

Masterarbeit  
im Studiengang  
Mobile Systeme (MOS)

# **Konzeption und Implementierung eines Inventarisierungssystem in einer Laborumgebung**

Referent : Prof. Dr. Elmar Cochlovius  
Koreferent : Judith Jakob  
Vorgelegt am : 17. August 2018  
Vorgelegt von : Helge Meiering  
Matrikelnummer: 256010  
Elzstraße 1, 79297 Winden im Elztal  
h.meiering@hs-furtwangen.de



## **Abstract**

The present study examines the suitability of different software distributions for the OSGi-based gateway OpenMUC. For this purpose, criteria for the evaluation were identified by analysing different application scenarios. Each tool was evaluated on the basis of the defined criteria. Hence, the most suitable tool was integrated prototypically into the OpenMUC-framework. Afterwards, the software deployment process was tested and analyzed based on two test cases. The results of this approach forms the basis for recommendations for the further development of the OpenMUC-framework. The aim of this study is, to gain expertise, which can be applied to other OSGi-based gateways.

Die vorliegende Arbeit untersucht die Eignung verschiedener Werkzeuge zur Softwareverteilung für das OSGi-basierte Gateway OpenMUC. Zu diesem Zweck wurden anhand verschiedener Anwendungsszenarien des Gateways Kriterien für die Bewertung der Tauglichkeit der Werkzeuge erhoben. Anschließend wurden die einzelnen Werkzeuge auf Grundlage der definierten Kriterien bewertet und das geeignetste Werkzeug ermittelt. Dieses ist prototypisch in das OpenMUC-Framework integriert worden. Im Anschluss wurde der Prozess zur Softwareverteilung anhand zweier Teststellungen eingehend analysiert. Das Ergebnis der Analyse bildet die Grundlage für eine Reihe von Empfehlungen zur Weiterentwicklung des OpenMUC-Frameworks. Die Arbeit zielt ab auf einen Erkenntnisgewinn, welcher sich im Weiteren auf andere OSGi-basierte Gateways anwenden lässt.



## Inhaltsverzeichnis

Abstract . . . . .	i
Inhaltsverzeichnis . . . . .	iv
Abbildungsverzeichnis . . . . .	v
Abkürzungsverzeichnis . . . . .	vii
1 Einleitung . . . . .	1
1.1 Problemstellung . . . . .	1
1.2 Ziel der Arbeit . . . . .	2
2 Vorgehen . . . . .	3
2.1 Versionsverwaltung . . . . .	3
2.2 Scrum . . . . .	3
2.2.1 Scrum als Einzelperson . . . . .	4
3 Konzeption . . . . .	5
3.1 Analyse des aktuellen Webauftritts . . . . .	5
3.1.1 Corporate Design . . . . .	6
3.1.2 Erweiterung des Webauftritts . . . . .	6
3.2 Konzeption des Inventarisierungssystem . . . . .	8
3.3 Vergleichbare Inventarisierungssysteme . . . . .	9
3.3.1 Magento . . . . .	9
3.3.2 WooCommerce . . . . .	10
3.3.3 Entscheidung zur eigenen Anwendung . . . . .	10
3.4 Anforderungen . . . . .	11
3.5 Datenbank . . . . .	12
3.6 MockUps . . . . .	13
3.7 User Experience . . . . .	14
3.8 Frameworks . . . . .	14
3.9 Wahl der Frameworks . . . . .	14

4	Umsetzung	17
4.1	Aufbau einer Testumgebung	17
4.1.1	Integration in das Netzwerk	18
4.1.2	MySQL	19
4.1.3	Nginx	20
4.2	XAMPP	20
4.3	Austausch von Daten	21
4.4	Frontend	21
4.4.1	Installation von Angular	22
4.4.2	Commandlineinterface von Angular	22
4.4.3	TypeScript	24
4.4.4	Module	24
4.4.5	Components	24
4.4.6	Services	24
4.5	Backend	24
4.5.1	Composer	24
4.5.2	Controller	24
4.5.3	REST	24
4.6	Security	24
4.6.1	CORS	24
4.6.2	Token	24
4.6.3	Salt	24
4.7	Deploy	24
5	Anwendung	25
5.1	Routing Strategie	25
5.2	E-Mail Reminder	25
6	Diskussion der Ergebnisse	27
6.1	Fazit	27
6.2	Ausblick	27
	Literatur	29
	Eidesstattliche Erklärung	31

## Abbildungsverzeichnis

Abbildung 1: Ablauf eines Scrums. [NS17] . . . . .	4
Abbildung 2: Die Startseite des Webauftritts . . . . .	5
Abbildung 3: Raumkarte . . . . .	7
Abbildung 4: Netzwerkkarte . . . . .	8
Abbildung 5: Magento Itemverwaltung . . . . .	9
Abbildung 6: ER-Diagramm . . . . .	12
Abbildung 7: MockUp von Items-Tabelle . . . . .	13
Abbildung 8: Button Gruppierung . . . . .	14
Abbildung 9: VM-Ware mit geladenem Abbild . . . . .	17
Abbildung 10: Netzwerkeigenschaften im VM-Ware-Player . . . . .	19
Abbildung 11: WinSCP . . . . .	21
Abbildung 12: Projektordner: Angularprojekt . . . . .	23
Abbildung 13: Sourceordner in Angular Projekt . . . . .	23
Abbildung 14: App Ordner in Src-Ordner des Angularprojektes . . . . .	23





## Abkürzungsverzeichnis

**bzw.** beziehungsweise

**CD** Corporate Design

**CLI** Command Line Interface

**CMS** Content Management System

**CSS** Cascading Style Sheets

**Ecma** Ecma International

**FTP** File Transfere Protocol

**GNU** General Public License

**GPL** General Public License

**HTML** Hypertext Markup Language

**HTTP** Hypertext Transfer Protocol

**IDE** integrated development environment

**IP** Internet Protocoll Adresse

**MAC** Media-Access-Control-Adresse

**MIT** Massachusetts Institute of Technology

**NPM** Node Package Manager

**REST** Representational-State-Transfer

**UI** User-Interface

**uvm.** und vieles mehr

**UX** User Experience

**WLAN** Wireless-Local-Area-Network

**PHP** PHP: Hypertext Preprocessor

**SASS** Syntactically Awesome Stylesheets

**SCP** Secure Copy

**SHL** Smart Home Labor

**SSH** Secure Shell

**SQL** Structured Query Language

**VM** Virtuelle Maschine

**z. Dt.** zu Deutsch

# **1 Einleitung**

Das Smarthome-Labor der Hochschule Furtwangen umfasst vier einzelne Räume und ein Arbeitsbereich, in welchem sich Studierende, Mitarbeiterinnen und Mitarbeiter sich auf ihre wichtigen Arbeiten im Labor konzentrieren können. Das noch junge Labor steckt voller Leben. Es wird mit Bachelor-Projekten, Bachelor-Thesen und Master-Thesen immer weiter ausgebaut und die Studierenden lernen praxisnah den Umgang mit Geräten in einer Smarthome-Umgebung. Diese betrifft nicht nur die Unterkunft in welcher Personen leben, sondern auch die Industrie 4.0.

Im Sommersemester 2017 entstand die Masterthesis von Ashiq Mohamed Akbar Ali mit dem Titel "Creating a website with Info-Terminal and Live CCTV Stream for the Smart Home Laboratory at the Hochschule Furtwangen University". Mit dieser Arbeit wurde das Labor mit einer umfangreichen Webpräsenz ausgestattet, welche einen Einblick über die Vielfalt im Labor gibt. Das Labor wird hierbei vorgestellt und es wird auf einzelne Details eingegangen. [Ash17]

Es gibt eine Vielzahl an Geräten, Sensoren, Mikrocomputer und Alltagsgegenstände, wie zum Beispiel das Bett, mit welchen die Studierenden experimentieren und einen Betrag für die Wissenschaft und Öffentlichkeit bieten können.

## **1.1 Problemstellung**

Durch diese große Anzahl von Geräten, Sensoren und Mikrocomputer entsteht auch eine Unübersichtlichkeit. Zwar wurden alle Gegenstände erfasst und Dokumentiert, jedoch ist das nur in einer Tabelle erfasst worden. Ein Durchsuchen dieser Tabelle kann Aufwendig sein wenn ein bestimmtes Gerät und dessen Status überprüft werden soll. Für ihre Arbeiten müssen sich Studenten auch Geräte reservieren oder gar ausleihen um sie in fremden Umgebungen testen zu können oder um daheim weiter arbeiten zu können. Dies in einer Tabelle zu erfassen ist nicht unmöglich, aber es handelt sich hierbei um einen großen Aufwand. Dieser gestaltet die Arbeit von einer Mitarbeiterin oder einem Mitarbeiter, als sehr Aufwendig. Diese Tabelle muss ständig kontrolliert und aktualisiert werden. Auch muss geprüft werden, ob die Leihe für ein Gerät schon verstrichen ist.

## **1.2 Ziel der Arbeit**

Ziel ist eine Übersicht in das umfangreiche Inventar des Smarthome-Labors zu geben. Dabei kann eine Webanwendung helfen, welche in einer Tabelle Alle Geräte, Sensoren und Mikrocomputer aufnimmt. Diese Tabelle kann mittels Software schnell durchsucht und einfach erweitert werden. Auch soll die Mitarbeiterin oder der Mitarbeiter, wie auch Studierende an das Ende einer Leihfrist erinnert werden. Auch hierbei ist die Software eine Lösung, indem sie Erinnerungsmails verschickt.

## **2 Vorgehen**

In der Vorbereitung wurden schon früh die Werkzeuge für die Umsetzung und Konzeption gewählt, welche das Projekt strukturieren und planen sollten. Dabei kamen Mittel wie Kugelschreiber, Notizbücher, Kalender oder auch die Mindmap zum Einsatz. Mittels Requirement Engineering wurde eine Anforderungsanalyse erstellt. Um einen übersichtlichen Workflow zu haben, fiel die Entscheidung Scrum als Vorgehensmodell für den künftigen Ablauf zu wählen.

### **2.1 Versionsverwaltung**

Für die Versionsverwaltung wurde Git gewählt. Es handelt sich um ein Dezentrales Versionsverwaltungssystem und ist nicht von einem bestimmten Server verbunden. In einem Versionsverwaltungssystem werden verschiedene Entwicklungszustände gesichert. Damit ist es möglich zu sehen, wie sich eine Software entwickelt hat. Sollte ein Fehler auftreten, kann zu einer älteren Version gewechselt werden, in welcher der Fehler nicht vorhanden ist. Diese Versionen werden in einem Repository verwaltet. [git]

### **2.2 Scrum**

Unter Scrum versteht man in der Projektplanung ein agiles Vorgehensmodell in der Softwareentwicklung. Es besteht aus mehreren Komponenten wie, Rollen Artefakte und Meetings. Sobald die Anforderungen und Eigenschaften eines Produktes angelegt worden sind, legt sie der Product Owner in einem sogenannten Product Backlog an. Daraufhin wird ein Sprint geplant. Es werden Anforderungen und Eigenschaften gewählt, welche in einem bestimmten Zeitraum erledigt werden können. Diese wiederum werden in das sogenannte Sprint Backlog abgelegt. Darauf erfolgt ein Sprint. Welcher nicht länger als ein Monat dauern sollte. Während des Sprints tauscht sich das Entwicklerteam über den täglichen Status der Entwicklung aus. Man spricht von einem Daily Meeting. Jeder ist auf dem aktuellen Stand und es kann sofort eingegriffen werden, wenn es an einer Stelle zu Schwierigkeiten kommt. Ist ein Sprint abgeschlossen kann ein funktionierender Softwareteil präsentiert werden. [NS17]

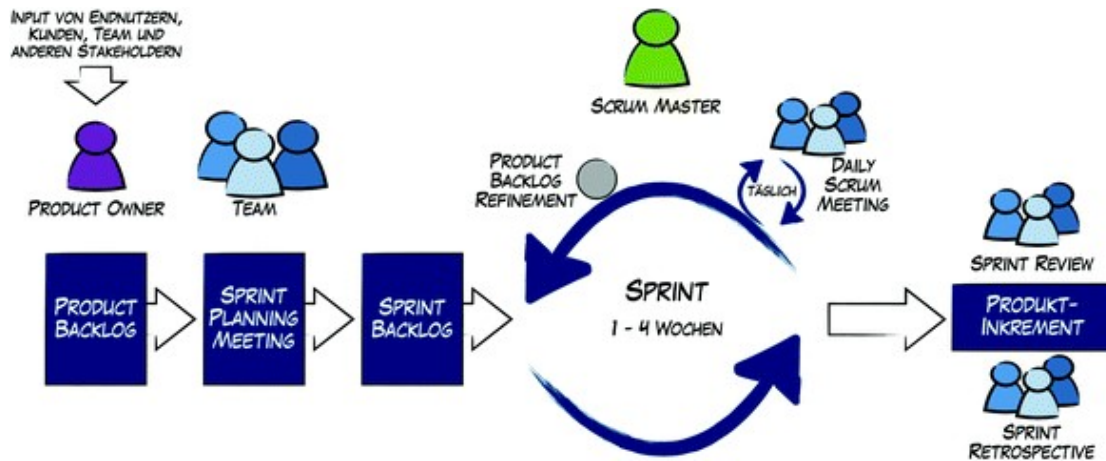


Abbildung 1: Ablauf eines Scrums. [NS17]

### 2.2.1 Scrum als Einzelperson

In dieser Arbeit wurde alleine und nicht im Team gearbeitet. Auch wenn es hierdurch keine Rollenverteilung gab, so konnte Scrum doch ideal eingesetzt werden. Das Backlog wurde mit Hilfe der Konzeption erzeugt. Mit der Planung wurde klar, welche Anforderungen an die Software bestehen. Diese wurden alle handschriftlich in ein Productbacklog niedergeschrieben. Darauf wurde mit dem Betreuer besprochen, welche Aufgaben innerhalb von zwei Wochen erledigt sein sollen und wurden, ebenfalls handschriftlich, in das Sprintbacklog aufgenommen. Während des Sprints sind die Daily Meetings entfallen, da diese nicht nötig waren. Als einzelne Person ist man immer über den aktuellen Stand seiner Entwicklung bewusst. Ein tägliches Austauschen mit Teammitgliedern ist nicht nötig, da es kein Team gibt. Nach den vergangenen zwei Wochen, wurde der Aktuelle Stand in einem Protokoll dokumentiert. Erfolge und Probleme wurden im Anschluss mit dem Erstbetreuer und der Zweitbetreuerin besprochen. Daraufhin wurde, wenn es nötig wurde, das Productbacklog erweitert. Aus diesem Backlog wurden erneut Aufgaben für den nächsten Sprint gewählt. Der Entwickler übernahm so alle Rollen, welche in dem Vorgehensmodell Scrum definiert sind.

### 3 Konzeption

Bei der Konzeption wurde der aktuelle Stand des bestehenden Webauftritts betrachtet und analysiert. Im weiteren Schritt wurden Erweiterungen für die Website geplant. Im nächsten großen Schritt erfolgte eine Analyse aktuell bestehender Software, welche Gegenstände verwaltet.

#### 3.1 Analyse des aktuellen Webauftritts

Der Webauftritt, erreichbar im Netzwerk der Hochschule Furtwangen <http://web.smarthome.hs-furtwangen.de/>, des Smart Home Labor (SHL) Labors, wird mit der Hilfe des Content Management System (CMS) WordPress verwaltet. WordPress eigne sich hier, weil es mittels User Experience (UX) so gestaltet wurde, das es sehr leicht zu erlernen ist. Unerfahrene Benutzer könne so schnell eigene Inhalte einpflegen.

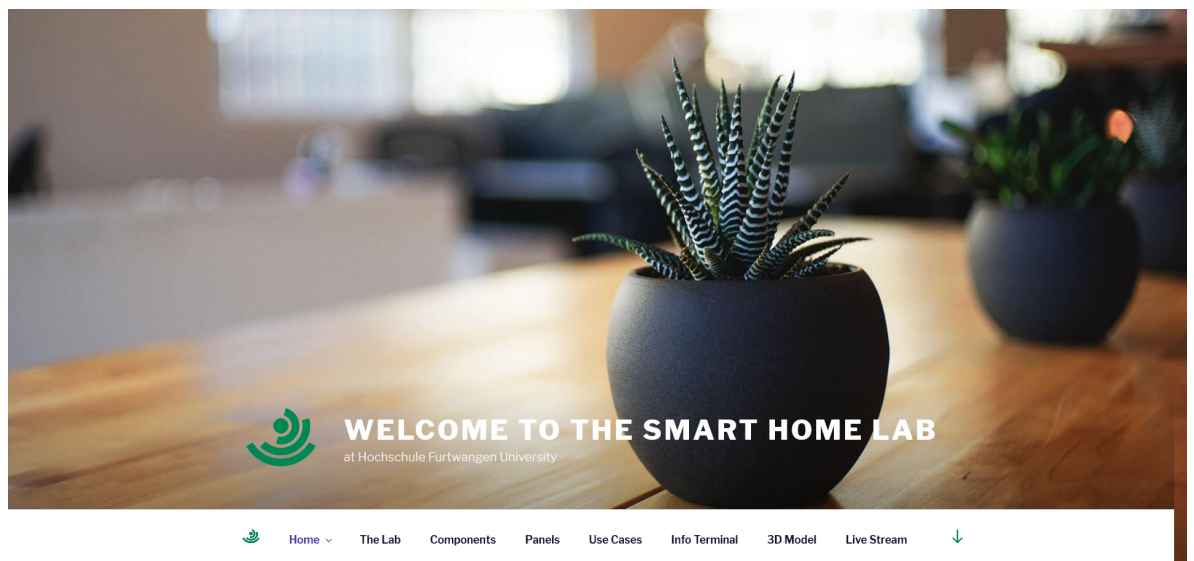


Abbildung 2: Die Startseite des Webauftritts

Die Website ist mit dem, von WordPress eigenem Designe Twentyseventeen gestaltet. Auf der Startseite sieht man ein großes Headerbild. Darunter folgt eine Navigationsleiste, welche die Links bei einem Besuch einer Unterseite in einem Lila Farbton darstellt. Die Navigation umfasst die Seiten:

- Home
- The Lab

- Components
- Panels
- Use Cases
- Info Terminal
- 3D Model
- Live Stream

Weiter folgt eine kurze Beschreibung über das Labor. Der nächste Abschnitt der Startseite stellt das Team vor. Im Anschluss sieht man noch ein Video, welches das Labor vorstellt, ein Kontaktformular und die Adresse mit einer Google Maps Karte. Am untersten Ende ist ein Footer, welche eine Copyright und einen Link zum Impressum enthält.

Die Unterseite „The Lab“ stellt das Labor mit Grundrissen etwas detailreicher vor. „Components“ stellt wenige, aber wichtige Geräte vor. In „Panels“ werden, die mit Sensoren und Geräte versehenen, Wände in den Räumen beschrieben. Eine Auswahl von umgesetzten Anwendungsfällen werden in der Unterseite „Use Cases“ beschrieben. Für das Infoterminal gibt es ebenfalls eine Unterseite, welche automatisiert eine Präsentation über das Labor abspielt. Unter dem Punkt „3D Model“, befindet sich eine 3D Ansicht des Labors, welches sich in einer Ego- und Vogelperspektive betrachten lässt. Auch können hier vereinzelt Geräte bedient werden. Abgeschlossen wird die Navigation mit einer Unterseite, welche einen Video Livestream zeigen kann.

### 3.1.1 Corporate Design

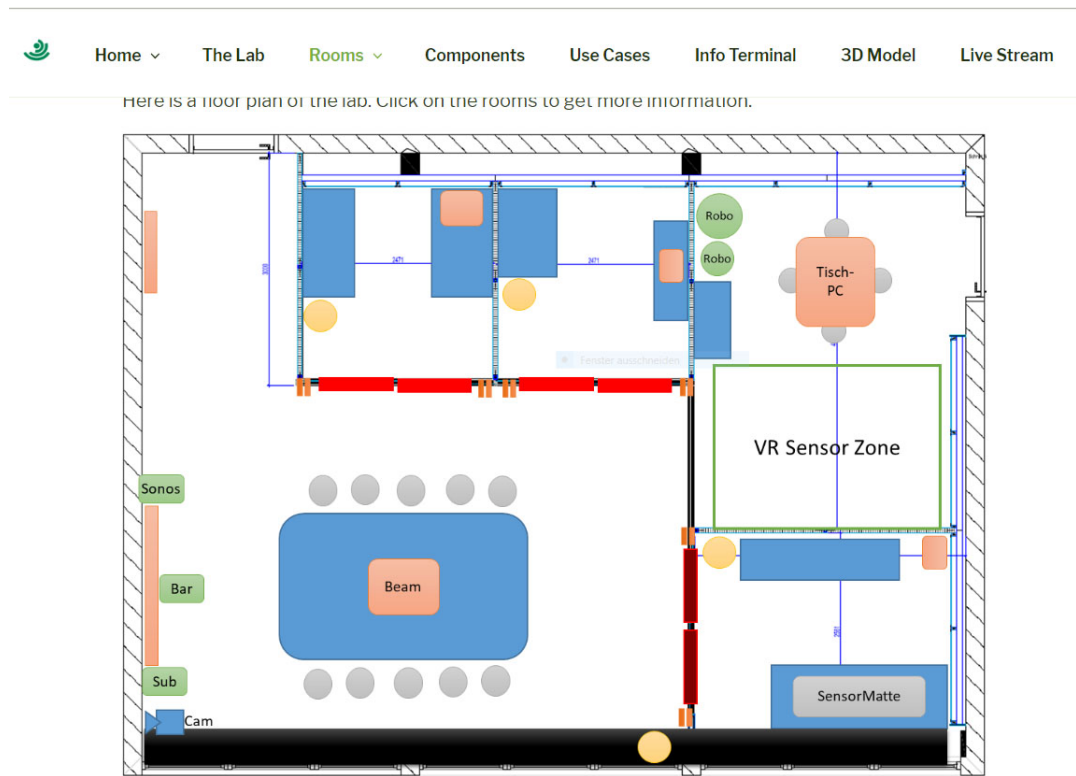
Mit den aktuellen Farben entspricht der Webauftritt des Labors nicht dem Corporate Design (CD) der Hochschule Furtwangen. So müssen die Lila Farben durch die Farbe Grün mit dem Hexwert 83b62d geändert werden. Dies ist wichtig um einen Wiedererkennungswert der Hochschule darzustellen. Die Navigation stellte hier einen größeren Bruch dar. Um die Seite vollständig im CD der Hochschule zu haben, ist ein Expertengespräch nötig.

### 3.1.2 Erweiterung des Webauftritts

Neben dem Corporate Design (CD) gab es noch weitere Punkte, welche den Auftritt noch Informativer und attraktiver gestalten konnten. So stehen im Labor die einzelnen Räume: Küche, Bad, Multimediaraum, IoT-Raum und der Arbeitsbereich stark im Vordergrund. Diese wurden bisher nur sehr dürftig auf der Homepage erwähnt. Daher wurde für die Räume eigene Unterseiten angelegt. Um auf User Experience (UX) zu



achten, wurde mittels des WordPress-Plugin Draw Attention eine Interaktive Karte erzeugt. Bei einem Klick auf einen Raum, erhält der User mehr Informationen. [WP ]

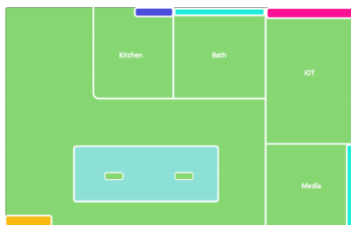


---

Abbildung 3: Raumkarte

Jeder Raum hat seine eigene Unterseite, welche die Aufgaben des Raumes und dessen Ausstattung beschreibt.

Die Netzwerkarchitektur im Labor ist sehr komplex. Wenn man nicht gerade mit am Aufbau dieser Architektur gearbeitet hat, kann schnell der Überblick verloren werden. Eine Übersichtskarte (siehe Abbildung 4), auf welcher man sieht wo sich Server, Router und Gateways befinden, kann helfen um herauszufinden, wo bei Problemen mal nachgesehen werden kann. Diese findet man in der Navigation unter dem neuen Punkt Räume. Neben der Karte gibt es zu jedem Raum eine eigene Unterseite, wo auf dessen Netzwerkeigenschaften genauer eingegangen wird. Die Karte wurde mit Adobe Illustrator CC 2017 erzeugt. Dabei wurde, um die Übersichtlichkeit nicht zu gefährden, sehr auf Schlichtheit geachtet. Nur das nötigste wurde eingezeichnet. Neben der Einfachheit halten sich die Farben Grün und Weiß an das CD der Hochschule Furtwangen.



On the graphic you can see the different rooms of the laboratory. The network runs around the lab using a network cable. Thus, in each room with the help of LAN, devices can be integrated into the network. There are also two WLANs available. So also wireless devices can be integrated.

Network in:

- IOT room
- Bath
- Kitchen
- Media room
- Workplace

Abbildung 4: Netzwerkkarte

### 3.2 Konzeption des Inventarisierungssystem

Das Inventarisierungssystem ist das Herzstück dieser Masterthesis. Mit ihm können alle Gegenstände verwaltet werden und dazu noch sehr einfach und übersichtlich. Dabei spielt User Experience (UX) und damit auch das User-Interface (UI) eine sehr große Rolle. Um einen Überblick zu bekommen was der aktuelle Markt zu bieten hat, müssen Vergleichbare Verwaltungssysteme(siehe Kapitel 3.3) analysiert werden. Damit eine reibungslose Programmierung erfolgen kann müssen Konzepte für die Architektur der Anwendung ausgearbeitet werden.

### 3.3 Vergleichbare Inventarisierungssysteme

Sucht man nach Inventarisierungssystemen stößt man immer wieder auf Software für Webshops. Mit ihnen hat man sehr häufig einen riesigen Umfang an Funktionen, mit welchen man nicht nur sein Inventar, sondern auch seine Verkäufe verwalten kann.

#### 3.3.1 Magento

Magento<sup>1</sup> ist eine Openen-Source-E-Commerce Plattform und steht unter der Open Software License<sup>2</sup>. Umgesetzt wurde es mit dem PHP: Hypertext Preprocessors (PHPs) Framework Zend und lässt sich durch zahlreiche Plug-Ins erweitern.[.2018]

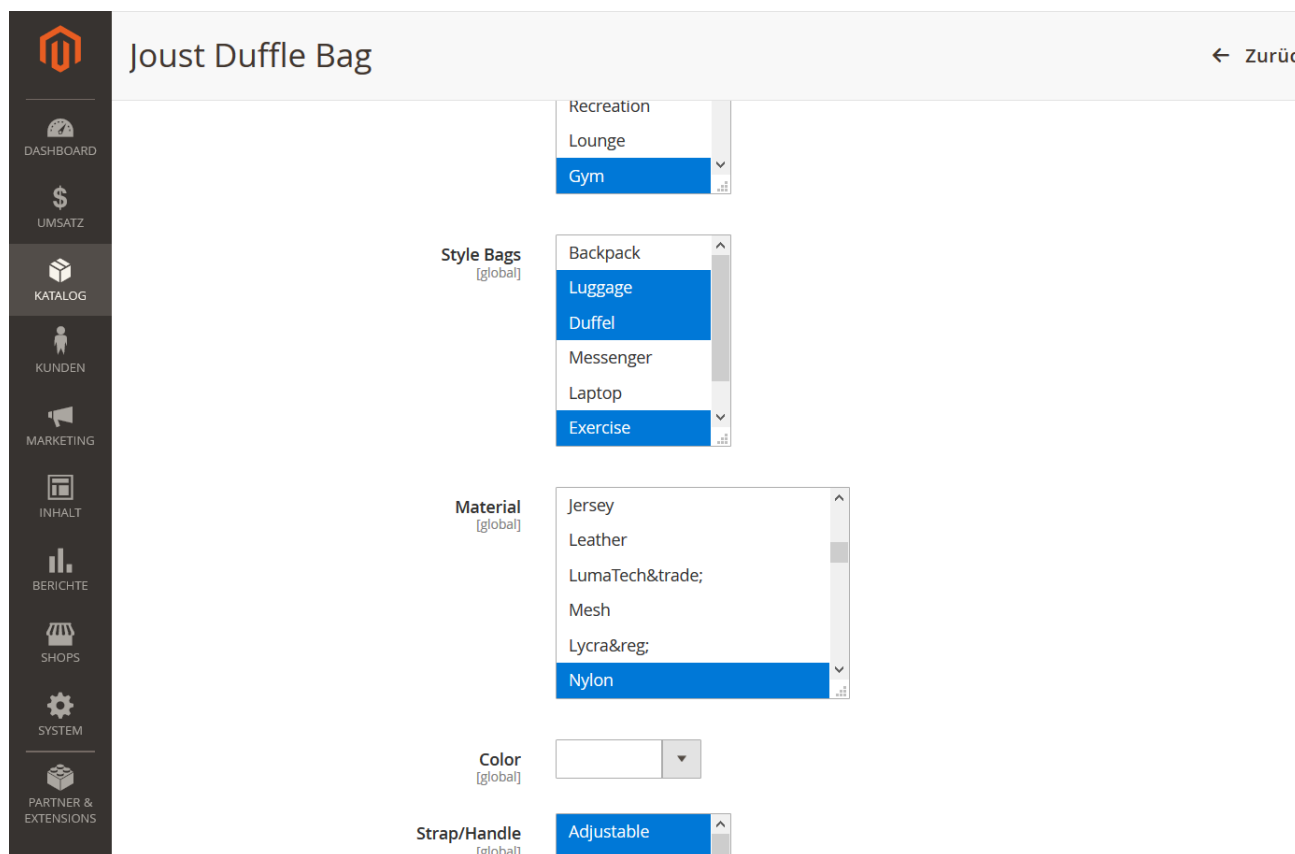


Abbildung 5: Magento Itemverwaltung

##### 3.3.1.1 Vorteile von Magento

Magento bringt zahlreiche Verwaltungsoptionen. Man kann bis in das kleinste Detail einen Webshop konfigurieren und verwalten. Eine Rollenverteilung der User wird ebenfalls geboten. Jeder Gegenstand kann mit vorgefertigten und eigen angelegten Eigenschaften versehen werden. Das UI von Magento ist ähnlich wie bei WordPress

<sup>1</sup><https://magento.com>

<sup>2</sup><https://opensource.org/licenses/osl-3.0.php>

sehr benutzerfreundlich. Mit ein wenig Erfahrung findet man sich schnell zurecht. [Koc12]

### **3.3.1.2 Nachteile von Magento**

In seiner Größe, liegt der Nachteil des Frameworks. Es bietet zu viele Funktionen, welche nicht für ein Inventursystem in einer Laborumgebung benötigt werden. Damit belegt es unnötig Speicherplatz und lädt Dateien, welche nicht gebraucht werden. Auch wenn die Lernkurve sehr flach ist, muss eine gewisse Einarbeitung doch erfolgen. Auch ist mit der aktuellen Version von Magento eine Aktualisierung der PHP Version nötig. [Koc12]

### **3.3.2 WooCommerce**

Anders als bei Magento (siehe Kapitel 3.3.1) steht das WordPress-Plugin, welches auch als E-Commerce Software dient unter der General Public License (GNU) Lizenz.

#### **3.3.2.1 Vorteile von WooCommerce**

Bei kostenlose WordPress-Plugin ist kostenlos und sehr einfach zu installieren. Mit einem Klick ist es aus dem Plugin Bereich gewählt und kann mit einem Installationsassistenten den eigenen Bedürfnissen nach angepasst werden. Ein erfahrener WordPress-Nutzer hat eine deutlich niedrigere Lernkurve, als bei Magento [Woo18]

#### **3.3.2.2 Nachteile von WooCommerce**

Die Nachteile sind ähnlich wie bei Magento. Es beinhaltet zu viele Funktionen, welche nicht gebraucht werden. Außerdem ist es sehr schwer möglich es nicht als Webshop, sondern als Verwaltungssystem zu nutzen. WooCommerce gibt hier sehr starke Richtlinien vor. So muss man schon im Installationsassistenten sich Gedanken über Maße, Bezahlmethoden und Versand machen. Was für ein Inventarisierungssystem nicht nötig ist. Mit WooCommerce ist man stark an WordPress gebunden. Sollte man das CMS wechseln wollen, ist ein mitnehmen der Anwendung nicht möglich.

### **3.3.3 Entscheidung zur eigenen Anwendung**

Die Entscheidung eine eigene Anwendung zu schreiben wurde schon früh getroffen. Es muss eine Anwendung zur Verfügung gestellt werden, welche nicht mit Funktionen überladen ist. Die Lernkurve muss für jeden Anwender möglichst flach gehalten sein. Der Aufwand Magento oder WooCommerce in das Hochschul CD zu bringen, kann ein großer Aufwand sein. Daher lohnt es sich die Anwendung selbst zu schreiben. So bleibt sie konfigurierbar und es werden nur PHP Kenntnisse und keine speziellen WordPress

oder Magento-Kenntnisse vorausgesetzt. Bei einem Wechsel zu einem anderen CMS kann die Anwendung ebenfalls übernommen werden.

### **3.4 Anforderungen**

Mit dem Anforderungsmanagement werden mögliche Features definiert. In der Anforderungsanalyse werden Eigenschaften, Funktionalitäten und die Qualität an die Software Festgehalten. [Gra14] Es werden funktionale und nicht-funktionale Anforderungen niedergeschrieben. Mit diesen Anforderung entsteht eine Definition und Ziele an die Anwendung.

- Das Inventarisierungssystem muss alle Items in einer Laborumgebung aufnehmen.
- Dabei soll bekannt sein wie der aktuelle Zustand der jeweiligen Items ist.
- Die Datenbank sollen durchsuchbar sein.
- Es müssen neue Items angelegt und alte gelöscht werden können.
- Es soll für jeden Benutzer einen persönlichen Bereich geben.
- Es soll Verschiedene Rollen für die Administration und User geben.
- Jeder Benutzer muss seine Persönlichen Daten verändern können.
- Benutzer können über den Administrator oder einem dazu befugten Mitarbeiter Items leihen.
- Benutzer sollen an die Rückgabe der geliehenen Items erinnert werden.

Ist eine solche Liste von Anforderungen erstellt worden, erhält man eine gute Übersicht über die zu entwickelnden Teilbereiche.

### 3.5 Datenbank

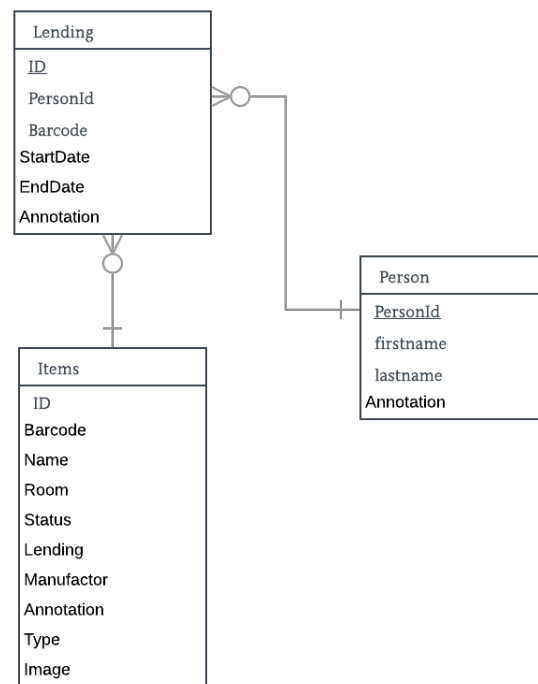


Abbildung 6: ER-Diagramm

Mit der Idee der Webanwendung, wurde schon sehr früh mittels einer Mindmap jede mögliche Eigenschaften aufgeschrieben. Aus dieser Mindmap konnte sehr komfortabel ein ER-Diagramm erzeugt. Um Redundanzen zu vermeiden, muss eine Normalisierung durchgeführt werden. In der Regel sind Redundanzen mit den ersten drei Normalformen abgeschlossen. Im ersten Schritt wurde darauf geachtet, dass alle Eigenschaften Atomar sind. Vorname, in der Datenbank firstname und Nachname, in der Datenbank lastname, müssen als einzelne Eigenschaften zur Verfügung stehen. Eine Eigenschaft, mit dem Namen fullname, ist nach der ersten Normalform nicht erlaubt. Um die zweite Normalform zu erfüllen, muss die erste Normalform erfüllt sein und jedes Nichtschlüsselattribut von jedem Schlüsselkandidaten voll funktional abhängig sein. So wird festgestellt welche Eigenschaften eindeutig sind. Das ist nicht immer möglich, wie auch im Fall der Items Tabelle. Jeder Barcode in der Items-Tabelle ist einzigartig, jedoch gibt es Produkte, welche bewusst keinen Barcode haben. Gelöst wurde das Problem mit dem Künstlichen Primärschlüssel ID.[Dat18a] Die höchste Priorität der Normalisierung ist das Vermeiden von Redundanzen und kann nur mit der 3. Normalform erreicht werden. Hierfür wurde die Hilfstabelle Lending eingefügt. Feststellen kann man das, in dem man überprüft, ob es viele-zu-viele existieren. Denn ohne diese Tabelle käme es zu Problemen, wenn sich eine Person mehrere Items leihen möchte.[Dat18b]

### 3.6 MockUps

Um die geplante Idee umzusetzen ist es nötig ein MockUp zu erstellen. Doch bevor das MockUp erstellt werden kann, muss Storytelling betrieben werden. Beim Storytelling werden die Anforderungen in realistische kleine Geschichten erzählt. So wird es leichter sich vorzustellen, wie ein Programm funktioniert. Diese Geschichten werden sachlich aufgeschrieben. An diesen Geschichten kann früh festgestellt werden, ob sich eine Idee nicht oder nur sehr aufwändig umsetzen lässt. Hier tritt auch die User Experience in den Vordergrund, da man sich beim Schreiben in den User hineinversetzen muss. [Sch17]

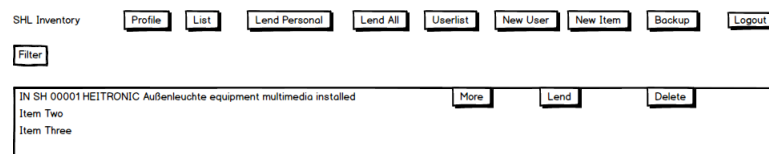


Abbildung 7: MockUp von Items-Tabelle

Ist das Storytelling abgeschlossen und die möglichen Fehler beseitigt wurden, so kann das MockUp beginnen. Beim MockUp, in der Webentwicklung, wird das Layout schemenhaft mit Bleistift und Papier erzeugt. Dabei tritt das Design stark in den Hintergrund, es ist in erster Linie wichtig, das verstanden wird, wie die Anwendung mit Button, Formulare und weiteren Elemente funktionieren. Das MockUp macht es später für die Entwicklung leichter, wenn es um die Umsetzung der Webanwendung geht. Bei der Erstellung des MockUp's hat man die Programmierung immer im Hinterkopf. Dabei wird überlegt wie Elemente angelegt werden.

### 3.7 User Experience

User Experience (UX) wurde häufiger in dieser Ausarbeitung verwendet und wurde noch nicht genauer erklärt. Das geschieht nun. Bei der User Experience, auf deutsch *Benutzererfahrung*, spricht man davon, wie der Benutzer mit Software Umgeht. Dabei greift man auf eigene Erfahrungen zurück, es gibt allerdings auch Werkzeuge um eine gute Benutzbarkeit zu messen. [Mos12] Dazu gehört Beispielsweise das Storyboard, welches mit Hilfe des Storytelling Geschichten mit Bildern erzählt. Für dieses Projekt wurde auf kurze Wege geachtet, das der User die Maus nicht allzu sehr die Maus bewegen. Auch wird dem User durch das Design gezeigt was klickbar ist und was nicht. Unbewusst wird er eine Gruppierung vornehmen. So sind Buttons, mit welchen etwas erzeugt wird, wie *New Item*, *New User* und *Backup*, mit einem anderen Hintergrund und einer Umrandung versehen.



Abbildung 8: Button Gruppierung

### 3.8 Frameworks

Wenn man Framework in die deutsche Sprache übersetzt, so erhält man das Wort Rahmengerüst. In der Softwareentwicklung spricht man von solchen Frameworks bei Programmierstrukturen, welche den Entwickler unterstützen. Dabei handelt es sich nicht um eine fertige Software. Es handelt sich um bewährte Programmierstrukturen, welche dem Entwickler die Arbeit abnehmen. So werden bekannte Entwurfsmuster übernommen werden. Auch aufwendige Funktionen sind in Frameworks schon häufig implementiert. Der Entwickler muss sich nicht mehr aufwendig um eine Datenverbindung kümmern oder kann komplizierte Animationen mit wenigen Zeilen Code erstellen. Eine wichtige Aufgabe während der Softwareentwicklung ist das Testen. Hier werden mithilfe des Frameworks Tests geschrieben, mit welchen kontrolliert werden kann, ob die Software korrekt funktioniert. Frameworks kommen meistens in der Objektorientierung zum Einsatz. Ihre Einsatzgebiete können vielseitig sein. Man findet sie in der Softwareentwicklung, bei der Entwicklung von Spielen und in der Webentwicklung.

### 3.9 Wahl der Frameworks

Bei der Wahl der Frameworks wurde analysiert welches sicher läuft, eine große und aktive Community hat, wenig Einrichtungen erfordern und in welchen die meisten Erfahrungen schon vorhanden sind. Bei der Wahl des Frontend-Frameworks fiel die Wahl



schon früh auf Angular. Es wird bis heute noch von Google weiterentwickelt und hat eine sehr große Community. Die Erfahrungen in JavaScript sind ebenfalls vorhanden und die meisten Einrichtungen sind schon vorhanden oder werden von dem Framework selbst übernommen. Bei der Wahl des Backend-Frameworks war die Wahl nicht so einfach. Es standen Tomcat, ein Java-Framework und Laravel, ein PHP-Framework, zur Auswahl. Es wurde Laravel gewählt, da es von sich aus viele Aufgaben, wie Sicherheit, Datenbankverwaltung oder Benutzerverwaltung mit Login-System übernimmt. PHP und MySQL waren auf dem Server installiert. Für Tomcat hätte eine separate Installation erfolgen müssen.



## 4 Umsetzung

Mit der Umsetzung des Konzeptes zeigt sich wie gut dieses Gelingen ist. Nachdem nun geklärt ist, wie die Anwendung aussehen soll und welche Werkzeuge verwendet werden, muss eine Testumgebung aufgebaut werden. Ebenfalls muss der Umgang mit den Frameworks und der integrated development environment (IDE), zu deutsch integrierte Entwicklungsumgebung, vertraut sein.

### 4.1 Aufbau einer Testumgebung

Der Server im Smarthome Lab erledigt rund um die Uhr Aufgaben. Dabei überwacht er den Zustand verschiedener Geräte und sorgt dafür dass diese erreichbar sind. Auf ihm liegen auch Medien und der Webaufritt, welcher über einen Webserver läuft. Dies muss immer zuverlässig funktionieren. Damit der Betrieb nicht eingeschränkt ist, muss eine virtuelle Testumgebung geschaffen werden. Für diesen Zweck wurde ein Abbild des Servers erzeugt, als alles zuverlässig gearbeitet hat. Auf dem Abbild war neben dem Betriebssystem auch der Webserver mit Webaufritt und die MySQL-Datenbank zu finden. Da diese identisch mit den Daten des Servers waren, konnte realistisch getestet werden. Durch das einfache erstellen von Sicherheitskopien konnte risikofrei Software installiert werden um zu schauen, wie sich diese auf das System auswirkt. Mit den verschiedenen Sicherheitskopien, konnte bei Problemen immer auf einen alten Stand zurückgegriffen werden.

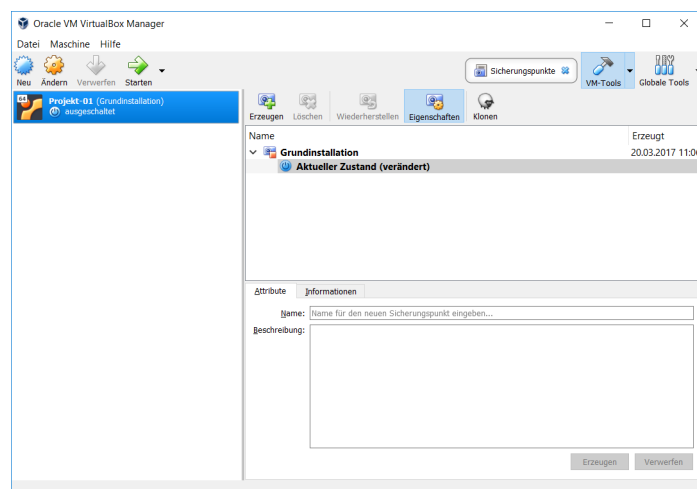


Abbildung 9: VM-Ware mit geladenem Abbild

In Abbildung 9 sieht man den VM-Ware Player. Dieser ist ein kostenloses Tool um Virtuelle Maschinen abzuspielen. Der VM-Ware Player simuliert dabei Hardware, welche im Live-betrieb verändert werden kann. [vmw18] In diesem Fall wird der Server aus dem Smarthome Lab nur Simuliert. Dabei kann ihm mehr Speicher hinzugefügt werden, oder auch Speicher entfernt werden. So kam es zu dem Problem, dass auf der Virtuellen Maschine (VM) nicht ausreichend Festplattenspeicher zur Verfügung stand. In diesem Fall musste die komplette VM vergrößert werden. Dies ist mit VM-Ware Player nicht machbar. Unter Windows wird bei der Installation von VM-Ware Player auch der Vdiskmanager installiert. Ein Werkzeug, welches sich über die Kommandozeile steuern lässt. So lässt sich mit dem Befehl `vmware-vdiskmanager -x 100Gb vm.vmdk`, im Verzeichnis der VM, die Größe verändern. [Tho18] Damit war ein Vergrößern der VM zwar möglich, der zusätzliche Speicher stand zwar physisch, durch eine größere VM Datei zur Verfügung, konnte jedoch nicht verwendet werden. In der VM wurde ein sogenannter Snapshot verwendet. Bei einem Snapshot, handelt es sich um eine Kopie der VM zu einem bestimmten Zeitpunkt. Dieser musste ebenfalls vergrößert werden. Da die Quellen hierzu rar waren, erforderte die Lösung eine lange Suche. Die Lösung ist, mit dem Vdiskmanager den Snapshot auf die gleiche Größe zu bringen. Der Speicher wird nun angezeigt, kann jedoch noch nicht verwendet werden, da er keiner Partition zugewiesen worden ist. Dies ist über das Betriebssystem, Ubuntu Server 16.04, welches auf der VM läuft nicht einfach zu bewerkstelligen. Eine Lösung ist es Ubuntu 18.04 zu verwenden. Dabei wird ein Abbild der DVD, welche für die Installation von Ubuntu benötigt wird, in das Virtuelle DVD Laufwerk gelegt. Ubuntu kann ohne Installation verwendet werden. Es liefert das Programm GParted, mit welchem sich der neue, noch nicht zugewiesene Speicher, der Serverpartition zuweisen lässt. Mit einem Neustart der VM und dem Entfernen des Images aus dem virtuellen Laufwerk, hat die VM nun ausreichend Festplattenkapazität.[**automatix.**]

#### 4.1.1 Integration in das Netzwerk

Da es sich bei der VM um ein direktes Abbild des Servers im Smarthome Lab ist, hat es auch die identischen Eigenschaften. So auch die Media-Access-Control-Adresse (MAC). Mit der MAC-Adresse soll der Netzwerkkarte eine eindeutige Identifikationsnummer zugewiesen werden. Dies geschieht schon bei der Herstellung und der Router kann damit Geräte identifizieren, welche im Netzwerk eingebunden sind. Damit ist dem Router bewusst welches Gerät, welche Pakete und welche Internet Protocoll Adresse (IP) erhält. [Zis16] Solche IPs sollen weltweit eindeutig sein, sind es aber nicht, wenn es nicht gewünscht ist. Bei einem Abbild erwartet der User ein identisches System. Möchte er dieses System allerdings im gleichen Netzwerk einsetzen, so wird der Router die Pakete an Router und an die VM senden, da der Router nicht unterscheiden kann, wer der Korrekte Empfänger ist. Wird das nicht korrigiert,

kann es leicht passieren das unerwünschte Änderungen am Server und nicht an der VM unternommen werden. Denn die VM, wie auch der Server werden über Secure Shell (SSH) angesteuert. Dabei wird ein Programm verwendet, in diesem Fall Putty, welches eine sichere Verbindung über das Netzwerk zum Server oder zur VM aufbaut. Ist man nun im Netzwerk des SHL ist nicht klar ob die VM oder der Server angesprochen wird. Denn schließlich haben beide die selbe IP zugewiesen bekommen. Eine Lösung für dieses Problem wird direkt in vmware-Player zur Verfügung gestellt. In Abbildung 10 sieht man das man die Media-Access-Control-Adresse einfach in einem Textfeld anpassen kann. Mit einer neuen Media-Access-Control-Adresse unterscheidet das Netzwerk nun beide *Netzwerkkarten*. Damit erhält die Virtuelle Maschine nun eine neue IP und ist damit eindeutig adressierbar.

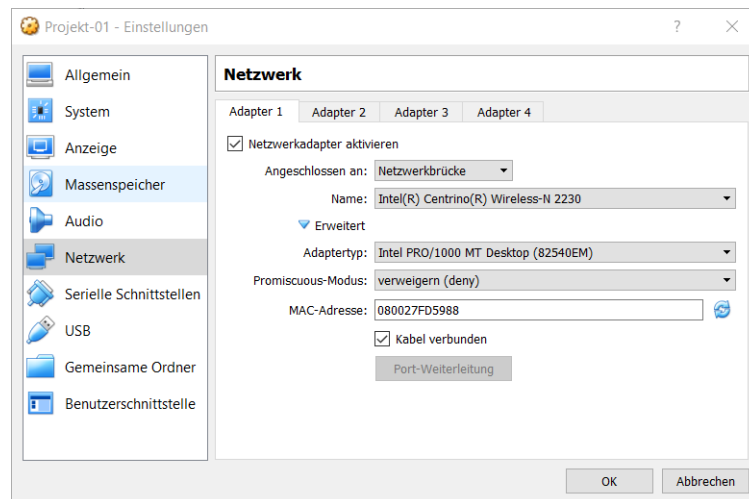


Abbildung 10: Netzwerkeigenschaften im VM-Ware-Player

#### 4.1.2 MySQL

Auf dem Server stand bereits eine MySQL Datenbank zur Verfügung. Es handelt sich hierbei um eine relationale Datenbank. [Ora] MySQL steht unter der General Public License (GPL) 2.0 und ist OpenSource. WordPress setzt eine MySQL oder eine MariaDB voraus.[Ash17] Die Daten der Tabellen, welche Wordpress genutzt hatte, waren alle weiter erreichbar und konnten verwaltet werden. Die Daten für die Datenbank selbst haben gefehlt. Wenn diese Daten fehlen, so ist es möglich die Datenbank, mit Adminrechte, ohne Passwort zu starten. Nach einem solchen Start, kann das Passwort beliebig verändert werden. In diesem Fall wurde auch das von der Datenbank untersagt. Eine Datenbank war für die kommende Arbeit notwendig und somit auch der Zugriff. Da dies nicht möglich war, musste die Datenbank neu installiert werden. Für diesen Zweck wurden alle Tabellen und Datensätze in eine Datei gesichert. Über ein Plugin, welches in WordPress installiert wurde, konnte eine solche Datei erzeugen.

Mit der Neuinstallation der MySQL-Datenbank wurde ein bekanntes Passwort vergeben. Die Daten wurden importiert und es mussten nur noch wenige Daten angepasst werden, da es sich um die identischen Daten handelte.

#### 4.1.3 Nginx

Vorinstalliert war auch der WebServer Nginx. Standardmäßig handelt es sich hier um einen Webserver, welcher HTTP anfragen entgegennimmt und GET Anfragen verarbeitet. Um einen WordPress Blog welcher PHP, JavaScript und Hypertext Markup Language (HTML) benötigt, ist das völlig ausreichen.[Ash17] Für weitere Verarbeitungen wie POST, PUT und DELETE müssen diverse Konfigurationen an Nginx vorgenommen werden.[MFB13]

## 4.2 XAMPP

Um lokal die Webanwendung entwickeln zu können, ist es nötig auf der lokalen Maschine MySQL und PHP installiert zu haben. Eine Lösung, welche für dieses Problem angeboten wird, ist XAMPP. Es handelt sich um eine frei Paketsammlung zu welcher auch PHP und MySQL gehören und wurde von den Apache Friends entwickelt.[Apa] Es steht unter der GPL. Durch seine UI ist das starten von PHP oder MySQL mit einem klick erledigt. Sollte eine Datenbankabfrage mal nicht möglich sein, könnte eine Fehlerquelle sein, das der MySQL-Server nicht läuft.

### 4.3 Austausch von Daten

Für einen sicheren Datentransfer von der Entwicklungsmaschine zur Virtuellen Maschine wird Secure Copy (SCP) verwendet. Um dies nicht über ein Command Line Interface (CLI) machen zu müssen, gibt es Software mit einer grafischen Oberfläche. Mit solch einer Software wird eine sichere Verbindung aufgebaut. Ein Datentransfer funktioniert so ähnlich wie bei einem File Transfer Protocol (FTP) Tool. Man zieht die zu übertragenden Dateien mit der Maus in das Zielverzeichnis. Ein solcher Vertreter ist, unter dem Betriebssystem Windows, WinSCP, welches in Abbildung 11 zu sehen ist.

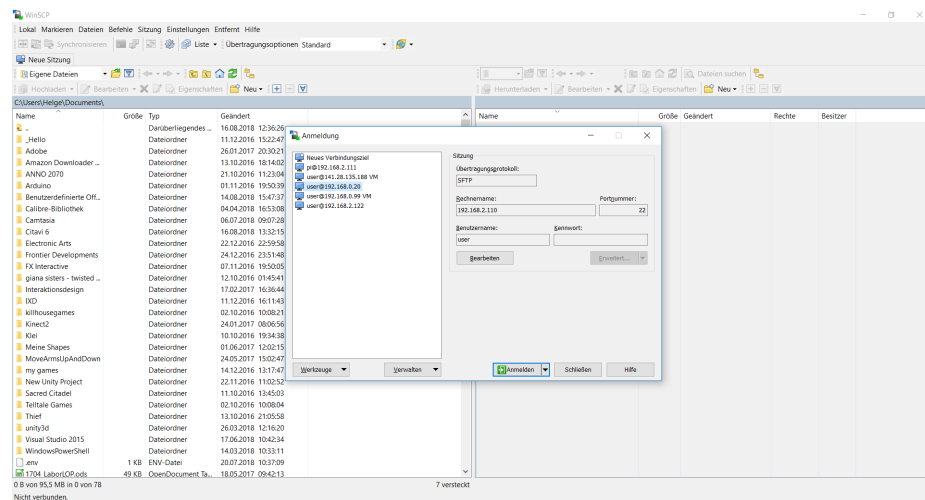


Abbildung 11: WinSCP

### 4.4 Frontend

Bei einem Frontend spricht man von dem was der Anwender sieht. Es handelt sich um Buttons, Links, Farben, Layouts und vieles mehr (uvm.). Wie in Kapitel 3.9 erwähnt, wurde Angular für die Umsetzung des Frontends verwendet. Zum Entwicklungszeitpunkt befand sich Angular in der Version 5, was sich im Laufe der Umsetzung geändert hat. Dies nahm jedoch keinen Einfluss auf die Entwicklung und die Webanwendung wurde mit Angular 5 fertig entwickelt. Es erfüllt die Aufgaben einer Komponentenbasierten Darstellung von verschiedenen Bausteinen. Durch seine Feste Struktur von Services, Components (zu Detusch (z. Dt.) Komponenten) und Views (z. Dt. Ansichten) [Clo18], ist es übersichtlich und Kenntnisse mit Angular reichen, um sich einzuarbeiten. Es wird von Google entwickelt, aber profitiert von einer sehr aktiven Community (z. Dt. Gemeinschaft). Diese entwickelt Module, welche den Funktionsumfang von Angular erweitern. Dies ermöglicht die Massachusetts Institute of Technology (MIT)-Lizenz, welche den Code frei einsehbar macht und erlaubt diesen zu verändern.[Bon17]

#### 4.4.1 Installation von Angular

Bevor man Angular installieren kann, muss erst einmal Node.js auf dem Entwicklersystem installiert werden. Es ist für die gängigen Betriebssysteme wie Windows, MacOS und Linux verfügbar. Es wird kostenfrei auf <https://nodejs.org/de/> Angeboten. Bei Node.js handelt es sich um eine Laufzeitumgebung. [Nod18] Dies ist vergleichbar mit Java, welches für die Entwicklung von Programmen benötigt wird, welche in Java geschrieben sind. Im Fall von Node.js wird JavaScript beziehungsweise (bzw.) EcmaScript verwendet. Mit der Installation von Node.js wird auch der Node Package Manager (NPM) installiert. Bei NPM handelt es sich um ein Tool mit welchem Programme, Frameworks uvm. installiert werden können. Die Angebotenen Dienste und Programme werden auf <https://www.npmjs.com> angeboten. Auch Angular wird hier angeboten und damit auch unzählige Erweiterungen für Angular. Möchte man Angular nun installieren geschieht das über das Command Line Interface. Hierbei kann mittels Parameter entschieden werden ob etwas für das ganze Betriebssystem global oder nur lokal für ein einzelnes Projekt installiert werden soll. Im Falle von Angular geschieht das global. Der Befehl zum installieren findet man ebenfalls auf <https://www.npmjs.com>. Werden keine weiteren Parameter angegeben, wird immer die neuste Version heruntergeladen.

```
1 $ npm install angular -g
```

#### 4.4.2 Commandlineinterface von Angular

Angular selbst liefert seit der Version 2 ein Command Line Interface mit. Mit diesem CLI ist es möglich Angular Projekte zu verwalten. Mit einer Zeile kann ein komplettes Projektsetup erzeugt werden. Mit Hilfe von Parameter kann angegeben werden, wie Anwendung gestylt wird. Dabei kann man beispielsweise wählen ob man die Anwendung mit Cascading Style Sheets (CSS) oder mit Syntactically Awesome Stylesheets (SASS) gestalten möchte. Bei SASS handelt es fast um die gleiche Syntax wie CSS. Es unterscheidet sich in dem Punkt, das es möglich ist Blöcke zu verschachteln und es gibt die Möglichkeit Variablen zu verwenden. Da noch kein Browser nativ SASS unterstützt, wird der SASS-Code in CSS-Code übersetzt. Angular wird über die Konsole verwendet indem ng angeführt wird. Im folgenden wird ein Beispielprojekt angelegt, welches zum Gestalten SASS verwendet.

```
1 $ ng new Beispielprojekt --style=sass
```



Name	Änderungsdatum	Typ	Größe
.git	17.08.2018 15:18	Dateiordner	
e2e	17.08.2018 15:17	Dateiordner	
node_modules	17.08.2018 15:18	Dateiordner	
src	17.08.2018 15:17	Dateiordner	
.angular-cli.json	17.08.2018 15:17	JSON-Datei	2 KB
.editorconfig	17.08.2018 15:17	EDITORCONFIG-D...	1 KB
.gitignore	17.08.2018 15:17	Textdokument	1 KB
karma.conf.js	17.08.2018 15:17	JavaScriptdatei	1 KB
package.json	17.08.2018 15:17	JSON-Datei	2 KB
package-lock.json	17.08.2018 15:18	JSON-Datei	422 KB
protractor.conf.js	17.08.2018 15:17	JavaScriptdatei	1 KB
README.md	17.08.2018 15:17	MD-Datei	2 KB
tsconfig.json	17.08.2018 15:17	JSON-Datei	1 KB
tslint.json	17.08.2018 15:17	JSON-Datei	3 KB

Abbildung 12: Projektordner: Angularprojekt

Name	Änderungsdatum	Typ	Größe
app	17.08.2018 15:17	Dateiordner	
assets	17.08.2018 15:17	Dateiordner	
environments	17.08.2018 15:17	Dateiordner	
favicon.ico	17.08.2018 15:17	Symbol	6 KB
index.html	17.08.2018 15:17	Opera Web Docu...	1 KB
main.ts	17.08.2018 15:17	TS-Datei	1 KB
polyfills.ts	17.08.2018 15:17	TS-Datei	4 KB
styles.SASS	17.08.2018 15:17	SASS-Datei	1 KB
test.ts	17.08.2018 15:17	TS-Datei	1 KB
tsconfig.app.json	17.08.2018 15:17	JSON-Datei	1 KB
tsconfig.spec.json	17.08.2018 15:17	JSON-Datei	1 KB
typings.d.ts	17.08.2018 15:17	TS-Datei	1 KB

Abbildung 13: Sourceordner in Angular Projekt

Name	Änderungsdatum	Typ	Größe
app.component.html	17.08.2018 15:17	Opera Web Docu...	2 KB
app.component.SASS	17.08.2018 15:17	SASS-Datei	0 KB
app.component.spec.ts	17.08.2018 15:17	TS-Datei	1 KB
app.component.ts	17.08.2018 15:17	TS-Datei	1 KB
app.module.ts	17.08.2018 15:17	TS-Datei	1 KB

Abbildung 14: App Ordner in Src-Ordner des Angularprojektes

Wie man in Abbildung 12 sehen kann wird nicht nur ein Projektsetup angelegt, sondern auch ein Ordner, in dem alle NPM Abhängigkeiten verwaltet werden und es wird auch

eine Versionverwaltung mit der Hilfe von Git übernommen, sofern Git auf dem System installiert ist. Es muss nur ein Repository angelegt und mit dem Projekt verbunden werden.

#### 4.4.3 TypeScript

Angular selbst ist TypeScript-basiert. Das bedeutet das Angular nicht direkt JavaScript verwendet, wie es noch in Angular.js gemacht worden ist, sondern es verwendet eine Art objektorientiertes JavaScript. Bei **Ecma!** (**Ecma!**) handelt es sich um eine Private Organisation, welche Programmiersprachen standardisiert. So handelt es sich bei JavaScript, wie es heute geläufig ist und von den meisten Browser unterstützt wird, eigentlich um EcmaScript 5. Seit wenigen Jahren existiert EcmaScript 6. Dies ist ein JavaScript welches Klassen, Interfaces, Arrow-Functions uvm. anbietet. Diese Features werden von vielen Entwickler gerne gesehen und lösen Probleme, mit welchen man schon länger zu tun hat und sie nur auf komplizierten Wegen lösen konnte. Allerdings unterstützen die meisten Browser noch nicht oder nur in Teilen den EcmaScript 6 Standard. An dieser stelle kommt TypeScript zum Einsatz. Eine Installation von TypeScript läuft, wie bei Angular, über NPM. Man kann nun alle Features von EcmaScript 6 verwenden und in eine TypeScript-Datei schreiben. TypeScript übersetzt diesen Code in eine EcmaScript 5 Datei, mit welcher jeder gängige Browser umgehen kann.

#### 4.4.4 Module

## HTTP

4.4.5 Components

4.4.6 Services

## **4.5 Backend**

4.5.1 Composer

4.5.2 Controller

4.5.3 REST

## **4.6 Security**

4.6.1 CORS

4.6.2 Token

4.6.3 Salt

## **4.7 Deploy**



## **5 Anwendung**

### **5.1 Routing Strategie**

### **5.2 E-Mail Reminder**



## **6 Diskussion der Ergebnisse**

### **6.1 Fazit**

### **6.2 Ausblick**





## Literatur

- [Apa] Apache Friends. *XAMPP*. URL: <https://www.apachefriends.org/de/index.html>.
- [Ash17] Ashiq Mohamed Akbar Ali. „Creating a website with Info-Terminal and Live CCTV Stream for the Smart Home Laboratory at the Hochschule Furtwangen University“. Masterthesis. Furtwangen: Hochschule Furtwangen, 30.08.2017.
- [Bon17] Bonny Kern. *Open Source Lizenzen - Die MIT-Lizenz*. 2017. URL: <https://wss-redpoint.com/open-source-lizenzen-die-mit-lizenz>.
- [Clo18] Mark Clow. *Angular 5 Projects: Learn to Build Single Page Web Applications Using 70+ Projects*. Berkeley, CA: Apress, 2018. ISBN: 978-1-4842-3278-1. DOI: 10.1007/978-1-4842-3279-8. URL: <http://dx.doi.org/10.1007/978-1-4842-3279-8>.
- [Dat18a] Datenbanken Verstehen. *Dritte Normalform*. 2018. URL: <http://www.datenbanken-verstehen.de/datenmodellierung/normalisierung/dritte-normalform/>.
- [Dat18b] Datenbanken Verstehen. *Zweite Normalform*. 2018. URL: <http://www.datenbanken-verstehen.de/datenmodellierung/normalisierung/zweite-normalform/>.
- [Gra14] Marcus Grande. *100 Minuten für Anforderungsmanagement: Kompaktes Wissen nicht nur für Projektleiter und Entwickler*. 2., aktualisierte Auflage. Wiesbaden: Springer Vieweg, 2014. ISBN: 978-3-658-06434-1. DOI: 10.1007/978-3-658-06435-8. URL: <http://dx.doi.org/10.1007/978-3-658-06435-8>.
- [Koc12] Daniel Koch. *Magento - Schritt für Schritt zum eigenen Online-Shop: [aktuell ab Version 1.7]*. München: Hanser, 2012. ISBN: 978-3-446-42307-7.
- [MFB13] MFB. *How do I allow a PUT file request on Nginx server?* 2013. URL: <https://stackoverflow.com/questions/16912270/how-do-i-allow-a-put-file-request-on-nginx-server>.
- [Mos12] Christian Moser. *User Experience Design: Mit Erlebniszentrierter Softwareentwicklung zu Produkten, die Begeistern*. X.media.press. Berlin: Springer, 2012. ISBN: 978-3-642-13363-3.
- [Nod18] Node.js. *Über Node*. 2018. URL: <https://nodejs.org/de/about/>.
- [NS17] Peter F.-J. Niermann und Andre M. Schmutte, Hrsg. *Managemententscheidungen: Methoden, Handlungsempfehlungen, Best Practices*. 2. Auflage. Wiesbaden: Springer Gabler, 2017. ISBN: 978-3-658-10181-7. DOI: 10.1007/978-3-658-10181-7. URL: <http://dx.doi.org/10.1007/978-3-658-10181-7>.

- [Ora] Oracle. *MySQL*. URL: <https://www.mysql.com/de/>.
- [Sch17] Annika Schach. *Storytelling: Geschichten in Text, Bild und Film*. Wiesbaden: Springer Fachmedien Wiesbaden, 2017. ISBN: 978-3-658-15232-1.
- [Tho18] Thomas Krenn AG. *Vergrößern der virtuellen Festplatte unter VMware*. 2018. URL: [https://www.thomas-krenn.com/de/wiki/Vergr%C3%B6%C3%9Fern\\_der\\_virtuellen\\_Festplatte\\_unter\\_VMware](https://www.thomas-krenn.com/de/wiki/Vergr%C3%B6%C3%9Fern_der_virtuellen_Festplatte_unter_VMware).
- [vmw18] vmware inc. *VM Ware Player*. 2018. URL: <https://www.vmware.com/de/products/workstation-player.html>.
- [Woo18] WooCommerce. *WooCommerce*. 2018. URL: <https://woocommerce.com/product-category/woocommerce-extensions/>.
- [WP ] WP Draw Attention. *WP Draw Attention*. Hrsg. von nsqua.red. URL: <https://wpdrawattention.com/>.
- [Zis16] Harald Zisler. *Computer-Netzwerke: Grundlagen, Funktionsweise, Anwendung*. 4., aktualisierte und erweiterte Auflage. Rheinwerk Computing. Bonn: Rheinwerk Computing, 2016. ISBN: 978-3-8362-4322-3.

## **Eidesstattliche Erklärung**

Ich versichere, dass ich die vorstehende Arbeit selbständig verfasst und hierzu keine anderen als die angegebenen Hilfsmittel verwendet habe. Alle Stellen der Arbeit die wörtlich oder sinngemäß aus fremden Quellen entnommen wurden, sind als solche kenntlich gemacht.

Die Arbeit wurde bisher in gleicher oder ähnlicher Form in keinem anderen Studiengang als Prüfungsleistung vorgelegt oder an anderer Stelle veröffentlicht.

Ich bin mir bewusst, dass eine falsche Erklärung rechtliche Folgen haben kann.

---

Winden im Elztal, den 17. August 2018 Helge Meiering