BETRIEBSPRAKTIKUM

Von Malte Rövemeier, IA3B



|  |  |
| --- | --- |
| Datum | 24.06.2024 bis zum 23.08.2024 |
| Ort | GE Vernova  Holsterfeld 16  48499 Salzbergen |
| Ansprechpartner | Herr Stegemann  (+) 49 5971 980 1061  Stefan.Stegemann@ge.com |
| Zu bewertendes Fach | Software |

Inhalt

[1. Vorspann 1](#_Toc171404878)

[1.1. Auswahl der Praktikums Stelle 1](#_Toc171404879)

[1.2. Die an das Praktikum geknüpfte Erwartungen 1](#_Toc171404880)

[1.3. Zusammenfassung der Praktikumserfahrungen 1](#_Toc171404881)

[2. Vorstellung des Betriebs 2](#_Toc171404882)

[2.1. Der Betrieb 2](#_Toc171404883)

[2.2. Beschreibung des eigenen Arbeitsplatzes 2](#_Toc171404884)

[3. Projekt: Wetterstation in *Kran Boards* mit einbinden 4](#_Toc171404885)

[3.1. Ausgangssituation 4](#_Toc171404886)

[3.1.1. Aufgabenstellung/Pflichtenheft/Kundenwünsche 4](#_Toc171404887)

[3.1.2. Ist-Analyse 4](#_Toc171404888)

[3.1.3. Schnittstellen 5](#_Toc171404889)

[3.2. Ressourcen- und Ablaufplan 5](#_Toc171404890)

[3.2.1. Soll-Konzept 5](#_Toc171404891)

[3.2.2. Ablaufplan/Meilensteine 6](#_Toc171404892)

[3.3. Durchführung und Auftragsbearbeitung 7](#_Toc171404893)

[3.3.1. *ESP32* Code 7](#_Toc171404894)

[3.3.2. Erstellen der Datenbank 8](#_Toc171404895)

[3.3.3. Website Oberfläche mit Fertigungsplan 9](#_Toc171404896)

[3.3.4. Position setzen 9](#_Toc171404897)

[3.3.5. Login-Page 9](#_Toc171404898)

[3.3.6. *SSL-Zertifikat* 10](#_Toc171404899)

[3.3.7. 3D Druck 10](#_Toc171404900)

[3.3.8. Abweichungen, Anpassungen, Entscheidungen 10](#_Toc171404901)

[3.4. Ergebnisse 11](#_Toc171404902)

[3.5. Verwendete Quellen innerhalb der Projektaufgabe 11](#_Toc171404903)

[3.6. Anlagen innerhalb der Projektaufgabe 12](#_Toc171404904)

[3.6.1. Glossar 12](#_Toc171404905)

[3.6.2. Quellcode/Testdaten 16](#_Toc171404906)

[4. Wochenberichte 16](#_Toc171404907)

[4.1. Woche 1 16](#_Toc171404908)

[4.2. Woche 2 17](#_Toc171404909)

[4.3. Woche 3 17](#_Toc171404910)

[4.4. Woche 4 17](#_Toc171404911)

[4.5. Woche 5 17](#_Toc171404912)

[4.6. Woche 6 17](#_Toc171404913)

[4.7. Woche 7 17](#_Toc171404914)

[4.8. Woche 8 17](#_Toc171404915)

[5. Bewertung 17](#_Toc171404916)

# Vorspann

## Auswahl der Praktikums Stelle

Bereits zu den Osterferien 2023 habe ich mich im Rahmen meiner Ausbildung zum Fachinformatiker für Systemintegration für ein Praktikum bei GE beworben. Mein Vater hatte mir erzählt, dass GE schon in den vergangenen Jahren diese Ausbildung unterstützt hat, was mir sehr geholfen hat, da das Unternehmen den Ablauf bereits gut kennt und erprobt ist. Nach meiner Bewerbung begann ich, mich intensiv über GE zu informieren. Besonders beeindruckt hat mich, dass *GE* Windkraftanlagen herstellt und damit aktiv zur Förderung erneuerbarer Energien beiträgt. Zudem finde ich es spannend, dass *GE* ein amerikanisches Unternehmen ist und sich in vielerlei Hinsicht von deutschen Firmen unterscheidet. Einige Wochen nach einem Bewerbungsgespräch mit Herrn Stefan Stegemann erhielt ich eine E-Mail, die mir mitteilte, dass ich angenommen wurde. Kurz darauf unterschrieb ich einen Ausbildungsvertrag über drei Jahre.

## Die an das Praktikum geknüpfte Erwartungen

Da ich nicht zum ersten Mal bei *GE* war, hatte ich bereits eine gute Vorstellung davon, was mich erwarten würde. In den Osterferien zuvor hatte ich bereits mit einem Arbeitskollegen, gesprochen. Wir hatten vereinbart, dass er sich ein interessantes Projekt überlegen würde, das ich während der Sommerferien bearbeiten könnte. Meine Erwartungen an das Praktikum waren hoch: Ich hoffte, meine Software-Kenntnisse weiter vertiefen zu können und viele neue Dinge zu lernen. Die Aussicht, an einem spannenden Projekt zu arbeiten, motivierte mich zusätzlich und ich war gespannt auf die Herausforderungen, die vor mir lagen.

## Zusammenfassung der Praktikumserfahrungen

# Vorstellung des Betriebs

## Der Betrieb

GE Vernova ist der Geschäftsbereich von General Electric (*GE*), der sich auf Energie konzentriert. Es umfasst:

- Erneuerbare Energien: GE Renewable Energy entwickelt Windkraftanlagen, Solarenergie- und Wasserkraftlösungen.

- Energieerzeugung: GE Power bietet Technologien wie Gasturbinen und Dampfturbinen für effiziente Energieerzeugung.

- Elektrische Netze: GE Grid Solutions verbessert die Zuverlässigkeit und Effizienz von Stromnetzen.

- Nuklearenergie: GE Hitachi Nuclear Energy bietet fortschrittliche Kernkraftwerkstechnologien.

GE Vernova fördert durch innovative und nachhaltige Lösungen die Energiewende und eine kohlenstoffarme Zukunft.

## Beschreibung des eigenen Arbeitsplatzes

Mein Arbeitsplatz liegt im dritten Stock des Bürogebäudes. Auf diesem Stockwerk befinden sich neben der IT-Abteilung auch die Personalabteilung. Zunächst teilte ich mir ein Büro mit zwei Fachinformatikern für Systemintegration (FiSis), die sich ausschließlich in der Ausbildung befinden und jeden Tag bei *GE* sind, außer an den Tagen, an denen sie Berufsschule haben. Am 1. August 2024 kam ein weiterer FiSi hinzu. Mein Arbeitsplatz ist mit einem Laptop ausgestattet, der an eine Dockingstation angeschlossen ist. An dieser Dockingstation sind zwei Monitore, eine Maus, eine Tastatur und ein Headset angeschlossen. Die Hauptarbeit erledige ich auf dem Laptop.



Ich nutze eine in meiner Tastatur integrierte *Smart Card* für den sicheren Server-Login. Zusätzlich besitze ich ein multifunktionales Badge, das folgende Aufgaben erfüllt:

1. Eingang und Zeiterfassung: Öffnet die Eingangstür und erfasst meine Arbeitszeit.

2. Zugang: Gewährt Zugang zum Serverraum und zum Fitnessstudio.

3. Druckerzugang: Ermöglicht sicheren Druck durch Verbindung zum Druckserver per Scannen des Badges.

4. *SSO*-Verifizierung: Enthält meine eindeutige Single Sign-On (*SSO*) für universelle Verifizierung.

Dieses System bietet hohe Sicherheit und vereinfacht die Abläufe durch die Kombination mehrerer Funktionen in einem Gerät. Außerdem bekam ich Stahlkappenschuhe, Schutzbrille, Handschuhe, Helm und eine Warnweste, um mich in der Fertigung und dem Außenbereich uneingeschränkt bewegen zu können.

# Projekt: Wetterstation in *Kran Boards* mit einbinden

## Ausgangssituation

### Aufgabenstellung/Pflichtenheft/Kundenwünsche

Meine Aufgabe bestand darin, in ein funktionierendes System (siehe 3.1.2. Ist-Analyse) eine Wetterstation zum Optimieren der Heizungskosten zu bauen. Dazu sollte ich an diese *ESP32* in der Fertigungshalle *DHT22* Sensoren hängen und eine einfache und gut übersichtliche Website programmieren, auf der gut sichtbar wird, wo es gerade wie Warm ist. Für diese Anwendungen ergaben sich folgende technische Aufgaben:

* Programmieren eines *ESP32,* um Temperatur zu lesen und an einen Server zu senden
* Datenbank auf einem Server erstellen, um Temperatur, Luftfeuchtigkeit, Mac Adressen, Zeit und Position (um auf der Website sagen zu können, wo sich welches Gerät befindet) speichern zu können
* Website in PHP programmieren, um Daten aus der Datenbank auszulesen und neue hereinzuschreiben, wenn man die Position der *ESP32* setzt
* Hinzufügen von JavaScript, um Mausklicks zu erkennen
* Login zum Setzen/Zurücksetzen der Positionen
* Designen eines Gehäuses für den *DHT22*
* *SSL-Zertifikat* erstellen und zuweisen

### Ist-Analyse

In der Fertigungshalle haben wir ganz viele Kräne, welche von unterschiedlichen Leuten genutzt werden. Also hat sich ein Azubi es zum Projekt gemacht, einen *ESP32* so zu programmieren, dass er auf einem Display anzeigt:

* ob der Kran aktuell überprüft werden muss, um festzustellen, ob er bei der letzten Benutzung kaputt gegangen ist oder nicht.
* ob der Kran voll funktionstüchtig ist.
* ob der Kran defekt ist.

### Schnittstellen

Da man auf dem *ESP32* nur einen Code aufspielen kann, musste ich den bereits vorhandenen Code, ändern und das Senden meiner Wetterdaten und Mac Adressen hinzufügen.

## Ressourcen- und Ablaufplan

Für dieses Projekt benötigte ich neben meinem Arbeitsplatz und Laptop eine Reihe spezifischer Softwareanwendungen, um die Arbeit effizienter zu gestalten. Die Programmierung der Anwendung erfolgt in Visual Studio Code und der Arduino IDE, wobei die Programmiersprachen *PHP*, *JavaScript*, C und *C++* zum Einsatz kommen. Für den Zugriff auf den Server verwende ich die Remotedesktopverbindung. Auf dem Server selbst nutze ich HeidiSQL als Datenbankmanagementsystem. Um die Website lokal zu testen, setze ich XAMPP ein. Der Webserver sowie die Erstellung und Zuordnung des *SSL-Zertifikats* werden über den Internet Information Services (*IIS*) realisiert. Das 3D-Modell wurde in FreeCAD entworfen.

### Soll-Konzept

#### 3.2.1.1 Terminplanung

Das Projekt fan in der Woche vom 01.07.2024 – 05.07.2024 statt. Der Zeitplan sah wie folgt aus:

|  |  |
| --- | --- |
| Arbeitsschritt | Benötigte Zeit |
| Recherche | 5 Stunden |
| ESP32 Code senden der Daten | 5 Stunden |
| Datenbank erstellen und testen | 0,5 Stunden |
| Datenbankverbindung zur Website herstellen | 2 Stunden |
| Website Oberfläche mit Fertigungsplan | 4 Stunden |
| Tabelle zum Darstellen der Temperatur und Luftfeuchtigkeitswerte mit Zeitstempel | 2 Stunden |
| Seite zum Setzen und Zurücksetzen der Positionen | 9 Stunden |
| Login | 3 Stunden |
| *SSL-Zertifikat* beantragen und zuweisen | 2 Stunden |
| Puffer für Bugfixing und zusätzliche Features | 7,5 Stunden |
| Summe | **40 Stunden** |

#### 3.2.1.2. Kostenplanung

Für das Projekt ist es nachher geplant, 15 solcher Wetterstationen zu haben. Dafür mussten folgende Sachen bestellt werden:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Artikel | Anzahl | Preis | Gesamtpreis | Link |
| DHT22 | 3x | 13,99€ | 41,97€ | [Amazon](https://www.amazon.de/Temperatursensor-Luftfeuchtigkeitssensor-kompatibel-Digitaler-Feuchtesensor/dp/B0CRB36DTJ?source=ps-sl-shoppingads-lpcontext&ref_=fplfs&psc=1&smid=A2SPO00TSCGKK7) |
| Kabel | 1x | 6,99€ | 6,99€ | [Amazon](https://www.amazon.de/Female-Female-Male-Female-Male-Male-Steckbr%C3%BCcken-Drahtbr%C3%BCcken-bunt/dp/B01EV70C78/ref=sr_1_4?dib=eyJ2IjoiMSJ9.vEpeDldThnaRPDTCkZ1A_JfAl2nuENh2H-ME5xxL6zUVxDJO4WS3arbPBZVfexaSnReld41GDmNDwDcOci_744d88xtNBhPanpnGzhl1tm6G2j2d_n4lc_76P6IEjIjMzR2BI_a5XjMWjcufkM2wTux8boATh9FoF32HkrgBbwqqbDNrydHRjCuse3WWW9DuaiOvl9MZfKda1S0S4LqEy517oftgelfGXEAdoNxDZkw.CjV5xybX9cgcnjEIWJhSD4keoEvtZLFfb7Da7A8m2KE&dib_tag=se&keywords=male+female&qid=1719927962&sr=8-4) |
| ESP32 | 15x | 10,99€ | 164,85€ | [[Amazon](https://www.amazon.de/iHaospace-ESP32-DevKitC-ESP32-WROOM-32U-Development-Bluetooth/dp/B08BZKZXLL?th=1)](https://www.amazon.de/iHaospace-ESP32-DevKitC-ESP32-WROOM-32U-Development-Bluetooth/dp/B08BZKZXLL?th=1) |
| Antenne | 15x | 7,64€ | 114,60€ | [Amazon](https://www.amazon.de/-/en/POPESQ%C2%AE-Antenna-ESP8266-Self-Adhesive-A5579/dp/B0C58MGCB3) |
| Gesamt |  |  | 328,41‬€ |  |

### Ablaufplan/Meilensteine

1. Zuerst plane ich den *ESP32* so zu programmieren, dass dieser Daten an eine Datenbank auf einem Server sendet. In dem *HeidiSQL* soll ein Zeitstempel (timestamp) hinzugefügt werden.
2. Die Daten im *PHP*-Code abfragen und auf der Website in einer Tabelle darstellen.
3. Setzen und Zurücksetzen Position-X und Position-Y in *HeidiSQL* in eine Tabelle schreiben uns auslesen für Oberfläche
4. Login-Page
5. *SSL-Zertifikat* erstellen und zuweisen

## Durchführung und Auftragsbearbeitung

### *ESP32* Code

Zunächst musste ich den Code schreiben, um die Temperatur zu lesen und an den Server zu senden.

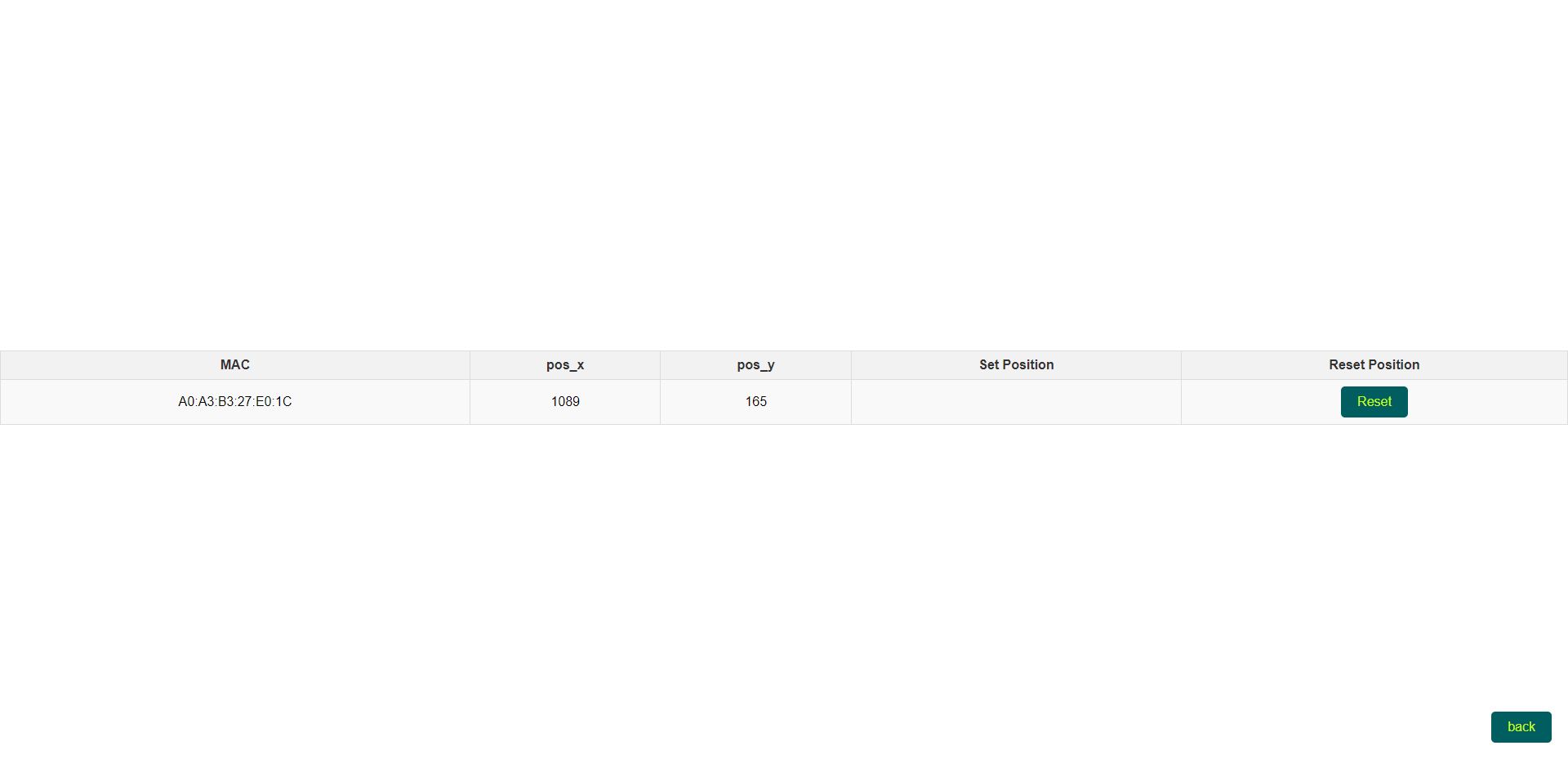


Dies tat ich, indem ich an die Variable „tmpName“ einfach noch die gelesenen Werte der Temperatur „temp“ und der Luftfeuchtigkeit „humidity“ anhing und mitsendete. Die Variable „current\_status“ muss mitgesendet werden für die andere Funktion der *Kran Boards*. Diese Daten sollten alle 5 min gesendet werden also erstellte ich zwei Variablen „lastTempHumidityTime“ und „tempHumidityInterval“. Die Variable „tempHumidityInterval“ setzte ich auf 300000. Das entspricht 5 Minuten. Und wenn die Zeit der Variable „lastTempHumidityTime“ größer war als die des Intervalls, war die Abfrage „true“ also wurde sie ausgeführt und die Variable „lastTempHumidityTime“ wieder auf die Aktuelle Zeit gesetzt.

### Erstellen der Datenbank

Im nächsten Schritt erstellte ich eine Datenbank mit zwei Tabellen. Die erste Tabelle, „temphumidity“, dient zum Speichern der Mac-Adresse, Temperatur und Luftfeuchtigkeit. Ein Zeitstempel wird automatisch hinzugefügt, sobald ein neuer Eintrag erstellt wird. Dies erreicht man ganz einfach, indem man den Standardwert auf „current\_timestamp()“ setzt und die Option „Allow NULL“ deaktiviert. So wird bei jedem neuen Eintrag die aktuelle Zeit eingetragen, da der Wert nicht „NULL“ sein darf.

Die zweite Tabelle, „buttonposition“, dient dazu, den MAC-Adressen spezifische Positionen zuzuweisen. In der nachfolgenden Abbildung ist die entsprechende PHP-Seite zu sehen, die diese Informationen anzeigt. Die Positionen werden aus der Datenbank abgerufen und können auf einer anderen PHP-Seite festgelegt werden. Die Tabelle enthält die MAC-Adressen, welche in der Tabelle „temphumidity“ entweder Temperatur- oder Luftfeuchtigkeitswerte (oder beide) aufzeichnen. Dies ist von großer Bedeutung, da zwar alle ESP32-Geräte mit diesem Code ausgestattet sind, jedoch nicht alle über einen DHT22-Sensor zur Datenerfassung verfügen.



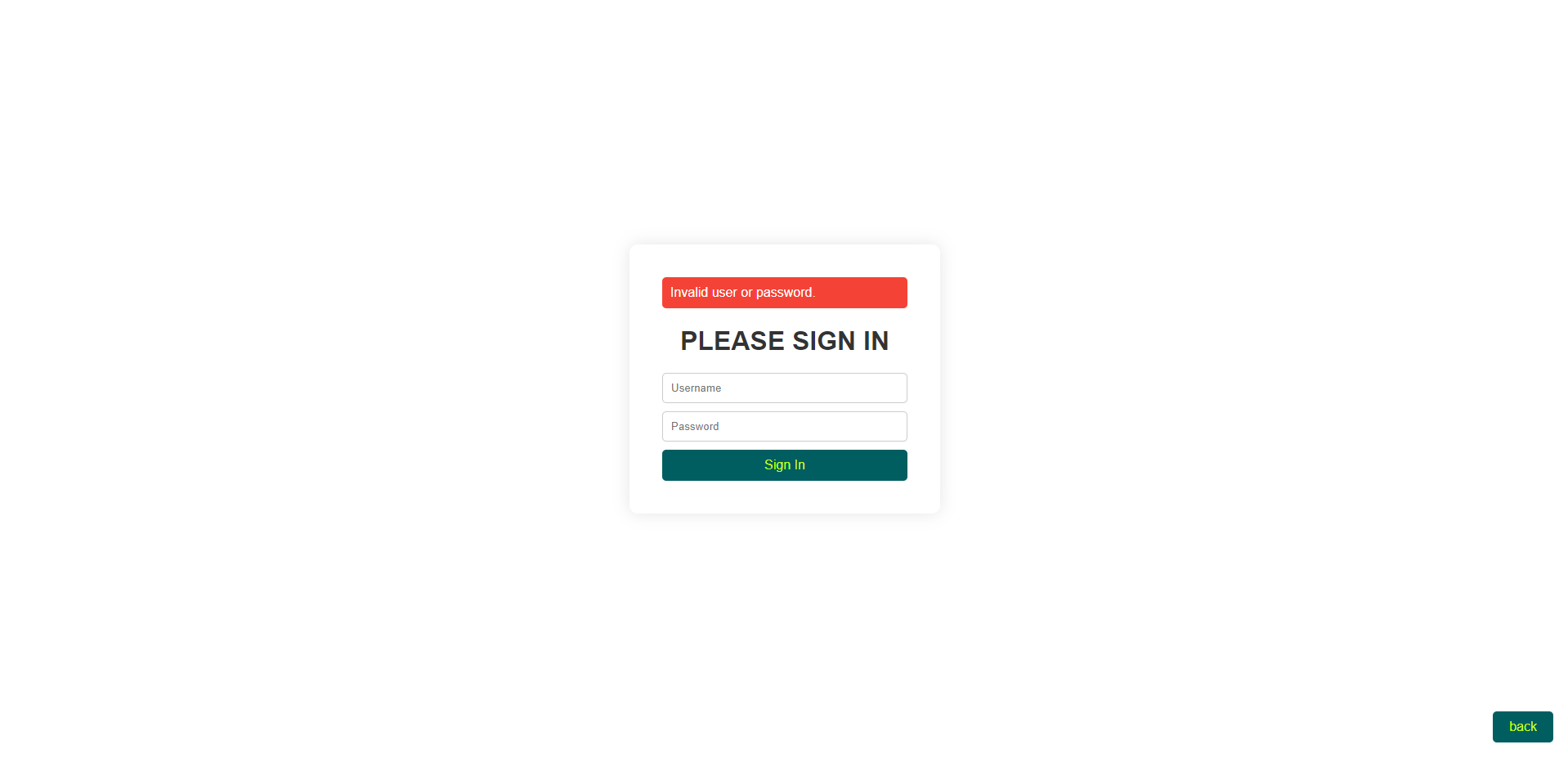
### Website Oberfläche mit Fertigungsplan

Zunächst wurde die Hauptseite der Website erstellt. Hierbei wurde ein Bild der Fertigung als `<img>`-Tag eingebunden, um sicherzustellen, dass es beim Zoomen entsprechend mit vergrößert wird. Auf der Seite ist ein Button sichtbar, dessen Position, durch die in der Datenbank definierten x- und y-Koordinaten festgelegt ist. Dieser Button befindet sich im selben `<div>` wie das Bild und wird daher beim Zoomen ebenfalls vergrößert. Zusätzlich ist in der Mitte der Seite ein weiterer Button platziert, der die Anzahl der MAC-Adressen anzeigt, die in der Tabelle „buttonposition“ keinen Eintrag für x- und/oder y-Koordinate haben. Die Abfrage erfolgt gezielt auf die Tabelle „buttonposition“, um sicherzustellen, dass nur die MAC-Adressen angezeigt werden, die Temperatur- und Luftfeuchtigkeitswerte erfassen.

### Position setzen

Im nächsten Schritt implementierte ich eine Mausklick-Erfassung mittels JavaScript. Ursprünglich hatte ich vor, ausschließlich PHP und HTML zu verwenden, jedoch bietet PHP keine Möglichkeit, Mausklicks zu erfassen. Nach einer Recherche fand ich eine passende Lösung in JavaScript. Jetzt führt ein Klick auf den Button "Set" den Benutzer auf eine neue Seite, auf der er die gewünschte Position durch einen Mausklick festlegen kann. Diese Position wird anschließend in der Datenbanktabelle „buttonposition“ gespeichert.

### Login-Page

Eine Login-Seite war von entscheidender Bedeutung, um sicherzustellen, dass nur autorisierte Personen die Position der Buttons setzen und zurücksetzen können. Obwohl die Website nur im internen Netzwerk zugänglich ist, soll dennoch nicht jeder die Möglichkeit haben, Änderungen an der Button-Position vorzunehmen. Die Zugangsdaten sind fest im Code verankert, da es lediglich einen Benutzer gibt. Eine alternative Lösung hätte den vorgegebenen Projektzeitraum von 40 Stunden überschritten. Bei Eingabe falscher Benutzerdaten wird eine Fehlermeldung angezeigt. 

### *SSL-Zertifikat*

CSR-Erstellung: Über den *IIS*-Manager, wobei genaue Angaben zu Namen, Organisation und Standort gemacht werden müssen. Während dieses Prozesses wird auch ein Kryptografie Anbieter und eine Bit länge ausgewählt.

SSL-Zertifikat Installation: Nach Erhalt des SSL-Zertifikats von „DigiCert“, wird dieses über den *IIS*-Manager auf dem Server, auf dem der CSR erstellt wurde, installiert. Dies beinhaltet das Hinzufügen des Zertifikats, die Zuweisung an die entsprechende Website und das Einrichten von HTTPS-Bindungen.

### 3D Druck

Zum Schluss designte ich noch ein 3D Objekt, welches auf der derzeitigen *Kran Boards* mit draufkommt, sodass der *ESP32* in der Fertigung auf jeden Fall gut geschützt ist.

### Abweichungen, Anpassungen, Entscheidungen

Im Rahmen meines Projekts gab es nur wenige Abweichungen vom ursprünglichen Plan, da die Zielsetzung sehr offen definiert war. Die wichtigsten Anpassungen betrafen die verwendeten Technologien: Ursprünglich plante ich, die Implementierung in PHP und HTML durchzuführen. Aufgrund technischer Beschränkungen bei der Mausklick-Erfassung musste ich jedoch auf JavaScript zurückgreifen, da diese Funktionalität mit PHP nicht realisierbar war. Abgesehen von dieser Änderung gab es keine nennenswerten Abweichungen. Das Design der Webseite habe ich weitgehend eigenständig gestaltet und mich für ein klar strukturiertes Layout in den Firmenfarben entschieden, um eine übersichtliche und ansprechende Benutzeroberfläche zu gewährleisten.

## Ergebnisse

## Verwendete Quellen innerhalb der Projektaufgabe

Quelle 1

Titel: Prozessoptimierung mit Hilfe von 2D-Hallenplänen

Autor: lklevenow

Datum: 11. Mai 2020

URL: [Laserscanning Europe](https://www.laserscanning-europe.com/de/news/prozessoptimierung-mit-hilfe-von-2d-hallenplaenen)

Zusammenfassung: Für Das Hintergrundbild

Quelle 2

Titel: PHP Manual (Deutsch)

URL: [PHP Manual](https://www.php.net/manual/de/)

Zusammenfassung: Die offizielle PHP-Dokumentation bietet umfassende Informationen und Anleitungen zur PHP-Programmierung, einschließlich Funktionen, Klassen, und Best Practices.

Quelle 3

Titel: CSR Creation and SSL Installation for IIS 10

URL: [DigiCert](https://www.digicert.com/kb/csr-creation-ssl-installation-iis-10.htm)

Zusammenfassung: Anleitung zur Erstellung eines Certificate Signing Request (CSR) und zur Installation eines SSL-Zertifikats auf einem IIS 10-Server, inklusive detaillierter Schritte und Screenshots zur Unterstützung.

## Anlagen innerhalb der Projektaufgabe

### Glossar

Arduino IDE

Die Arduino Integrated Development Environment (IDE) ist eine plattformübergreifende Anwendung, die zur Programmierung und Entwicklung von Arduino-Boards verwendet wird. Sie bietet eine einfache Schnittstelle zum Schreiben, Kompilieren und Hochladen von Code auf Arduino-Mikrocontroller.

C++

C++ ist eine leistungsfähige, universelle Programmiersprache, die sowohl objektorientierte als auch generische Programmierung unterstützt. Sie wird häufig für Systemsoftware, Spieleentwicklung und Anwendungen mit hohen Leistungsanforderungen verwendet.

CSR-Datei

Eine Certificate Signing Request (CSR) Datei ist eine Datei, die bei der Erstellung eines SSL-Zertifikats verwendet wird. Sie enthält Informationen wie den öffentlichen Schlüssel und die Daten des Antragstellers, die von einer Zertifizierungsstelle überprüft werden.

DHT22

Der DHT22 ist ein Sensor, der Temperatur und Luftfeuchtigkeit misst. Er ist bekannt für seine hohe Genauigkeit und wird häufig in Umweltüberwachungssystemen eingesetzt.

DigiCert

DigiCert ist eine führende Zertifizierungsstelle, die SSL/TLS-Zertifikate und andere digitale Sicherheitslösungen anbietet. Sie sorgt dafür, dass die Kommunikation im Internet sicher verschlüsselt ist.

ESP32

Der ESP32 ist ein kostengünstiger Mikrocontroller mit integriertem Wi-Fi und Bluetooth. Er wird häufig in IoT-Projekten (Internet of Things) verwendet und bietet umfangreiche Funktionen für die Entwicklung drahtloser Anwendungen.

GE

GE steht für General Electric, ein multinationales Konglomerat, das in verschiedenen Industriezweigen tätig ist, darunter Energie, Gesundheitswesen und Finanzdienstleistungen.

Hartkodiert

Hartkodiert bezieht sich auf Daten oder Werte, die direkt in den Quellcode eines Programms eingebettet sind. Dies macht Änderungen an diesen Werten schwierig, da der Code neu kompiliert werden muss.

HeidiSQL

HeidiSQL ist ein kostenloses Open-Source-Tool für die Verwaltung von MySQL-, MariaDB- und PostgreSQL-Datenbanken. Es ermöglicht Benutzern, Datenbanktabellen zu erstellen, zu bearbeiten und zu durchsuchen sowie Abfragen auszuführen.

HTML

Hypertext Markup Language (HTML) ist die Standard-Auszeichnungssprache für die Erstellung von Webseiten und Webanwendungen. Sie beschreibt die Struktur eines Dokuments und ermöglicht die Einbettung von Text, Bildern, Links und anderen Elementen.

IIS

Internet Information Services (IIS) ist ein Webserver von Microsoft für Windows-Server-Betriebssysteme. Er dient zur Bereitstellung von Webseiten und anderen Webdiensten.

JavaScript

JavaScript ist eine vielseitige Programmiersprache, die hauptsächlich zur Erstellung von interaktiven und dynamischen Inhalten auf Webseiten verwendet wird. Sie kann sowohl auf dem Client- als auch auf dem Server-Side eingesetzt werden.

Kran Boards

Kran Boards sind Steuerungseinheiten, die in Kranen und anderen Hebegeräten verwendet werden, um deren Bewegungen und Funktionen zu kontrollieren. Sie können programmierbare Logikcontroller (PLCs) und andere elektronische Komponenten enthalten.

MAC-Adresse

Eine MAC-Adresse (Media Access Control) ist eine eindeutige Kennung, die einer Netzwerkschnittstelle zugewiesen wird. Sie dient zur Identifizierung von Geräten in einem Netzwerk und besteht aus einer 48-Bit-Zahl, die häufig in Hexadezimalform dargestellt wird.

PHP

PHP (Hypertext Preprocessor) ist eine serverseitige Skriptsprache, die speziell für die Webentwicklung entwickelt wurde. Sie wird verwendet, um dynamische Webseiten zu erstellen und kann in HTML eingebettet werden.

Smart Card

Eine Smart Card ist eine Plastikkarte mit einem eingebetteten Mikrochip, der persönliche Daten speichern und verarbeiten kann. Sie wird häufig in Sicherheitssystemen, Identifikationskarten und Kreditkarten verwendet.

SSO

Single Sign-On (SSO) ist ein Authentifizierungsverfahren, das es Benutzern ermöglicht, mit nur einer Anmeldung auf mehrere Anwendungen und Systeme zuzugreifen. Dies verbessert die Benutzerfreundlichkeit und Sicherheit.

SSL-Zertifikat

Ein SSL-Zertifikat (Secure Sockets Layer) ist ein digitales Zertifikat, das die Verschlüsselung und Authentifizierung von Daten zwischen einem Webbrowser und einem Server ermöglicht. Es stellt sicher, dass die übertragenen Daten sicher und vor unbefugtem Zugriff geschützt sind.

Visual Studio Code

Visual Studio Code (VS-Code) ist ein kostenloser, quelloffener Code-Editor, der von Microsoft entwickelt wurde. Er bietet Unterstützung für eine Vielzahl von Programmiersprachen und Tools, einschließlich Debugging, Versionskontrolle und Erweiterungen.

### Quellcode/Testdaten

GitHub: <https://github.com/mroev/tempHumidity>

# Wochenberichte

## Woche 1

Als ich im Betrieb ankam verbrachte ich die erste Zeit damit, meine ungelesenen Mails zu lesen und ggf. zu beantworten. Danach beantragte ich eine „Removable Block Exception“ um die Möglichkeit zu haben Daten auf beispielsweise eine SD Karte zu ziehen um dann einen Raspberry-Pi aufzusetzen. Am Mittwoch fuhren wir rüber zum anderen Gebäude und räumten sämtliche alte Geräte auf, da dieser Standort nicht mehr genutzt wird. Eine weitere Aufgabe war es dann, eine Art Wetterstation zu bauen. Diese sollen dann an den Server die aktuellen Werte schicken. Die Werte werden dann in einer Datenbank gespeichert und auf einer Website ausgegeben. Im besten Fall wird noch eine Karte geplant, auf welcher dann die aktuellen Werte zu der passenden Position in der Fertigung angezeigt werden. Dies besprachen wir so weit, dass ich in der nächsten Woche passend damit anfangen konnte.

## Woche 2

Die Woche begann damit, dass ich mich an das Programmieren der Wetterstation setzte. Ich hatte zunächst einen Pfad Fehler und dann ein Problem mit der Verbindung zur Datenbank. Dies ließ sich schnell beheben. Im Laufe der Woche programmierte ich ein paar mehr .php Seiten, auf denen man erstens die aktuellen Temperaturen anzeigen lassen konnte. Zweitens, Mac Adressen aus der Datenbank ohne x und y Koordinaten in der Datenbank konnte auf einer Seite eine neue Position gesetzt bzw. zurückgesetzt werden. Drittens eine Login Seite, auf der man sich mit „username“ und „passwort“ anmelden muss. Außerdem veränderte ich den Code, welcher dann auf den ESP32 geflasht wurde, nochmal so dass nun alle Daten richtig an den Server gesendet werden. In der Woche hatten wir noch ein Meeting mit dem Geschäftsführer, in dem wir besprachen, wie man am besten für junge Leute werben kann, ein/e Ausbildung/Dualstudium anzufangen. Zum Ende der Woche beantragte ich noch ein SSL-Zertifikat für die Website und Designte ein Gehäuse für den DHT22.

## Woche 3

## Woche 4

## Woche 5

## Woche 6

## Woche 7

## Woche 8

# Bewertung