

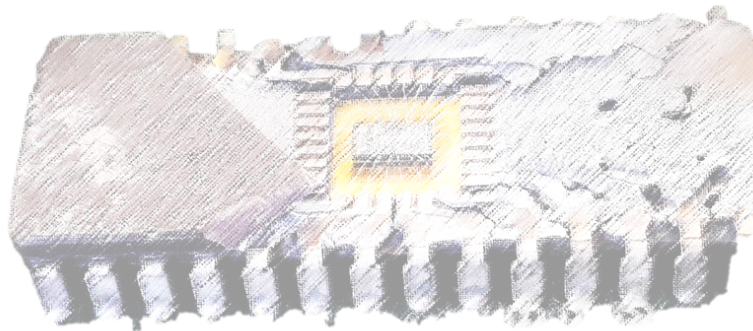
Höhere technische Schule für Informationstechnologie  
Shkolla e mesme profesionale private për teknologji informacioni

**Österreichische Schule Peter Mahringer**  
Shkolla Austriake Shkodër

## **Infotainment System**

**Diplomarbeit Nr. 20.08**

Klasse 5ay, Schuljahr 2019/20



Ausgeführt von: Irena Bala  
Aldo Sheldija

Projektbetreuer 1: Ing. Welat Abdall, MSc  
Projektbetreuer 2: DIPL.-ING (FH) Dominik Stocklasser MSc  
Projektbetreuer 3: Frenk Kasmi, MSc

Shkoder, 12. Dezember 2019

# Eidesstattliche Erklärung

Wir versichern, dass wir die vorliegende Arbeit selbstständig und ohne fremde Hilfe angefertigt haben. Wir haben uns keiner anderen als der im beigefügten Quellenverzeichnis angegebenen Hilfsmittel bedient. Alle Stellen, die wörtlich oder sinngemäß aus Veröffentlichungen entnommen wurden, sind als solche kenntlich gemacht.

Irena Bala

Ort, Datum

Unterschrift

Aldo Sheldija

Ort, Datum

Unterschrift

Approbation Datum u. Unterschrift	PrüferIn	IT-Koordinator/Direktion
--------------------------------------	----------	--------------------------

Sämtliche in dieser Diplomarbeit verwendeten personenbezogenen Bezeichnungen sind geschlechtsneutral zu verstehen.

# Kurzfassung

In der vorliegenden Diplomarbeit geht es um die Entwicklung eines Systems, wo die wichtigsten Ankündigungen des Tages auf einem Bildschirm dargestellt werden. Auf diese Weise wird eine Innovation im Erhalt der Informationen für die Schule erreicht.

Deswegen war es notwendig, ein System zu erstellen, wo alle diese Informationen in einer digitalisierten Form angezeigt werden. Dieses System heißt Infotainment-System. Der erstellte Prototyp besteht aus einem Raspberry PI Client und einem Raspberry PI Server. Im Raspberry PI Client befindet sich eine Website, die mit dem Server kommuniziert. Der Administrator kann sich bei der Webseite einloggen und um dort die verschiedenen Funktionalitäten zu verwalten. Er erhält die Möglichkeit, die Informationen auszuwählen, die er auf dem Bildschirm anzeigen lassen will, sowie auch das entsprechende Layout.

Die Informationen werden direkt aus einer von uns erstellten Datenbank ausgewählt, die schulrelevante Informationen wie Unterrichtsplan, Lehrerplan usw. enthält. Am Bildschirm können auch noch Kalenderinformationen (Termine, Olympiade), Wetterdaten und letzter Post von der Webseite der Schule dargestellt werden. Die Wetterdaten und den letzten Post bekommen wir mithilfe von APIs. Unser System kann mehrere Bildschirme mit unterschiedlichen Inhalten haben. Alle Bildschirme können auf relativ einfache und effiziente Weise von der Webseite aus verwaltet werden.

Das Infotainment-System ist auch zu Unterhaltungszwecken zu verwenden. Über Chatbot haben die Schüler die Möglichkeit, verschiedene Fotos zu posten, die auf dem Bildschirm angezeigt werden. Dies können Bilder des Alltags sein, die den Schultag auf die eine oder andere Weise interessanter machen.

# Abstract

This diploma thesis deals with the development of a system where the most important announcements of the day are displayed on a screen. In this way, an innovation in obtaining information for the school is achieved. The previous method of writing information on paper was very tiresome and time consuming. Every day, new papers were printed, which were then hung on the information board.

That was not an appealing task to the students. Therefore, it was necessary to create a system where all this information is displayed in a digitized form. This system is called infotainment system and consists of a Raspberry PI Client and a Raspberry PI Server. The Raspberry PI Client has a Website that communicates with the server. The administrator can log in to the website where they have the opportunity to select the information they want to display on the screen, as well as the corresponding layout. The information is selected directly from a database created by our team that contains school-related information such as lesson plan, teacher plan, and so on.

Calendar information (dates, Olympics), weather data and the latest posts from the school's website can also be displayed on the screen. Weather data and the latest posts can be received using APIs.

Our system can have multiple screens with different content. All screens can be managed in a relatively simple and efficient way from the website. The infotainment system can also be used for entertainment purposes. Chatbot allows students to post various photos which are going to be displayed on screen. The photographs will depict the daily school life of a Peter Mahringer student.

## Përmbledhje

Ideja e këtij projekti është realizimi i një sistemi, që ka si funksion paraqitjen e njoftimeve më të rëndësishme të ditës në një ekran. Në këtë mënyrë, do të arrihet një risi në mënyrën e transmetimit së informacioneve përkatëse për shkollën.

Metoda e mëparshme e përcjelljes së informacionit ka qenë tepër e lodhshme dhe kërkonte një kohë të konsiderueshme. Prandaj, ka qenë i nevojshëm krijimi i një sistemi ku të gjitha informacionet të paraqiten në një formë të dixhitalizuar. Prototipi që ne kemi krijuar përbëhet nga dy minikompjutera Raspberry PI, ku njëri kryen funksionin e klientit dhe tjetri atë të serverit. Tek klienti ndodhet një uebit që komunikon në mënyrë direkte me serverin. Administratori mund të logohet aty dhe të gjejë funksionalitete të ndryshme që i disponohen. Ai ka mundësinë që të zgjedhë informacionet që dëshiron të paraqesë në ekran, duke i selektuar ato direkt nga një bazë të dhënash.

Kjo bazë e dhënash është krijuar nga ne dhe mbart informacionet relevante për shkollën. Administratori mund të zgjedhë edhe një strukturë përkatëse për mënyrën e paraqitjes së këtyre të dhënave.

Në ekran do të tregohen edhe njoftime të rëndësishme si për shembull datat e olimpiadave apo pushimeve. Një funksionalitet i mëtejshëm konsiston në paraqitjen e të dhënave për motin apo edhe posti i fundit i uebit të shkollës. Kjo arrihet me ndihmë të API-ve. Një tjetër tipar inovativ është fakti që sistemi ynë mund të përdoret edhe për qëllime argëtuese. Nëpërmjet Chatbotit, që është një komponent i inteligjencës artificiale, studentët mund të postojnë vetë foto të ndryshme në ekran. Këto mund të jenë fotografi të jetës së përditshme, që në një mënyrë apo në një tjetër thyejnë monotoninë gjatë rutinës ditore shkollore.

# Danksagung

# Inhaltsverzeichnis

<b>1</b>	<b>Allgemeines</b>	<b>1</b>
1.1	Idee, Thema, Aufgabenstellung . . . . .	1
1.2	Team . . . . .	1
1.3	Allgemeines . . . . .	2
<b>2</b>	<b>Planung</b>	<b>3</b>
2.1	Projektziele . . . . .	3
2.1.1	Muss-Ziele . . . . .	3
2.1.2	Optionale-Ziele . . . . .	4
2.1.3	Nicht-Ziele . . . . .	4
2.2	Projektplanung . . . . .	4
2.2.1	Meilensteine . . . . .	4
2.2.2	Projektstrukturplan . . . . .	4
2.2.3	Arbeitspakete . . . . .	5
2.3	Projektmanagementmethode . . . . .	9
2.3.1	Structed Design . . . . .	9
2.3.2	Wasserfall . . . . .	9
<b>3</b>	<b>Dokumentation des Projektverlaufs</b>	<b>12</b>
3.1	Beschreibungen des Arbeitsverlaufs . . . . .	12
3.1.1	Aldo Sheldija . . . . .	12
<b>4</b>	<b>Name</b>	<b>16</b>
<b>5</b>	<b>Umsetzung - Aldo Sheldija</b>	<b>17</b>
5.1	Allgemeine Beschreibung . . . . .	17
5.1.1	Admin Webseite . . . . .	17
5.1.2	Client . . . . .	18
5.2	Technologien . . . . .	18
5.2.1	Admin Webseite . . . . .	18
5.2.2	Client . . . . .	19

---

5.3	Technische Lösung . . . . .	21
5.3.1	Basis Seite . . . . .	21
5.3.2	Displays . . . . .	21
5.3.3	Layouts . . . . .	21
5.3.4	Rechte in die Webseite . . . . .	21
5.3.5	Webseite auf mehrere Sprachen . . . . .	21



# Kapitel 1

## Allgemeines

### 1.1 Idee, Thema, Aufgabenstellung

Diese Diplomarbeit wird von zwei Schülerinnen der Österreichischen Schule “Peter Mahringer” in Shkodra geschrieben. Die Idee des Projekts ist, ein System zu entwickeln, wo die wichtigsten Benachrichtigungen des Tages für unsere Schule auf einem Bildschirm dargestellt werden. Das war notwendig, weil die frühere Arbeit sehr aufwendig und ungünstig war.

Dieses Thema ist sehr wichtig, weil in der heutigen Zeit es eine weitverbreitete Umgebung für die Anwendungen dieses Systems gibt. Bei vielen Unternehmen ist es erforderlich, die Informationen so schnell wie möglich darzustellen, damit die Kunden immer auf dem Laufenden sind. Momentan wird dieses System für die Schule angepasst.

Es wird eine Webseite mit Login programmiert, wobei der Administrator die Möglichkeit hat, verschiedene Informationen mit dem passenden Layout auf dem Bildschirm anzeigen zu lassen. Diese Informationen werden direkt aus einer selbsterstellten Datenbank selektiert. Zusätzlich, werden auch Kalenderinformationen wie z.B Olympiaden, wichtige Termine aber auch Wetterdaten oder der letzte Post von der Webseite dargestellt. Die Wetterdaten und der letzte Post von der Homepage der Schule werden mithilfe von APIs aus dem Internet geholt. Dieses System bietet viele Bildschirme mit unterschiedlichen Inhalten an. Der Administrator kann die Inhalte von fern für jeden Bildschirm ändern.

Eine weitere Funktionalität des Systems wird Chatbot sein. Chatbot ist ein sehr wichtiger Teil von künstlicher Intelligenz, deswegen war es wichtig, diesen Komponent zu involvieren.

### 1.2 Team

Das Projektteam besteht aus zwei Personen: Irena Bala und Aldo Sheldija. Irena Bala ist die Projektleiterin und Aldo Sheldija ist stellvertretender Projektleiter. Seit zwei Jahren sind sie in einer Klasse zusammen. Sie haben aber auch früher zusammengearbeitet, deswegen kennen sie sich gut. Die Mitglieder dieses Teams haben sich in der vierten Klasse für denselben Schwerpunkt entschlossen; nämlich für den Schwerpunkt

Systemtechnik. Sie haben gemeinsame Interesse an Softwareprogrammierung und an eingebettete Systeme. Allgemein haben sie auch andere Fähigkeiten. Aldo Sheldija hat in der Vergangenheit viele Websites erstellt, während Irena Bala viel Erfahrung mit Datenbanken hat. Auf diese Weise ergänzen sie ihre Kompetenzen gegenseitig, um das Projekt erfolgreich abzuschließen. Sie verfolgen den gleichen Zweck, um dieses Projekt optimal durchzuführen.

Die Aufgabenteilung in dieser Diplomarbeit ist folgende:

**Irena Bala** führt unter Zuhilfenahme eines Structed Design die Planung des Systems durch. Sie wird für die Konfiguration vom Raspberry PI Server, sowie für die Erstellung der SSL-Zertifikate und für die Einrichtung von der Datenbank verantwortlich sein. Das Design von der Datenbank, die Einrichtung des Systems und die API-Integration für Wetterdaten werden auch von ihr durchgeführt. Zusätzlich wird sie Chatbot einrichten und die Sperr-Funktion von unpassenden Bildern programmieren.

**Aldo Sheldija** ist zuständig für die Planung des Systems mithilfe von einem Big Picture und für die Entwicklung von dem Logo. Er wird für die Konfiguration vom Raspberry PI Client, sowie für die Erstellung des Admin-Panels und für die Erstellung von Bilder-Funktionalität beim Chatbot verantwortlich sein. Der Entwurf der Datenbank und die API-Integration für den letzten Post von der Webseite der Schule wird von ihm durchgeführt. Zusätzlich wird er das System aufbauen und testen.

## 1.3 Allgemeines

Damit das Projekt optimal umgesetzt werden kann, werden einige zusätzliche Funktionen berücksichtigt. Beispielsweise wird die Website mit einem SSL-Zertifikat verschlüsselt. Bei einer Unterbrechung der Netzwerkverbindung werden anstelle eines schwarzen Bildes, die zuletzt dargestellten Informationen für die Dauer dieser Unterbrechung am Bildschirm dargestellt. Die Lösungen dafür werden in den folgenden Kapiteln beschrieben.

# Kapitel 2

## Planung

### 2.1 Projektziele

#### 2.1.1 Muss-Ziele

- Es werden Bilder auf der Webseite hochgeladen und automatisch angezeigt.
- Es werden Kalenderinformationen wie Feiertage, wichtige Termine (Olympiaden usw.) auf dem Bildschirm dargestellt.
- Der Stundenplan der Klassen wird angezeigt.
- Der aktuelle Supplierplan wird angezeigt.
- Das System soll ein flexibles Layout haben. Ein Vollbild, zwei oder mehr Panels, Größen und Platzierungen wie Informationen auf Bildschirmen werden angezeigt.
- Der tägliche Wetterbericht wird angezeigt.
- Die aktuelle Uhrzeit und das Datum werden auf dem Bildschirm dargestellt.
- Die leeren Klassen werden angezeigt.
- Verschiedene Bildschirmgrößen werden unterstützt.
- Ein SSL Zertifikat für die Client-Serververbindung wird eingerichtet.
- Es werden mehrere Anzeigen mit verschiedenen Inhalten angeboten.
- Chatbot wird auch implementiert, um die Bilder hochzuladen und sperren.
- Es werden unterschiedliche Anzeigzeiten für die unterschiedlichen GUI Bereiche unterstützt.
- Es wird ein funktionierendes System auch im offline Betrieb erreicht.

### 2.1.2 Optionale-Ziele

- Videos können hochgeladen und freigegeben werden. Die gängigsten Videoformate werden unterstützt.
- Das System kann auf mehrere Sprachen angeboten werden.
- Die Notfallwarnungen können dargestellt werden.

### 2.1.3 Nicht-Ziele

- Es wird Interaktion des Benutzers mit dem Bildschirm geben.
- Der Login auf die Webseite mit Gesichtserkennung und Office365 wird nicht möglich sein.
- Die Dateien können nicht von anderen Plattformen geholt werden.
- Es wird Audiounterstützung geben.

## 2.2 Projektplanung

### 2.2.1 Meilensteine

Die Meilensteine sind Orientierungspunkte, die am Ende der Projektplanung definiert werden. Durch diese Punkte wird der Weg vom Beginn bis zum Ende des Projekts in strukturierter Form beschrieben. Die Meilensteine sind sehr wichtig, weil sie die Weiterführung des Projekts bestimmen.

Datum	Meilenstein
04.10.2019	Implementierung der Anzeige-Struktur
18.10.2019	Initialisierung des Systems (Datenbank)
31.10.2019	Raspberry PI Konfiguration (Server, Client)
29.11.2019	Erstellung der Admin-Webseite
29.11.2019	Bilder-Verwaltung
05.12.2019	Chatbot Implementierung
20.12.2019	Verschlüsselung der Webseite durch SSL-Zertifikate

Tabelle 2.1: Meilensteine

### 2.2.2 Projektstrukturplan

Bei einem Projektstrukturplan ist das Projekt in Teilaufgaben und Arbeitspaketen unterteilt. Jedes Arbeitspaket gibt eine Aufgabe an, für die einer von der Gruppe verantwortlich ist. Der Projektstrukturplan ist ein sehr wichtiger Teil der Projektplanung, weil es bei der Definition der Ziele und bei der Darstellung der Arbeitspakete in strukturierter Form hilft.

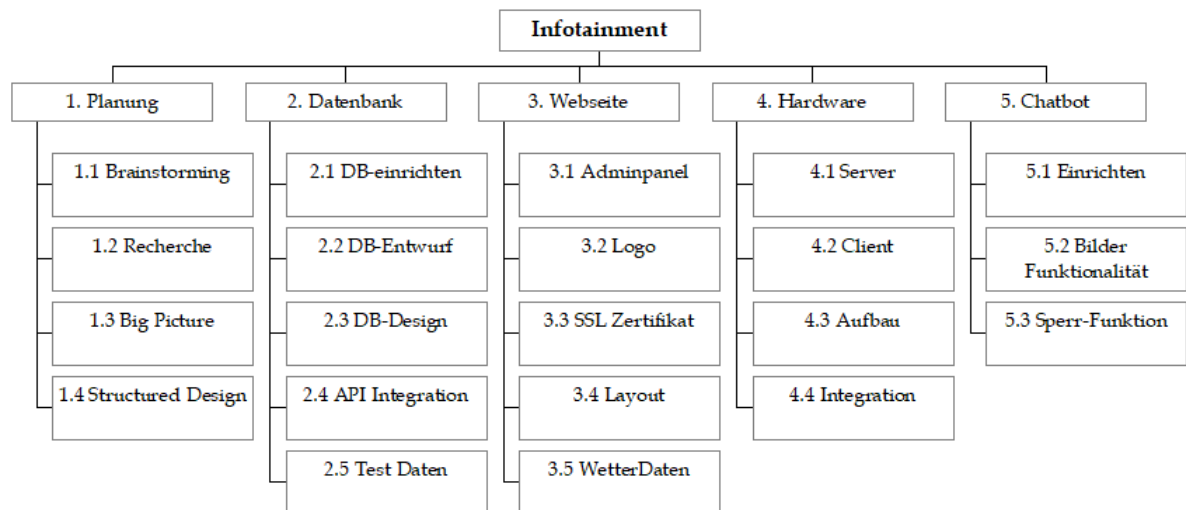


Abbildung 2.1: Projektstrukturplan

### 2.2.3 Arbeitspakete

Die Arbeitspakete sind die Hauptelemente eines Projekts, die ein definiertes Ergebnis, Start-und Endzeitpunkt haben. Sie werden nicht weiter unterteilt. Jedes Mitglied des Projektteams ist für einige Arbeitspakete zuständig.

<b>Arbeitspaket AP 1.1: Brainstorming</b>	<b>Verantwortung: Aldo</b>
Beginn: 16/10/2019 Ende: 22/10/2019 Mitarbeit: Aldo, Irena	Beschreibung: - Idee suchen - Idee Generierung - Idee sammeln - Idee bearbeiten
Arbeitspaket AP 1.2: Recherche	Verantwortung: Irena
Beginn: 24/10/2019 Ende: 29/10/2019 Mitarbeit: Aldo, Irena	Beschreibung: - Informationen für ähnliche Systeme suchen - Informationen finden - Informationen bearbeiten
Arbeitspaket AP 1.3: Big Picture	Verantwortung: Aldo
Beginn: 30/09/2019 Ende: 03/10/2019 Mitarbeit: Aldo, Irena	Beschreibung: - Umwelt und Rahmenbedingungen definieren - Daten bzw. Steuerungsanweisungsbeschreibungen - Iterationen - Digitalisierung
Arbeitspaket AP 1.4: Structured Design	Verantwortung: Irena

Beginn: 03/10/2019 Ende: 06/10/2019 Mitarbeit: Aldo, Irena	Beschreibung: - Festlegung der Schnittstellen und Grenzen jedes Modules - Schnittstellenbeschreibung - Funktionsbeschreibung - Modes - Iterationen - Digitalisierung
Arbeitspaket AP 2.1: DB einrichten	Verantwortung: Irena
Beginn: 07/10/2019 Ende: 13/10/2019 Mitarbeit: Irena	Beschreibung: - DB erstellen - User anlegen - Rechte vergeben
Arbeitspaket AP 2.2: DB Entwurf	Verantwortung: Aldo
Beginn: 14/10/2019 Ende: 18/10/2019 Mitarbeit: Aldo	Beschreibung: - Konzeptueller Entwurf - Implementationsentwurf - Physischer Entwurf
Arbeitspaket AP 2.3: DB-Design	Verantwortung: Irena
Beginn: 22/10/2019 Ende: 26/10/2019 Mitarbeit: Aldo, Irena	Beschreibung: Entwurf der Tabellen für: - Administrator - Bilder - Klassenplan - Lehrerplan - Supplierplan - Stundenplan - Lehrer - Wetter
Arbeitspaket AP 2.4: API Integration	Verantwortung: Aldo
Beginn: 10/12/2019 Ende: 13/12/2019 Mitarbeit: Aldo	Beschreibung: - API Dokumentation lesen - API Funktionalität anschauen - API als Schnittstelle verwenden - Daten aus vorhandenen und älteren Datenquellen durch API bekommen
Arbeitspaket AP 2.5: Test Daten	Verantwortung: Aldo
Beginn: 02/12/2019 Ende: 22/12/2019 Mitarbeit: Aldo	Beschreibung: - Dummy Daten erzeugen - Tabelle füllen - Proben durchführen
Arbeitspaket AP 3.1: Admin Panel	Verantwortung: Aldo

Beginn: 01/11/2019 Ende: 28/11/2019 Mitarbeit: Aldo, Irena	Beschreibung: - Login - Einstellungen - Gewünschte - Informationen selektieren - Informationen anzeigen lassen
Arbeitspaket AP 3.2: Logo	Verantwortung: Aldo
Beginn: 09/10/2019 Ende: 13/10/2019 Mitarbeit: Aldo	Beschreibung: - Wortmarke finden - Symbol Marke finden - Kombination (Wort und Symbol) auswählen - Entwicklung - Entwurf - Digitalisierung
Arbeitspaket AP 3.3: SSL Zertifikate	Verantwortung: Irena
Beginn: 11/12/2019 Ende: 13/12/2019 Mitarbeit: Irena	Beschreibung: - SSL Zertifikate Keys anlegen - SSL Zertifikate einrichten - HTTPS verwenden
Arbeitspaket AP 3.4: Layout	Verantwortung: Irena
Beginn: 16/12/2019 Ende: 22/12/2019 Mitarbeit: Aldo, Irena	Beschreibung: - Bootstrap - Responsives - Webdesign - Navigationsleiste - Bilder - Typographie
Arbeitspaket AP 3.5: Wetterdaten	Verantwortung: Irena
Beginn: 25/11/2019 Ende: 30/11/2019 Mitarbeit: Irena	Beschreibung: - API Integration - Format wählen (XML, JSON) - Datenbank erstellen - Daten auf die Webseite anzeigen lassen
Arbeitspaket AP 4.1: Server	Verantwortung: Irena
Beginn: 30/10/2019 Ende: 3/11/2019 Mitarbeit: Irena	Beschreibung: - Betriebssystem installieren - Programme herunterladen und installieren - Apache Server einrichten - MySQL Datenbank anlegen
Arbeitspaket AP 4.2: Client	Verantwortung: Aldo

Beginn: 21/10/2019 Ende: 2/11/2019 Mitarbeit: Aldo	Beschreibung: - Betriebssystem installieren - Programme herunterladen und installieren - Apache Server einrichten - MySQL Datenbank anlegen
Arbeitspaket AP 4.3: Aufbau	Verantwortung: Aldo
Beginn: 04/11/2019 Ende: 08/11/2019 Mitarbeit: Aldo, Irena	Beschreibung: - RaspberryPi Gehäuse - RaspberryPi am Bildschirm anbringen - Bildschirm an die Wand hängen
Arbeitspaket AP 4.4: Einrichtung des Systems	Verantwortung: Irena
Beginn: 11/11/2019 Ende: 30/11/2019 Mitarbeit: Irena	Beschreibung: - WLAN Verbindung - Verbindung mit der Datenbank
Arbeitspaket AP 5.1: Chatbot Einrichten	Verantwortung: Irena
Beginn: 18/11/2019 Ende: 24/11/2019 Mitarbeit: Irena	Beschreibung: - API Dokumentation anschauen - Account anlegen - Registrieren - Bilder, Videos posten
Arbeitspaket AP 5.2: Bilder Funktionalität	Verantwortung: Aldo
Beginn: 25/11/2019 Ende: 01/12/2019 Mitarbeit: Aldo	Beschreibung: - Bilder hochladen - Bilder in der Datenbank speichern - Bilder von der Datenbank selektieren und auf der Website anzeigen lassen
Arbeitspaket AP 5.3: Sperr-Funktion	Verantwortung: Irena
Beginn: 03/12/2019 Ende: 07/12/2019 Mitarbeit: Irena	Beschreibung: - ID des Bildes schicken - Das Telefonnummer sperren, damit er keine Bilder mehr hochladen kann - Das Bild am Bildschirm löschen - Das Bild aus der Datenbank entfernen

Tabelle 2.2: Arbeitspakete



## 2.3 Projektmanagementmethode

### 2.3.1 Structed Design

Structed Design ist eine systematische Methode, die verwendet wird, um eine Software so gut wie möglich zu beschreiben. Mit dieser Methode wird das Architekturdesign, die fließenden Daten und Signale, sowie alle Schnittstellen sehr leicht und deutlich beschrieben. Ein weiterer Vorteil dieser Methode ist, dass es sehr leicht zu verstehen ist. In diesem Projekt wurde Structed Design verwendet, aufgrund der guten Möglichkeit, dass sie bietet, Iterationen zu verwenden und tiefer in die wichtigsten Ebenen der Arbeit zu gehen.

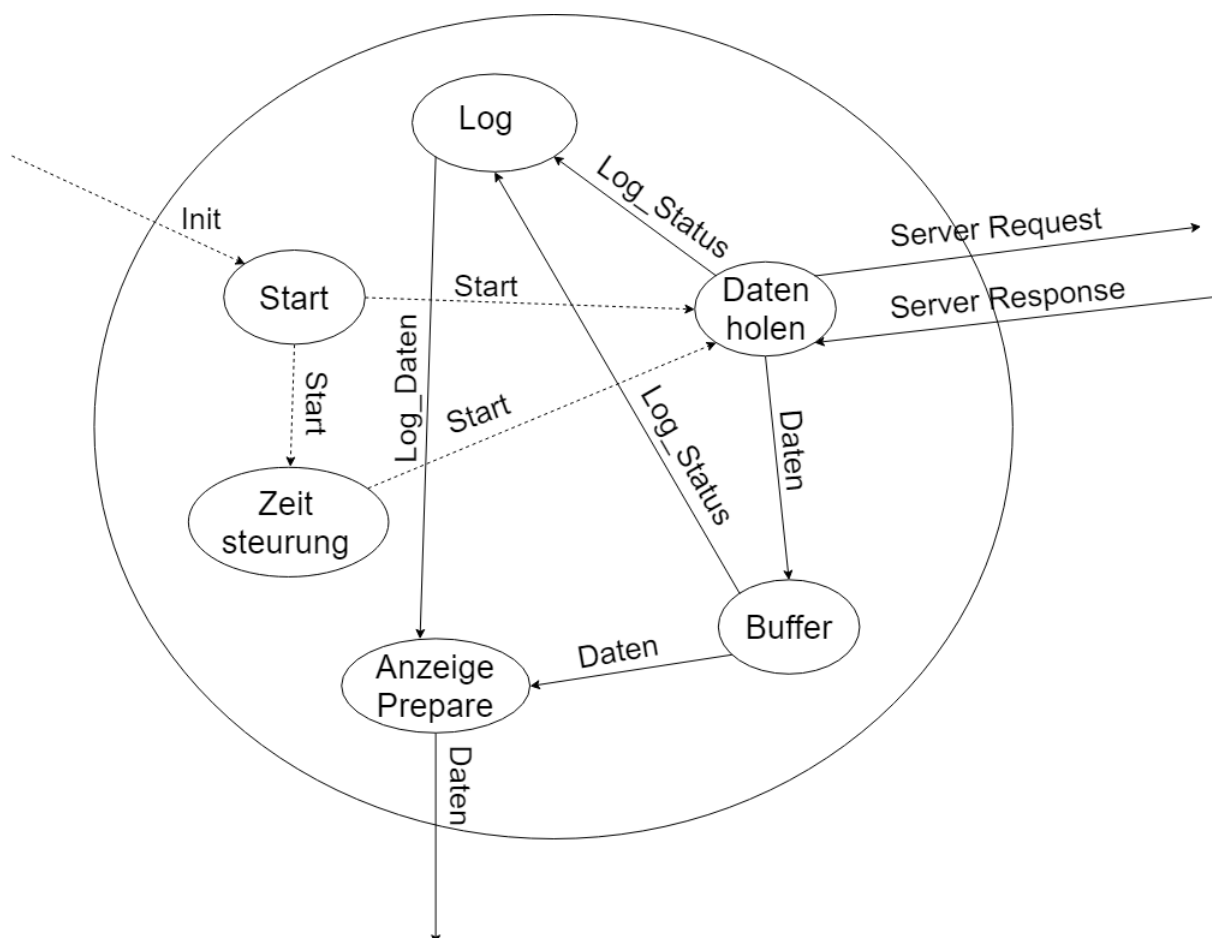


Abbildung 2.2: Structed Design - Client

### 2.3.2 Wasserfall

Das Wasserfall Modell (Abbildung 2.3.2) ist die zweite Projektmanagement Methode, die von uns gewählt wurde. Diese Methode teilt die wichtigsten Prozesse und Phasen

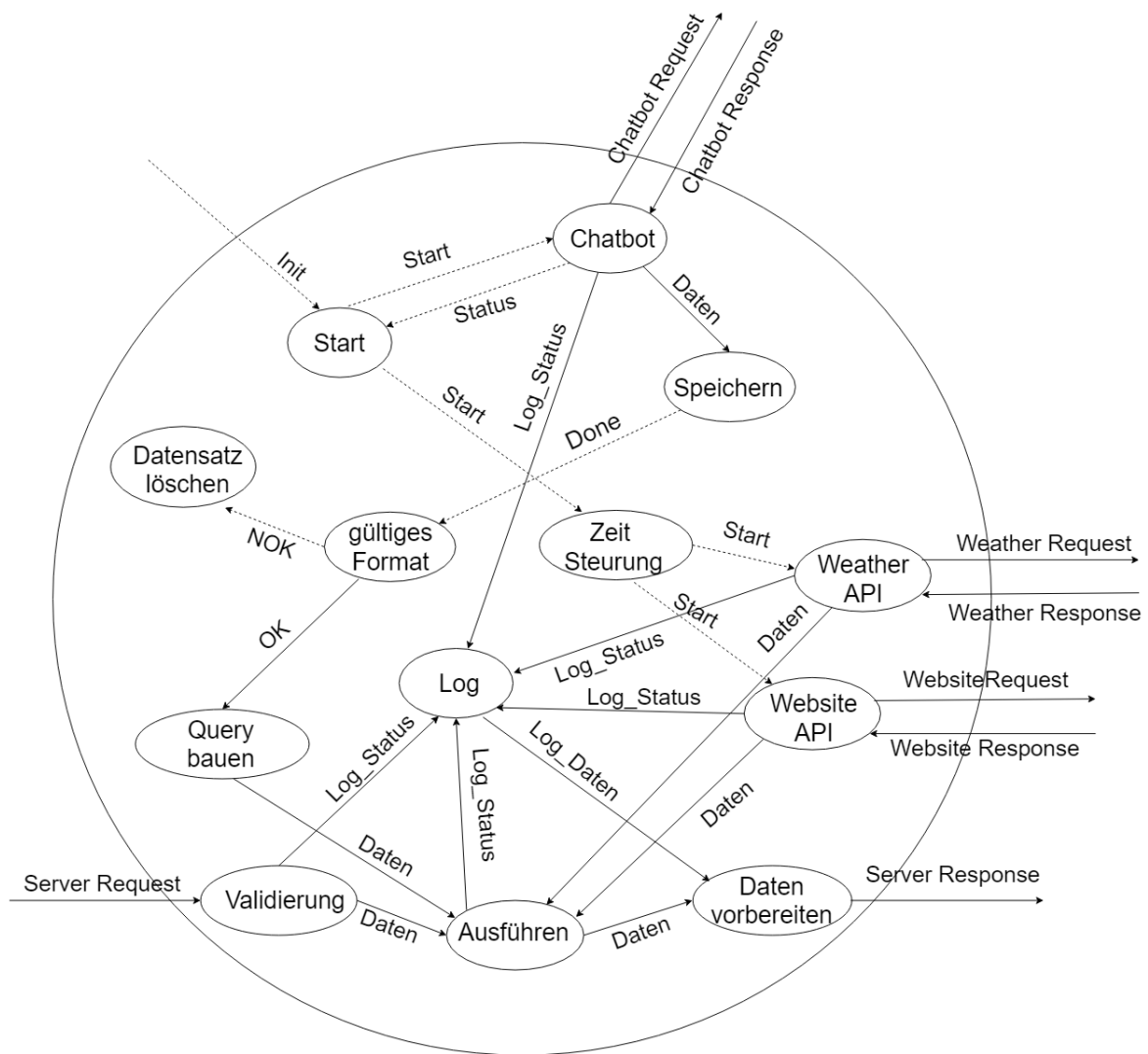


Abbildung 2.3: Structed Design - Server

des Projekts so, dass sie sequenziell nacheinander bearbeitet werden und dass jede Phase von der vorherigen Phase abhängig ist. Die wichtigsten Schritte eines Wasserfallmodells sind: Anforderung, Analyse, Entwurf, Realisierung, Test, Systemintegration und Systemabnahme. Das Projekt geht durch diese sieben Schritte bzw. Phasen sequentiell durch, bis die Arbeit abgeschlossen ist. Der Name dieses Modells leitet sich aufgrund des Fließens des Projektes durch alle Phasen ab. Das Modell wird sehr gerne bei der Erstellung von Webseiten oder bei Entwurf von Datenbanken angewendet. Obwohl diese Methode sehr langwierig sein kann und nicht sehr flexibel ist (Fehlerkorrektur), ermöglicht es eine sehr klare und leichte Planung und Organisation der Arbeit, sowie eine gute Zielabgrenzung und Aufwandsabschätzung. Die oben genannten Merkmale sind der Grund, warum diese Methode für dieses Projekt ausgewählt wurde.

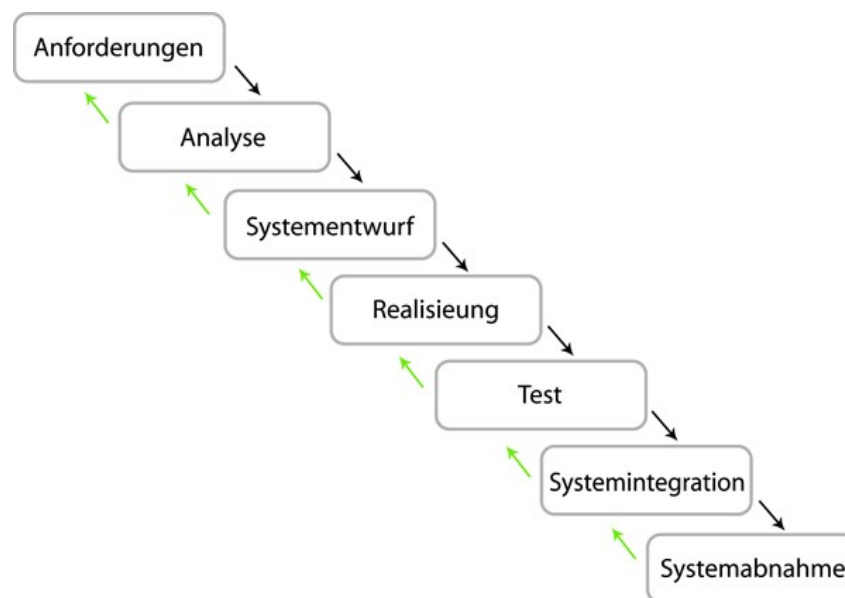


Abbildung 2.4: Wasserfallmodell

# Kapitel 3

## Dokumentation des Projektverlaufs

### 3.1 Beschreibungen des Arbeitsverlaufs

#### 3.1.1 Aldo Sheldija



# Detailed report

2019-09-10 - 2019-12-08

Total 111 h 20 min

Date	Description	Duration	User
09-10	<b>Recherche für andere ähnliche Projekte, um die Zeile unseres Projekts zu definieren</b>	<b>1:35:00</b>	Aldshe14
	Infotainment - [Dokumentation]	19:25-21:00	
09-11	<b>#Dokumentation - DA-Antrag ausfüllen</b>	<b>1:00:00</b>	Aldshe14
	Infotainment - [Dokumentation]	17:15-18:15	
09-12	<b>#Dokumentation - DA-Antrag verbessern</b>	<b>1:30:00</b>	Aldshe14
	Infotainment - [Dokumentation]	18:00-19:30	
09-16	<b>#AdminPanel - Recherche für die Features der AdminPanel</b>	<b>1:00:00</b>	Aldshe14
	Infotainment - [AdminPanel]	17:30-18:30	
09-19	<b>#AdminPanel - Recherche für die Darstellung von Informationen in verschiedene GUI Bereiche</b>	<b>1:40:00</b>	Aldshe14
	Infotainment - [AdminPanel]	18:00-19:40	
09-23	<b>#Logo - Erste Konzept</b>	<b>3:00:00</b>	Aldshe14
	Infotainment - [Logo]	18:30-21:30	
09-25	<b>#Logo - Verbesserung der Logo</b>	<b>1:30:00</b>	Aldshe14
	Infotainment - [Logo]	17:30-19:00	
09-26	<b>#Logo - Digitalisierung des Logo</b>	<b>2:30:00</b>	Aldshe14
	Infotainment - [Logo]	20:30-23:00	
09-27	<b>#AdminPanel - Grunstruktur der PHP Dateien</b>	<b>4:30:00</b>	Aldshe14
	Infotainment - [AdminPanel]	18:00-22:30	
09-28	<b>#Display - Layout 1 / 5 Sections Layout mit Kopf und Fußzeile</b>	<b>1:30:00</b>	Aldshe14
	Infotainment - [Display]	10:20-11:50	
09-28	<b>#Display - Layout 2 / 3 Sections Layout mit Kopf und Fußzeile</b>	<b>1:15:00</b>	Aldshe14
	Infotainment - [Display]	11:20-12:35	
09-29	<b>#Client - Raspbian Image auf SD-Karte spielen und das System konfigurieren</b>	<b>5:25:00</b>	Aldshe14
	Infotainment - [Client]	15:00-20:25	
10-02	<b>#Datenbank - ERD Diagramm</b>	<b>3:00:00</b>	Aldshe14
	Infotainment - [Datenbank]	17:30-20:30	
10-03	<b>#Datenbank - Tabellen erstellen</b>	<b>2:45:00</b>	Aldshe14
	Infotainment - [Datenbank]	18:15-21:00	
10-05	<b>#AdminPanel - Datatables Plug-in für jQuery konfiguriert</b>	<b>6:00:00</b>	Aldshe14
	Infotainment - [AdminPanel]	13:00-19:00	
10-08	<b>#Client - Apache, MariaDB Server, PHP, Git, VIM Installiert am System</b>	<b>2:00:00</b>	Aldshe14
	Infotainment - [Client]	16:00-18:00	

<b>10-10</b>	<b>#AdminPanel - Fontawesome Icons integriert im Offline Betrieb</b>	<b>1:00:00</b>	Aldshe14
	Infotainment - [AdminPanel]	18:00-19:00	
<b>10-11</b>	<b>#AdminPanel - Login Bereich</b>	<b>6:30:00</b>	Aldshe14
	Infotainment - [AdminPanel]	14:30-21:00	
<b>10-12</b>	<b>#AdminPanel - Die Verbindung wurde mit der neuen Datenbank aktualisiert, die auf dem Server in der Schule gehostet wird</b>	<b>0:30:00</b>	Aldshe14
	Infotainment - [AdminPanel]	16:30-17:00	
<b>10-12</b>	<b>#Datenbank - Export gemacht und in Git Repository gespeichert, um diese Datenbank in der neuen Server zu importieren</b>	<b>1:00:00</b>	Aldshe14
	Infotainment - [AdminPanel]	17:00-18:00	
<b>10-16</b>	<b>#Display - Datum und Uhrzeit mit jQuery darstellen auf Deutsch</b>	<b>1:00:00</b>	Aldshe14
	Infotainment - [Display]	19:00-20:00	
<b>10-18</b>	<b>#AdminPanel - Admin und Einfacher Benutzer unterschieden</b>	<b>2:00:00</b>	Aldshe14
	Infotainment - [AdminPanel]	11:00-13:00	
<b>10-19</b>	<b>#AdminPanel - Passwort zurücksetzen</b>	<b>4:45:00</b>	Aldshe14
	Infotainment - [AdminPanel]	17:45-22:30	
<b>10-22</b>	<b>#AdminPanel - Stundplan für eine bestimmte Tag anzeigen mit Hilfe von einer Dropdown</b>	<b>3:00:00</b>	Aldshe14
	Infotainment - [AdminPanel]	19:00-22:00	
<b>10-24</b>	<b>#AdminPanel - Supplierlehrer bei der Supplierplan zeigen, die die Kriterien erfüllen</b>	<b>1:00:00</b>	Aldshe14
	Infotainment - [AdminPanel]	18:30-19:30	
<b>10-25</b>	<b>#AdminPanel - Supplierplan für ein Lehrer machen und bestimmte Stunden</b>	<b>4:30:00</b>	Aldshe14
	Infotainment - [AdminPanel]	18:00-22:30	
<b>10-27</b>	<b>#AdminPanel - Neue User anlegen mit unterschiedlichen Rechten und löschen</b>	<b>6:00:00</b>	Aldshe14
	Infotainment - [AdminPanel]	16:00-22:00	
<b>10-30</b>	<b>#AdminPanel - Fehlende Lehrer selektieren aus der Stundenplantabelle ( Editieren und Löschen )</b>	<b>2:20:00</b>	Aldshe14
	Infotainment - [AdminPanel]	17:40-20:00	
<b>11-02</b>	<b>#Display - Welcome Page hinzugefügt, wenn der Player noch nicht aktiviert wurde</b>	<b>5:30:00</b>	Aldshe14
	Infotainment - [Display]	14:30-20:00	
<b>11-03</b>	<b>#AdminPanel - Supplierplan ändern und löschen</b>	<b>3:30:00</b>	Aldshe14
	Infotainment - [AdminPanel]	19:00-22:30	
<b>11-06</b>	<b>#AdminPanel - Logout Funktionalität</b>	<b>2:00:00</b>	Aldshe14
	Infotainment - [AdminPanel]	18:40-20:40	
<b>11-07</b>	<b>#AdminPanel - Passwort zurücksetzen beim Login Updaten</b>	<b>1:30:00</b>	Aldshe14
	Infotainment - [AdminPanel]	13:30-15:00	
<b>11-09</b>	<b>#Display - Letzte Post der Webseite der Schule anzeigen</b>	<b>4:45:00</b>	Aldshe14
	Infotainment - [Display]	14:30-19:15	
<b>11-11</b>	<b>#Client - Chromium Installieren für die Anzeige von Informationen</b>	<b>1:00:00</b>	Aldshe14
	Infotainment - [Client]	16:45-17:45	

<b>11-13</b>	<b>#AdminPanel - Kopfzeile der Tabele Fett darstellen Updated</b>	<b>1:00:00</b>	Aldshe14
	Infotainment - [AdminPanel]	15:00-16:00	
<b>11-15</b>	<b>#Client - Webseite beim Hochfahren öffnen</b>	<b>2:00:00</b>	Aldshe14
	Infotainment - [Client]	15:47-17:47	
<b>11-16</b>	<b>#AdminPanel - Stundenplan importieren und alte Daten löschen</b>	<b>2:00:00</b>	Aldshe14
	Infotainment - [AdminPanel]	18:10-20:10	
<b>11-21</b>	<b>#Datenbank - Automatische ausführung von PHP Sceipts mit hilfe von Cronjobs um die neue Posts von der Webseite zu bekommen</b>	<b>1:30:00</b>	Aldshe14
	Infotainment - [Datenbank]	17:50-19:20	
<b>11-26</b>	<b>#Display - Logo im Header platzieren und darstellen</b>	<b>1:00:00</b>	Aldshe14
	Infotainment - [Display]	20:00-21:00	
<b>11-29</b>	<b>#AdminPanel - Bilder Hochladen</b>	<b>2:00:00</b>	Aldshe14
	Infotainment - [AdminPanel]	18:45-20:45	
<b>11-30</b>	<b>#Display - Anzeige von Bildern</b>	<b>2:00:00</b>	Aldshe14
	Infotainment - [Display]	16:12-18:12	
<b>12-01</b>	<b>#AdminPanel - Settings Seite erstellt um die API-Key, Telegram token und URL der Webseite zu editieren</b>	<b>3:00:00</b>	Aldshe14
	Infotainment - [AdminPanel]	17:05-20:05	
<b>12-04</b>	<b>#AdminPanel - Anzeige hinzufügen, editieren und löschen</b>	<b>2:00:00</b>	Aldshe14
	Infotainment - [AdminPanel]	18:00-20:00	
<b>12-07</b>	<b>#AdminPanel - Kalendarinformationen hinzufügen</b>	<b>2:50:00</b>	Aldshe14
	Infotainment - [AdminPanel]	09:52-12:42	

Created with toggl.com

# Kapitel 4

Name



# Kapitel 5

## Umsetzung - Aldo Sheldija

In diesem Kapitel wird die Vorgehensweise der Technische Lösung bei der Admin Webseite und Client erklärt. Weiter werden auch die verwendeten Technologien aufgelistet.

### 5.1 Allgemeine Beschreibung

#### 5.1.1 Admin Webseite

Ein sehr wichtiger Teil dieser Diplomarbeit ist auch die Admin Webseite. In dieser Webseite hat der Admin die Möglichkeit, alle Anzeigen zu verwalten und den mit unterschiedlichen Inhalten zu befüllen. Durch diese Webseite wird die Verwaltung einfacher und übersichtlicher gemacht. Dort wird die Gelegenheit zu dem Admin gegeben, Anzeige hinzufügen, ändern und löschen, verschiedenen Layouts für unterschiedliche Anzeigen anzuordnen, Bilder zu verwalten und Benutzer zu administrieren. Ein anderes Feature zu dieser Webseite ist auch der Supplierplan. Hier kann der Stundepan für bestimmte Tagen gemacht werden, damit er in die Anzeige dargestellt werden kann. Es wurde so gemacht, dass der Admin nur in ein paar Klicks das erledigen kann. Da unser Diploma nicht nur aus dieser Webseite besteht, wird unsere Webseite eine Schnittstelle sein, um die API-Key von Wetterbericht, URL der Webseite, von dem die letzten Posts gespeichert werden, und Chatbot Token zu verwalten.

Diese Webseite wurde so konzipiert, dass der Benutzer, der das verwenden wird, die Informationen und die Bedienung dieser Webseite sehr leicht und ohne Probleme folgt. Die Website ist auch in mehrere Sprachen angeboten, so dass noch leichter wird.

Wie gesehen, der Inhalt in dieser Webseite ändert ständig und deshalb ist unsere Webseite, eine dynamische Website, die mit PHP und MySQL gemacht wird. Für die Darstellung und das Design wurde Bootstrap verwendet. Um die Tabellen am besten darzustellen wurde auch Datables, ein JavaScript Plugin, verwendet. Regex wurde verwendet um die Validierung von unterschiedlichen User Input zu überprüfen und sicherstellen, dass der Input richtig ist.

Damit nicht jeder diesen Inhalten ändern und löschen kann, ist diese Webseite auch mit einem Login Bereich geschützt. Nur bestimmte Personen, abhängig von welchem Rechten sie haben können dann die zugehörigen Änderungen machen. Falls ein Be-

nutzer das Passwort vergisst, hat er die Möglichkeit das Passwort zurückzusetzen und wartet bis der Admin das bestätigt hat.

Der Zugriff auf dieser Webseite ist auf jedes Gerät und von jedem Ort aus möglich. Um diese Webseite anzuschauen ist nur ein Internet Verbindung notwendig.

### 5.1.2 Client

Zu diese Diplomarbeit gehört auch der Client dazu. Als Client wird bezeichnet die Anzeige, in dem die Informationen dargestellt werden. Der Client besteht aus einem RaspberryPi, der mit Internet verbindet wird (WLAN<sup>1</sup> oder LAN<sup>2</sup> Verbindung), und mit dem Server kommunizieren wird. Der RaspberryPi hat einen HDMI<sup>3</sup> Anschluss, der mit einem Bildschirm verbindet wird. Um das System mehr kompatibel und umfassend, werden unterschiedliche Bildschirmgrößen unterstützt. Der Anzeige wird automatisch registriert in der Datenbank mithilfe von einem Skript, der die Informationen beim Hochfahren des Geräts zu Datenbank schickt. Diese Anzeige wird beim Hochfahren des Betriebssystems die Webseite von der Anzeige öffnen und das Anzeigen lassen. Diese Webseite wird im Vollbild geöffnet. Zusätzlich werden die Bildschirmaustastung, Bildschirmschoner und Energieverwaltungssystem ausgeschaltet, damit die Anzeige immer eingeschaltet bleibt. Damit das System nicht abhängig von dem Internet ist, werden die Informationen für diese Anzeige von der Datenbank, die im Server liegt, kopiert und auf der lokalen Datenbank gespeichert.

In dem Client laufen eine Apache Server, MySQL Datenbank und PHP, die für die Anzeige notwendig sind um die Informationen anzuzeigen. Falls der Anzeige noch nicht von der Admin freigegeben wurde, wird die Anzeige zu eine Welcome Seite umgeleitet, in dem nur die MAC-Adresse der Anzeige dargestellt wird.

## 5.2 Technologien

Dieses Kapitel setzt sich mit der Beschreibung der Technologien, die für die Erstellung des Admin Webseite und der Client verwenden wurden auseinander.

### 5.2.1 Admin Webseite

**Bootstrap** (Abb. 5.1) ist eine Open-Source und Kostenloses CSS<sup>4</sup> Framework, die für die Herstellung von unterschiedlichen responsiven Webseiten hilft. Dieses Framework ist nicht nur aus CSS basiert, sondern die enthält auch JavaScript, die die Interaktion von dynamischer Webseite erhöht. Bootstrap stellt fertige CSS Klassen zur Verfügung, die für unterschiedliche Bildschirmgrößen automatisch angepasst werden. Das wird durch ein Grid System ermöglicht.

---

<sup>1</sup>Wireless LAN

<sup>2</sup>Local Area Network

<sup>3</sup>High Definition Multimedia Interface

<sup>4</sup>Cascading Style Sheet



Abbildung 5.1: Bootstrap Logo

**PHP** (Abb. 5.2) stand für Personal Home Page Tools und jetzt heißt PHP Hypertext Preprocessor. Die aktuellste Version von PHP ist 7.4.0, die im Dezember 2019 veröffentlicht. PHP ist eine Serverseitige Programmiersprache, die für die Erstellung von dynamischen Webseiten dient. Mit Hilfe von PDO wird die Verbindung mit der Datenbank gemacht. PHP ist fast von allen Webhosting Services verfügbar und unterstützt.



Abbildung 5.2: PHP Logo

**JavaScript** (JS) ist eine kompakte, interpretierte oder just-in-time kompilierte Programmiersprache mit erstklassigen Funktionen. Während es als Skriptsprache für Webseiten am bekanntesten ist, wird es auch von vielen anderen Umgebungen als Browsern verwendet, z. B. Node.js, Apache CouchDB und Adobe Acrobat. JavaScript ist eine prototypbasierte, auf mehreren Paradigmen basierende, dynamische Sprache mit einem Thread, die objektorientierte, imperative und deklarative Stile (z. B. funktionale Programmierung) unterstützt. Lesen Sie mehr über JavaScript. Verwechseln Sie JavaScript nicht mit der Programmiersprache Java. Sowohl "Java" als auch "JavaScript" sind Marken oder eingetragene Marken von Oracle in den USA und anderen Ländern. Die beiden Programmiersprachen haben jedoch eine sehr unterschiedliche Syntax, Semantik und Verwendung.

## Datatables

### 5.2.2 Client

#### MySQL<sup>5</sup>

Für die Relationale Datenbank Management System wurde MySQL (Abb. 5.4) gewählt. Dieses Datenbankverwaltungssystem wird gerne für kleine Projekte verwendet, weil es Open-Source und kostenlos ist. MySQL ist eine Software um Datenbank

<sup>5</sup>My Structured Query Language



Abbildung 5.3: JavaScript Logo

zu erstellen und zu verwalten. MySQL ermöglicht kontrollierten Zugriff auf die Datenbank. Die Daten werden auf Tabellen Struktur gespeichert, die in Dateien auf der Festplatte gespeichert sind. Diese Software ist kompatibel auf mehrere unterschiedlichen Plattformen wie z.B. Windows, Linux, MacOS usw.



Abbildung 5.4: MySQL Logo

Das **Apache** Apache HTTP Server-Projekt ist ein Versuch, einen Open-Source-HTTP-Server für moderne Betriebssysteme wie UNIX und Windows zu entwickeln und zu warten. Ziel dieses Projekts ist es, einen sicheren, effizienten und erweiterbaren Server bereitzustellen, der HTTP-Dienste synchron mit den aktuellen HTTP-Standards bereitstellt. Dieser Server verwendet die Port 80 für HTTP und Port 443 für HTTPS. Die erste Version wurde im Jahr 1995 und der letzte stabile Version ist 2.4.41, der im August 2019 veröffentlicht wurde. “[test]



Abbildung 5.5: Apache HTTP Server Logo

## **5.3 Technische Lösung**

### **5.3.1 Basis Seite**

### **5.3.2 Displays**

### **5.3.3 Layouts**

### **5.3.4 Rechte in die Webseite**

### **5.3.5 Webseite auf mehrere Sprachen**

# Abbildungsverzeichnis

2.1	Projektstrukturplan . . . . .	5
2.2	Structed Design - Client . . . . .	9
2.3	Structed Design - Server . . . . .	10
2.4	Wasserfallmodell . . . . .	11
5.1	Bootstrap Logo . . . . .	19
5.2	PHP Logo . . . . .	19
5.3	JavaScript Logo . . . . .	20
5.4	MySQL Logo . . . . .	20
5.5	Apache HTTP Server Logo . . . . .	20

# Tabellenverzeichnis

2.1	Meilensteine . . . . .	4
2.2	Arbeitspakete . . . . .	8

# Literatur