Erweiterungsmodul genannt) benötigt, um später die RFID bzw. NFC-Tags auslesen zu können. Für dieses Projekt wurde ein Modul von dem Hersteller⁶ erworben. Dieses Erweiterungsboard verfügt bereits über eine integrierte Hochleistungsantenne. Es wurde speziell diese Erweiterungsplatine verwendet, da hier die Antenne nicht über mehrere Bauteile verläuft. Die Antenne ist hier separat am Rand positioniert, was eine Manipulation beziehungsweise eine Verbindung mit selbstgewickelten Antennen leicht ermöglicht. Um das Erweiterungsboard mit dem Raspberry Pi zu verbinden, muss zuerst dafür gesorgt werden, dass der Raspberry Pi nicht in Betrieb ist (mit dem Befehl `sudo shutdown -h now` kann der Raspberry Pi heruntergefahren werden). Danach kann das Erweiterungsmodul auf den Raspberry Pi gesteckt werden. Abbildung 14 zeigt den Raspberry Pi inkl. dem NFC-Erweiterungsmodul an.

Der Reader muss unbedingt so angeschlossen werden, dass dieser über dem Raspberry Pi liegt. *Achtung:* Ein fehlerhaftes Verbinden des Readers mit dem Raspberry Pi kann diverse Schäden verursachen.

⁶ www.element14.com zul. geladen am 16.03.2017

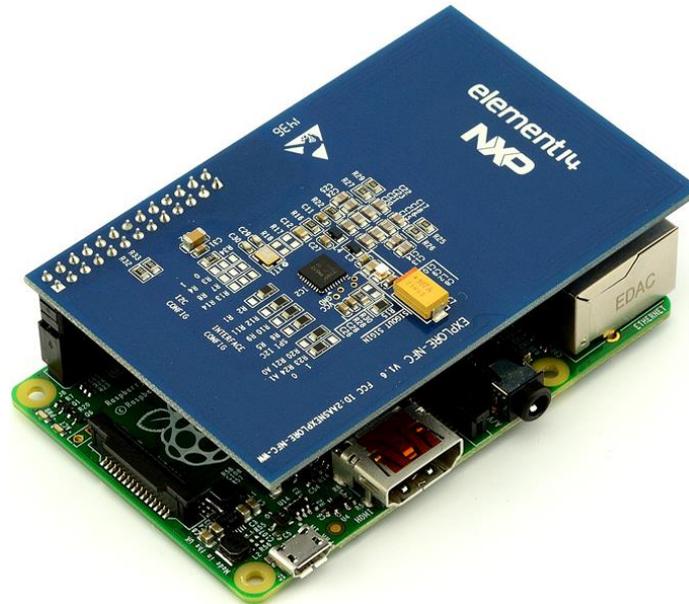


Abbildung 14 Ein Raspberry Pi inkl. einem Explore-NFC Erweiterungsmodul.
Entnommen aus (Kiwi Electronics B.V., 2017).

2.1.2.1 KONFIGURATION DER NFC-ERWEITERUNGSPLATINE

Nachdem die Erweiterungsplatine erfolgreich und fehlerfrei (siehe Abbildung 14) mit dem Raspberry Pi verbunden worden ist, müssen sämtliche Einstellungen konfiguriert, werden. Dafür muss der Raspberry Pi zuerst hochgefahren werden und man muss versichert sein, dass nachfolgende Einstellungen und Konfigurationen mit höheren Rechten ausgeführt werden (mit dem Befehl *sudo*). Als erstes wird ein Update und auch ein Upgrade der Paketquellen, beziehungsweise der Pakete, durchgeführt werden. Dies geschieht zuerst mit dem Befehl *sudo apt-get update* danach muss man den Befehl *sudo apt-get upgrade* eingeben, um die Pakete auch upzugraden. Man muss einige Operationen zusätzlich mit einem Y für „yes“ bestätigen, dies ist nur eine Warnung über zusätzlich benötigten Speicherplatz einer Operation. Nachdem diese Grundschrirte erledigt wurden, kann nun mit der eigentlichen Konfiguration begonnen werden.

Zuerst muss das Serial Peripheral Interface(SPI) aktiviert werden. Dies ist ein synchrones Datenbussystem, mit welchem digitale Schaltungen nach dem Master-Slave-Prinzip miteinander verbunden werden können (raspberrypi.tips, 2014).

Nun muss man den Befehl `sudo raspi-config` in die Konsole eintippen, um das Konfigurationstool des Raspberry Pi's zu starten. Nachdem man den Befehl erfolgreich ausgeführt hat, muss man im Menü den vorletzten Punkt „Advanced Options“ mit der Eingabetaste auswählen. Danach wählt man den Unterpunkt „SPI“ aus und aktiviert diesen, indem man mit „Yes“ bestätigt.

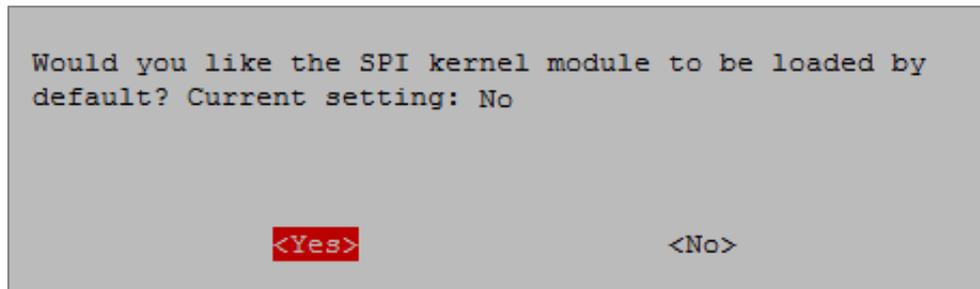


Abbildung 15 Unterpunkt "SPI"

Nachdem das SPI aktiviert wurde, muss man den Raspberry Pi neustarten, damit die Änderungen aktiviert werden. Dies kann man mit dem Befehl `sudo reboot` machen. Nun muss man sich die Software für den Reader herunterladen. Diese findet man kostenlos auf der NXP Webseite⁷. Wenn man die Software heruntergeladen hat, muss man diese auf einen USB-Stick kopieren, um diese später auf dem Raspberry Pi entpacken zu können. Nachdem man das Paket erfolgreich entpackt hat, muss man nun in das zuvor entpackte Verzeichnis wechseln (`cd $Pfad_zum_Verzeichnis`). Wenn man sich im Verzeichnis befindet, muss man die Dateien auch installieren, dies macht man mit folgenden Befehlen:

```
sudo dpkg -i libnearDAL0_<version>_armhf.deb
```

```
nearDexplorenc_<version>_armhf.deb
```

Der Tag „<version>“ muss mit der eigenen Versionsnummer ersetzt werden. Wenn dieser Schritt erfolgreich ausgeführt wurde, kann man nun mit dem Befehl `explorenc-basic -k` den Reader auf Funktionalität testen. Nun wartet das Programm auf einen Tag, der über die Antenne platziert, beziehungsweise gehalten, werden muss. Der Reader, beziehungsweise die Software, funktioniert erfolgreich, wenn die ID des Tags nun in einer Dauerschleife ausgegeben wird (NXP Semiconductors N.V., 2016).

⁷ www.nxp.com

```
001489014280 09.02.2017-00:04:40 $B827EB506E12$ UID: @0EC7515A@
UID: 0E C7 51 5A
UID: 04 AA 61 5A 6D 3B 80
001489014288 09.02.2017-00:04:48 $B827EB506E12$ UID: @04AA615A6D3B80@
UID: 04 AA 61 5A 6D 3B 80
```

Abbildung 16 Ausgabe der UIDs, bei zwei erfassten RFID-Tags

2.1.2.2 NXPPY PYTHON-WRAPPER

Nun wird ein Python-Wrapper für die Erweiterungsplatine benötigt, damit man später eigene Befehle (umschalten der GPIOs) einbauen kann. Für die Schnittstelle mit dem Reader wird die Bibliothek (library) NXPPY verwendet. Dies ist ein schlichter Python Wrapper, welcher den Code der öffentlichen NXP Bibliothek heranzieht. Die Grundvoraussetzung für diesen Wrapper ist, dass das SPI aktiviert ist, dies ist bereits geschehen, da der Reader sonst nicht ordnungsgemäß funktionieren würde (Vitale(svvitale), 2017). Nun muss CMake heruntergeladen und installiert werden. CMake ist ein Programmierwerkzeug, welches insbesondere als Build-System genutzt wird (hrvojeheavy, 2013). Um das neueste CMake zu installieren, muss man zuerst einen eigenen Ordner dafür erstellen. Dazu wird folgender Befehl benötigt: `mkdir cmake`. Danach muss in den Ordner `cmake` gewechselt werden, das geschieht mit dem Befehl `cd cmake`. Als nächstes müssen die benötigten Dateien, beziehungsweise der komprimierte Ordner, heruntergeladen und extrahiert werden. Dazu werden folgende Befehle benötigt:

```
wget https://cmake.org/files/v3.4/cmake-3.4.1.tar.gz
```

```
tar -xvzf cmake-3.4.1.tar.gz
```

Danach muss man in das zuvor extrahierte Verzeichnis wechseln und schlussendlich CMake kompilieren und installieren (jeden Befehl einzeln eingeben, ohne „;“):

```
cd cmake-3.4.1/; sudo ./bootstrap; sudo make; sudo make install
```

Dieser Vorgang kann einige Minuten in Anspruch nehmen. (MakerDev, 2016)

Nun muss ein Update und auch Upgrade der Paketquellen beziehungsweise Pakete durchgeführt werden. Dies geschieht zuerst mit dem Befehl *sudo apt-get update*, danach muss man den Befehl *sudo apt-get upgrade* eingeben, um die Pakete auch upzugraden. Als nächstes kann mit der eigentlichen Installation des Wrappers fortgefahren werden. Die nachfolgenden Schritte richten sich nach der Anleitung⁸ des Python-Wrappers. Nun müssen einige Pakete bzw. Dependencies installiert werden. Dies geschieht mit dem Befehl:

```
sudo apt-get install build-essential python3-dev unzip wget
```

Nachdem die benötigten Pakete heruntergeladen und installiert wurden, muss schlussendlich mittels dem Python Package Index(PyPI) die NXP Bibliothek heruntergeladen werden. Dies geschieht mit dem Befehl *pip install nxppy*. Da die komplette Bibliothek heruntergeladen wird, kann dies einige Zeit in Anspruch nehmen. (Vitale(svvitale), 2017) Nachdem dieser Schritt abgeschlossen wurde, ist im Homeverzeichnis ein Ordner namens „nxppy-master“ erstellt worden. Mit dem Befehl *cd nxppy-master/* wechselt man in dieses Verzeichnis. Dort kann man nun ein Python-Script zum Testen des Python-Wrappers einfügen. Mit *sudo nano test.py* kann man sich eine Python-Datei erzeugen. Dort kann man nun den Beispielpcode des Autors einfügen, um die Funktionalität des Wrappers zu testen.

⁸ www.github.com/svvitale/nxppy

```
import nxppy
import time

mifare = nxppy.Mifare()

# Print card UIDs as they are detected
while True:
    try:
        uid = mifare.select()
        print(uid)
    except nxppy.SelectError:
        # SelectError is raised if no card is in the field.
        pass

    time.sleep(1)
```

Abbildung 17 Beispielcode für den Wrapper

Anhand der Abbildung 17 werden zuerst zwei Module importiert, nämlich das nxppy- und das time-Modul. Das nxppy-Modul stellt die Verbindung zur Erweiterungsplatine her. Als nächstes wurde eine Variable erstellt, die von der nxppy-Bibliothek die Funktion *Mifare()* aufruft. Danach werden in einer while-Schleife mit *mifare.select()* die RFID-Tags ausgelesen und mit *print(uid)* wird die jeweilige UID eines Tags in der Console ausgegeben. Falls ein Lesefehler auftritt, wird die entsprechende Fehlermeldung dazu ausgegeben. Das time-Modul wird hier für eine Verzögerung zwischen dem Auslesen der RFID-Tags benötigt. In diesem Fall wurde diese mit *time.sleep(1)* auf eine Sekunde festgelegt. Das heißt, dass pro Sekunde einmal gelesen wird. Schlussendlich kann das Script mit dem Befehl *sudo python test.py* ausgeführt werden. Mit der Tastenkombination *STRG(CTRL)+C* kann das Script wieder beendet werden.

```
pi@NFCraspi ~/nxppy-release-1.4.3 $ sudo python test.py
0EC7515A
0EC7515A
0EC7515A
0EC7515A
0EC7515A
0EC7515A
0EC7515A
```

Abbildung 18 Auslesen von RFID-Tags mit dem Python-Wrapper

2.1.2.3 ERWEITERUNGSMODUL BEARBEITEN

Es wird eine Antenne erstellt bzw. gewickelt, damit der Reader später mit der Leiterplatte verbunden werden kann. Schlussendlich sollen die Antennen durch die Multiplexer abwechselnd mit dem Reader verbunden werden. Um eine eigene Antenne später mit dem Erweiterungsmodul verbinden zu können, muss das Erweiterungsmodul bearbeitet werden. Die integrierte Antenne des Readers muss entfernt (siehe Abbildung 19) werden. Dies kann mit einer handelsüblichen Handsäge geschehen, man muss ungefähr entlang der roten Markierung sägen. Wichtig dabei ist, dass man nicht zu Nah an den anliegenden Bauteilen sägt, damit diese nicht beschädigt werden (die integrierte Antenne wird nicht mehr benötigt, also wirken sich Schäden auf dieser nicht negativ aus).

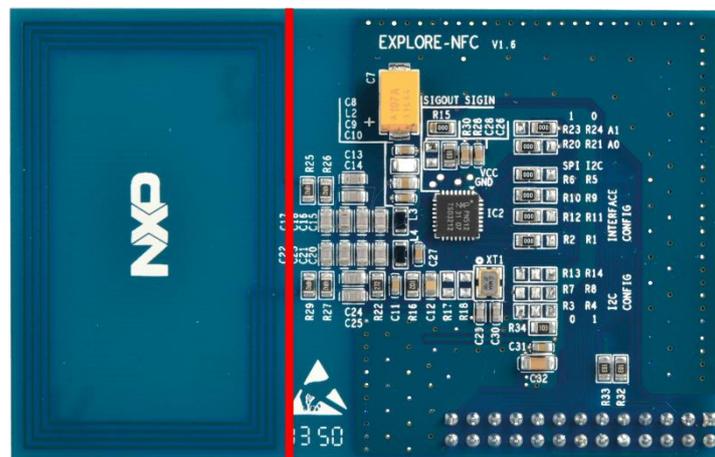


Abbildung 19 Zeigt den Schneideverlauf(rot) auf dem Erweiterungsmodul an.
(reichelt elektronik GmbH & Co. KG, 2017)

Nachdem die Antenne erfolgreich entfernt wurde, muss das Erweiterungsmodul ungefähr wie in der nachfolgenden Abbildung (Abbildung 20) aussehen. In der nachfolgenden Abbildung sind auch die Kontakte für eine Antenne gekennzeichnet worden (roter Kreis).

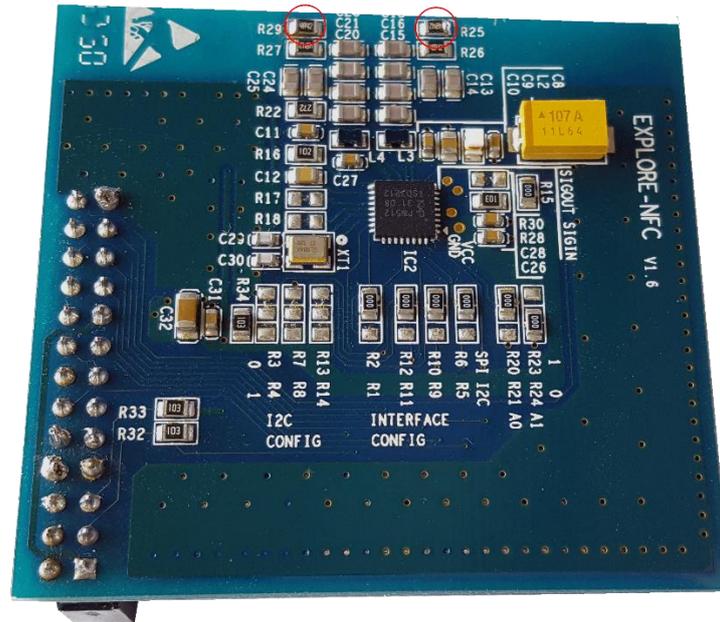


Abbildung 20 Das bearbeitete Erweiterungsmodul

Nun besteht die Möglichkeit, dass man eine selbsterstellte Antenne direkt an die gekennzeichneten Kontakte anlötet. Jedoch sollte hierbei auf Flexibilität geachtet werden, deshalb kommt hier eine 2-Pin-Stecker-Klemme, oder auch 2-Pin Schraubklemmenblock genannt (engl. 2 pin/way pitch), zum Einsatz.



Abbildung 21 Eine 2-Pin-Stecker-Klemme in frontaler und seitlicher Ansicht

Damit die 2-Pin-Stecker-Klemme auf das Erweiterungsmodul platziert werden kann, müssen für die entsprechenden Stifte, beziehungsweise Pins, Löcher gebohrt werden. Diese Löcher sollten am besten auf einer Position gebohrt werden, auf der keine anderen Bauteile des Erweiterungsmoduls beschädigt werden können.

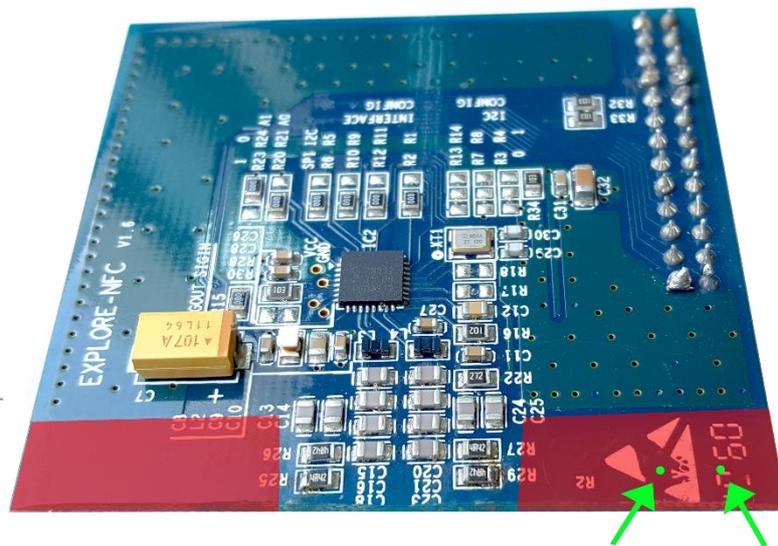


Abbildung 22 Eine mögliche Positionierung der 2-Pin Stecker-Klemme

Die obige Abbildung zeigt Positionsmöglichkeiten der 2-Pin-Stecker-Klemme an (roter Farbbereich). Die möglichen Bohrstellen werden mit zwei grünen Punkten und Pfeilen verdeutlicht. Die Bohrstelle kann auch im linken Bereich (ebenfalls roter Bereich) gesetzt werden.

Anschließend muss vorsichtig mit einem Bohrer, mit einem Durchmesser von ca. 1 mm (hängt auch von den Stiften bzw. Pins der Stecker-Klemme ab), durch die Erweiterungsplatine an den markierten Stellen gebohrt werden.

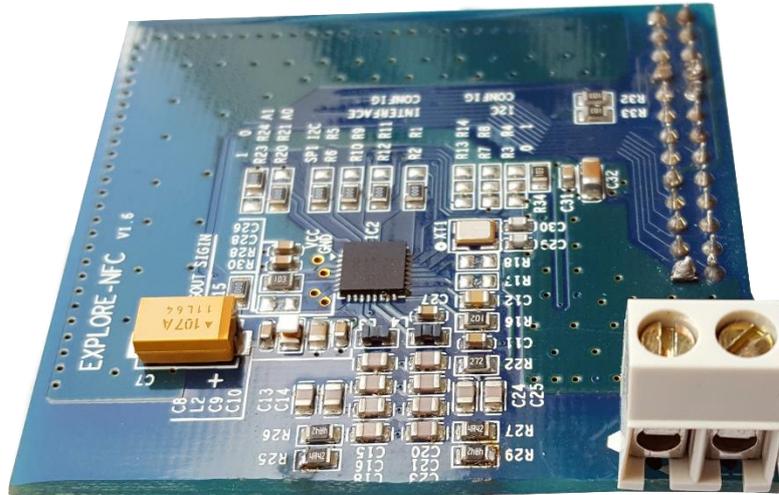


Abbildung 23 Die Erweiterungsplatine mit positionierter 2-Pin-Stecker-Klemme

Nun wurde die 2-Pin-Stecker-Klemme erfolgreich positioniert. Jetzt müssen die Pins dieser Klemme mit den Kontakten des Erweiterungsmoduls verbunden werden. Dafür werden Kupferlackdrähte (CuL) verwendet. Ein Kupferlackdraht ist ein Kupferdraht, der bei der Herstellung mit einer elektrisch isolierenden Schicht überzogen wurde (Kupferlackdraht, Wikipedia, 2017). Dadurch werden unbeabsichtigte Kurzschlüsse auf dem Erweiterungsmodul vermieden. Um die Pins der 2-Pin-Stecker-Klemme mit den Kontakten des Erweiterungsmoduls zu verbinden, müssen Kupferleitdrähte an die Pins und an die Kontakte angelötet werden. Zuerst sollte man jedoch die Enden des Kupferlackdrahtes erhitzen um die Lackschutzschicht zu lösen, damit das Lötzinn besser darauf haftet. **Achtung:** Nur der Teil, der angelötet werden soll, sollte erhitzt werden, da sonst die Gefahr eines Kurzschlusses besteht, da die Lackschutzschicht nicht mehr vorhanden wäre. Den Kupferlackdraht gibt es in verschiedenen Durchmessergrößen, in diesem Fall wurde ein Kupferlackdraht mit einem Drahtdurchmesser von 0,3mm verwendet.

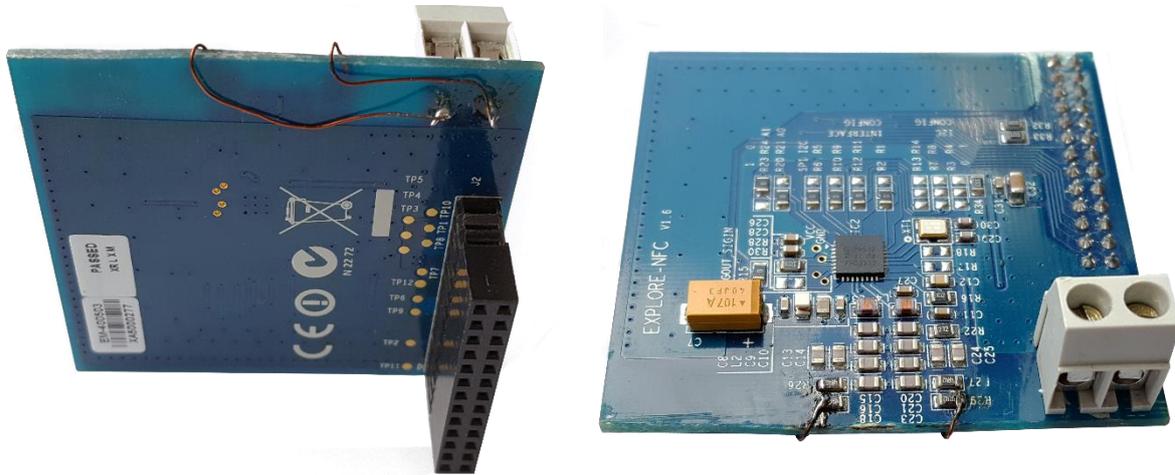


Abbildung 24 2-Pin-Stecker-Klemmen an die Kontakte des Erweiterungsmoduls mit einem Kupferlackdraht angelötet

2.1.2.3.1 ANTENNE(N) ERSTELLEN

Nun wird (vorerst) eine Antenne erstellt, um den Reader nach dem Umbau beziehungsweise nach der Bearbeitung, auf Funktionalität zu testen. Für die Antenne wird ebenfalls ein Kupferlackdraht verwendet. Die Antenne sollte ungefähr einen Durchmesser von 2 cm haben. Damit die Antenne eine passende Form annimmt, sollte diese um einen runden Gegenstand gewickelt werden. Die Antenne sollte ungefähr 5-7 Mal um einen runden Gegenstand gewickelt werden. Nachdem die Antenne eine runde Form angenommen hat, müssen die Enden so zurechtgebogen werden, damit diese in die 2-Pin-Stecker-Klemmen passen. Auch in diesem Fall müssen die Enden mit einem Feuerzeug erhitzt werden, damit der Draht an den Enden leitfähig wird. Somit kann ein Kontakt zwischen den 2-Pin-Stecker-Klemmen hergestellt werden. Die nachfolgende Abbildung (Abbildung 25) zeigt das Endergebnis an dem Erweiterungsmodul an.

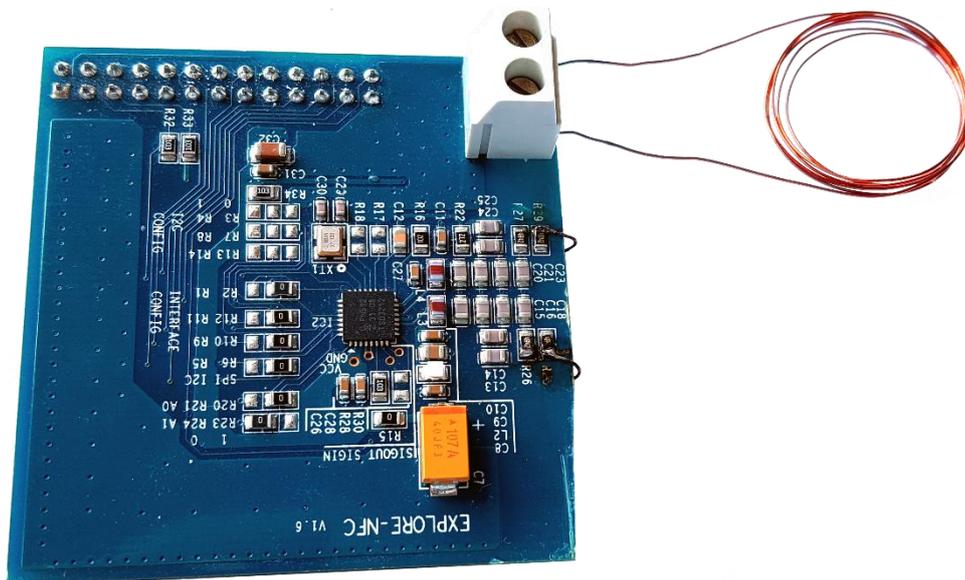


Abbildung 25 Erweiterungsmodul mit selbsterstellter Antenne

Wenn das Erweiterungsmodul weiterhin funktioniert (Tags werden gelesen), ist das ein sehr gutes Zeichen. Falls der Reader nicht mehr funktioniert, dann muss geprüft werden, ob die Kontakte mit den Pins der Stecker-Klemmen sicher angelötet wurden. Es kann auch an der Antenne liegen, mögliche Fehlerquelle wäre, dass die Lackschicht von dem Draht nicht komplett bzw. nicht genügend entfernt wurde. Ein Fehler könnte auch aufgrund der Windungen der Antenne entstehen, da die Form nicht rund genug ist, oder die Umwicklungen nicht nah genug beisammen sind.

Nun kann das Erweiterungsmodul wieder mit dem Raspberry PI verbunden werden.

2.2 MULTIPLEXER

2.2.1 EINLEITUNG MULTIPLEXER

In den folgenden Abschnitten wird beschrieben welche Aufgaben ein sogenannter Multiplexer im diesem Projekt hat. Zuerst wird auf den allgemeinen Verwendungszweck eines Multiplexer eingegangen, darauf folgt eine spezifisch auf die Diplomarbeit passende Erklärung.

2.2.2 ALLGEMEINE ERKLÄRUNG

Mit Hilfe eines Multiplexer ist es möglich, für ein Eingangssignal einen bestimmten Ausgang auswählen zu können. Dies funktioniert nach dem Prinzip einer Selektionsschaltung. Beispiele dafür werden weiter unten in der spezifischen Erklärung beschrieben. Im Unterschied zu Relais funktionieren Multiplexer nicht mechanisch, sondern elektronisch, dabei wird mit sogenannten Halbleiterschaltungen gearbeitet (Multiplexer, Wikipedia, 2016).

„Neben mehreren Eingängen und einem Ausgang verfügt ein Multiplexer über ein oder mehrere Steuersignale, über die festgelegt wird, welcher Eingang ausgewählt wird. Es wird derjenige Eingang zum Ausgang durchgeschaltet, der die Kennung hat, die in Form einer Dual Zahl als Steuersignal anliegt.“ (unbekannt) So hat z.B. ein parallel angesteuerter Multiplexer mit dem Bezeichnungsschlüssel n-MUX n Steuersignale, $2n$ Eingänge und einen Ausgang. Eingänge werden meist mit den Zahlen 0 bis $2n-1$ durchnummeriert. Im Folgenden soll auf die konkrete Anwendung des Multiplexer im Chess4U Projekt eingegangen werden (Multiplexer, Wikipedia, 2016) .

2.2.3 SPEZIFISCHE ERKLÄRUNG

Es wird auch bei der Diplomarbeit Chess4U mit insgesamt neun Multiplexern gearbeitet. Diese ermöglichen das Umschalten zwischen den einzelnen Antennen, von denen es insgesamt 64 Stück unter dem gesamten Spielfeld gibt, da sich ebenso viele Spielfelder auf einem Schachbrett befinden. Dabei spielt der sogenannte Mastermultiplexer die wichtigste Rolle, der in der Lage ist zwischen den anderen acht Multiplexer umzuschalten.

Dies ist essentiell, da jeder einzelne Multiplexer in der Lage ist acht Antennen anzusteuern. Wenn beispielsweise eine Bit Kombination der MSB (Most significant Bits), 000 auf den GPIO Pins ausgewählt ist, wird der erste Multiplexer vom Mastermultiplexer angesteuert. Tabelle 1 veranschaulicht für welchen Eingangswert, welcher Multiplexer ausgewählt wird.

Eingangswert	Ausgewählter Multiplexer
000	CI2
001	CI3
010	CI4
011	CI5
100	CI6
110	CI7
101	CI8
111	CI9

Tabelle 1 Mögliche Auswahl eines Multiplexer

Die LSB (Least Significant Bit) der GPIO Pins steuern anschließend die richtigen Antennen an. Wenn beispielsweise insgesamt ein Eingangswert von 000 000 ausgewählt wird, ist somit die erste Antenne angesteuert. Die Abbildung 26 zeigt, wenn ein Eingangswert von 000 000 angelegt wird.

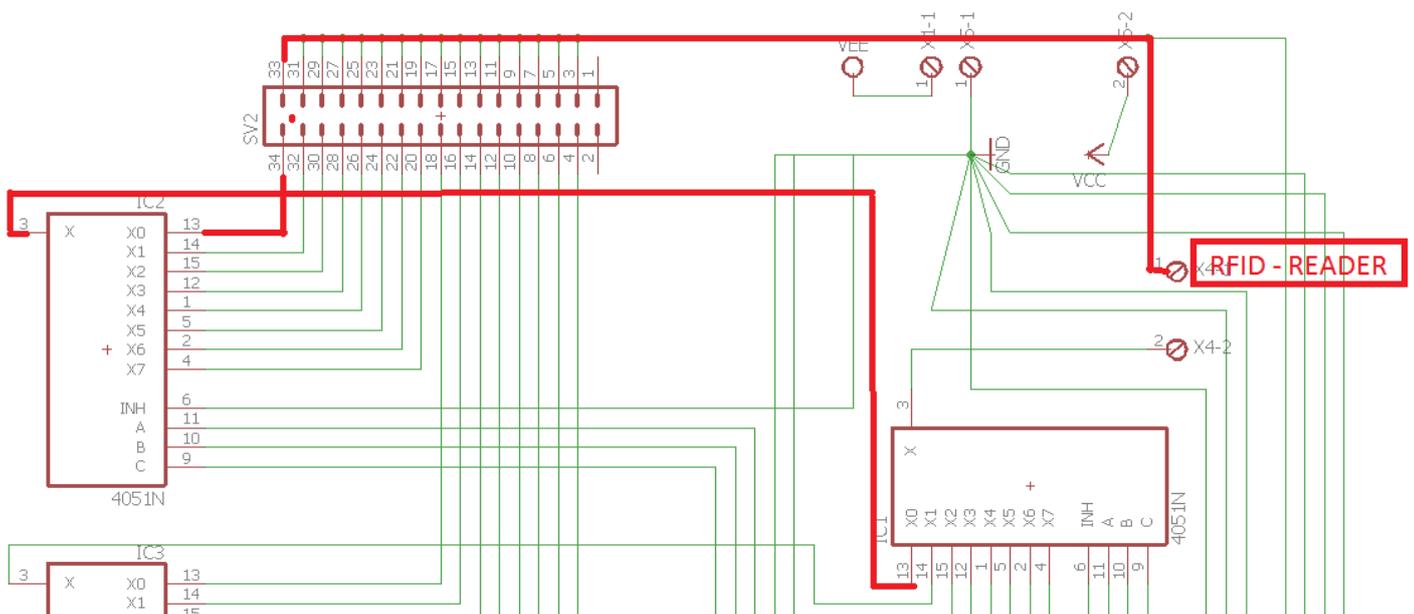


Abbildung 26 Kombination von "000 000" am Mastermultiplexer

Die LSB wählen immer alle Antennen zwischen X0 und X7 aus, deshalb kann nur durch die Angabe der ersten drei Bits die korrekte Antenne angesteuert werden, da der Mastermultiplexer den richtigen Multiplexer auswählt.

Ausgewählter Multiplexer	Belegung der Antennen
C12	X0 - X7
C13	X0 - X7
C14	X0 - X7
C15	X0 - X7
C16	X0 - X7
C17	X0 - X7
C18	X0 - X7
C19	X0 - X7

Tabelle 2 Mögliche Auswahl einer Antenne

Somit besteht die Möglichkeit alle 64 Antennen, die sich unter jedem Spielfeld befinden, anzusteuern.

Um das Platzieren der Multiplexer auf die eigens dafür entworfene Leiterplatte zu ermöglichen wurde auf die speziell dafür vorgesehenen Software EAGLE zurückgegriffen. Auf diese Software wird im Abschnitt 2.3.4 noch expliziter eingegangen werden.

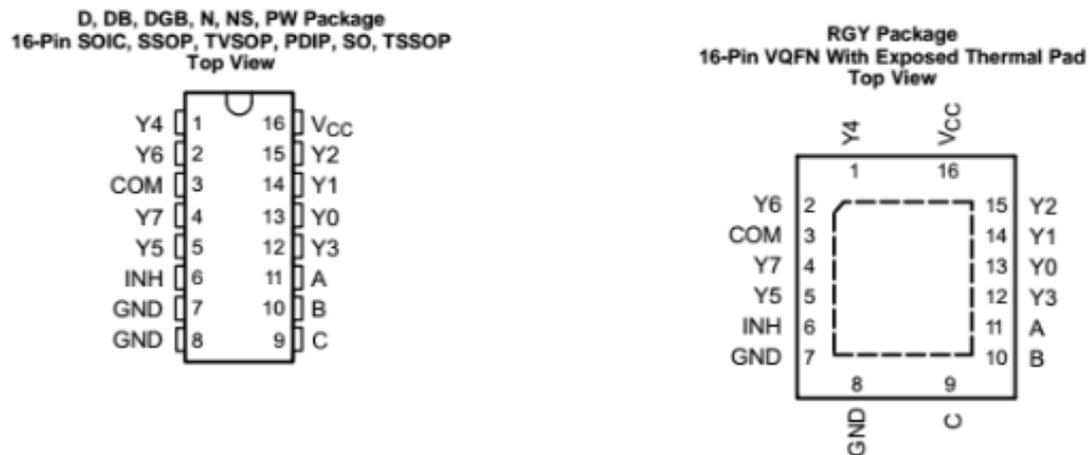


Abbildung 27 Pin-Belegung eines Multiplexer. Entnommen aus (Instruments, 2016).

Die Abbildung 27 veranschaulicht die Pin-Belegung eines Multiplexer und zeigt welche Pins mit welcher Eigenschaft belegt sind. Diese Informationen waren essentiell bei der richtigen Platzierung, sowie dem korrekten Löten auf der fertigen PCB Leiterplatte. Die Einkerbung an der obersten Kante des Multiplexers zeigt, aus welcher Richtung das Bauelement zu betrachten ist. Die Pinbelegung A, B und C steht in Verbindung mit den GPIO Pins, welche für das Umschalten der Multiplexer verantwortlich sind. Dies wird im Bereich „GPIO“ genauer beschrieben. Jeder Multiplexer ist über COM miteinander verbunden. Die Erdung oder Masse wird hier durch GND veranschaulicht. Die Pins Y0 bis Y7 stehen in direkter Verbindung mit den Antennen. Wie oben bereits erwähnt ist jeder Multiplexer in der Lage bis zu acht Antennen zu verwalten. Die INH Belegung jedes Multiplexer läuft ebenso über GND zusammen (Instruments, 2016).

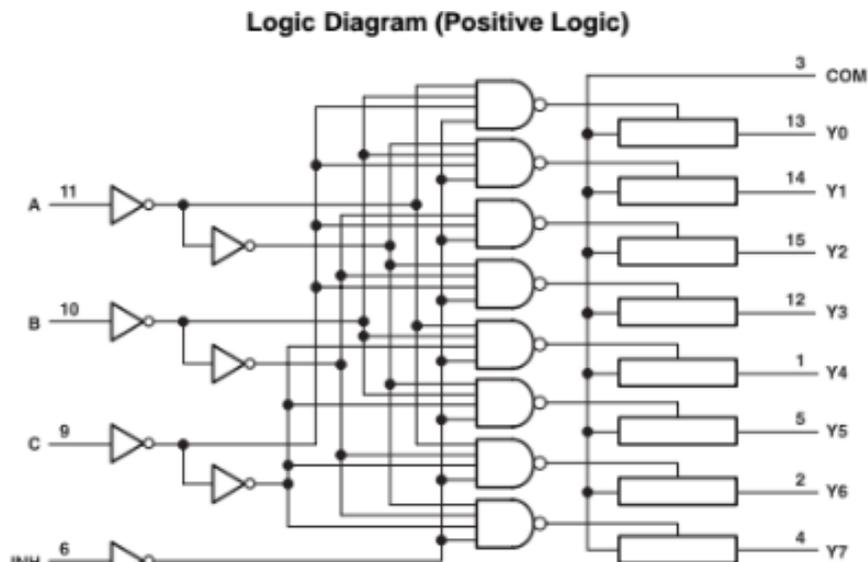


Abbildung 28 Schematische Darstellung eines Multiplexer. Entnommen aus (Instruments, 2016).

Das Logikdiagramm, das auch Logikgatter genannt werden kann, stellt die schematische Darstellung grafisch dar. Es gibt verschiedene Gattertypen wie beispielsweise UND, ODER, NICHT oder EXKLUSIV-ODER, die zu einem einzigen logischen Ergebnis umgewandelt werden können, und durch das Ausgangssignal Spannungswerte repräsentieren. Die logischen Zustände werden oftmals mit „0“ oder „1“, „low“ oder „high“ (L oder H) bezeichnet. Im Allgemeinen lassen sich alle logischen Verknüpfungen als Gatter realisieren (Instruments, 2016).

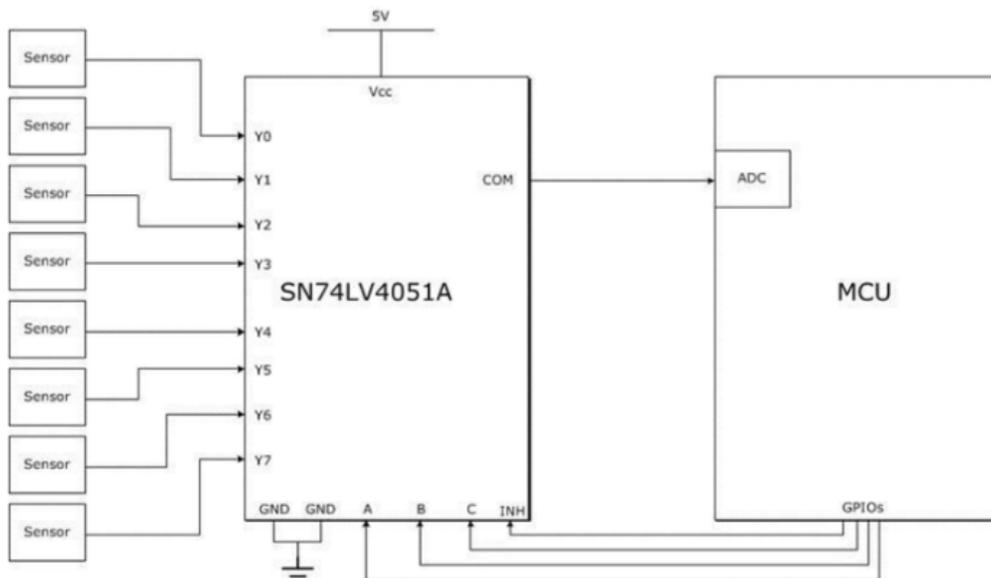


Abbildung 29 Beispiel eines Multiplexer mit Sensor und Strom.
Entnommen aus (Instruments, 2016).

Die Abbildung 29 veranschaulicht wie ein Multiplexer beispielsweise eingesetzt werden kann. Im Falle von Chess4U werden die Ausgänge Y0 bis Y7 mit Antennen statt Sensoren belegt. Der Mastermultiplexer wird hier durch einen Microcontroller angesteuert. Dies wird bei Chess4U durch den Raspberry PI übernommen.

2.3 LEITERPLATTE

2.3.1 EINLEITUNG LEITERPLATTE (PCB)

In den folgenden Abschnitten wird beschrieben, welche Herausforderungen das Designen einer PCB Leiterplatte in diesem Projekt mit sich brachte. Zuerst wird auf den allgemeinen Verwendungszweck einer Leiterplatte eingegangen, danach folgt eine spezifisch auf die Diplomarbeit passende Erklärung.

2.3.2 ALLGEMEINE ERKLÄRUNG

Beider Leiterplatte (Leiterkarte, Platine oder gedruckte Schaltung; englisch printed circuit board, PCB) handelt es sich um einen Träger für elektronische Bauteile. Ihre Aufgabe ist die mechanische Befestigung und elektrische Verbindung von Bauteilen. Praktisch jedes elektronische Gerät verfügt über eine oder mehrere Leiterplatten (vgl. Hermann 2016).

Ganz allgemein bestehen Leiterplatten aus elektrisch isolierendem Material, auf dem wiederum leitende Verbindungen (Leiterbahnen) montiert sind. Im Normalfall, beziehungsweise im Falle des höheren Qualitätsstandards, besteht dieses isolierende Material aus faserverstärktem Kunststoff, bei niedrigeren Anforderungen wird oft auf Hartpapier zurückgegriffen. Die Leiterplatten dagegen werden häufig aus einer dünnen Kupferschicht aus 35 µm geätzt. Im darauffolgenden Schritt werden die Bauelemente auf die dafür vorgesehenen Lötäugen gelötet (Hermann, 2016).

2.3.3 SPEZIFISCHE ERKLÄRUNG

Im Laufe des Projektes ist man auf das Problem gestoßen, dass das Platzieren der benötigten Bauelemente auf einem Steckbrett sehr unübersichtlich, eng und komplex ist. Im Zuge dessen ist man um das eigene Designen einer PCB-Leiterplatte nicht herumgekommen. Diese wurde in der Software namens EAGLE designed, welche explizit dazu zur Verfügung steht, PCB Leiterplatten für den eigenen Gebrauch und nach eigenen Wünschen zu erstellen. Mittels „Drag and Drop“ ermöglicht diese Software, die benötigten Bauelemente auf einer 100 mm x 80 mm großen Fläche so zu platzieren, dass es für den späteren Gebrauch ideal liegt. Jedes Element muss dabei über sogenannte Leiterbahnen miteinander, sowie mit GND, VCC und auch INH verbunden sein. Nachdem besagte Elemente platziert und angeordnet wurden, ist man in der Lage, das ganze so Zeichnen zu lassen, dass sie auf einem Bereich von 100 x 80 mm (in der Free Version) so angeordnet werden, dass der Platz möglichst gut genutzt wird. Dies wird zu 100% von der Software übernommen. Anschließend hat man die Möglichkeit, einzelne Leiterbahnen manuell zu bearbeiten. Leiterbahnen haben eine Standardbreite von 1mm. Anschließend folgte die Entscheidung, ob man die Platine einseitig oder beidseitig drucken lässt. Chess4U musste sich für den beidseitigen Druck entscheiden, da der Platz sonst nicht gereicht hätte. Als Vorlage für das Design galt die im Vorfeld erstellte Steck-Platine, welche als Testobjekt hervorragend diente. Nach der Fertigstellung des gesamten Layouts, wurde im Falle dieser Diplomarbeit der Druck der Leiterplatte einer externen Firma in Auftrag übertragen.

2.3.4 EAGLE 7.7.0 (SOFTWARE)

Die Firma CadSoft veröffentlichte ein Programm namens EAGLE, welches zur Erstellung und designen von Leiterplatten dient. Dieser Name bildet sich aus den Worten: „Einfach Anzuwendender Grafischer Layout Editor (engl.: Easily Applicable Graphical Layout Editor) (Computer CadSoft, GmbH, 2016).

Die Software besteht aus mehreren Komponenten. Mittels der Bauteil-Datenbank, die man in der Schaltplanansicht über Add aufrufen kann, ist es dem Nutzer möglich, die benötigten Bauelemente auszuwählen und im sogenannten Schaltplan Editor nach Belieben zu bearbeiten. Neben dem Bewegen, Rotieren, Spiegeln und dem Kopieren gibt es viele weitere Optionen, Bauelemente nach Belieben zu bearbeiten. Schlussendlich muss man jedes Bauelement mittels „Invoke“ über den richtigen Ausgang verbinden, sodass das Endprodukt funktionieren kann. In der nachstehenden Abbildung ist ein Bildausschnitt des Arbeitsbereiches des Schaltplan Editors von EAGLE zu sehen.

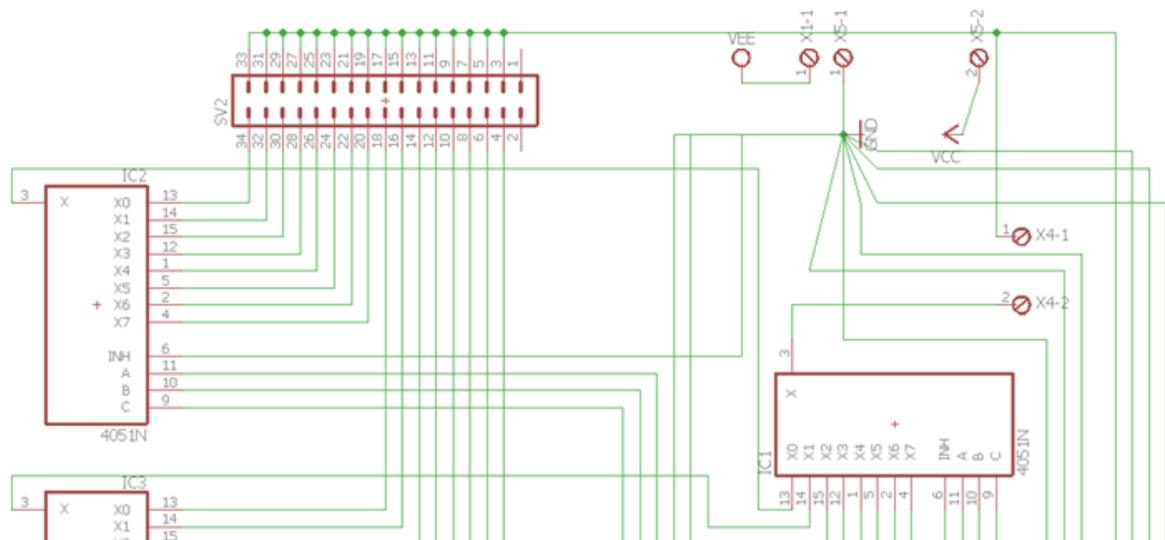


Abbildung 30 Schaltplan Editor Ansicht EAGLE

In der Ansicht des Layout-Editors werden per Klick die idealen Leiterbahnen berechnet. Es besteht jedoch jederzeit die Möglichkeit Leiterbahnen manuell zu bearbeiten, beziehungsweise nach Belieben zu verschieben. Nach fertiger Berechnung aller Leiterbahnen ist es dem Nutzer möglich, das fertige Leiterplatten Layout zu speichern. Bei dieser Diplomarbeit wurde der Druck der Leiterplatte von einer externen Firma durchgeführt (Computer CadSoft, GmbH, 2016).

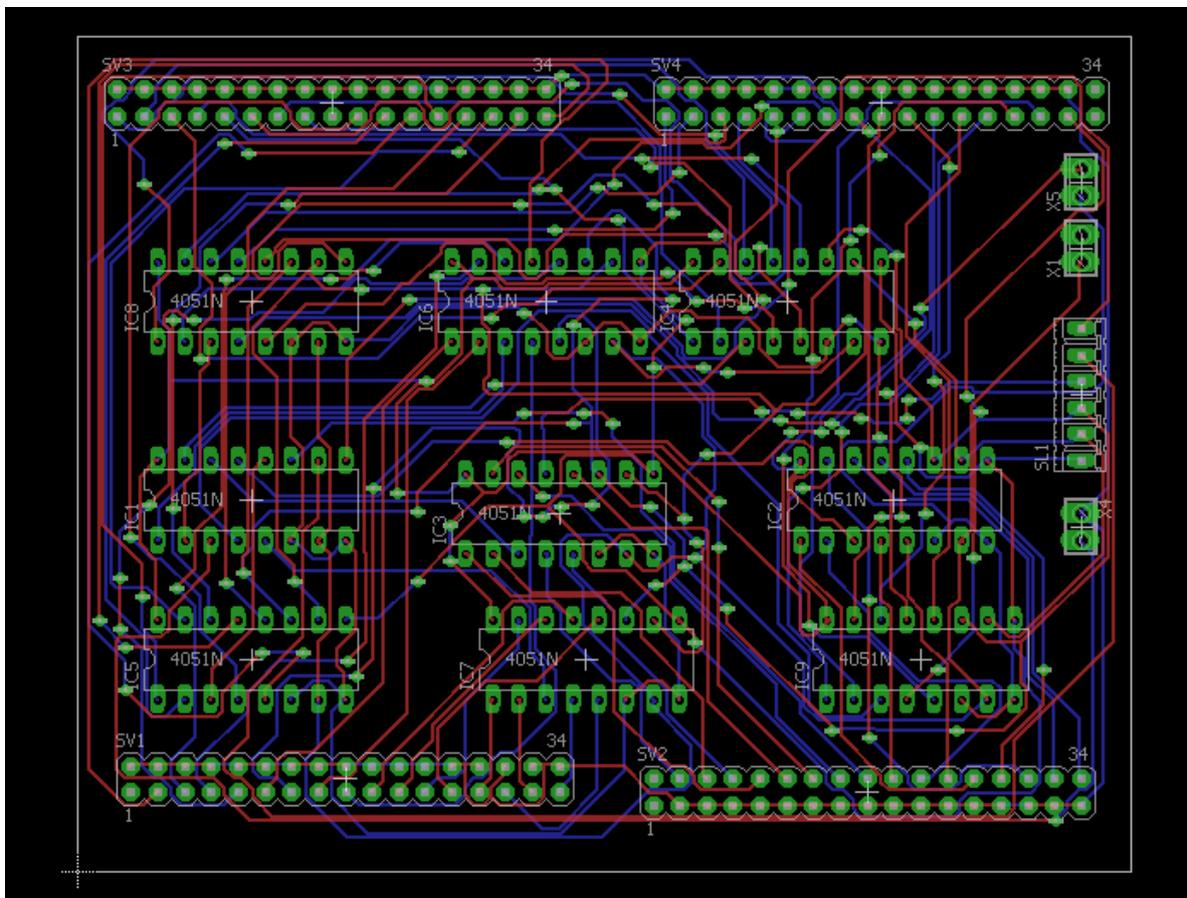


Abbildung 31 Layout Editor Ansicht EAGLE

3 LÖTEN AM PCB-LAYOUT

Da die Konstruktion auf einem Steckbrett sich als sehr unübersichtlich herausgestellt hatte, wurde entschieden, alle Bauelemente auf eine von einer Firma produzierten Leiterplatte zu löten. Insgesamt wurden der Druck von insgesamt zwei Leiterplatten in Auftrag gegeben, da die zweite Leiterplatte nicht nur kostengünstiger war, sondern ebenso als mögliches Backup dienen hätte können, falls ein Lötfehler das erste Exemplar zerstört hätte. Die nachstehende Abbildung veranschaulicht eine Leiterplatte, auf welcher drei Sockel als Befestigung für Multiplexer zu sehen sind, sowie eine Pin-Leiste welche als Schnittstelle zwischen Leiterplatte und Antennen dient.

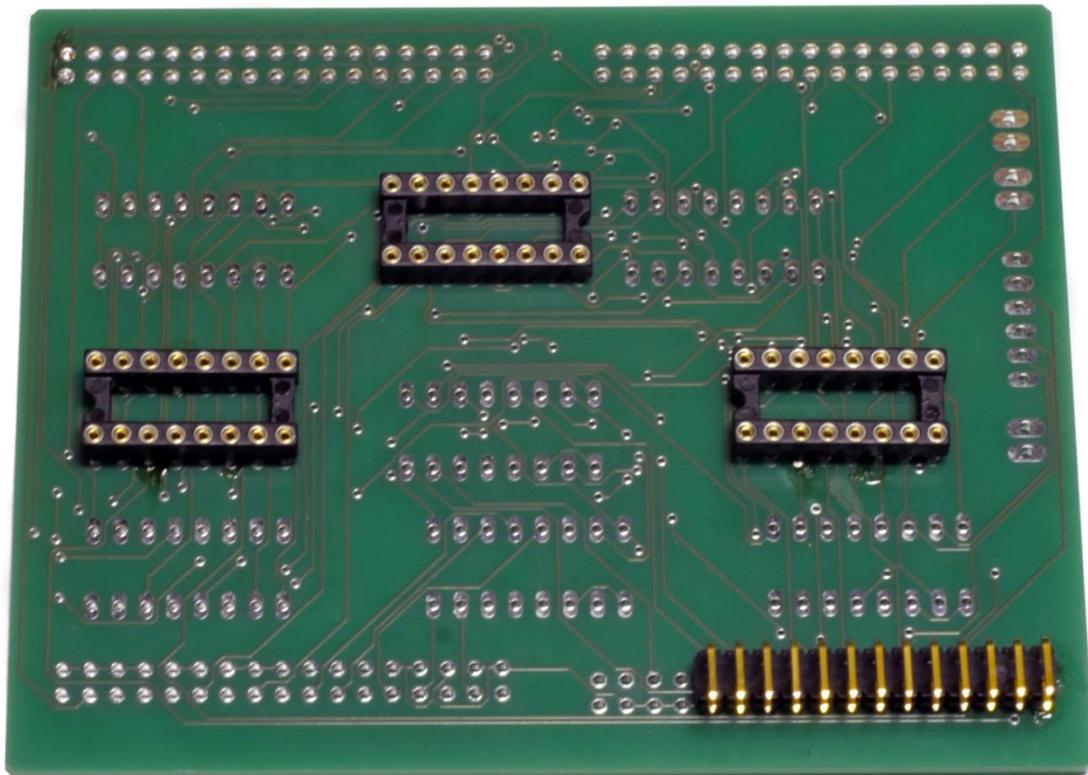


Abbildung 32 Leiterplatte samt Sockel für Befestigung der Multiplexer

3.1.1 PCB MIT READER UND ANTENNEN VERBINDEN

Damit die Platine mit dem Reader verbunden werden kann, benötigt man ein Erweiterungsboard (engl. expansion board), da die Pins auf dem Raspberry Pi bereits von dem Reader belegt sind. Jedoch werden nicht alle Pins genutzt, und diese können nun zur Ansteuerung der Multiplexer, zur Antennenauswahl, verwendet werden. Mit GPIOs (General Purpose Input Output) werden programmierbare Ein- und Ausgänge (In- and Outputs) für allgemeine Zwecke bezeichnet. (Elektronik-Kompodium.de, 2017)

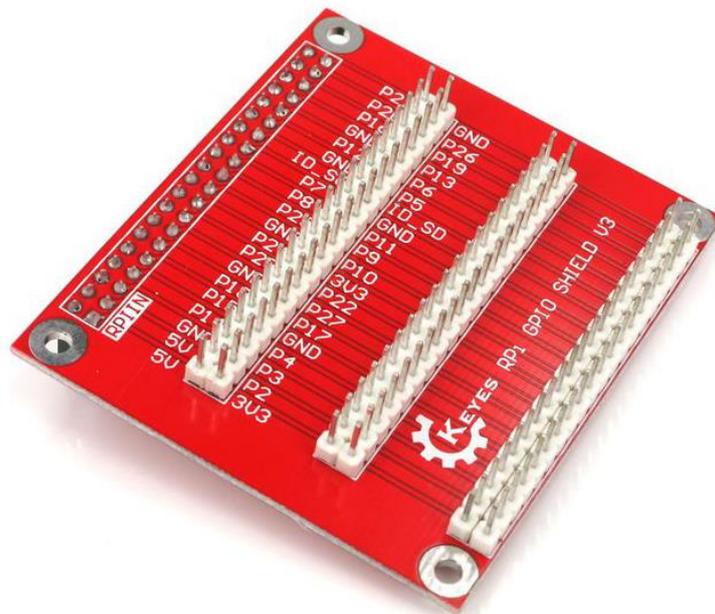


Abbildung 33 Zeigt ein Erweiterungsboard für einen Raspberry Pi an.

Es müssen sechs GPIOs mit dem PCB verbunden werden, damit $64(2^6=64)$ Antennen angesteuert werden können. Es wird in einem kurzen Zeitintervall aus einer binären Kombination nur eine Antenne angesteuert. Schlussendlich sollen in einem kurzen Zeitrahmen alle Antennen angesteuert werden. *Achtung:* Das Erweiterungsboard muss sachgemäß mit dem Raspberry Pi verbunden werden, damit es später bei der Verbindung zu keinem Kurzschluss kommt. *Hinweis:* Für dieses Projekt wurde ein Erweiterungsmodul für einen Raspberry Pi 3 verwendet (der Raspberry Pi 3 hat 18 GPIOs mehr). Da das System von Chess4U auf einem Raspberry Pi 1 läuft, bleiben diese zusätzlichen Pins unbenutzt und wurden

abisiert, damit es beim fehlerhaften Verbinden nicht zu einem Kurzschluss kommt.

Es wird die Leiterplatte mit dem Raspberry PI, beziehungsweise mit dem Erweiterungsboard (auf dem Raspberry PI), verbunden.

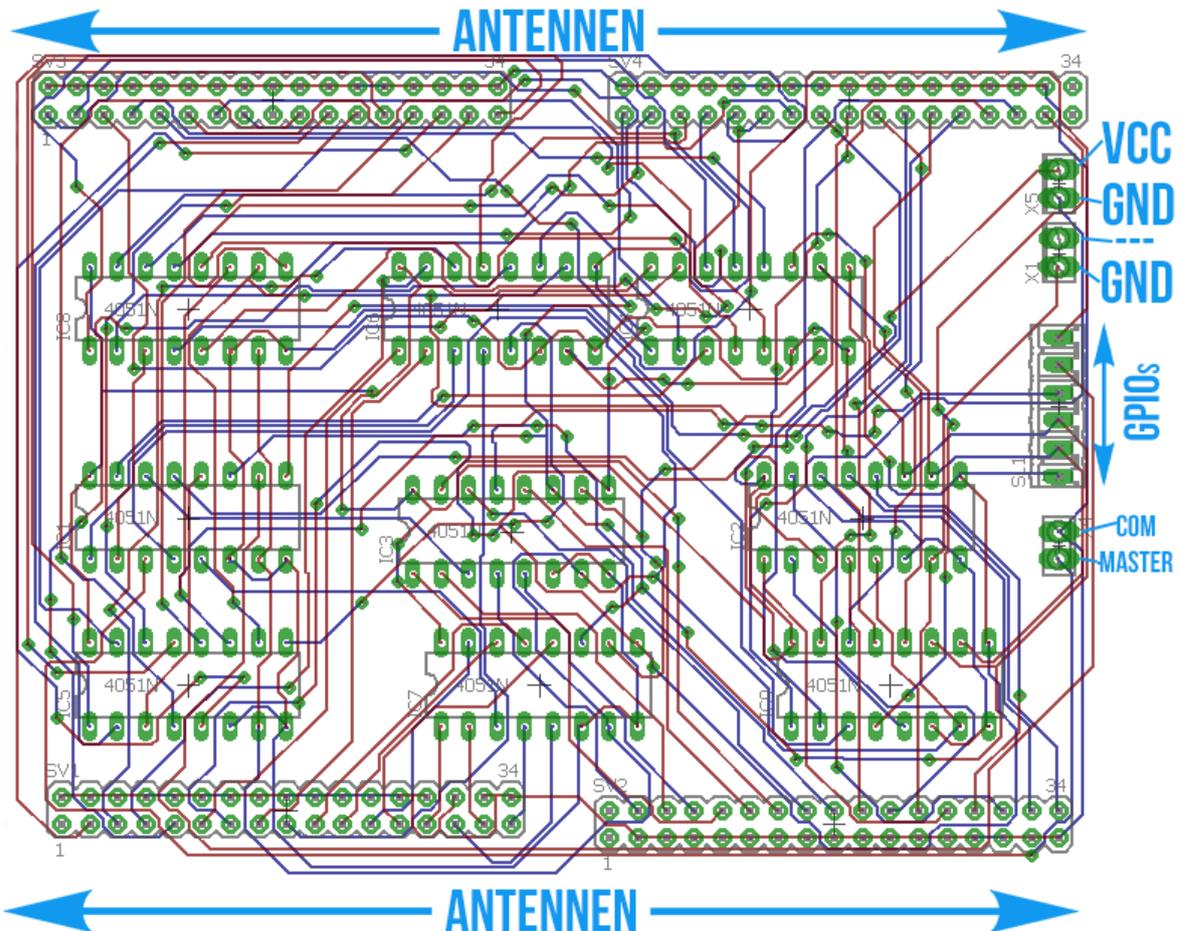


Abbildung 34 Zeigt die Anschlüsse des PCBs an

Es werden zwei Antennen gewickelt, um das PCB auf Funktion zu testen. Es muss beachtet werden, dass jeweils ein Anschluss einer Antenne mit dem Anschluss der nächsten Antenne verbunden ist. Schlussendlich führen diese zum COM. Durch das Erweiterungsboard wurden die Pins physisch verdreifacht. Jedoch stehen nicht alle Pins beziehungsweise GPIOs zur Verfügung, da einige von dem Erweiterungsmodul benötigt werden.

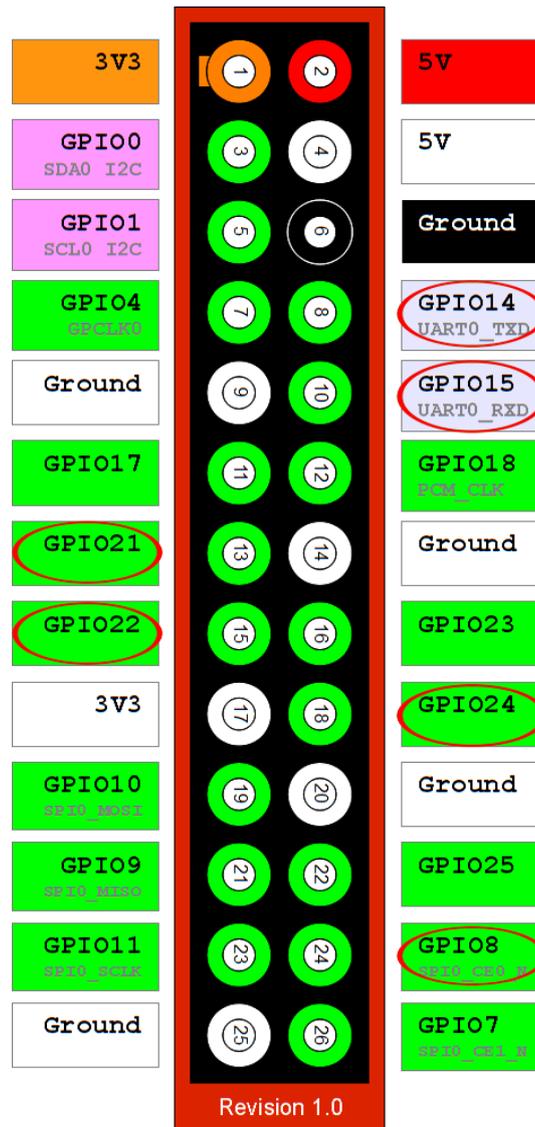


Abbildung 35 Zeigt freie GPIOs an. Entnommen von (Matt, 2012).

Die Abbildung 35 zeigt die freien GPIOs an, die verwendet werden können. Es wird auch VCC (+5V) und GND (Ground) benötigt, jedoch können diese hier problemlos verwendet werden. Für die Verbindung zwischen dem PCB und der Erweiterungsplatine eignen sich Überbrückungskabel (weiblich) auf engl. „jumper wires“ sehr gut. Die Antennen werden auch auf die Überbrückungskabel gesteckt. Für die Verbindung müssen zuerst die Ground-Kabel mit dem Erweiterungsboard nach den Abbildungen verbunden werden. Danach die Erdung, Ground und schlussendlich die 6 GPIOs. Am Ende bleiben auf dem PCB noch zwei Anschlüsse übrig, nämlich COM und Master, diese müssen in die 2-Pin-Stecker-Klemme gesteckt und leicht angezogen werden.

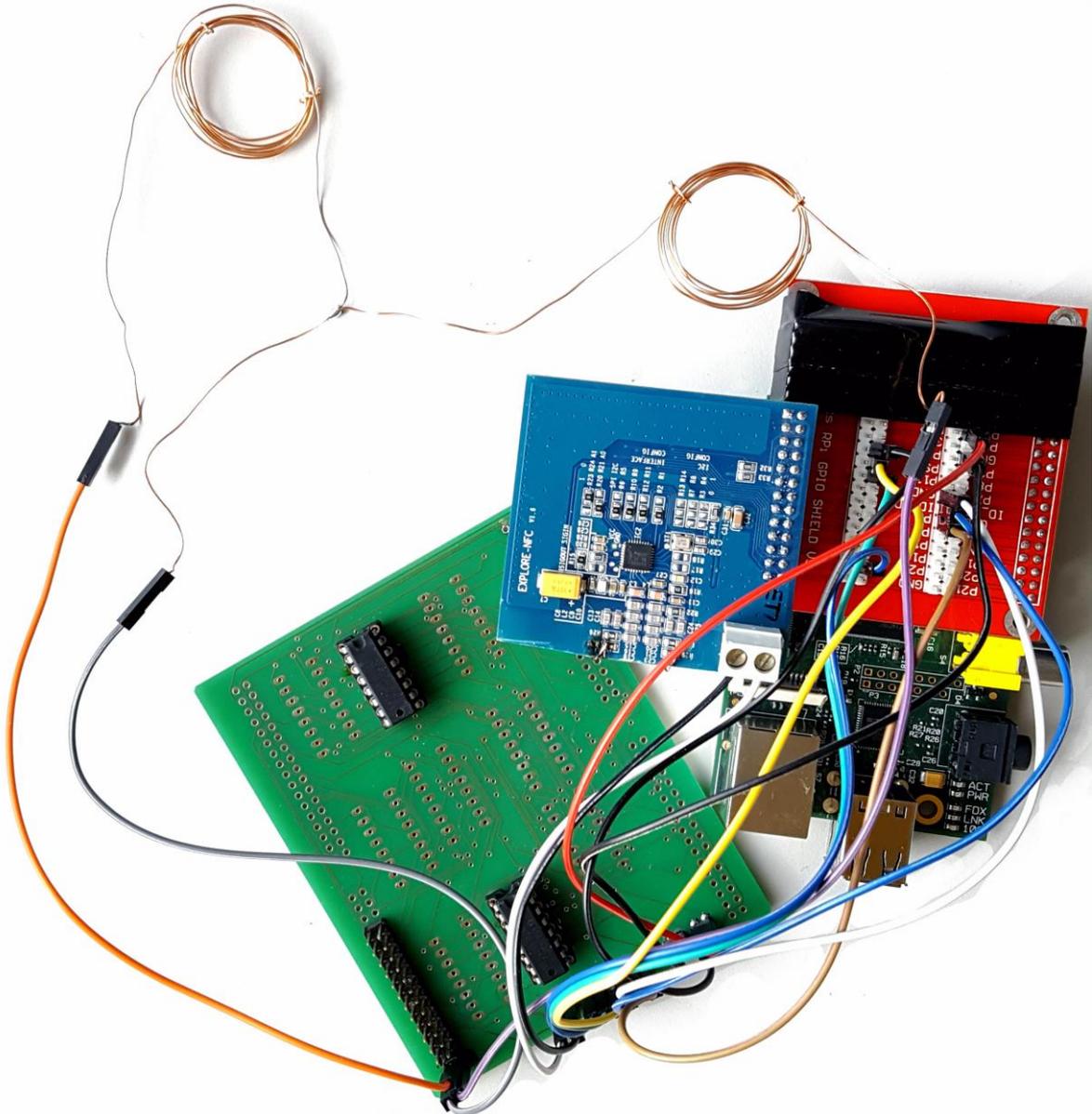


Abbildung 36 Zeigt das PCB verbunden mit dem Raspberry PI an.

4 SOFTWARE

4.1 ANTENNEN SWITCH

GPIOs werden zur Ansteuerung der Multiplexer zur Antennenauswahl verwendet werden. Es wird aus einer bestimmten binären Kombination genau eine Antenne angesteuert.

```
#!/usr/bin/env python3
import nxppy
import time
import RPi.GPIO as GPIO
import requests

GPIO.setmode(GPIO.BCM)
mifare = nxppy.Mifare()

while True:
    GPIO.setup(14, GPIO.OUT)
    GPIO.output(14, GPIO.LOW)
    try:
        uid=mifare.select()
        sendId='B2:A1:'+uid
        print(sendId)
        send={'id': sendId}
        r=requests.post("http://192.168.0.1/chess4u/da/client2.php", params=send)
        print(r.url)
        print(r.text)
    except nxppy.SelectError:
        pass
    time.sleep(1)
    GPIO.output(14, GPIO.HIGH)
    try:
        uid=mifare.select()
        sendId='A1:B2:'+uid
        print(sendId)
        send={'id': sendId}
        r=requests.post("http://192.168.0.1/chess4u/da/client2.php", params=send)
        print(r.url)
        print(r.text)
    except nxppy.SelectError:
        pass
    time.sleep(1)
```

Abbildung 37 Script zum Auslesen von Tags und umschalten von GPIOs

Es wird das Script aus der Abbildung 17 erweitert. Es kommen zwei Module dazu, nämlich das GPIO und das requests Modul. Das GPIO-Modul wird die Spannungen setzen, somit die Antennen umschalten. Das requests-Modul greift dann auf den Server zu und übermittelt dem Server die Position beziehungsweise die ID des Tags, per Querystring. In einer while-Schleife wird in diesem Fall der GPIO 14 auf „low“ gesetzt, somit entsteht die Kombination „000000“ und es wird die erste Antenne angesteuert. Danach wird der Tag beziehungsweise die Spielfigur erfasst und es wird die Position und die ID an den Server übermittelt und ebenfalls im Script ausgegeben. Nach einer Sekunde wird der GPIO 14 auf „high“ gesetzt, somit entsteht die binäre Kombination „000001“ und es wird die zweite Antenne angesteuert. Die Position der Figur wird durch die Ansteuerung der Antenne ermittelt, wobei jede Antenne über eine eigene binäre Kombination verfügt. Die Antennen sollten jedoch eine fixe Position unter dem Schachbrett haben, damit eine bestimmte binäre Kombination für jedes Feld zuständig ist, somit funktioniert auch die Positionsermittlung.

4.2 WEBSOCKETS

Im Web laufen die meisten Client Server Anfragen über HTTP (Hyper Text Transfer Protocol). Dabei ruft der Client die Seite einmal auf, stellt so eine Verbindung zum Server her, der Server schickt die verlangten Daten und die Verbindung wird wieder geschlossen. Sobald der Client oder der Server eine Änderung vornehmen will, muss eine neue Anfrage gestartet werden. Um diesem Problem entgegenzusteuern und Nachrichten vom Server beziehungsweise vom Client in Echtzeit zu bekommen, gibt es Websockets.

Websockets sind bidirektionale, vollduplexe, länger anhaltende Verbindungen von einem Webbrowser zu einem Server. Wenn hier einmal eine Verbindung vorhanden ist, bleibt diese solange bestehen, bis entweder der Client, oder der Server die Verbindung beendet. Solange die Verbindung besteht, kann der Server dem Client Nachrichten schicken, und umgekehrt. (Boden, 2013)

Bei dieser Diplomarbeit werden laufend neue Positionen übertragen und diese müssen dann sofort in die Datenbank eingetragen werden. Hierfür eignen sich Websockets perfekt.

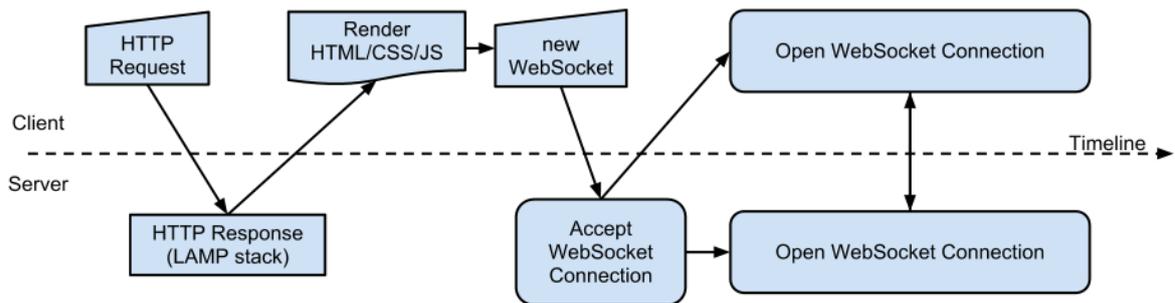


Abbildung 38 Entstehung einer WebSocket Verbindung. Entnommen aus (Boden, 2013).

Anhand Abbildung 38 kann man erkennen, wie eine WebSocket-Verbindung entsteht. Der Client sendet zuerst eine Anfrage, einen HTTP Request, an den Server, dieser antwortet mit einem HTTP-Response.

Nun kann beim Client HTML, CSS, etc. gerendert werden und ein neuer WebSocket erstellt werden, der Server akzeptiert diesen und es gibt eine bestehende WebSocket Verbindung.

Um diese Verbindung herstellen zu können, gibt es viele Libraries, bei dieser Diplomarbeit wird eine von Ratchet⁹ verwendet.

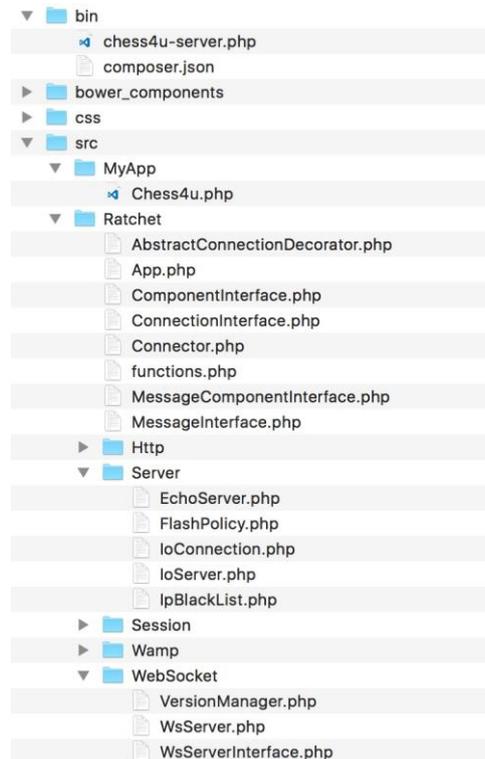


Abbildung 39 Library Struktur

In Abbildung 39 kann man die Library Struktur mit den Daten dieser Diplomarbeit sehen.

⁹<http://socketo.me/> zul. geladen am: 01.02.2017

4.3 WEBSOCKET CHESS4U SERVER

In der Datei Chess4U.php befinden sich die allgemeinen Server Einstellungen. Die wichtigsten Funktionen sind *onOpen*, *onClose*, *onMessage* und *saveMessage*

- Über die Funktion *onOpen* wird eine neue Verbindung zum Client hergestellt.
- Über die Funktion *onClose* wird diese Verbindung dann wieder beendet.

```
public function onMessage(ConnectionInterface $from, $msg)
{
    $numRecv = count($this->clients) - 1;
    echo sprintf('Connection %d sending message "%s" to %d other connection%s' . "\n"
        , $from->resourceId, $msg, $numRecv, $numRecv == 1 ? '' : 's');
    if ($this->saveMessage($msg)) {
        echo 'Saved message to DB';
    } else {
        echo '!';
    }
    foreach ($this->clients as $client) {
        if ($from !== $client) {
            $client->send($msg);
        }
    }
}
```

Abbildung 40 PHP Server Funktion *onMessage*

- Über die Funktion *onMessage* (Abbildung 40) werden neue Nachrichten, also die neuen Positionen der einzelnen Schachfiguren, empfangen und an die Clients versendet. Außerdem wird in dieser Funktion *saveMessage* aufgerufen und gleichzeitig geprüft, ob diese auch korrekt ausgeführt wurde.
- Über die Funktion *saveMessage* werden die Daten in der Datenbank gespeichert.

Der Server bekommt die Daten von dem Python Script, welches am Raspberry PI läuft.

```
import requests
sendd={'id': positionen[33]}
resp=requests.post('http://localhost/chess4u/da/client2.php',params=sendd)
```

Abbildung 41 Abschnitt des Python Scripts

Wie man anhand Abbildung 41 erkennen kann, sendet dieses Python Script per HTTP-POST Request die Daten an ein PHP Script.

```
require __DIR__ . '/vendor/autoload.php';
Ratchet\Client\connect('ws://127.0.0.1:8080')->then(function ($conn) {
    $conn->on('message', function ($msg) use ($conn) {
        echo "Received: {$msg}\n";
        $conn->close();
    });
    $conn->send($_GET['id']);
    $conn->close();
}, function ($e) {
    echo "Could not connect: {$e->getMessage()}\n";
});
```

Abbildung 42 Verbindung Python Script -> PHP Server

An Abbildung 42 kann man sehen, dass dieses Script ein PHP Client ist, der per HTTP GET die Daten vom Raspberry PI empfängt und dann automatisch per Websocket Nachricht an den Server sendet. Dieses Script wird jedes Mal, wenn das Python Script Daten sendet, im Hintergrund ausgeführt.

4.4 WEBSOCKET CHESS4U CLIENT

Der Client besteht aus mehreren Teilen. Der erste Teil ist das Login Script, welches über die Datei `index.php` aufgerufen wird. In diesem gibt es zwei Möglichkeiten, entweder man meldet sich als User an, oder man beginnt ein neues Spiel. Wenn man sich anmeldet, kommt man auf die Userseite, auf der man all seine bisher gespielten Spiele gelistet anschauen kann. Will man ein neues Spiel beginnen, müssen beide Spieler ihren Benutzernamen und ihr Passwort eingeben und wenn dies bei beiden Spieler korrekt eingegeben wurde, wird man auf die Seite `spiel.php` weitergeleitet, während im Hintergrund ein neues Spiel in die Datenbank eingetragen wird.

- Beide Möglichkeiten werden über das Login Script `User.php` ausgeführt. In diesem befindet sich die Klasse `User` mit den Funktionen `login`, `logout`, `IsLoggedIn`, `spielStarten` und `IsGameActiv`.

```
public function login($email, $pwd) {
    $pw = "";
    $userdb = "root";
    $server = "localhost";
    $db = "chess4u";
    $charset = "utf8";
    $opt = [PDO::ATTR_ERRMODE => PDO::ERRMODE_EXCEPTION, PDO::ATTR_DEFAULT_FETCH_MODE => PDO::FETCH_ASSOC];
    $connection = new PDO("mysql: host=$server; dbname=$db; charset=$charset", $userdb, $pw, $opt);
    if (!$connection) {
        error_log("Error!");
        die("Datenbankfehler");
    }
    $control = 0;
    $statement = $connection->prepare("SELECT * FROM `spieler` WHERE email = :email and password = :pw");
    $statement->bindParam(':email', $email, PDO::PARAM_STR);
    $statement->bindParam(':pw', $pwd, PDO::PARAM_STR);
    $statement->execute();
    while ($row = $statement->fetch()) {
        $control++;
        extract($row);
        $user = $id;
        echo "user: " + $user;
    }
    if ($control > 0) {
        $_SESSION["username"] = $user;
        header("Location: login.php");
        return true;
    }
    else {
        return false;
    }
}
```

Abbildung 43 login Funktion der PHP User Klasse

- In Abbildung 43 kann man die Funktion *login* sehen. Hier wird der Benutzername und das Passwort übergeben und in einem SELECT Befehl geprüft, ob diese zwei Daten in ein und derselben Spalte in der Datenbank vorhanden sind. Falls dies der Fall ist, wird man zu der User Seite weitergeleitet und eine neue Session wird gestartet.

- In der Funktion *spielStarten* werden ebenfalls Benutzername und Passwort übergeben, in dem Fall aber zweimal, also von zwei verschiedenen Benutzern und in der Funktion wird wieder das Vorhandensein der Spalten geprüft, diesmal wieder zweimal, einmal für den ersten Benutzer und einmal für den zweiten. Falls dies der Fall ist, wird man auf die Spiel-Seite weitergeleitet. Dieser Seite werden auch die ID's der beiden Benutzer per `$_GET` übergeben und sobald die Seite aufgerufen ist, besteht eine neue Websocket Verbindung zum Server.

```
public function logout() {
    session_destroy();
    return true;
}

public function IsLoggedIn() {
    if(isset($_SESSION["username"])) {
        return true;
    }
}
```

Abbildung 44 *logout* und *isLoggedIn* Funktion der PHP User Klasse

- In der Abbildung 44 kann man die Funktionen *Logout* und *IsLoggedIn* sehen. In der Funktion *Logout* wird die aktuelle Session beendet und man kann sich mit einem neuen Benutzer anmelden bzw. ein neues Spiel starten.
- In der Funktion *IsLoggedIn* wird geprüft, ob die aktuelle Session noch läuft. In der Funktion *IsGameActiv* wird dasselbe geprüft, nur mit der aktuellen Game Session.

```

<form class="navbar-form" action="?page=log" method="POST">
  <li>
    <div class="form-group">
      <label class="sr-only" for="exampleInputEmail3">Email address</label>
      <input type="email" class="form-control" id="inputEmailLogin" name="inputEmailLogin" placeholder="Email">
    </div>
  </li>
  <li role="separator" class="divider"></li>
  <li>
    <div class="form-group">
      <label class="sr-only" for="exampleInputPassword3">Password</label>
      <input type="password" class="form-control" id="inputPasswortLogin" name="inputPasswortLogin" placeholder="Password">
    </div>
  </li>
  <li role="separator" class="divider"></li>
  <li>
    <button type="submit" class="btn btn-default">Sign in</button>
  </li>
</form>

```

Abbildung 45 Anmelde-Formular der Seite index.php

```

<?php
  session_start();
  include("User.php");
  $x = new User;
  $x -> IsLoggedIn();
  $page;
  if(!isset($_GET["page"])) {
    $page="";
  }
  else
    $page = $_GET["page"];
  if($page == "log") {
    if($x->login($_POST["inputEmailLogin"],
      $_POST["inputPasswortLogin"]))
      echo "PASST";
    else {
      $falschesLogin = "Die Email und das Passwort
        stimmen nicht über ein";
    }
  }
  if($page == "startgame") {
    $x->spielStarten($_POST["inputEmail1"],
      $_POST["inputPasswort1"],
      $_POST["inputEmail2"], $_POST["inputPasswort2"]);
  }
?>

```

Abbildung 46 Ausführung der PHP User Klasse

Wie in den daliegenden Codestücken zu sehen ist, werden zu Beginn aus der Datei `index.php` die Funktionen der Klasse ‚User‘ ausgeführt (Abbildung 46) mit den Werten, welche von dem Login Formular (Abbildung 45) und dem Spiel Starten Formular übergeben werden.

In der Datei `login.php` werden alle bisher gespielten Spiele des jeweiligen Users angezeigt. Klickt man dann auf ein Spiel, so bekommt man genauere Informationen. Es werden alle Spielzüge des Spiels in kurzer und langer Notation, sowie grafisch auf einem Schachbrett angezeigt.

In der Datei `spiel.php` befinden sich zwei Tabellen für die Notationen, zwischen denen man umschalten kann, und eine Tabelle für das Schachbrett. Die zwei Tabellen für die Notationen sind für die kleine und die lange Notation.

Wie oben erwähnt, besteht, sobald man die Seite aufruft, eine Websocket-Verbindung zum Server. Dies geschieht in dem Fall über ein eingebundenes, externes Javascript-File, in dem die Verbindung hergestellt wird. Sobald der Server Daten sendet, werden diese einerseits in den Tabellen für die Notationen hinzugefügt, andererseits im Schachbrett visuell angezeigt.

```

function init() {
    var host = "ws://localhost:8080";
    try {
        socket = new WebSocket(host);
        console.log('WebSocket - status '+socket.readyState);
        socket.onopen = function(msg) {
            console.log("Welcome - status "+this.readyState);
        };
        socket.onmessage = function(msg) {
            console.log("Received: "+msg.data);
            var altesFeld = msg.data.substr(0,2);
            var feld = msg.data.substr(3,2);
            var figur = msg.data.substr(6,8);
            var farbe = msg.data.substr(6,1);
            var figurGenau = msg.data.substr(7,1);
            addNotationsListe(altesFeld, feld, farbe, figurGenau);
            for(var i = 8; i>=1; i--)
            {
                var abc = 'ABCDEFGH';
                for(var j=0;j<abc.length;j++)
                {
                    var feldx = ''+abc[j]+''+i;
                    if(altesFeld == feldx)
                    {
                        altePosition(altesFeld);
                    }
                    if(feld == feldx)
                    {
                        welchePosition(feld, figur);
                        console.log(feld);
                        console.log(figur);
                    }
                }
            }
        };
        socket.onclose = function(msg) {
            console.log("Disconnected - status "+this.readyState);
        };
    }
    catch(ex){
        log(ex);
    }
}

```

Abbildung 47 Implementierung des PHP Clients

- In der Abbildung 47 kann man die Funktion *init* sehen, über die alle Websocket Funktionen des Clients implementiert werden. Die Funktion *init* wird im *body*-Tag der Seite *Spiel.php* über den Befehl *onLoad* aufgerufen.

4.5 DIE BENUTZEROBERFLÄCHE

Implementiert werden die Seiten mittels Bootstrap. Dies ermöglicht uns das Erstellen dynamischer Webseiten. Bootstrap skaliert Webseiten und Anwendungen mit einer einzigen Code-Basis für Handys, Tablets und Desktops. Man bekommt ebenso eine umfangreiche Dokumentation für gängige HTML-Elemente, dutzende benutzerdefinierte HTML- und CSS Komponenten und jQuery Plug-Ins (Twitter Inc., 2011).

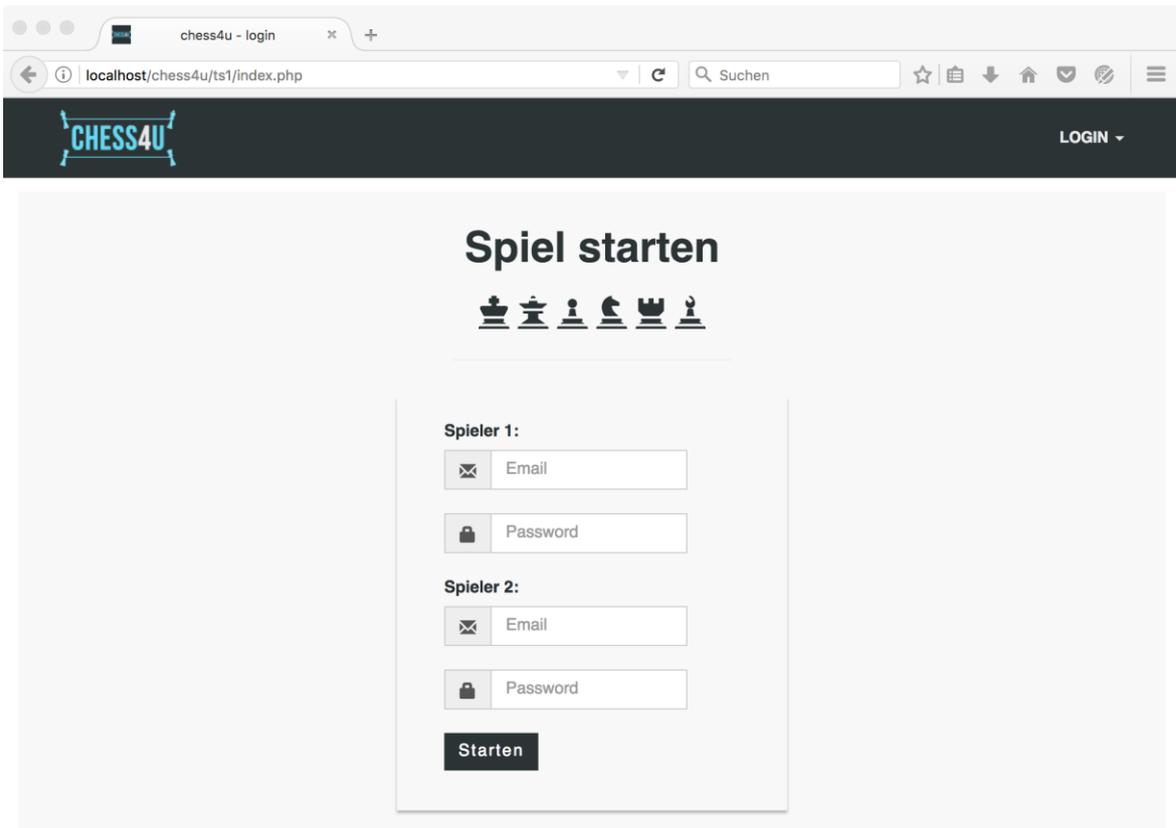


Abbildung 48 die Seite index.php

- Abbildung 48 zeigt die Seite index.php. Die Titelleiste wird durch die Bootstrap Klasse *navbar* erstellt, damit, wenn das Fenster verkleinert wird, die Seite dynamisch kleiner wird.

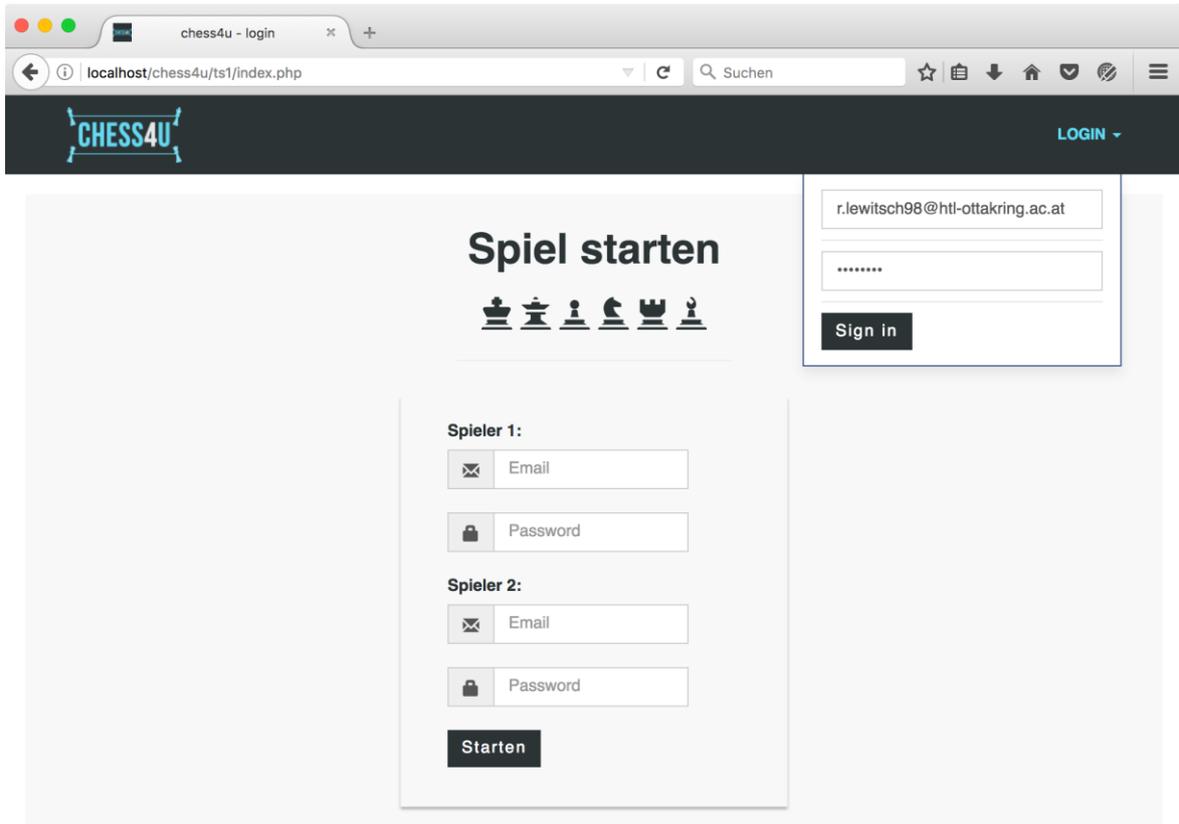


Abbildung 49 Die Seite `index.php` mit dem Login-Formular

Wie in Abbildung 49 zu sehen, erscheint per Klick auf den Button Login das Login Formular. Dieses besteht aus zwei Eingabefeldern und einem Button.

- Die Schachfiguren und die Zeichen vor den Eingabefeldern werden mittels der Klasse `glyphicon` der Bootstrap Library erstellt. Das Formular wird ebenso mit Bootstrap implementiert, damit man auch, wenn man das Fenster verkleinert, noch eine schöne, skalierte Ansicht zu sehen bekommt.

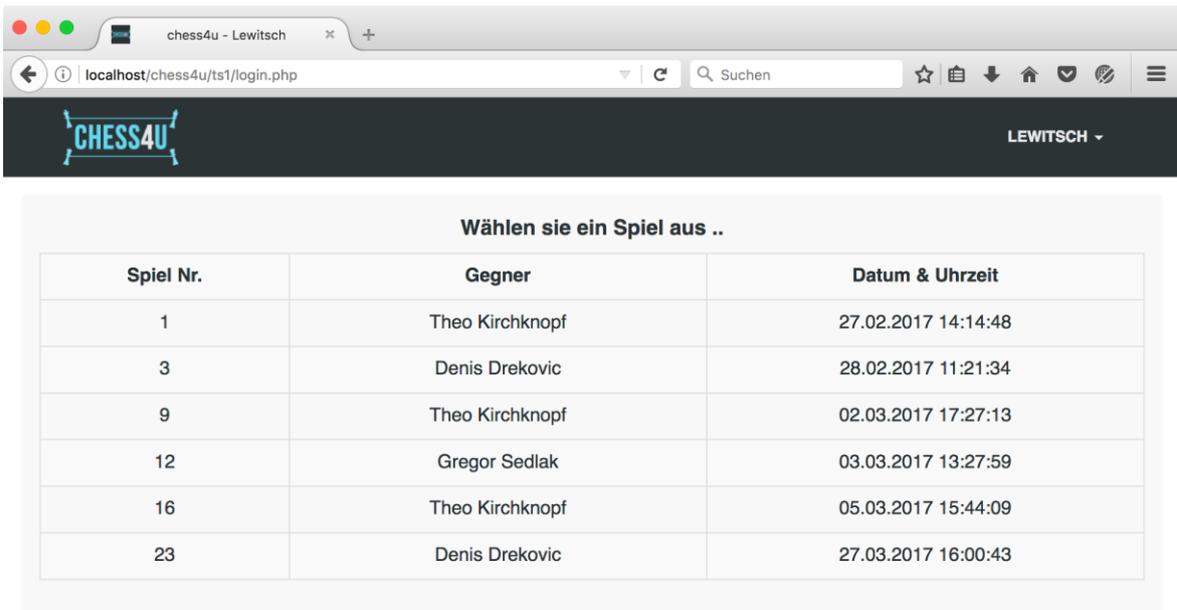
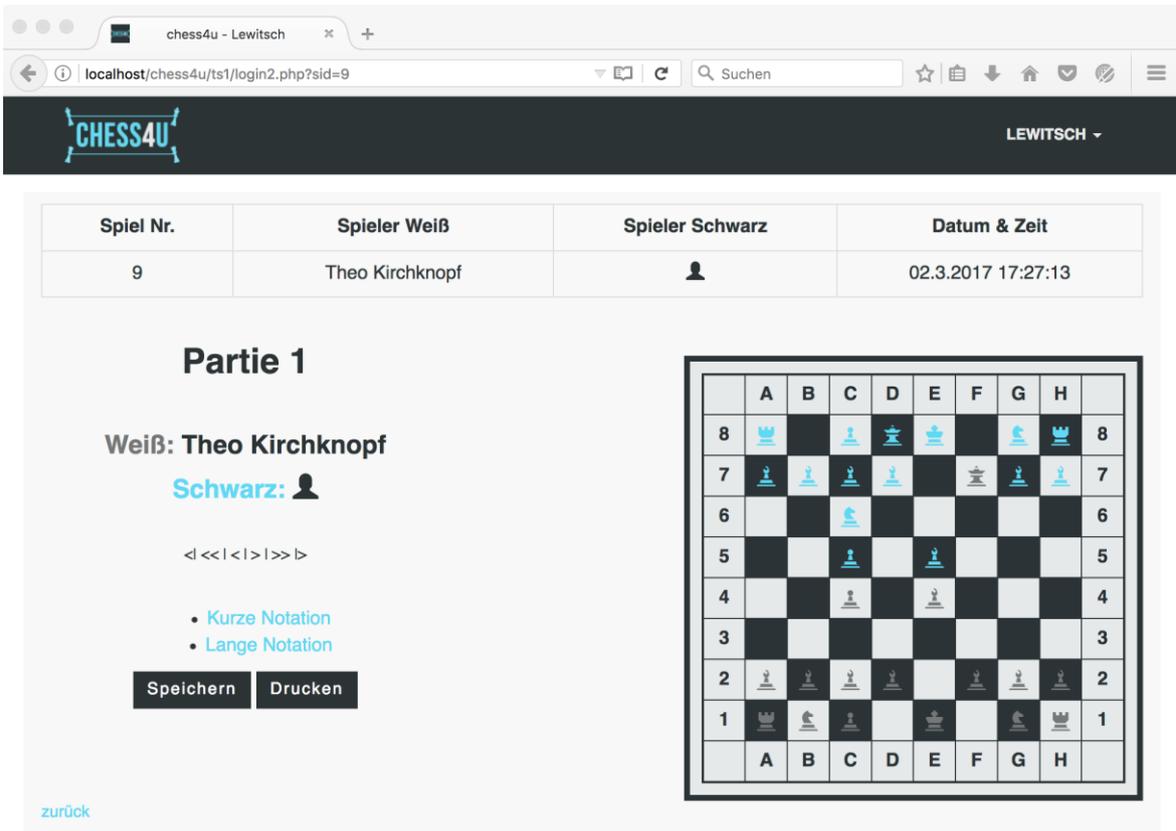


Abbildung 50 Die Seite login.php

- Sobald man sich angemeldet hat, kommt man zur Login-Seite. Auf dieser befindet sich eine Auflistung aller bisher gespielten Spiele, mit dem Gegner, dem Datum und der Uhrzeit des Spiels. Wenn man auf eine Zeile klickt, erfährt man mehr über das ausgewählte Spiel. Die Tabelle ist mittels der Bootstrap Klasse *table* definiert und kann somit dynamisch verkleinert werden.



Spiel Nr.	Spieler Weiß	Spieler Schwarz	Datum & Zeit
9	Theo Kirchknopf		02.3.2017 17:27:13

Partie 1

Weiß: Theo Kirchknopf

Schwarz: [empty]

<< | < | > | >>

- Kurze Notation
- Lange Notation

Speichern **Drucken**

zurück

	A	B	C	D	E	F	G	H	
8	♔		♙	♚	♖		♜	♗	8
7	♘	♙	♙	♙		♚	♙	♙	7
6			♜						6
5			♙		♙				5
4			♙		♙				4
3									3
2	♙	♙	♙	♙		♙	♙	♙	2
1	♔	♘	♙		♚		♜	♗	1
	A	B	C	D	E	F	G	H	

Abbildung 51 Die Seite login2.php

Bei den genaueren Informationen eines Schachspiels bekommt der User, wie in Abbildung 51, die Notationen und ein visuelles Schachbrett zu sehen. Mit den Pfeilbuttons kann man zwischen den einzelnen Spielzügen umschalten. Die Informationen sieht man dann einerseits in den Notationstabellen, andererseits im Schachbrett. Die Notationstabellen erscheinen per Klick auf den Button ‚Kurze Notation‘ oder den Button ‚Lange Notation‘. Das Schachbrett wird dynamisch mittels PHP eingebunden.

```

<table id="schachbrettWeiss">
  <tr>
    <th class="text-center"></th>
    <th class="text-center">A</th>
    <th class="text-center">B</th>
    <th class="text-center">C</th>
    <th class="text-center">D</th>
    <th class="text-center">E</th>
    <th class="text-center">F</th>
    <th class="text-center">G</th>
    <th class="text-center">H</th>
  </tr>
  <?php
  for ($i = 8; $i >= 1; $i--) {
    ?>
    <tr>
      <th class="text-center"><?= $i ?></th>
      <?php
      for ($j = "A"; $j <= "H"; $j++) {
        $farbeHintergrund = "black";
        if ($j == "B" || $j == "D" || $j == "F" || $j == "H") {
          if ($i % 2 != 0) {
            $farbeHintergrund = "white";
          } else {
            $farbeHintergrund = "#2b3336";
          }
        } else if ($j == "A" || $j == "C" || $j == "E" || $j == "G") {
          if ($i % 2 == 0) {
            $farbeHintergrund = "white";
          } else {
            $farbeHintergrund = "#2b3336";
          }
        }
      }
      $tdId = "weiss" . $j . $i;
      ?>
      <td class="<?= $tdId ?>" style='background-color: <?= $farbeHintergrund ?>; text-align: center;'></td>
      <?php
    }
    ?>
    <th class="text-center"><?= $i ?></th>
  </tr>
  <?php
}
?>
<tr...>
</table>

```

Abbildung 52 Die Tabelle zur Implementierung des Schachbrettes

Wie in der Abbildung 52 zu sehen, benötigt man zur Erstellung des Schachbrettes zwei Schleifen, da ein Schachbrett 8x8 Felder groß ist und somit wie ein zweidimensionales Array angesehen werden kann. Innerhalb der beiden Schleifen wird jedes Feld mit der jeweiligen Farbe ausgegeben. Bei der ersten Schleife wird bei acht begonnen zu zählen und hinabgezählt, weil diese Zahl gleichzeitig ein Teil der Feldidentifikation ist und man bei einem Schachbrett vertikal gesehen von unten die Felder zu definieren beginnt. Die beiden Spalten zu Beginn und zum Ende der Tabelle dienen zur Beschriftung des Schachbrettes.

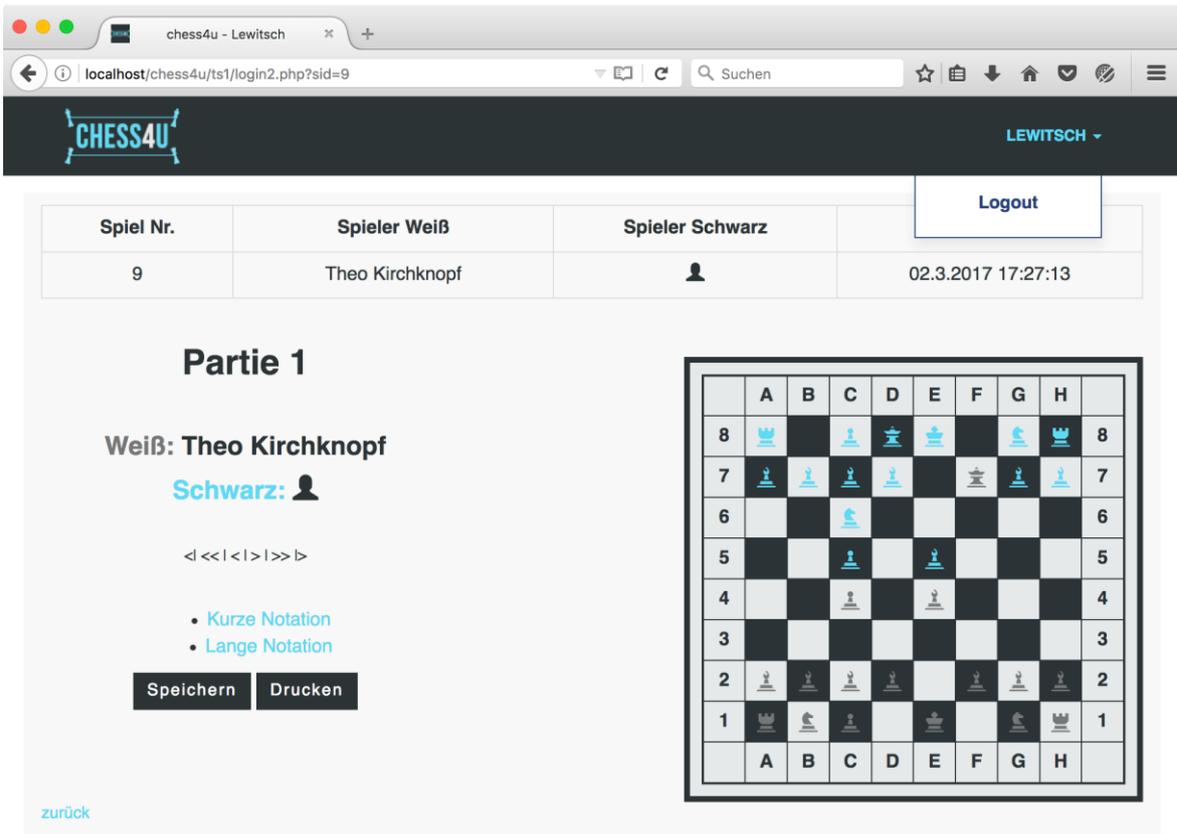


Abbildung 53 Die Seite login2.php mit dem logout-Button

Wie in der Abbildung 53 zu sehen, bekommt man per Klick auf den User-Namen den Button logout zu sehen, mit welchem man sich aus der Login-Seite ausloggt.

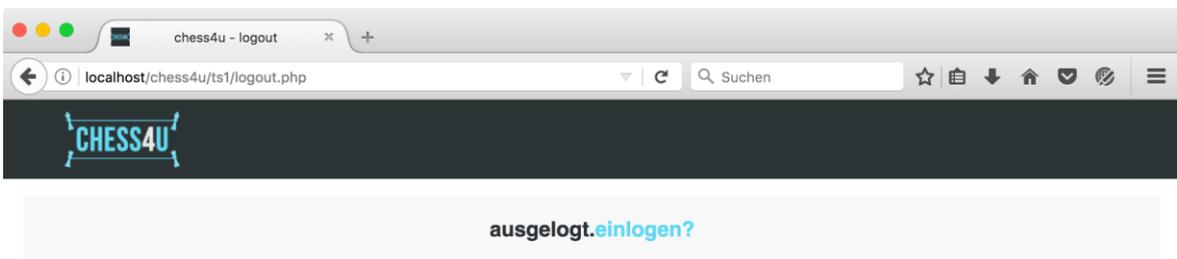


Abbildung 54 Die Seite logout.php

In Abbildung 54 kann man die logout Seite sehen. Wenn man auf den Button ‚einloggen?‘ klickt, kommt man wieder zur Startseite zurück.

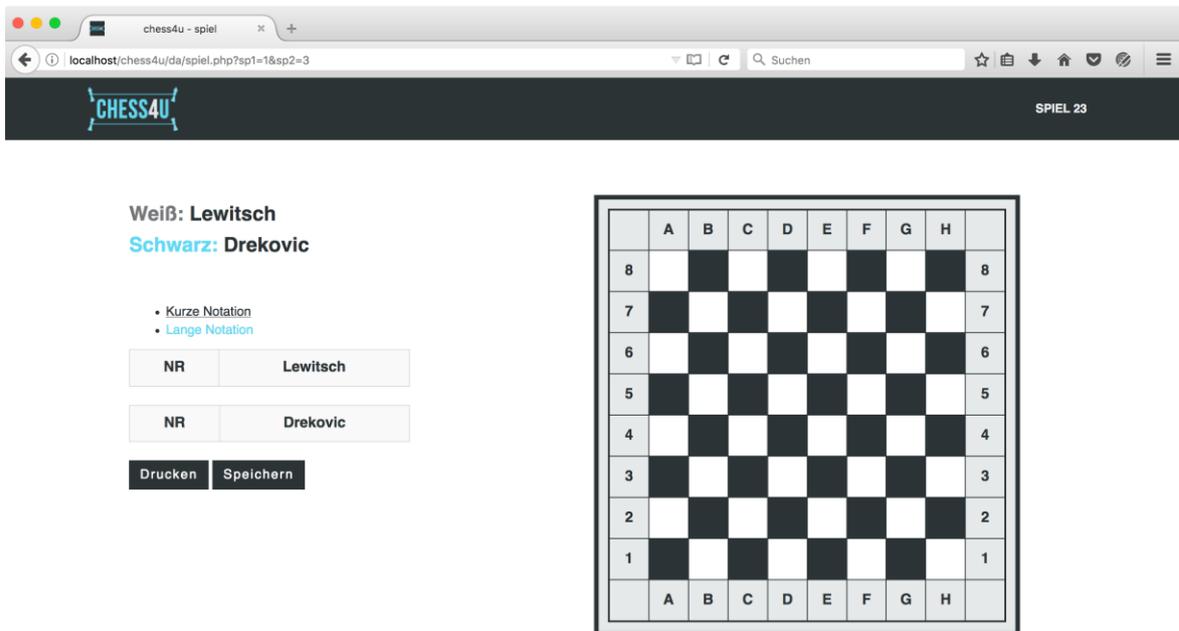


Abbildung 55 Das Spiel

Wenn man ein neues Spiel startet, bekommt man ein Bild wie in der Abbildung 55 zu sehen. Die Spielernamen werden über die URL übergeben. Das Schachbrett wird wieder mit dem obigen Codestück eingebunden und in den Notationstabellen bekommt man die Positionsangaben zu sehen. Die Positionen werden per Websockets übertragen, man bekommt somit jeden Positionswechsel live auf dem Schachbrett mit.

5 DATENBANK

5.1 ALLGEMEIN

Zur Sicherung aller Schachzüge und Spiele braucht es eine Datenbank.

Eine Datenbank ist eine organisierte Datensammlung und wird verwendet zur Speicherung, Formatierung, Verwaltung, Zugriff, Manipulation und Darstellung der Daten. Sie ist ein elektronisches Archiv und kann von mehreren Anwendern und Anwendungsprogrammen genutzt werden.

Die Abfrage der Daten erfolgt mit Abfragesprachen, bei denen die bekannteste für relationale Datenbanken die Structured Query Language (SQL) ist. Die Implementierung findet beispielsweise in MySQL statt. (Lipinski, 2016)

5.2 RELATIONALE DATENBANK

Relationale Datenbanken sind auf ein Modell des IBM-Forschungsinstitut-Mitarbeiter E.F. Code zurückzuführen, der dies 1970 beschrieben und definiert hat.

Beliebig viele Einzeltabellen können dabei in beliebiger Art und Weise miteinander verknüpft werden. Die Daten werden in Spalten und Reihen strukturiert. Die einzelnen Reihen enthalten die Datensätze und die Spalten werden durch Attribute definiert (beispielsweise Name, Straße, Geburtsdatum) und haben immer den gleichen Datentyp. Spalten und Reihen dürfen jeweils nur einmal vorkommen.

Man unterscheidet im Allgemeinen zwischen einer spaltenorientierten, also Column-oriented Database oder Columnar Database, und zeilenorientierten, also Row-oriented, Datenbank.

Relationale Datenbanken erfüllen einige Voraussetzungen in Bezug auf die Relation der Daten, daher können sie als besonders leistungsfähig angesehen werden. Sie können in mehreren Tabellen ausgedehnt und durch ihre Attribute miteinander kombiniert werden. Solche Kombinationen entstehen durch identische Attribute, beispielsweise die ID-Nummer (Lipinski, 2016).

5.3 DATENBANKMANAGEMENTSYSTEM

Datenbankmanagementsysteme (DBMS) werden zur Informationsverwaltung eingesetzt und bestehen aus den Daten und Programmen, die diese Daten verwalten und verarbeiten. Genauer gesagt sind sie zuständig für die Speicherung, Integrität, Wiederherstellung und den Zugriff auf die Daten.

Ein DBMS hat ein Datenverzeichnis, welches sozusagen Metadaten, also Name, Struktur, Attribute, Lage und Typ, der Daten enthält. In diesen Metadaten werden alle logischen und physikalischen Informationen und Änderungen solange die Daten bestehen, also vom Einfügen bis zum Löschen, gespeichert.

Ein Vorteil von DBMS ist die Sicherheit, da es in einem DBMS zusätzlich zum Betriebssystem und der Netzwerksicherheit meistens noch mehrere Sicherheitslevel gibt. Den Zugang zur Datenbank erlangt man dann nur noch mittels Passwort Authentifizierung und User Login (Lipinski, 2012).

5.4 RELATIONALES DATENBANKMANAGEMENTSYSTEM

Zur Verwaltung von relationalen Datenbanken gibt es relationale Datenbankmanagementsysteme (RDBMS), welche Inhalte in Beziehungen und Verhältnisse zueinander stellt. Die Daten werden in einfach strukturierten Tabellen gespeichert und lassen sich in verschiedenen Konstellationen miteinander kombinieren, dadurch kann man von einer äußerst flexiblen Datenspeicherung sprechen. Dazu werden komplexe Datenobjekte in viele einfache Datenobjekte zerlegt und in den Reihen und Spalten einer Tabelle gespeichert.

Über Beziehungen kann der Zugriff hergestellt werden und durch die Zerlegung der Datenobjekte können schematische Änderungen gut durchgeführt werden (Lipinski, 2009).

5.5 MYSQL

MySQL ist das am häufigsten im Web verwendete RDBMS, mit dem Webseiten dynamisch geändert werden können. Es ist ein robustes und schnelles Mehrbenutzersystem für die Erstellung dynamischer Webseiten und auf allen gängigen PC-Betriebssystemen lauffähig (Lipinski, 2011).

Auch in dieser Diplomarbeit wird mit MySQL gearbeitet.

5.6 ENTITY RELATIONSHIP DIAGRAM

Als Hilfe für die Erstellung einer Datenbank kann man ein Entity Relationship Diagram (ERD oder ER-Diagramm) erstellen, welches eine grafische Darstellung der Tabellenstruktur einer Datenbank ist. Ein ER-Diagramm stellt Entitäten und Beziehungen zwischen diesen dar, wobei die Entitäten dabei die Tabellen in einer Datenbank sind.

In relationalen Datenbanken werden Beziehungen zwischen Datenbanken über Primärschlüssel und Fremdschlüssel definiert. Der Primärschlüssel ist eine eindeutige Identifikation einer Zeile und der Fremdschlüssel definiert die eindeutige Verbindung zwischen den Tabellen.

Bei einem ER-Diagramm sind die einzelnen Tabellen als Rechteck dargestellt und all ihre Attribute als Kreise, wobei die Primärschlüssel unterstrichen sind. Die Tabellen sind mittels Verbindungslinien und einem Karo miteinander verbunden, außerdem ist die Art der Beziehung eingezeichnet. Dabei unterscheidet man zwischen:

- 1:1 Beziehung - Jeder Fremdschlüssel kann in der miteinander verbundenen Tabelle nur einmal vorkommen.
- 1:n - Jeder Fremdschlüssel kann in der miteinander verbundenen Tabelle ein- oder mehrmals vorkommen.
- m:n - Beide miteinander verbunden Tabellen können mehrmals aufeinander verweisen. (Lipinski, 2012)

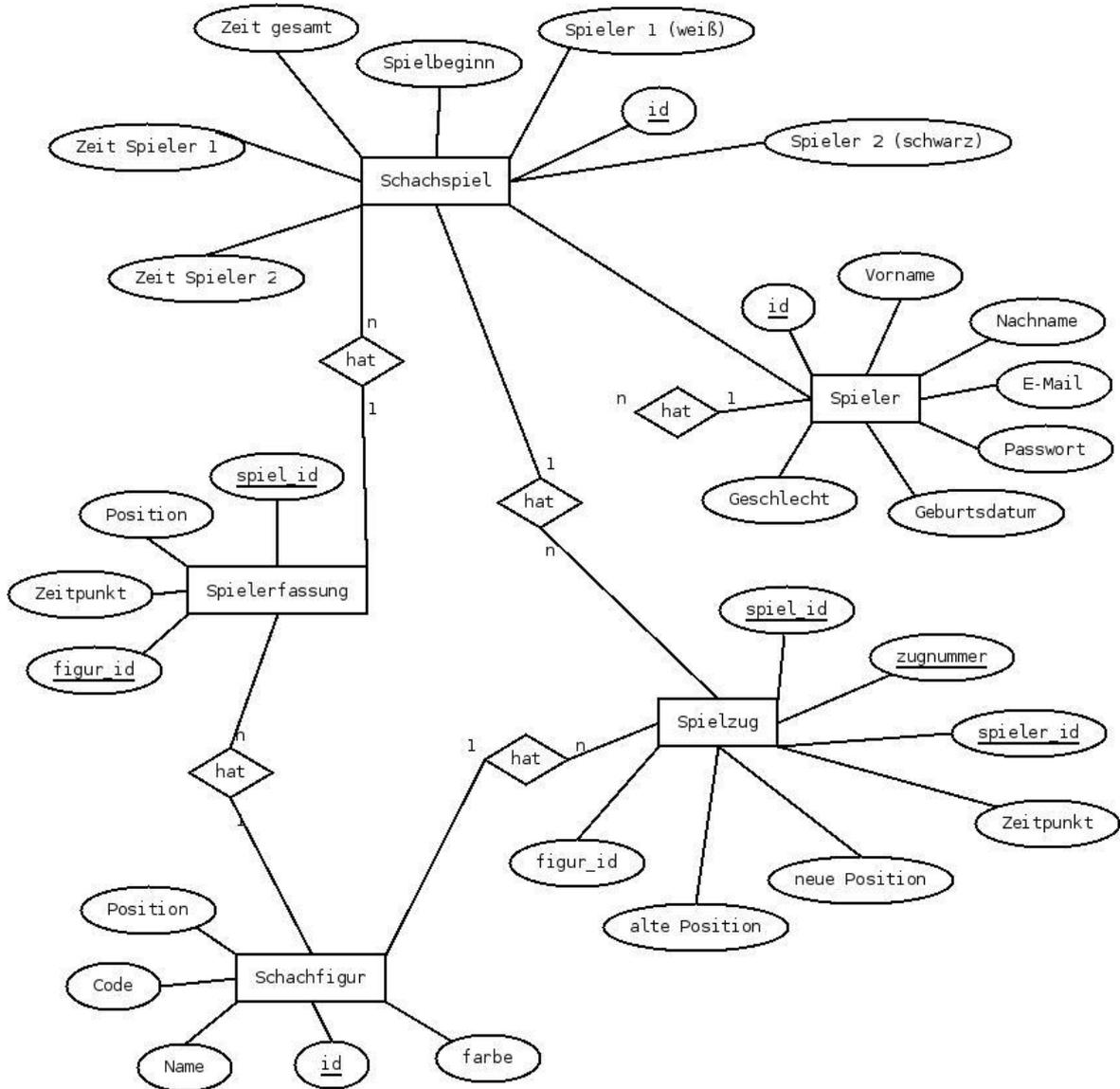


Abbildung 56 ER-Diagramm Chess4U

Abbildung 56 zeigt das ER-Diagramm dieser Diplomarbeit.

5.7 TABELLEN - CHESS4U

Nachdem das ER-Diagramm erstellt wurde, werden die Tabellen erstellt. Wie oben erwähnt, wird hierfür mit MySQL gearbeitet.

#	Name	Typ	Kollation	Attribute	Null	Standard
1	schachspiel_id 🗝️	int(11)			Nein	kein(e)
2	spieler_schwarz 🔑	int(11)			Ja	NULL
3	spieler_weiß 🔑	int(11)			Ja	NULL
4	spielbeginn	timestamp			Nein	CURRENT_TIMESTAMP
5	zeit_gesamt	time			Ja	NULL
6	zeit_spielerweiß	time			Ja	NULL
7	zeit_spielerschwarz	time			Ja	NULL

Abbildung 57 Tabelle Schachspiel

In der Abbildung 57 kann man die Attribute der Tabelle Schachspiel sehen, wobei `spieler_schwarz` und `spieler_weiß` Fremdschlüssel zur Tabelle Spieler sind. In dieser Tabelle werden die Zeiten und Spieler der einzelnen Schachspiele gespeichert.

#	Name	Typ	Kollation	Attribute	Null	Standard	Kommentare	Extra
1	id 🗝️	int(11)			Nein	kein(e)		AUTO_INCREMENT
2	vorname	varchar(20)	latin1_swedish_ci		Nein	kein(e)		
3	nachname	varchar(30)	latin1_swedish_ci		Nein	kein(e)		
4	email	varchar(50)	latin1_swedish_ci		Nein	kein(e)		
5	passwort	varchar(20)	latin1_swedish_ci		Nein	kein(e)		
6	geburtsdatum	date			Nein	kein(e)		
7	geschlecht	enum('weiblich', 'maennlich')	latin1_swedish_ci		Nein	kein(e)		

Abbildung 58 Tabelle Spieler

Die Attribute der Tabelle Spieler kann man in Abbildung 58 sehen. Alle vorhandenen Spieler sind in dieser Tabelle gespeichert.

#	Name	Typ	Kollation	Attribute	Null	Standard	Kommentare	Extra
1	id_schachspiel	int(11)			Nein	kein(e)		
2	id_schachfigur	int(11)			Nein	kein(e)		
3	position	enum('A1', 'A2', 'B1', 'B2', 'C1', 'C2', 'D1', 'D2...')	latin1_swedish_ci		Nein	kein(e)		
4	zeitpunkt	timestamp		on update CURRENT_TIMESTAMP	Nein	CURRENT_TIMESTAMP		ON UPDATE CURRENT_TIMESTAMP

Abbildung 59 Tabelle Spielerfassung

Wie in der Abbildung 59 zu erkennen ist, besteht der Primärschlüssel der Tabelle Spielerfassung aus den Identifikationsnummern des Schachspiels und der Schachfigur. Diese beiden Schlüssel sind mit der Tabelle Schachspiel und mit der Tabelle Schachfigur verbunden. Die Tabelle dient zur Erkennung, welche Figur auf welcher Position zu Beginn des Spiels steht, deswegen wird die ersterfasste Position und der Zeitpunkt jeder Figur gespeichert.

#	Name	Typ	Kollation	Attribute	Null	Standard	Kommentare	Extra
1	farbe	enum('weiss', 'schwarz')	latin1_german1_ci		Nein	kein(e)		
2	id	int(11)			Nein	kein(e)		AUTO_INCREMENT
3	name	varchar(30)	latin1_german1_ci		Nein	kein(e)		
4	position	enum('A8', 'B8', 'C8', 'D8', 'E8', 'F8', 'G8', 'H8...')	latin1_german1_ci		Nein	kein(e)		
5	code	varchar(30)	latin1_german1_ci		Nein	kein(e)		

Abbildung 60 Tabelle Schachfigur

Wie man anhand Abbildung 60 erkennen kann, werden in der Tabelle Schachfigur die Farbe, der Name, die Position und der Code aller Spielfiguren gespeichert. Der Code ist die Nummer des jeweiligen RFID-Tags.

#	Name	Typ	Kollation	Attribute	Null	Standard
1	spiel_id 🔑	int(11)			Nein	kein(e)
2	zugnummer 🔑	int(11)			Nein	kein(e)
3	spieler_id 🔑 🔗	int(11)			Nein	kein(e)
4	alte_position	varchar(20)	latin1_german1_ci		Nein	kein(e)
5	neue_position	varchar(20)	latin1_german1_ci		Nein	kein(e)
6	figur_id 🔗	int(11)			Nein	kein(e)
7	zeitpunkt	timestamp			Nein	CURRENT_TIMESTAMP

Abbildung 61 Tabelle Spielerfassung

In der Abbildung 61 sind die Attribute der Tabelle Spielzug zu sehen. Die ID des Schachspiels, die Zugnummer und die ID des Spielers bilden dabei den Primärschlüssel. Zusätzlich werden die alte und neue Position und die ID der Schachfigur und der Zeitpunkt gespeichert.

6 WETTBEWERBE

6.1 ALLGEMEIN

Das Team von Chess4U hat sich für einen Wettbewerb sowie einer Förderaktion angemeldet. Für die Förderaktion, genannt Netidee, hat das Team sich angemeldet, um die Projektkosten abzudecken. Dort hat das Team eine Fördersumme von 3000,00€ erhalten. Zusätzlich hat sich das Team für den Sonderpreis „Internet of things“ angemeldet. Chess4U hat sich außerdem dafür entschieden, sich bei Jugend Innovativ in der Kategorie “Engineering 2” anzumelden.

6.2 NETIDEE

Netidee ist eine seit 2006 bestehende Förderaktion, welche von der gemeinnützigen IPA gesponsert wird. Das Ziel der IPA ist die Förderung des Internet in Österreich und des freien und geordneten Zugangs zu dessen Netzen und Diensten unter Wahrung internationaler Verpflichtungen (Internet Foundation Austria (IPA), 2016).

6.2.1 ABLAUF

Für die Förderrate muss man zu Beginn einen ausgefüllten Förderantrag an die Zentrale von Netidee senden. Nach der Bestätigung zur Förderung des Projekts müssen die Teams den Projektverlauf sowie alle Projektkosten dokumentieren und diese an Netidee senden. Die geförderte Geldsumme wird in drei Stufen freigegeben:

- bei der ersten Stufe erhält man 50 % des Förderbetrags
- bei der zweiten Stufe erhält man 30% des Förderbetrags
- bei der dritten Stufe erhält man 20% des Förderbetrags.

Für all diese Etappen müssen zuvor alle Projektkosten und der Projektverlauf an Netidee gesendet werden. Die Projektkosten sollten am Ende des Projektes die Summe der Förderung decken.

6.2.2 VERANSTALTUNGEN:

Jährlich bietet Netidee einen zweitägigen Open-Source Workshop sowie eine eintägige „Best of“ Veranstaltung, bei welcher die besten Projekte des Jahres präsentiert werden. Außerdem werden bei der „Best of“ Veranstaltung zwei Sonderpreise für die Kategorien „Internet of things“ und „Internet for refugees“ verliehen. Chess4U wurde für die Kategorie „Internet of things“ nominiert.

6.3 JUGEND INNOVATIV

Jugend Innovativ ist ein seit dem Jahre 1987 jährlich stattfindender Wettbewerb, bei welchem innovative Projektideen von Schülerinnen und Schülern mit Preisgeldern und Preisen ausgezeichnet wird. Das Team von Jugend Innovativ unterstützt die Schülerinnen und Schüler von der Idee bis zur Realisierung ihrer Projekte. Jugend Innovativ unterteilt im Jahr 2017 ihre Preise in den Kategorien Engineering, Design, Science, Young Entrepreneurs und Sustainability. Chess4U hat, sich für die Kategorie Engineering 2 angemeldet und wurde innerhalb von zwei Monaten für die Teilnahme bestätigt. (Austria Wirtschaftsservice Gesellschaft mbH, kein Datum)

6.4 LIZENZEN

Netidee fordert eine vollständige Veröffentlichung der Ergebnisse (Inhalte, Sourcecode, sowie Dokumentation) für jedes geförderte Projekt.

Für die Inhalte müssen Creative Common Lizenzen¹⁰ verwendet werden. (IPA, 2016) Derzeit gibt es sechs Abstufungen von Creative Common Lizenzen, welche angeben, unter welchen rechtlichen Bedingungen ein Werk genutzt werden darf.

Chess4U hat die Lizenz CC-BY-0.3 AT in ihrer Dokumentation und ihren Projektbericht verwendet. Mit der CC-BY-0.3 AT Lizenz kann das Werk verbreitet, vervielfältigt, bearbeitet, abgewandelt und sogar kommerziell genutzt werden. (Duchamps, kein Datum)

¹⁰ <https://creativecommons.org>

Dem Sourcecode des Projekts wurde eine General Public Lizenz (GNU GPL -3.0) vergeben. Mit dieser Lizenzform können Dritte den Sourcecode der Software ausführen, studieren, modifizieren und verbreiten. Software, welche diese Rechte gewährt, nennt man Freie Software. Bei der Weitergabe von Software, welche die GPL Lizenz enthält, müssen die Lizenzen beibehalten werden (Copyleft). ("Trustable", 2017)

7 MARKETING

7.1 ALLGEMEIN



Abbildung 62 Rollup-Netidee



Abbildung 63 Plakat- Chess4U

Um das Projekt zu vermarkten, wurde es mittels Social Media, einer Webseite und einem Werbevideo beworben. Außerdem wurden für bestimmte Veranstaltungen, wie dem Tag der offenen Tür der HTL Ottakring, oder der „Best of“ Veranstaltung von Netidee-Plakate erstellt. Abbildung 62 zeigt das für Netidee erstellte Rollup und Abbildung 63 zeigt das für den Tag der offenen Tür erstellte Plakat.

7.2 SOCIAL MEDIA

Eine Facebook- Seite¹¹, welche den Projektverlauf und interessante Ereignisse des Projekts dokumentieren, wurde mit dem Namen „Chess4U“ erstellt. Auf der Seite werden Werbevideos, Veranstaltungen und Projektfortschritte in Form von Posts und Fotos veröffentlicht.

7.3 Webseite

Die Webseite¹² dient in erster Linie dafür, mehr über das Vorhaben und das Team zu erfahren. Außerdem werden auf der Webseite die Dokumentation und die Software am Ende des Projektverlaufes zur Verfügung gestellt.

7.4 WERBEVIDEO

Ein Werbevideo wurde zur Veranschaulichung der Idee von Chess4U erstellt, um der Förderorganisation von Netidee sowie anderen Interessenten einen Einblick über Netidee zu geben.

¹¹ www.facebook.com/chess4u zul. geladen am: 25.03.2017

¹² chess4u.at zul. geladen am: 25.03.2017

7.4.1 SCRIPT

Vid1: Sc1 (Intro)**Aud1: Angenehme Musik leise über das ganze Video**

Schon mal vom Chess4U Schachbrett gehört?

Vid 2: Sc3 Sc4 Sc5

Aud3: Das intelligente Schachbrett, welches mittels Sensoren Spielzüge aufzeichnen kann und diese auf einem GUI ausgeben. Dies geschieht mittels RFID Tags und 64 Antennen, welche unterhalb der Spielfiguren und des Schachbretts integriert werden. Die Spielzüge werden dann auf einem benutzerfreundlichen GUI ausgegeben und sind außerdem als Protokoll abrufbar.

Vid3: Outro**Aud3:** Chess4u - die Zukunft des Schachspiels

Abbildung 64 Script, Werbevideo

7.4.2 STORYBOARD

	<p>Scene Start: _____</p> <p>Duration: _____</p> <p>Figur und Antenne wird gezeigt</p>		<p>Scene Start: _____</p> <p>Duration: _____</p>
	<p>Scene Start: _____</p> <p>Duration: _____</p> <p>Zug wird gemacht Felder leuchten auf</p>		<p>Scene Start: _____</p> <p>Duration: _____</p>
	<p>Scene Start: _____</p> <p>Duration: _____</p> <p>Schachbrett: Schwenk über das Schachbrett</p>		<p>Scene Start: _____</p> <p>Duration: _____</p> <p>Outro</p>
	<p>Scene Start: _____</p> <p>Duration: _____</p>		<p>Scene Start: _____</p> <p>Duration: _____</p> <p>GUI wird gezeigt</p>

From www.IndependentMusicAdvice.com

Abbildung 65 Storyboard, Werbevideo

7.4.3 IDEE

Das Werbevideo soll nicht nur das Produkt bewerben, sondern auch einen groben Überblick über die Idee und die technische Umsetzung der Diplomarbeit von Chess4U vermitteln.

7.4.4 BEARBEITUNG DES FOOTAGE

Das Werbevideo wurde mit zwei Programmen, After Effects und Adobe Premiere, geschnitten und bearbeitet.

7.4.4.1 AFTER EFFECTS

After Effects ist eine Software zum Animieren von Grafiken sowie für Digital Composing. Mit der Software wurden die visuellen Effekte und Animationen im Werbevideo erstellt, welche hier unten angeführt werden.

7.4.4.1.1 STABILISIERUNG DER AUFNAHMEN

Mittels einem Plug-In werden mehreren Aufnahmen, welche mit freier Hand gefilmt wurden, stabilisiert. Dies wird mit dem Effekt "Verkrümmungsstabilisierung VFX"(engl. warp stabilizer) in der Animationsleiste behoben. Der Effekt bewirkt, dass Bewegungen mehrerer Kameras aufeinander abgestimmt werden und somit der Hintergrund ruckelfrei wird.

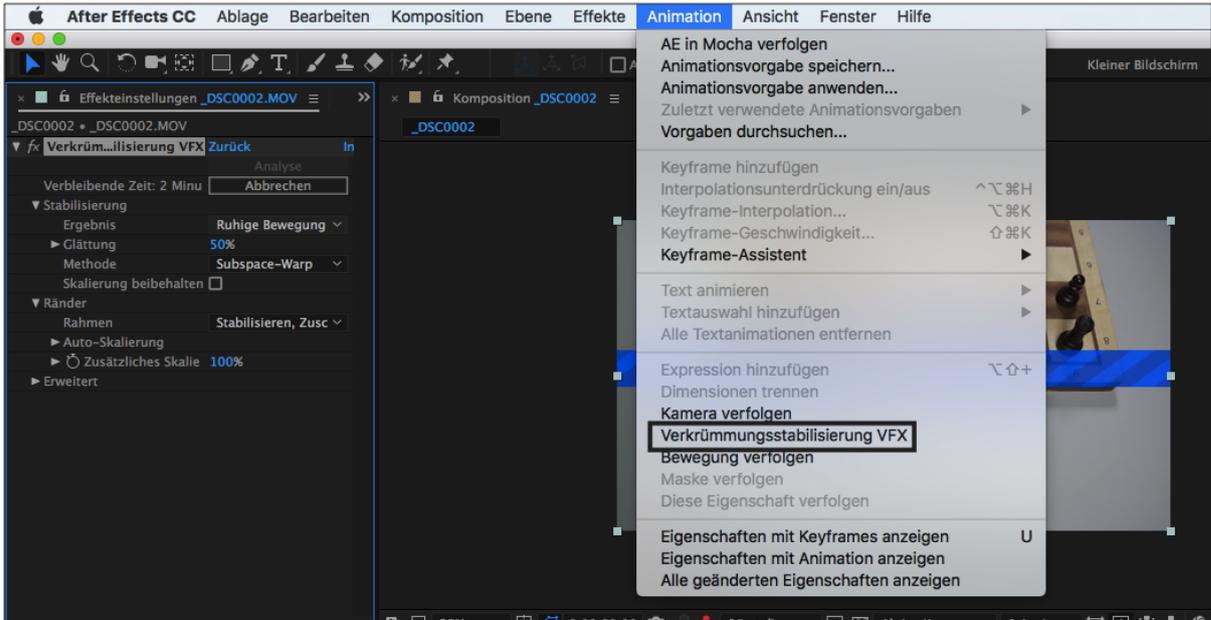


Abbildung 66 Stabilisierung, After Effects

7.4.4.1.2 INTRO & OUTRO

Für die Erstellung des Intros wurden mehrere Schritte mit den Programmen After Effects und Adobe Photoshop benötigt. Zuallererst wurde der Text vom Logo mittels Photoshop entnommen. Mit dem magnetischen Lasso wurde der Text markiert und freigestellt. Damit die Transparenz nicht verloren geht wurde das File als .PNG abgespeichert.



Abbildung 67 Chess4U-Logo freigestellt

Zuerst wurde das Logo freigestellt, siehe Abbildung 67. Danach wurde dazu ein passender Hintergrund für das Intro gesucht. In diesem Fall war es ein Schachbrett aus den Aufnahmen, die gemacht wurden. Beide Dateien, das freigestellte Logo sowie das Bild vom Schachbrett, wurden daraufhin in ein neu erstelltes Projekt in After Effects hineingeladen und es wurde eine Komposition erstellt.

Die Idee war es, den Hintergrund unscharf zu machen. Realisiert wurde dies mit den Gaußschen Weichzeichner. Diesen findet man unter Effekte → Weich und Scharfzeichnen → Gaußscher Weichzeichner.

Um den unscharfen beziehungsweise verschwommenen Effekt zu bekommen, wurde die Einstellung „Weichzeichnung“ von 0 auf 50,9 geändert. Zusätzlich wurde die Master- Sättigung auf -66 gesenkt, da die Farbe in dieser Stärke nicht zum Logo gepasst hätte.

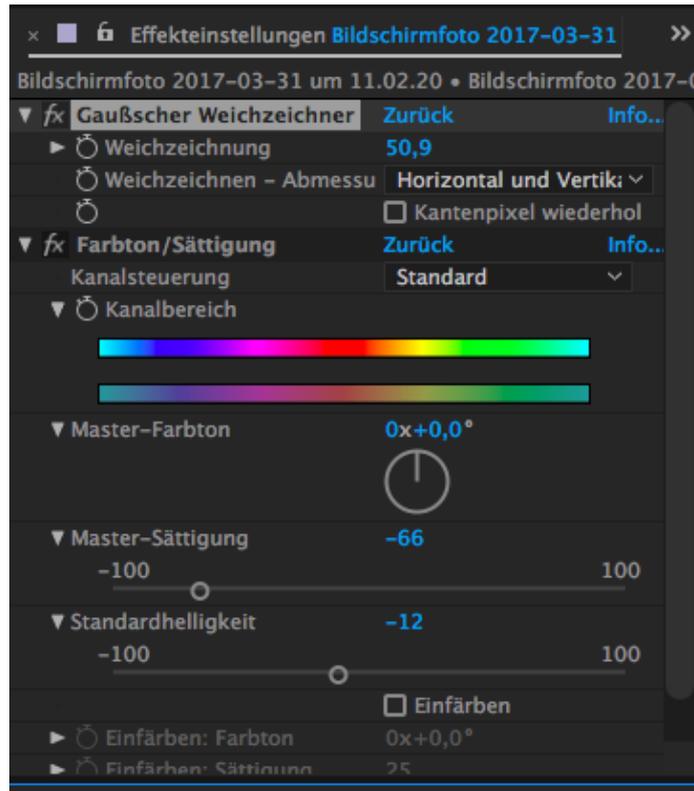


Abbildung 68 Gaußscher Weichzeichner (After Effects)

Der letzte Schritt war der Einbau von Keyframes in das Projekt. Keyframes legen einen Zeitpunkt fest, an welchem ein Wert für eine Ebene-eigenschaft geändert wird. Der Wert dazwischen wird interpoliert. (Adobe, 2016)

Sie wurden eingebaut, um eine kleine Animation zu erstellen. Die Animation beinhaltet mehrere Keyframes mit unterschiedlichen Deckkräften, damit der Text im Intro langsam eingeblendet und wieder ausgeblendet wird.

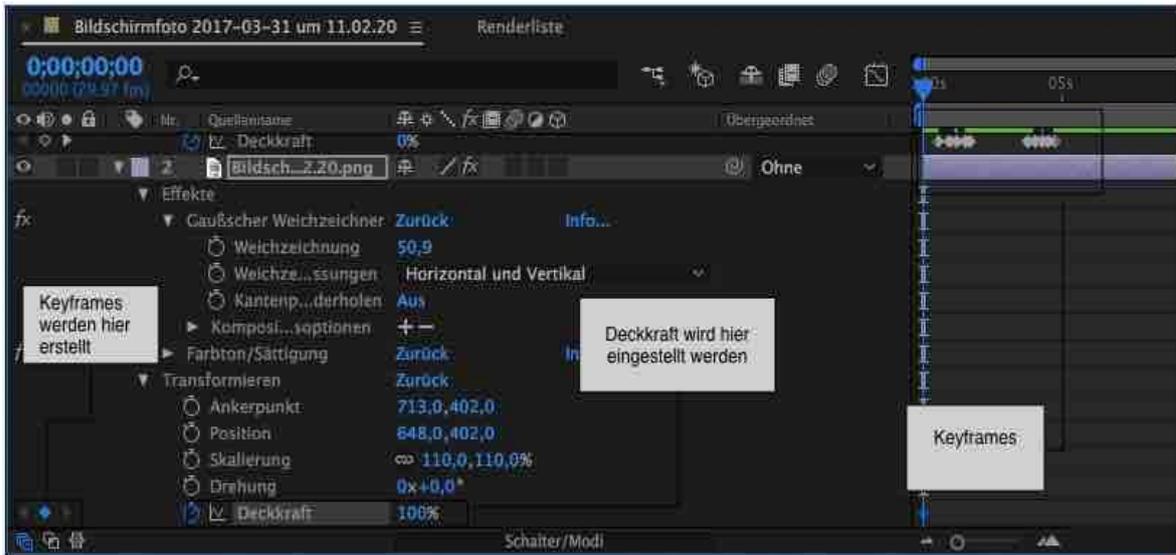


Abbildung 69 Keyframes (After Effects)

Nachdem all diese Schritte getätigt wurden, war das Intro fertig animiert.



Abbildung 70 fertiges Intro bzw. Outro

7.4.4.1.3 SPIELZUG MIT GLOW-EFFEKT

Für das Werbevideo wurde auch eine Aufnahme gemacht, welche einen Spielzug zeigt. In dieser Aufnahme sollte die RFID-Technik hinter dem Projekt grafisch dargestellt werden, um so einen besseren Einblick ins Projekt zu bekommen. Diese Szene ist mit den Programmen After Effects und Photoshop erstellt worden.



Abbildung 71 Spielzug-Aufnahme

Bevor die Aufnahme mit After Effects animiert wurde, wurde eine RFID-Schleife aus dem Internet heruntergeladen und mittels Photoshop (Schnellauswahlwerkzeug→ Ebene durch kopieren) freigestellt. Außerdem wurden die Farben der freigestellten RFID-Schleife verändert.

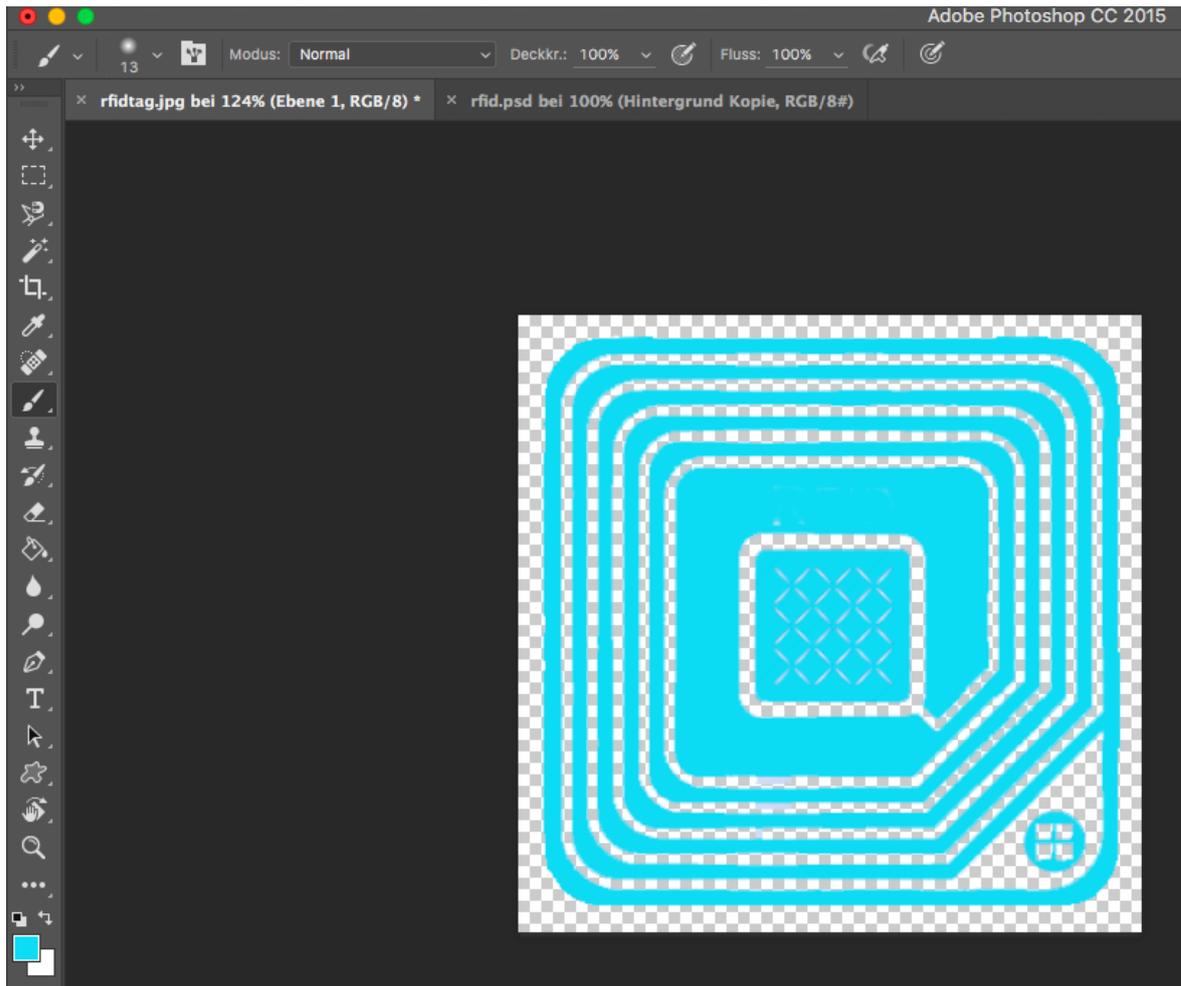


Abbildung 72 RFID- Schleife freigestellt

Damit die RFID-Schleifen später nicht die Figuren verdecken, wurden auch ein Screenshot der Spielzugaufnahme gemacht und bei diesem die Spielfiguren in Photoshop freigestellt.

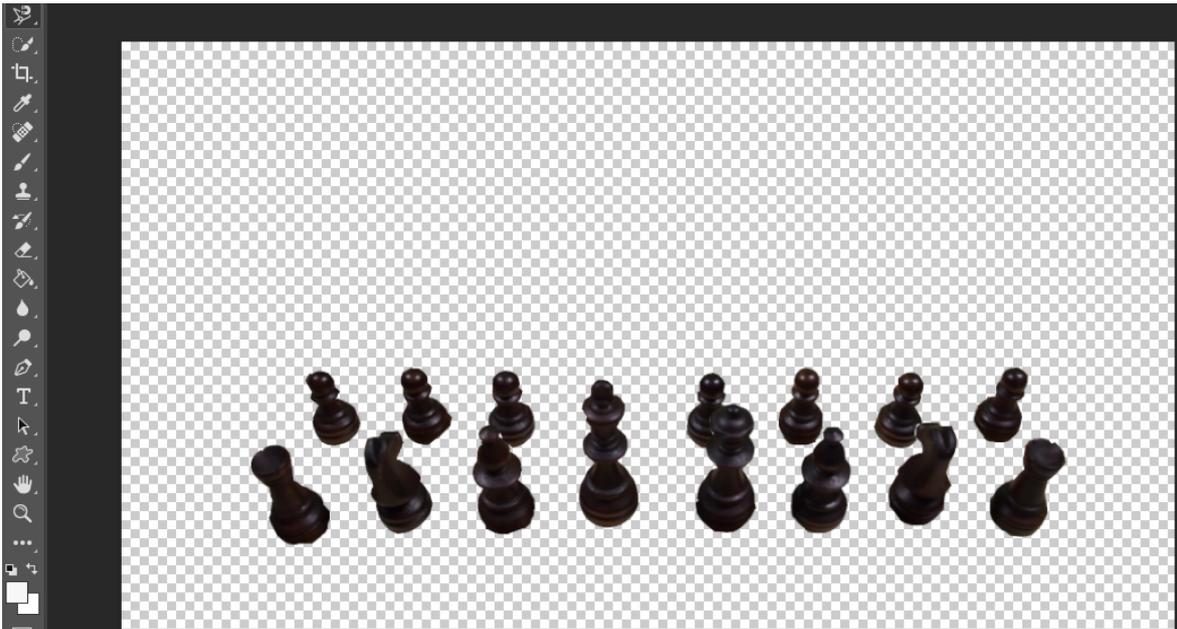


Abbildung 73 Figuren freigestellt

Die Dateien wurde abgespeichert und es wurde ein neues Projekt in After Effects erstellt, in welchem die freigestellten Bilder und die Aufnahme (RFID-Schleife, Spielzugaufnahme und die freigestellten Figuren) hineingeladen wurden.

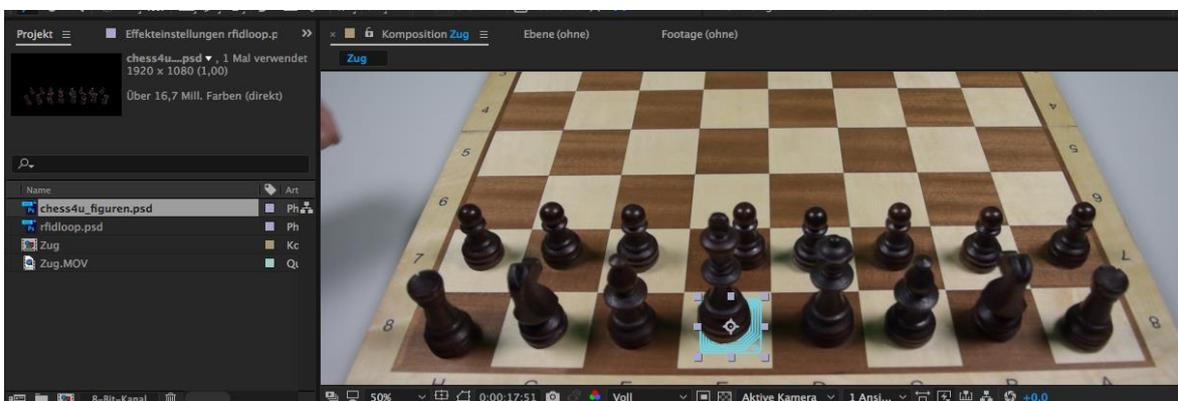


Abbildung 74 Figuren über RFID Schleifen

Ziel war es von Anfang an, die RFID-Schleife zum Leuchten zu bringen. Dies wurde mit dem Effekt „Leuchten“ (engl. Glow) gemacht.

Damit jedes Feld der ersten zwei Reihen eine RFID Schleife bekommt, wurde diese öfters dupliziert. Nachdem jedes gewünschte Feld eine Schleife bekommen hat, wurden die Schleifen mit dem Effekt „Eckpunkte verschieben“ (engl. Corner pin) so angepasst, dass ihre Fläche perfekt auf ihr eigenes Feld passt.

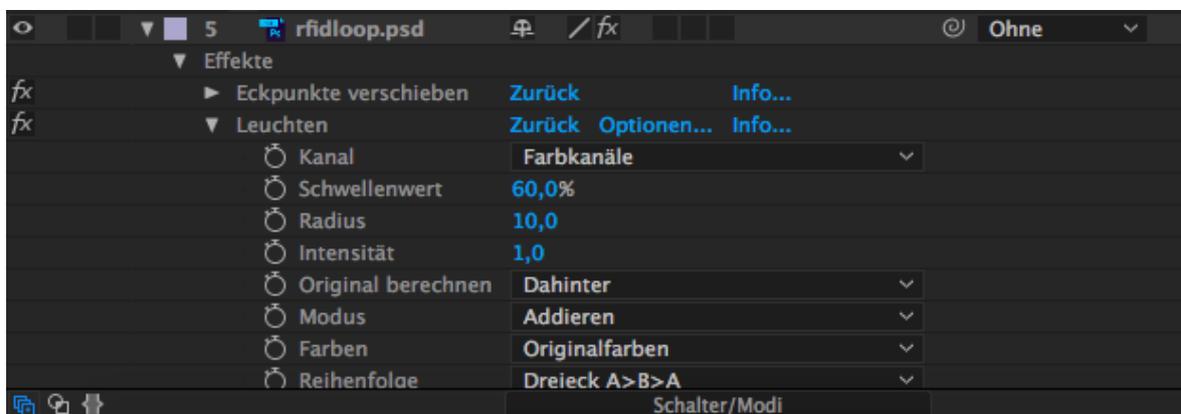


Abbildung 75 Effekte

Damit die Spielfigur am Ende des Zuges nicht auch verdeckt wird, wurde wieder ein Screenshot gemacht, dieses Mal vom Ausgangspunkt des Zuges, und die Figur wurde freigestellt.



Abbildung 76 RFID - Schleife über Figur

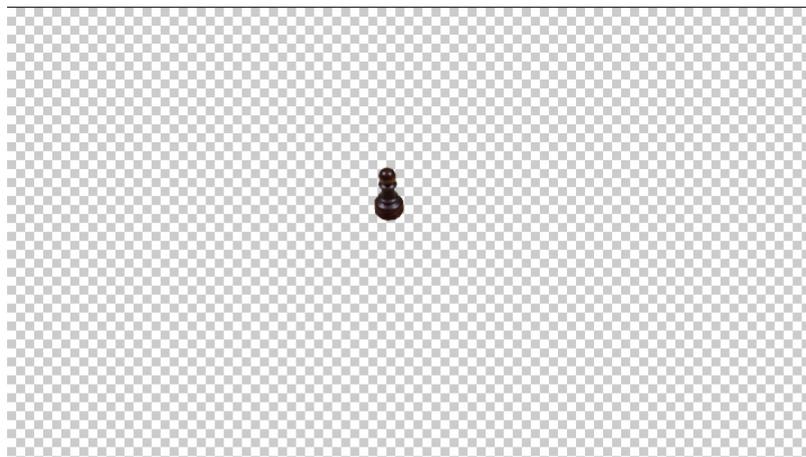


Abbildung 77 Spielzugfigur freigestellt

Die freigestellte Figur wurde in die Komposition hinzugefügt und positioniert.

Damit die Figur nicht von Anfang an sichtbar ist, wurden Keyframes hinzugefügt, sodass sie erst auftaucht, nachdem die Figur auf der Schleife steht.

Keyframes wurden auch benützt, um die Schleifen zu bestimmten Zeiten aufleuchten zu lassen und somit eine schöne Animation entsteht.

QUELLENVERZEICHNIS

"Trustable", B., 2017. *Wikipedia*. [Online]

Available at: https://de.wikipedia.org/wiki/GNU_General_Public_License

Adobe, 2016. *Adobe*. [Online]

Available at: <https://helpx.adobe.com/de/after-effects/using/animation-basics.html>

AGILLOX GmbH, 2017. <http://www.agillox.com>. [Online]

Available at: <http://www.agillox.com/de/was-ist-rfid/>

Austria Wirtschaftsservice Gesellschaft mbH, n.d. *Jugend Innovativ*. [Online]

Available at: <http://www.jugendinnovativ.at/Wettbewerb/Wettbewerb.html>

Boden, C., 2013. *Ratchet - What is a WebSocket?*. [Online]

Available at: <http://socketo.me/docs/>

[Accessed 28 02 2017].

Computer CadSoft, GmbH, 2016. *Wikipedia*. [Online]

Available at: [https://de.wikipedia.org/wiki/Eagle_\(Software\)](https://de.wikipedia.org/wiki/Eagle_(Software))

Duchamps, D., n.d. *CC your EDU*. [Online]

Available at: <http://www.cc-your-edu.de/die-cc-idee/die-cc-lizenzen/>

Elektronik-Kompendium.de, 2017. *elektronik-kompendium.de*. [Online]

Available at: <http://www.elektronik-kompendium.de/sites/kom/0902021.htm>

Fakir, S., 2017. <http://www.fakir.it>. [Online]

Available at: <http://www.fakir.it/unterschied-rfid-und-nfc-eine-kurze-erklaerung/>

Hermann, G., 2016. *Wikipedia*. [Online]

Available at: <https://de.wikipedia.org/wiki/Leiterplatte>

hrvojeheavy, 2013. *ubuntuusers*. [Online]

Available at: <https://wiki.ubuntuusers.de/CMake/>

i-keys.de, 2016. *i-keys.de*. [Online]

Available at: <https://www.i-keys.de/de/Chip-Codes.html>

Instruments, T., 2016. [Online].

Internet Foundation Austria (IPA), 2016. *Netidee*. [Online]

Available at: <https://www.netidee.at/die-internet-foundation-austria-ipa/missionstatement/>

IPA, 2016. *Netidee*. [Online]

Available at: https://www.netidee.at/fileadmin/www.netidee.at/documents/2016/20160411_RichtlinienBedingungen_Call11.pdf

Kiwi Electronics B.V., 2017. *www.kiwi-electronics.nl*. [Online]

Available at: <https://www.kiwi-electronics.nl/explore-nfc-ww-voor-raspberry-pi?lang=en>

Kupferlackdraht, Wikipedia, 2017. *Kupferlackdraht*. [Online]

Available at: <https://de.wikipedia.org/wiki/Kupferlackdraht>

Lipinski, K., 2009. *Relationales Datenbankmanagementsystem*. [Online]

Available at: <http://www.itwissen.info/RDBMS-relational-database-management-system-Relationales-Datenbankmanagementsystem.html>

[Accessed 18 03 2017].

Lipinski, K., 2011. *MySQL*. [Online]

Available at: <http://www.itwissen.info/MySQL.html>

[Accessed 18 03 2017].

Lipinski, K., 2012. *Datenbankmanagementsystem*. [Online]

Available at: <http://www.itwissen.info/DBMS-database-management-system-Datenbankmanagementsystem.html>

[Accessed 16 03 2017].

Lipinski, K., 2012. *Entity Relationship Diagram*. [Online]

Available at: <http://www.itwissen.info/ERD-entity-relationship-diagram.html>

[Accessed 18 03 2017].

Lipinski, K., 2016. *Datenbank*. [Online]

Available at: <http://www.itwissen.info/Datenbank-database-DB.html>

[Accessed 15 03 2017].

Lipinski, K., 2016. *Relationale Datenbank*. [Online]

Available at: <http://www.itwissen.info/Relationale-Datenbank-database-relational.html>

[Accessed 15 03 2017].

MakerDev, 2016. *http://osdevlab.blogspot.co.at*. [Online]

Available at: <http://osdevlab.blogspot.co.at/2015/12/how-to-install-latest-cmake-for.html>

Matt, 2012. *Simple Guide to the RPi GPIO Header and Pins*. [Online]

Available at: <http://www.raspberrypi-spy.co.uk/2012/06/simple-guide-to-the-rpi-gpio-header-and-pins/>

[Accessed 30 03 2016].

Multiplexer, Wikipedia, 2016. *Wikipedia*. [Online]

Available at: <https://de.wikipedia.org/wiki/Multiplexer>

NXP Semiconductors N.V., 2016. *NXP*. [Online]

Available at: http://www.nxp.com/documents/application_note/AN11480.pdf

raspberrypi.tips, 2014. *Raspberry.Tips*. [Online]

Available at: <https://raspberrypi.tips/faq/raspberry-pi-spi-und-i2c-aktivieren/>

reichelt elektronik GmbH & Co. KG, 2017. *www.reichelt.de*. [Online]

Available at: https://cdn-reichelt.de/bilder/web/xxl_ws/A300/RASP_EXPLORE_NFC_03.png

RFID#Speicherkapazität, Wikipedia, 2017. *RFID#Speicherkapazität*. [Online]

Available at: <https://de.wikipedia.org/wiki/RFID#Speicherkapazit.C3.A4t>

Schnellschach, Wikipedia, 2017. *Schnellschach*. [Online]

Available at: <https://de.wikipedia.org/wiki/Schnellschach>

Söldner, M., 2013. *PC Welt*. [Online]

Available at:

http://www.pcwelt.de/ratgeber/Die_besten_Verwendungsmoeglichkeiten_fuer_Raspberry_Pi-Mini-PC-7663628.html

STMicroelectronics, 2016. *st.com*. [Online]

Available at: http://www.st.com/content/ccc/resource/technical/document/application_note/d9/29/ad/cc/04/7c/4c/1e/CD00221490.pdf/files/CD00221490.pdf/jcr:content/translations/en.CD00221490.pdf

TAGnology RFID GmbH, 2016. [Online]

Available at: <http://www.tagnology.com>

tec-trends.de, 2016. [Online]

Available at: <http://www.tec-trends.de/hardware/rfid-im-alltag-was-verbirgt-sich-dahinter-und-wo-wird-es-eingesetzt/>

Transponder, Wikipedia, 2017. *Transponder*. [Online]

Available at: <https://de.wikipedia.org/wiki/Transponder>

Twitter Inc., 2011. *Bootstrap*. [Online]

Available at: <http://getbootstrap.com/>

[Accessed 29 03 2017].

Vitale(svvitale), S., 2017. *Github*. [Online]

Available at: <https://github.com/svvitale/nxppy>

ANHANG

Arbeitsaufteilung

Person	Folgende Punkte des Diplomarbeitshandbuches wurden von folgenden Personen geschrieben, inklusive aller Unterpunkte
Sedlak Gregor	<ul style="list-style-type: none"> -7 Marketing -7.1 Allgemein -7.2 Social Media -7.3 Webseite -7.4 Werbevideo -7.4.1 Script -7.4.2 Storyboard -7.4.3 Idee -7.4.4 Bearbeitung des Footage -7.4.4.1 After Effects -7.4.4.1.1 Stabilisierung der Aufnahmen -7.4.4.1.2 Intro & Outro -7.4.4.1.3 Spielzug mit Glow-Effekt
Drekovic Denis	<ul style="list-style-type: none"> -Kurzfassung -Abstract -Problemstellung -Ehrenwörtliche Erklärung -Vorwort -1.1 RFID-Technologien -1.1.2 Funktion der RFID-Technologie -1.1.3 Arten von Transpondern -1.1.4 Frequenzbereiche -1.1.4.1 LF – Low Frequency -1.1.4.2 HF – High Frequency -1.1.4.3 UHF – Ultra High Frequency -1.1.4.4 SUHF – Ultra High Frequency

	<ul style="list-style-type: none"> -1.1.5 NFC -2 Hardwareplattform -2.1 Raspberry PI -2.1.1 Raspberry Software -2.1.1 Raspberry Grundkonfiguration -2.1.2.1 Konfiguration der NFC-Erweiterungsplatine -2.1.2.2 NXPPY Python-Wrapper -2.1.2.3 Erweiterungsmodul bearbeiten -2.1.2.3.1 Antenne(n) erstellen -3.1.1 PCB mit Reader und Antennen verbinden - 4.1 Antennen Switch
Kirchknopf Theodor	<ul style="list-style-type: none"> -2.2 Multiplexer -2.2.1 Einleitung Multiplexer -2.2.2 Allgemeine Erklärung -2.2.3 Spezifische Erklärung -2.3 Leiterplatte -2.3.1 Einleitung Leiterplatte -2.3.2 Allgemeine Erklärung -2.3.3 Spezifische Erklärung -2.3.4 Eagle 7.7.0 (Software) -3 Löten am PCB-Layout
Lewitsch Raphael	<ul style="list-style-type: none"> -4.2 Websockets -4.3 Websocket Chess4U Server -4.4 Websocket Chess4U Client -4.5 Die Benutzeroberfläche -5 Datenbank -5.1 Allgemein -5.2 Relationale Datenbank -5.3 Datenbankmanagementsystem -5.4 Relationales Datenbankmanagementsystem

	<ul style="list-style-type: none">-5.5 MySQL-5.6 Entity Relationship Diagram-Tabellen – Chess4U
--	---

Auszug aus dem Diplomarbeitsantrag

PROJEKTIDEE

AUSGANGSSITUATION

Das Projekt soll die altbekannte Methode der Spielzugaufzeichnung auf Papier ersetzen. Dieses Projekt soll das Problem einer versäumten Spielzugaufzeichnung in der Schach Welt behandeln. Ein Schachbrett welches mittels Sensoren, im Schachbrett und an Figuren, Spielzüge aufzeichnen kann. Auf Anfrage sollen besagte Spielzüge angezeigt werden können. Ein Werbevideo mit ansprechenden Animationen soll das Projekt visuell veranschaulichen.

Mittel Sensoren werden Spielzüge auf einem Schachbrett aufgezeichnet, welche dann auf Anfrage angezeigt werden können. Mit dieser Technik müssen professionelle Schachspieler ihre Spielzüge nicht mehr per Hand notieren, da dies nun automatisch gemacht wird. Auch wenn es die Technik schon gibt, ist diese Variante günstiger und somit auch für den Massenmarkt geeignet.

PROJEKTZIELE

MUSS ZIELE

SPIELZUG AUFZEICHNUNG

Jeder Spielzug wird mittels RFID- Tags und den umgebenden Antennen eindeutig identifiziert und auf einem GUI ausgegeben bzw. aufgezeichnet

OPTIONALE ZIELE (SOLL, KANN ZIELE)

Falls ein Zug gemacht wird, welcher den Spielregeln nicht entspricht, wird eine Fehlermeldung angezeigt. Z.B: Ein Bauer geht drei Felder → Fehlermeldung

NICHT ZIELE

Die Vermarktung und ebenso der Verkauf der selbst umgebauten und programmierten Schachbretter ist ein absolutes NICHT Ziel von Chess4U.

PROJEKTORGANISATION

PROJEKTTEAM

Funktion	Name	Kürzel	E-Mail
Projektleiter	Gregor Sedlak	SEDG	g.sedlak98@htl-ottakring.ac.at
Stellvertretender Projektleiter	Denis Drekovic	DRED	d.drekovic97@htl-ottakring.ac.at
Projektmitarbeiter	Theodor Kirchknopf	KIRT	t.kirchknopf96@htl-ottakring.ac.at
Projektmitarbeiter	Raphael Lewitsch	LEWR	r.lewitsch98@htl-ottakring.ac.at

8.3.2 INDIVIDUELLE AUFGABENSTELLUNG

Gregor Sedlak - Projektmanagement, Dokumentation, Webseite (Design), RFID Antennen, Produktvideo (Filmen), Antennen-Switch

Denis Drekovic - Webseite, Produktvideo (Schneiden & Filmen), Stv. Projektmanagement, Programmierung

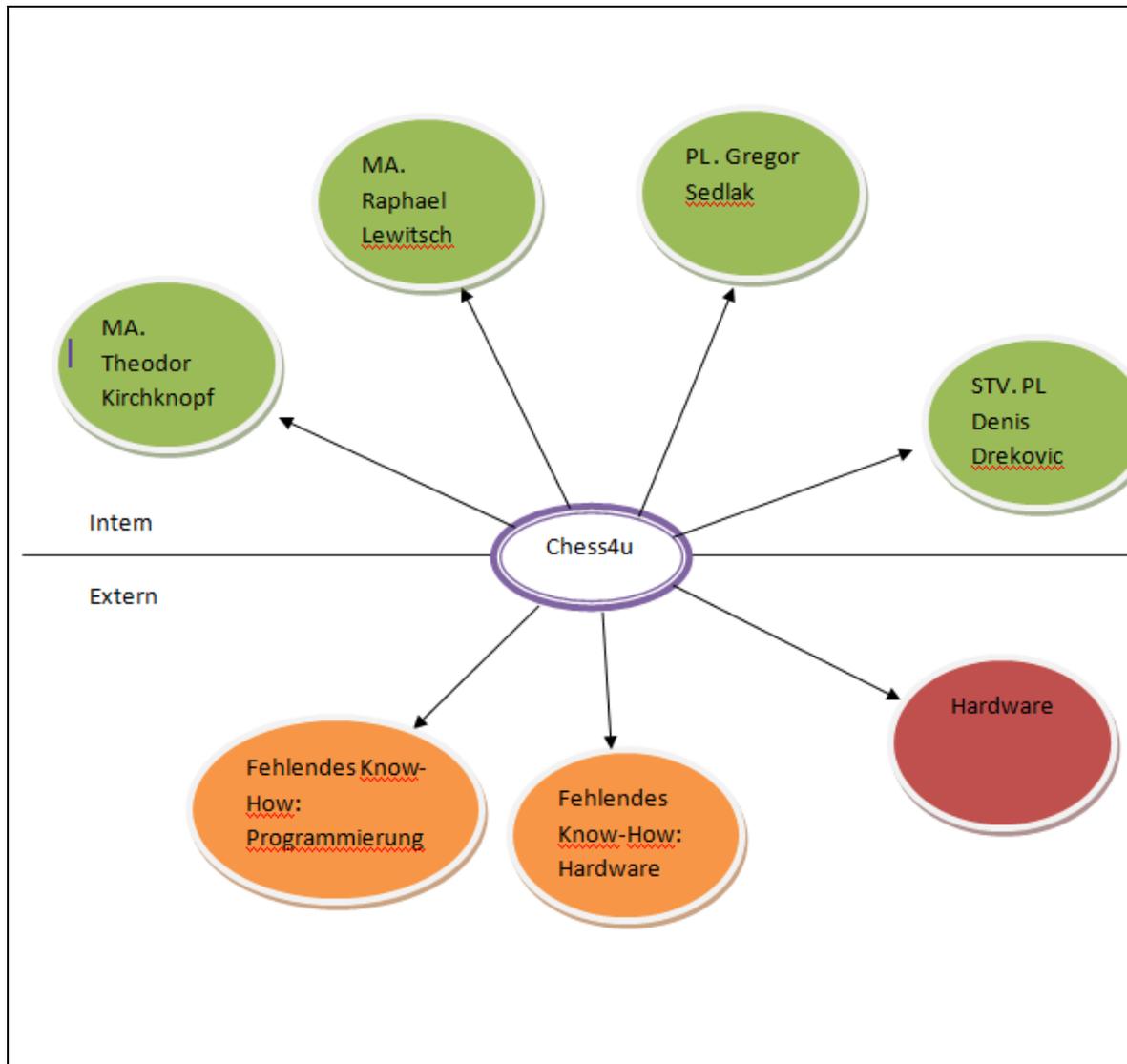
Raphael Lewitsch - Programmierung (Verantwortlicher), Antennen-Switch, Netzwerkanbindung, RFID Mesh

Theodor Kirchknopf - Präsentation, Dokumentation, RFID Antennen, RFID Mesh Vorname, Nachname, Funktion und detaillierte Aufgabenstellung

Theodore Kirchknopf - Präsentation, Dokumentation, RFID Antennen, RFID Mesh

PROJEKTUMWELTANALYSE

GRAFISCHE DARSTELLUNG



BESCHREIBUNG DER WICHTIGSTEN UMWELTEN

#	Bezeichnung	Beschreibung	Bewertung
1	Hardware	Hardware könnte nicht funktionieren oder kaputt gehen	-
2	Fehlendes Know-How: Hardware	Das fehlende Wissen über das Benutzen der Hardware muss angesammelt werden	~
3	Fehlendes Know-How: Hochfrequenztechnik	Falls Fehler im Programm entstehen	~
4	Lieferanten	Lieferanten liefern zu spät	~
5	Werkstatt	Werkstatt zum Bauen der einzelnen Teile	~

RISIKOANALYSE

BESCHREIBUNG DER WICHTIGSTEN RISIKEN

#	Bezeichnung	Beschreibung des Risikos	P	A	RF
1	Ausfall von Mitarbeiter	Wenn ein Mitarbeiter krank wird für eine längere Zeit oder einfach öfter fehlt.	30	40	1200
2	Hardware wird kaputt	Die Hardware wird während des Arbeitens beschädigt (zb. Antenne, Reader...)	10	20	200
3	Know-How: Hochfrequenztechnik	Fehlendes Wissen über HFT	60	30	1800
4	Know-How: Hardware	Das fehlende Wissen über das Benutzen der Hardware kann zu Zeitverlust führen	30	40	1200
5	Lieferanten	Lieferanten liefern zu spät oder bei der Lieferung fehlt etwas	40	80	1200
6	Werkstatt	Werkstatt könnte nicht zugänglich sein	20	40	400

Alle negativen Einflüsse der Umfeldanalyse kommen in die Risikoanalyse.

P...Eintrittswahrscheinlichkeit des Risikos

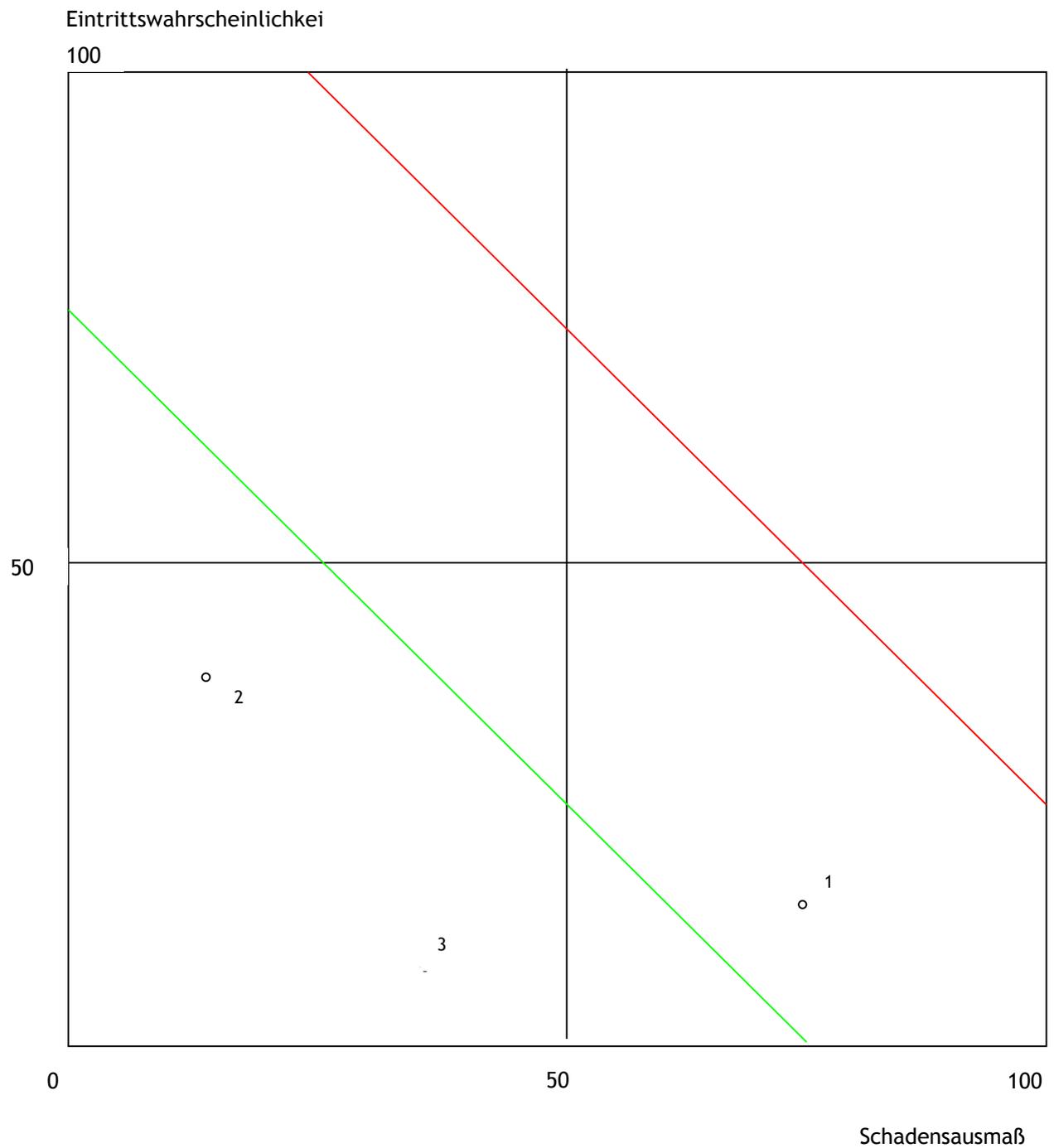
A...Schadensausmaß bei Eintritt des Risikos

RF...berechneter Risikofaktor

Die Tabelle wird im Anschluss nach absteigendem Risikofaktor geordnet.

Danach werden die Risiken aus der Analyse in das Portfolio übertragen und A / B / C Risiken identifiziert.

RISIKOPORTFOLIO



RISIKO GEGENMAßNAHMEN

#	Bezeichnung	Gegenmaßnahme
1	Ausfall von Mitarbeiter	Falls der Mitarbeiter grundlos nicht auftaucht, wird der Projektleiter ein ernstes Gespräch mit ihm führen
2	Hardware wird kaputt	Falls ein Gerät während des Arbeitens kaputt geht wird ein Mitarbeiter dafür sorgen das dieses neu bestellt wird
3	Know-How: HF-Technik	Wenn fehlendes Wissen zum Problem wird und man sich ein MA selbst nicht weiterhelfen kann, dann wird dieser einen passenden Lehrer/ Schueler, welcher sich in der HF- Technik auskennt, suchen und diesem um Hilfe bitten.
4	Know-How: Hardware	Fehlendes Wissen über die Hardware wird mit intensiver Recherche und ständiger Nachfrage bei Professoren behoben.
5	Lieferanten	Der Versand wird telefonisch oder per Email über das Problem informiert.
6	Werkstatt	Damit keine Missverständnisse mit anderen Schülern aufkommen werden für Räume freie Termine gewählt und dann mit einem zuständigen Werkstätten-Lehrer fixiert.

Für die höchsten Risiken werden ganz konkrete Gegenmaßnahmen geplant - einerseits um die Eintrittswahrscheinlichkeit zu verringern, andererseits um das Schadensausmaß zu verringern.

MEILENSTEINLISTE

Darstellung der Meilensteine mit geschätzten Terminen

Datum	Meilenstein
11.10.2016	Funktionsfähige RFID-Antennen
12.10.2016	RFID-Mash
13.10.2016	Webseite
26.10.2016	Antennen-Switch
20.11.2016	Netzwerkanbindung
15.12.2016	Produktvideo

KOSTENABSCHÄTZUNG

Abschätzung der Kosten des Projekts

#	Beschreibung der Kostenursache	Kosten
1	KKMOON BERÜHRUNGSLOSE 14443A KARTE ENCODER IC KARTENLESER SCHREIBER KARTENLESEGERÄT MIT 5PCS KARTEN 5PCS SCHLÜSSELANHÄNGER USB SCHNITTSTELLE 13.56MHZ RFID	Ca. 25€
2	125 kHz RFID Reader	Ca. 20€
3	125 kHz Tags	Ca. 15€
4	Schachbrett	Ca. 50€
5	div. Elektronik Bauteile	Ca. 50€
6	Raspberry Pi	Ca. 50€
	3* SparkFun Mux Shield 2	Ca. 90€
SUMME		Ca. 300€

FINANZIERUNG

Abschätzung der Kosten des Projekts

#	Beschreibung der Kostenursache	Kosten
1	13.56MHZ RFID Reader	Ca. 25€
2	125 kHz RFID Reader	Ca. 20€
3	125 kHz Tags	Ca. 15€
4	Schachbrett	Ca. 50€
5	div. Elektronik Bauteile	Ca. 50€
6	Raspberry Pi	Ca. 50€
7	3* SparkFun Mux Shield 2	Ca. 90€
SUMME		Ca. 300€

FINANZIERUNG

Wie werden die Kosten finanziert?

Die Schule kommt für den Großteil der Kosten auf. Kleine Dinge, welche in den Ferien benötigt werden, besorgen wir Privat

MOTIVATION

Gregor Sedlak

Als ich von dieser Idee Wind bekommen habe, war mir gleich klar, dass dies ein Projekt für mich ist, da ich selbst öfters Schach mit Freunden und Verwandten spiele. Auch angetan hat mir das Ergebnis des Projektes, da es eine Lösung für ein Problem ist, welches seit langem bekannt ist.

Denis Drekovic

Meine Motivation für dieses Projekt fängt schon bei der technischen Umsetzung an, denn wir als Team möchten eine RFID-Technologie in ein handelsübliches Schachbrett integrieren, welche mittels RFID-Tags die Schachfiguren eindeutig identifizieren und mit jedem Schachzug aufzeichnen bzw. in Echtzeit anzeigen soll. Ein ähnliches Prinzip gibt es zwar schon, jedoch ist dieses veraltet, da es auf Resonanzspulen basiert und somit auch sehr anfällig auf Störungen ist. Deshalb motiviert es mich umso mehr, bei unserem Projekt eine neuere Technologie zu implementieren. Darüber hinaus macht es Spaß mit dem Hintergrundgedanken, dass es sowas so noch nicht auf dem Markt gibt, zu arbeiten. Deshalb steigt meine Motivation und Lust etwas Neues zu entwickeln enorm.

Theodor Kirchknopf

Ich persönlich war sehr begeistert von der Idee als sie uns zum ersten Mal vom Herr Prof. Wurzer vorgestellt wurde, da ich speziell in diesem Projekt meine Vorliebe und Begeisterung mit Hardware zu arbeiten, zeigen kann. Das Projekt erfordert eine gute Zusammenarbeit zwischen Software und Hardwarekomponenten, was ein großer zusätzlicher Reiz war, dieses Projekt mit drei weiteren Schulkollegen als Diplomarbeit zu wählen.

Raphael Lewitsch

Wenn man ein Schachspiel analysieren will, vereinfacht man sich das Leben, indem man alle Schritte mitschreibt. Da dies allerdings recht aufwändig ist, gerade auch für mich, einem leidenschaftlichen Schachspieler, war ich von dieser Idee von Anfang an begeistert. Die Kombination von Hard- und Software ist bei diesem Projekt sehr interessant und auf seine eigene Art und Weise sehr speziell, was dieses Projekt für mich einzigartig macht!