



**Regensburg**  
für Oberpfalz / Kelheim

Abschlussprüfung Sommer 2022

Fachinformatiker für Anwendungsentwicklung

**Dokumentation zur betrieblichen Projektarbeit**

## **Entwicklung einer elektronischen Informationstafel mit Verwaltungsfunktionen**

### **Prüfungsbewerber:**

Maximilian Merkl

Halbergmooserstraße 17a

85356 Freising

Identnummer: 944073



### **Ausbildungsbetrieb:**

Berufsbildungswerk St. Franziskus Abensberg

Regensburger Straße 60

93326 Abensberg

## Inhaltsverzeichnis

1. Projektbeschreibung .....	1
1.1. Projektumfeld .....	1
1.2. Analyse des IST-Zustands .....	1
1.3. Ermittlung des SOLL-Konzepts .....	1
1.4. Projektziele .....	2
1.5. Make-or-Buy Entscheidung .....	2
Kandidaten .....	2
Entscheidung .....	2
2. Projektphasen .....	2
3. Projektbegründung .....	3
4. Projektplanungsphase .....	3
5. Wirtschaftlichkeitsanalyse und Kostenplanung .....	4
6. Entwurfsphase .....	5
6.1. Vorüberlegungen .....	5
6.2. Programmiersprachen und Erweiterungen .....	5
7. Frontend .....	5
7.1. Vorwort zum Frontend .....	5
7.2. Entwurf der Infotafel .....	6
7.3. Entwurf des Infotafel-Managers .....	7
8. Backend .....	8
8.1. Backend Funktionen .....	8
8.2 Datenbank .....	8
8.3. Bereitstellung der Daten .....	8
8.4. Ablegen von Dateien .....	8
8.5. Sicherheit mit PHP .....	8
9. Testphase .....	9
9.1. Entwicklungstests .....	9
9.2. Funktionstests .....	9
10. Soll-/ist-Vergleich .....	10
11. Abnahme .....	11
12. Fazit .....	11
13. Anhänge .....	12
13.1. Benutzerhandbuch .....	12
13.2. Glossar .....	15
13.3. Screenshots .....	16

## **1. Projektbeschreibung**

Entwicklung einer elektronischen Informationstafel mit Verwaltungsfunktionen.

### **1.1. Projektumfeld**

Das Projekt wird im Rahmen meiner Ausbildung im Berufsbildungswerk St. Franziskus Abensberg, in der IT-Abteilung, durchgeführt. Begleitet wird es fachlich von meinem Ausbilder, Oliver Freiseisen. Das B.B.W. hat ein Team von ca. 400 Ausbilder\*innen, pädagogischen Fachkräften, Lehrer\*innen und Psycholog\*innen. Darüber hinaus tragen etwa 70 weitere Mitarbeiter\*innen aus den Bereichen Verwaltung, Haustechnik und Dienstleistung zum Miteinander in und um das B.B.W. bei. Nicht zu vergessen sind auch die über 400 Auszubildenden, die mit ihrer Arbeit den Betriebsablauf teilweise ebenfalls unterstützen. Zusätzlich finden auch berufsvorbereitende Bildungsmaßnahmen und Arbeitserprobungen statt, bei denen Teilnehmer\*innen sich stetig ändern, was sowohl das kontinuierliche Anlegen und Löschen ihrer Accounts, als auch deren Rechteverwaltung erfordert.

### **1.2. Analyse des IST-Zustands**

Im Eingangsbereich unseres Ausbildungsbetriebs ist eine Pinnwand befestigt, die dazu genutzt wird, wichtige und ausbildungsrelevante Informationen, wie z.B. Termine oder Ankündigungen an die Auszubildenden bereit zu stellen. Diese wird jedoch nicht regelmäßig auf dem aktuellen Stand gehalten und ist dementsprechend nicht immer aktuell bzw. vollständig. Die Information wird auf Papierblättern ausgedruckt und mit Reißzwecken befestigt. Bei mehreren Blättern kann es dazu führen, dass die Pinnwand schnell unübersichtlich wird. Das führt oftmals dazu, dass man nicht direkt erkennt, wenn sich die Informationen geändert haben oder neue Informationen aufgehängt wurden. Ein weiteres Problem ist der begrenzte Platz, insbesondere wenn die Blätter größer sind oder übereinander aufgehängt werden. Aufgrund der steigenden Anzahl der Auszubildenden ist die Pinnwand unzureichend und somit nicht für den laufenden Betrieb geeignet.

### **1.3. Ermittlung des SOLL-Konzepts**

Die Pinnwand soll in die Form einer elektronischen und digitalen Tafel umgewandelt werden. Die Informationen sollen nicht mehr gedruckt, sondern digital auf einen interaktiven möglichst peripherielosen (Touch-)Bildschirm bzw. Oberfläche dargestellt werden. Aufgrund dessen soll die Administration nur innerhalb des lokalen Netzwerks von autorisierten Personen möglich sein, um die Informationen schnell und einfach zu aktualisieren bzw. zu bearbeiten. Das Interface bzw. Design soll einfach und übersichtlich sein, um jede Information schnell finden und lesen zu können. Eine Änderung der Information an der Infotafel soll nicht möglich sein, um Manipulation zu unterbinden. Zudem sollen die Informationen und deren Anzeigeparameter der Infotafel persistent gehalten werden. Die Wirtschaftlichkeitsbetrachtung erfolgt im Laufe des Projektes.

## **1.4. Projektziele**

Ziel des Projektes ist die Einführung einer digitalen Lösung für eine Infotafel, welche den folgenden Anforderungen der IT-Ausbildung des BBWs entspricht. Diese Anforderungen sollen das Projekt erfüllen:

- Ein interaktiver und peripherieloser Touchscreen.
- Das Anzeigen von Inhalten in Form von Kacheln.
- Beliebige Inhalte können von Mitarbeiter\*innen von ihrem Arbeitsplatz hinzugefügt werden.
- Eine managebare Oberfläche mit der die Infotafel verwaltet werden kann.
- Die Infotafel soll folgende Formate unterstützen: PDF, HTML, URLs.
- Die Infotafel soll ein übersichtliches und modernes Design haben.
- Die Möglichkeit Inhalte zu erstellen, zu bearbeiten und wieder zu löschen.
- Informationen sollen schnell zu finden sein.

## **1.5. Make-or-Buy Entscheidung**

Um festzustellen, ob vielleicht schon geeignete Produkte existieren, wurden konkrete Anforderungen bereits in den Projektzielen definiert.

### **Kandidaten**

Nach langer und ausgiebiger Recherche im Internet fanden sich zwar einige Produkte, welche manchen Anforderungen entsprechen, jedoch nicht allen. Oftmals werden auch Open-Source Projekte nicht mehr weiterentwickelt oder wurden schon vor langer Zeit eingestellt.

### **Entscheidung**

Da sich keine bereits entwickelten Produkte fanden, welche den Anforderungen weit genug entsprechen würden, habe ich mich mit Absprache meines Ausbilders für eine Eigenentwicklung entschieden. Ob sich eine Eigenentwicklung auch wirtschaftlich lohnt wird in der Wirtschaftlichkeitsanalyse und Kostenplanung ermittelt.

## **2. Projektphasen**

Die Umsetzung des Projekts erfolgt nach Vorgabe in einem Zeitraum von 70 Stunden und wird in den folgenden Phasen gegliedert.

### **3. Projektbegründung**

Da es soweit ist, mit der Zeit der Digitalisierung zu gehen, soll es nun eine digitale Infotafel geben, die von den Auszubildenden interaktiv und ohne Peripherie, sondern nur mit dem Finger gesteuert wird. Es soll ermöglicht werden, dass sie ihre benötigten Informationen lesen können und das jetzige Papierchaos beseitigt wird.

### **4. Projektplanungsphase**

Das Projekt durchläuft mehrere Phasen. Dabei wurde das vorgegebene Pensum von 70 Stunden beansprucht. Eine genaue Auflistung der Projektphasen kann der folgenden Darstellung entnommen werden.

#### **Projektplanung (6 Std.)**

- Analyse des IST-Zustands **(1 Std.)**
- Ermittlung des SOLL- Konzepts **(2 Std.)**
- Marktanalyse nach vorhandenen Lösungen **(2 Std.)**
- Make-or-Buy-Entscheidung **(1 Std.)**

#### **Projektentwurf (15 Std.)**

- Entwurf der Programmlogik **(8 Std.)**
- Entwurf der Benutzeroberfläche **(5 Std.)**
- Datenbankentwurf **(2 Std.)**

#### **Projektdurchführung (30 Std.)**

- Implementierung der Benutzeroberfläche **(12 Std.)**
- Implementierung der Programmlogik **(16 Std.)**
- Umsetzung der Datenbank **(2 Std.)**

#### **Tests (9 Std.)**

- Entwicklertests **(4 Std.)**
- Funktionstests **(3 Std.)**
- Protokollierung der Testergebnisse **(2 Std.)**

#### **Projektabschluss (10 Std.)**

- SOLL/IST-Vergleich **(1 Std.)**
- Erstellung der Projektdokumentation **(8 Std.)**
- Übergabe, Schulung und Abnahme **(1 Std.)**

## 5. Wirtschaftlichkeitsanalyse und Kostenplanung

Nun werden die bei der Projektdurchführung entstandenen Kosten berechnet. Als Ausbildungslohn berechne ich für mich selbst den geläufigen Stundensatz für einen Auszubildenden in der IT-Branche von 15 Euro und den Stundensatz für eine/n Mitarbeiter\*in von 50 Euro. Die Ressourcennutzung verursacht während der Ausführung des Projekts einmalig 50 Euro.

Es sind folgende Personen, im nachfolgend dargestellten Umfang, an dem Projekt beteiligt:

Auszubildender:

- Projektplanung (6 Std.)
- Projektentwurf (15 Std.)
- Projektdurchführung (30 Std.)
- Tests (9 Std.)
- Projektabschluss (10 Std.)

**Gesamt 70 Std. (15 € x 70 = 1050 €)**

Mitarbeiter\*in:

- Projektplanung (3 Std.)
- Projektabschluss (2 Std.)

**Gesamt 4 Std. (50 € x 5 = 250 €)**

Die damit errechneten Gesamtkosten können der folgenden Tabelle entnommen werden.

Vorgang:	Mitarbeiter:	Zeit:	Personal:	Sonstige Kosten:	Gesamt:
Projektplanung	1x Auszubildender	6 Std.	90,00 €	0,00 €	90,00 €
Projektplanung	1 x Mitarbeiter*in	3 Std.	150,00 €	0,00 €	150,00 €
Projektentwurf	1x Auszubildender	15 Std.	225,00 €	0,00 €	225,00 €
Projektdurchführung	1x Auszubildender	30 Std.	450,00 €	50,00 €	500,00 €
Tests	1x Auszubildender	9 Std.	135,00 €	0,00 €	135,00 €
Projektabschluss	1x Auszubildender	10 Std.	150,00 €	0,00 €	150,00 €
Projektabschluss	1x Mitarbeiter	2 Std.	100,00 €	0,00 €	100,00 €
<b>Gesamtpreis</b>					<b>1350,00 €</b>

## **6. Entwurfsphase**

### **6.1. Vorüberlegungen**

Die Infotafel soll jeden Werktag für ca. 8 Stunden betriebsfähig sein. Somit ist auf die Schonung von Ressourcen und Hardware zu achten.

### **6.2. Programmiersprachen und Erweiterungen**

Ich habe mich für HTML, CSS, JavaScript und PHP in Verbindung mit MySQL entschieden, da PHP sich sehr gut für jegliche webbasierte Anwendung mit einer Datenbank eignet. Als Schnittstelle zwischen dem Frontend aus HTML und CSS wird Ajax verwendet.

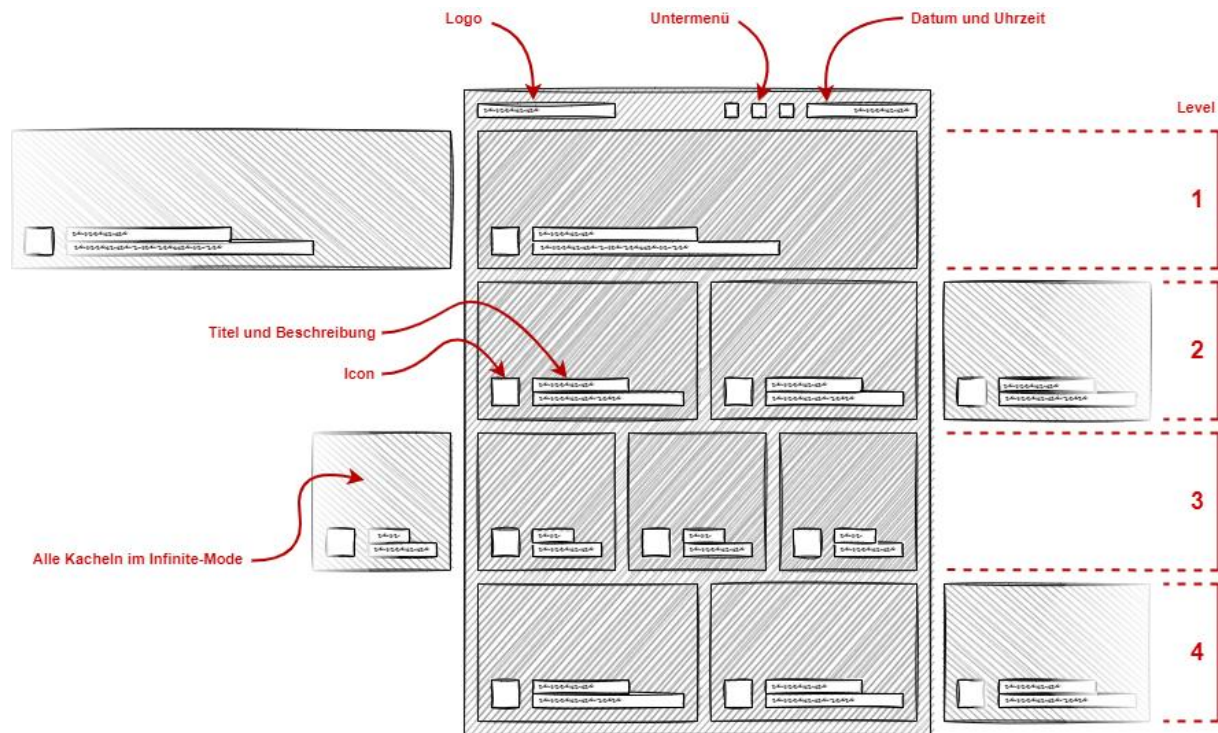
## **7. Frontend**

### **7.1. Vorwort zum Frontend**

Die Entwürfe und Skizzen der Benutzeroberflächen habe ich mittels draw.io durchgeführt und dann final in HTML mit CSS implementiert. Beim Implementieren habe ich jedoch noch kleinere Änderungen am gesamten Farbbild der Webseite vornehmen müssen, da mir das Ergebnis nicht so gut gefiel, weil die Farbabstimmung nicht zusammen mit dem Stil der Oberfläche harmonierte.

## 7.2. Entwurf der Infotafel

Das Frontend der Infotafel soll nur auf einem Endgerät verwendet werden und nicht extern erreichbar sein, daher wurde eine Web-Anwendung im Hochformat als praktischstes Mittel festgelegt. Die Weboberfläche muss dabei nicht nach dem Schema des Responsive Designs für normale Smartphones, PCs und Notebooks mit variierenden Bildschirmgrößen geeignet sein. Touch-Eingaben sind hierbei allerdings dringend notwendig, da der Bildschirm über keinerlei andere Peripheriegeräte verfügt.

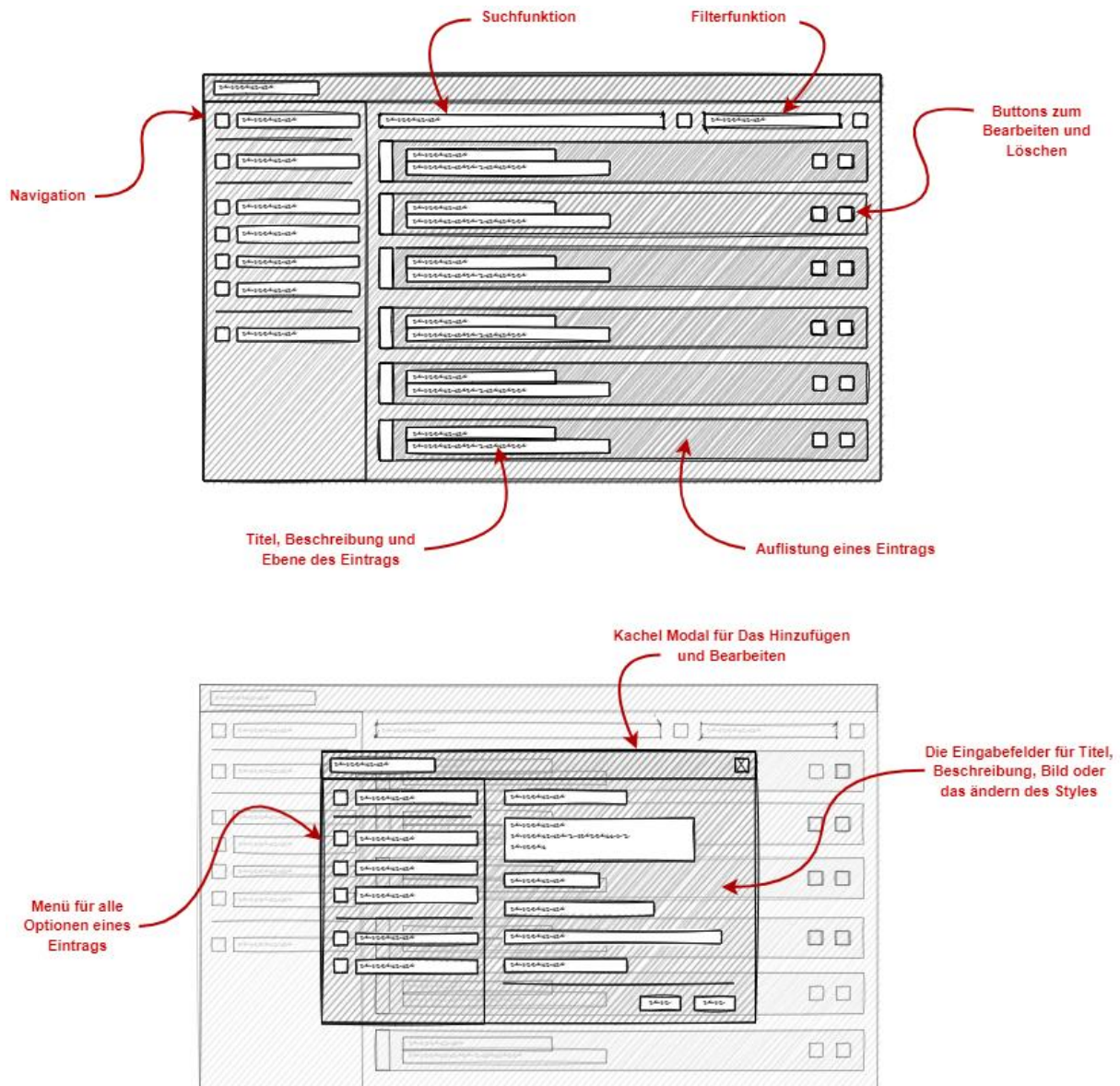


Die Seitenstruktur der Infotafel wurde mithilfe von Frontend-Skizzen digital entworfen, sie besteht aus dem strukturellen Aufbau der Oberfläche sowie dem dynamischen Inhalt, welcher abhängig von der aktuell abgerufenen Seite ist.



### 7.3. Entwurf des Infotafel-Managers

Das Frontend des Infotafel-Managers soll nur auf Desktop-Endgeräten verwendet werden und auch nicht extern erreichbar sein. Daher wurde eine Web-Anwendung als praktischstes Mittel festgelegt. Die Weboberfläche muss dabei nach dem Schema des Responsive Designs nur für normale Desktop-PCs und Windows-Tablets mit variierenden Bildschirmgrößen geeignet sein. So sollen insbesondere alle Mitarbeiter\*innen die Seite von ihrem Desktop-PC oder Windows-Tablet verwenden können. Dementsprechend ist eine Skalierung abhängig von der Bildschirmgröße und eine für Touch-Eingaben geeignete Bedienung hierbei notwendig.



Die Seitenstruktur des Infotafel-Managers wurde mithilfe von Frontend-Skizzen digital entworfen, sie besteht aus dem strukturellen Aufbau der Oberfläche sowie dem dynamischen Inhalt, welcher abhängig von der aktuell abgerufenen Seite ist.

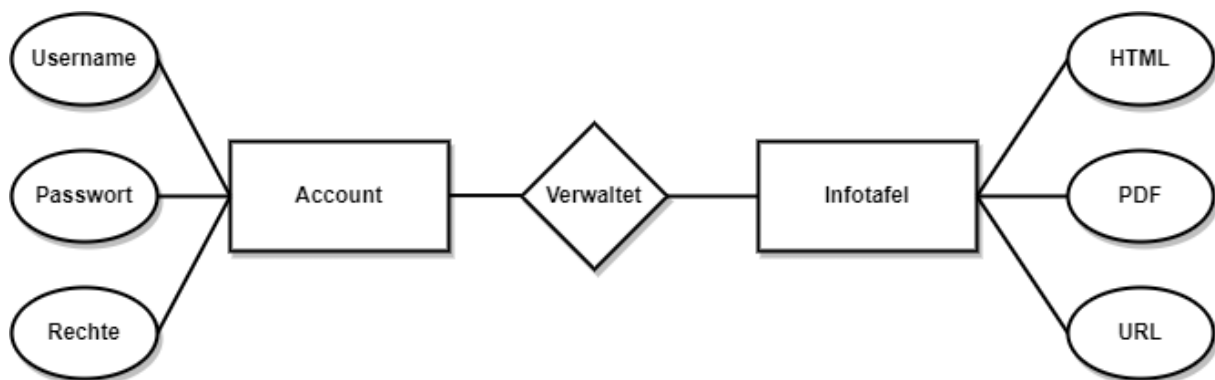
## 8. Backend

### 8.1. Backend Funktionen

Das Backend muss zwei primäre Funktionen erfüllen. Erstens die Synchronisierung der Datenbank mit dem Infotafel-Manager und zweitens die Bereitstellung der Daten für das Frontend der Infotafel. Dies geschieht mit PHP und Ajax.

### 8.2 Datenbank

Die Datenbankstruktur orientiert sich am Aufbau der Einträge. Jeder Eintrag besitzt eine eigene eindeutige UCID. Es muss jedoch berücksichtigt werden, dass manche Dateien unabhängig gespeichert und verändert werden können.



### 8.3. Bereitstellung der Daten

Die Daten werden über das Aufrufen bestimmter vordefinierter PHP-Funktionen bereitgestellt. Dabei können die Daten sowohl vor Seitenaufbau serverseitig vorformatiert oder auch in den entsprechenden Anwendungsfällen -wie der Suche oder der Administratoransicht- asynchron über JavaScript geladen werden.

### 8.4. Ablegen von Dateien

Dateien -wie Bilder und PDFs-, die über den Infotafel-Manager hochgeladen werden, werden über bereits vordefinierte PHP-Funktionen mit Simple-PHP in einem Separaten Ordner mit den jeweiligen Dateitypen in die UCID des Eintrags umbenannt und gespeichert.

### 8.5. Sicherheit mit PHP

Um zu gewährleisten, dass der Infotafel-Manager nicht durch unbefugte Personen genutzt werden kann, benötigt man ein Passwort, welches vorher in der Datenbank hinterlegt wurde, um sich im Login-Bildschirm des Managers anzumelden. Alle relevanten Seiten des Infotafel-Managers sind durch das Auslesen des Login-Cookies in der PHP-Session geschützt, so dass man immer auf die Login-Seite zurückkehrt, wenn kein Login-Cookie eines Users gesetzt ist.

## 9. Testphase

### 9.1. Entwicklungstests

Schon bei der Entwicklung der Infotafel wurden im größeren Stil kleinere Tests angewendet, um Fehler im Code auszuschließen. Die Tests waren erfolgreich und die Funktionalität des getesteten Codes wurde somit gesichert.

Darstellung im Browser:	Testergebnis:
Google Chrome	Erfolgreich
Mozilla Firefox	Erfolgreich
Microsoft Edge	Teilweise Erfolgreich

Sowohl die Infotafel als auch der Manager wurden im Google Chrome, Mozilla Firefox und Microsoft Edge Browser getestet. Die Darstellung funktioniert ohne Darstellungsfehler. Nur im Google Chrome und Mozilla Firefox Browser. Microsoft Edge verursachte bei der Darstellung vereinzelt Fehler bei einigen HTML und CSS Komponenten.

### 9.2. Funktionstests

Beim Funktionstest wurde getestet, ob man mit dem Manager einen Eintrag anlegen, bearbeiten und anschließend wieder löschen kann. Es wurde dabei auch darauf geachtet, dass währenddessen auf der Infotafel sich der Inhalt aktualisieren und wiedergeben lässt. Der Test der Infotafel wurde erfolgreich abgeschlossen und erfüllt somit die Anforderungen und kann übergeben werden.

Funktionalität der Infotafel:	Testergebnis:
Interface wird korrekt dargestellt	Erfolgreich
Kacheln mit dem Touchscreen schieben	Erfolgreich
Kachel mit einem Doppeltippen öffnen	Erfolgreich
Kachelinhalt wird korrekt dargestellt	Erfolgreich
Animationen laufen flüssig	Erfolgreich

Der Test der Infotafel verlief erfolgreich. Das Interface wurde korrekt dargestellt und man kann durch alle Kacheln mit dem Finger wischen und sich mit einem Doppeltippen den Inhalt jeder Kachel anzeigen lassen und wieder schließen.

<b>Funktionalität des Managers:</b>	<b>Testergebnis:</b>
Anmeldung	Erfolgreich
Auflistung aller Einträge	Erfolgreich
Erstellen eines Eintrags	Erfolgreich
Bearbeiten eines Eintrags	Erfolgreich
Löschen eines Eintrags	Erfolgreich
Ausloggen	Erfolgreich

Der Manager-Funktionstest verlief schon wie bei der Infotafel erfolgreich. Die Anmeldung kann nicht umgangen werden und funktioniert nur mit einem schon eingerichteten Account. Beim Anlegen eines Eintrags werden alle Daten erfolgreich ohne Fehler an die Datenbank übertragen. Das Editieren eines Eintrags erfolgt auch erfolgreich ohne Fehler sowie die vollständige Löschung eines Eintrags. Das Ausloggen des Users funktioniert auch ohne Probleme.

## 10. Soll-/Ist-Vergleich

Die nachfolgende Tabelle zeigt die Gegenüberstellung des geplanten Zeitaufwands zur tatsächlich erforderlichen Zeit des Projekts:

<b>Projektphasen:</b>	<b>Soll:</b>	<b>Ist:</b>	<b>Begründung:</b>
Planung	6 Std.	6 Std.	
Entwurf	15 Std.	8 Std.	Die Anforderungen wurden teilweise angepasst, daraus ergaben sich neue Ansätze die mehr Zeit beanspruchten.
Durchführung	30 Std.	45 Std.	Es wurden bereits Test während der Entwicklung durchgeführt, welche viel Zeit erforderten.
Tests	9 Std.	1 Std.	Hier wurde ein Großteil bereits bei der Durchführung bewerkstelligt, daher die Zeitersparnis.
Abschluss	10 Std.	10 Std.	
<b>Gesamt</b>	<b>70 Std.</b>	<b>70 Std.</b>	

## **11. Abnahme**

Nach dem Fertigstellen der Infotafel gemäß den Anforderungen wurde die Software dem Ausbilder demonstriert. Dafür wurde die Datenbank mit Beispiel-Daten gefüllt und verwendet, die bereits in den Entwicklungsarbeiten genutzt wurden. Nach einer kurzen Demonstration der Software und ihren Features wurde sie auf einen Mini-PC hinter dem großen Touchscreen installiert und entsprechend mit eigenen Daten gefüllt.

## **12. Fazit**

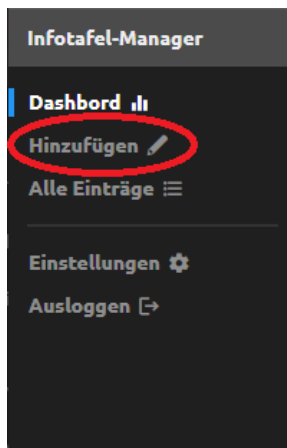
Für die Durchführung des Projekts sollte in Zukunft mehr Zeit eingeplant werden, um Zeit für mehr und auch bessere Features zu bieten. Es sind aber Verbesserungen und Erweiterungen nach dem Projekt geplant.

## 13. Anhänge

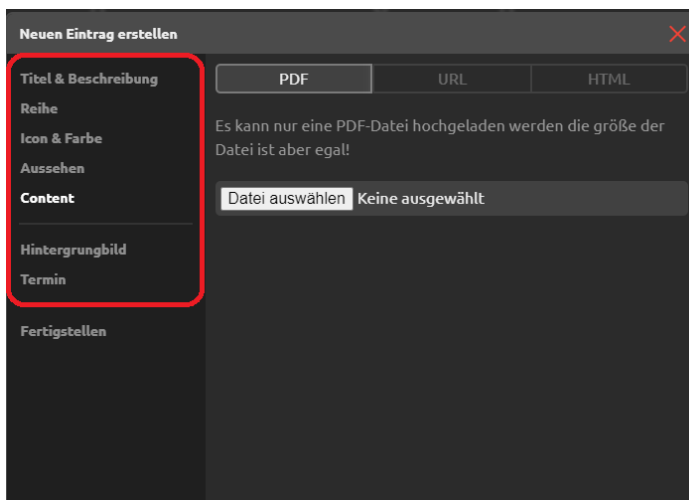
### 13.1. Benutzerhandbuch

Einen Eintrag erstellen:

1. Drücken Sie auf der Hauptseite des Managers links in der Navigation auf „**Hinzufügen**“.



2. Sobald sich das Fenster **Neuen Eintrag erstellen** geöffnet hat, befindet sich links ein Menü mit allen Optionen eines Eintrags. Die Optionen **Titel**, **Beschreibung**, **Farbe**, **Reihe**, **Icon** und **Content** sind allerdings hierbei Pflichtfelder. Die Felder **Farbe**, **Icon**, **Termin** und **Hintergrundbild** sind Optional und mit einem Default Wert belegt. Auf der rechten Seite befinden sich die jeweils dazugehörigen Eingabefelder der Optionen.

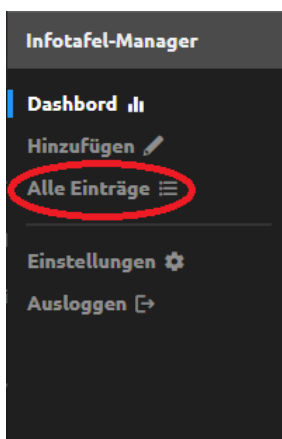


3. Gehen Sie alle Optionen durch, die für Ihren neuen Eintrag relevant sind und füllen sie die Felder aus.
4. Wenn Sie fertig sind, drücken Sie zum Abschluss im linken Menü auf „**Fertigstellen**“ und bestätigen sie mit „**Okay**“.

5. Nun ist der Eintrag angelegt und kann auf der Infotafel von den Benutzer\*innen angesehen werden.

Einen Eintrag bearbeiten:

1. Drücken Sie auf der Hauptseite des Managers links in der Navigation auf „**Alle Einträge**“.



2. Navigieren Sie zum gewünschten Eintrag oder suchen Sie ihn mit der Suchzeile oben und drücken dann rechts auf die Bearbeiten Schaltfläche.

Titel	Beschreibung	CardID	Level	Erstellt	
Beispiel Titel	Beispiel Beschreibung	8952467886	1	24.05.2022	<span>Bearbeiten</span> <span>X Löschen</span>
Beispiel Titel	Beispiel Beschreibung	9254927401	2	24.05.2022	<span>Bearbeiten</span> <span>X Löschen</span>
Beispiel Titel	Beispiel Beschreibung	0230024183	3	24.05.2022	<span>Bearbeiten</span> <span>X Löschen</span>

3. Sobald sich das Fenster „**(Eintragsname) bearbeiten**“ geöffnet hat, befindet sich links ein Menü mit allen Optionen eines Eintrags. Die Optionen **Titel**, **Beschreibung**, **Farbe**, **Reihe**, **Icon** und **Content** sind allerdings hierbei Pflichtfelder. Die Felder **Farbe**, **Icon**, **Termin** und **Hintergrundbild** sind Optional und mit einem Default Wert belegt. Auf der rechten Seite befinden sich die jeweils dazugehörigen Eingabefelder mit den bereits ausgefüllten Werten der Optionen.

**Beispiel Titel bearbeiten** [X]

**Titel & Beschreibung**

Reihe

Icon & Farbe

Aussehen

Content

Hintergrundbild

Termin

Fertigstellen

**Titel**  
 Lege den Titel der Kachel fest. Nehme ein Thema oder ein Schlagword was den Inhalt sehenswert macht.

Beispiel Titel

**Beschreibung**  
 Beschreibe kurz den Inhalt der Kachel und um was es sich handelt oder einfach was interessantes.

Beispiel Beschreibung

4. Gehen Sie alle Optionen durch, die für das bearbeiten des Eintrags relevant sind und ändern Sie sie wenn nötig.
5. Wenn Sie fertig sind, drücken Sie zum Abschluss im linken Menü auf „**Fertigstellen**“ und bestätigen mit „**Okay**“.

**Beispiel Titel bearbeiten** [X]

**Titel & Beschreibung**

Reihe

Icon & Farbe

Aussehen

Content

Hintergrundbild

Termin

**Fertigstellen**

**Titel**  
 Lege den Titel der Kachel fest. Nehme ein Thema oder ein Schlagword was den Inhalt sehenswert macht.

Beispiel Titel

**Beschreibung**  
 Beschreibe kurz den Inhalt der Kachel und um was es sich handelt oder einfach was interessantes.

Beispiel Beschreibung

6. Nun ist der Eintrag bearbeitet und kann auf der Infotafel von den Benutzer\*innen angesehen werden.

Einen Eintrag löschen:

1. Drücken Sie auf der Hauptseite des Managers links in der Navigation auf „**Alle Einträge anzeigen**“.
2. Navigieren Sie zum gewünschten Eintrag oder suchen Sie ihn mit der Suchzeile oben und halten Sie dann rechts die Löschen-Schaltfläche für 5 Sekunden gedrückt.



Titel	Beschreibung	CardID	Level	Erstellt	
Beispiel Titel	Beispiel Beschreibung	8952467886	1	24.05.2022	<input type="button" value="Bearbeiten"/> <input checked="" type="button" value="Löschen"/>
Beispiel Titel	Beispiel Beschreibung	9254927401	2	24.05.2022	<input type="button" value="Bearbeiten"/> <input type="button" value="Löschen"/>
Beispiel Titel	Beispiel Beschreibung	0230024183	3	24.05.2022	<input type="button" value="Bearbeiten"/> <input type="button" value="Löschen"/>

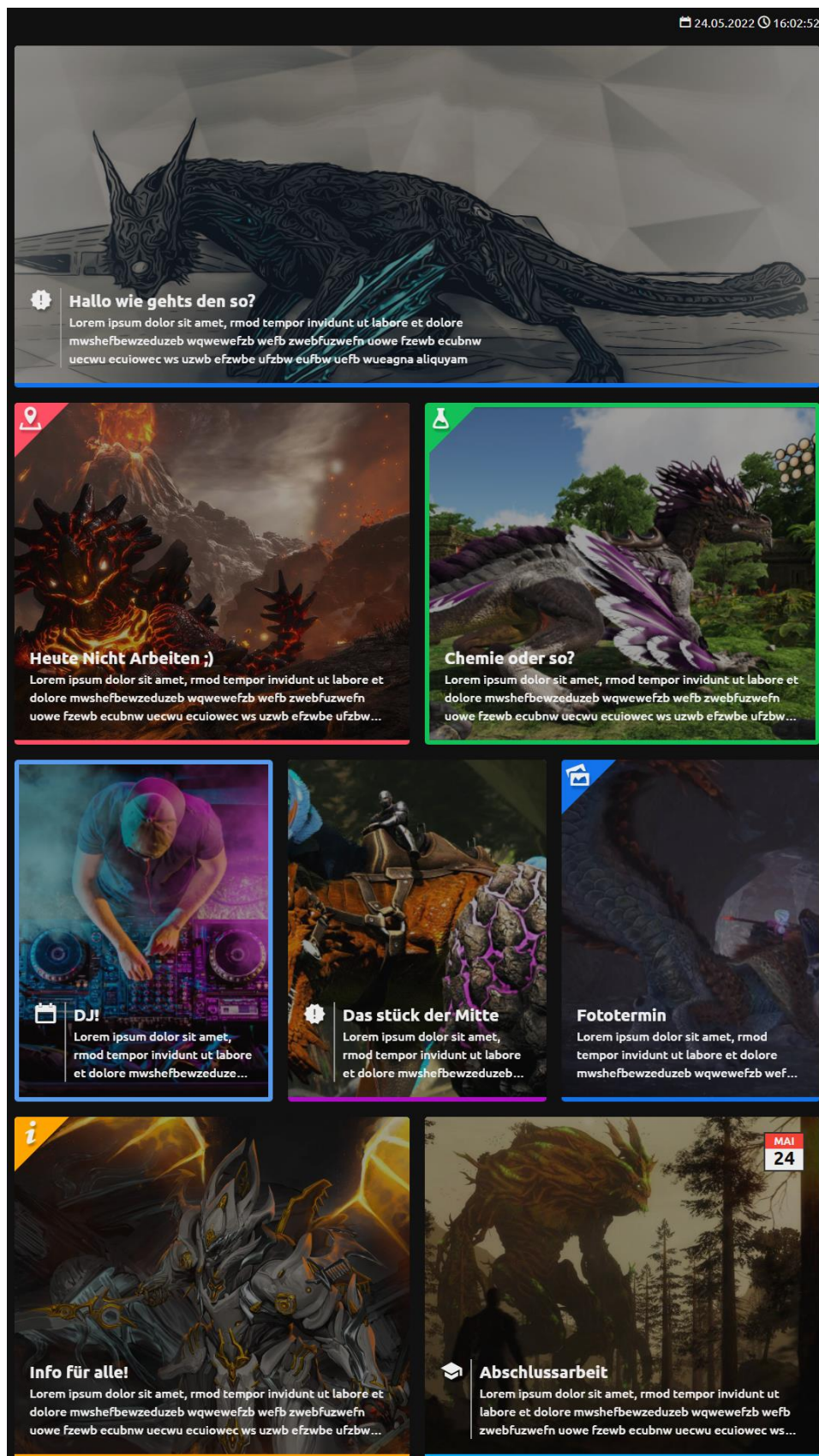
3. Zum Abschluss müssen Sie die Permanente Löschung des Eintrags mit „**Okay**“ bestätigen.
4. Nun ist der Eintrag gelöscht und kann auf der Infotafel von den Benutzer\*innen nicht mehr angesehen werden.

## 13.2. Glossar

Wort:	Erklärung:
UCID	Unique-Content-Identifikationsnummer
Simple-PHP	Simple-PHP ist eine Library mit vorgefertigten PHP-Funktionen, die vom mir bereits vor dem Projekt entwickelt und erstellt wurde, Sie enthält grundlegende Funktionen, die mehrfach in sämtlichen meiner Projekte zum Einsatz kommen und darin enthalten sind z.B. Funktionen für Datenbankzugriffe, Generieren einer Unique-ID und das Abspeichern von Dateien.
JSON	Die JavaScript Objekt Notation ist ein Datenformat zur Serialisierung von strukturierten Daten. Es dient zum Datenaustausch zwischen Anwendungen.
CSS	Cascading Style Sheet, sie werden für die stilistische Gestaltung für Webseiten verwendet.
PHP	PHP Hypertext Präprozessor, serverseitige Programmiersprache für Webanwendungen
MySQL	Relationales Datenbankmanagementsystem
JavaScript	Frontend-Programmiersprache, welche vom Browser interpretiert wird.
Ajax	Asynchrones JavaScript
BBW	Berufsbildungswerk St. Franziskus Abensberg – Das Unternehmen, in dem das Projekt durchgeführt wird.

### 13.3. Screenshots

#### Infotafel mit Beispiel-Inhalten



Neue Werte einer Datenbank hinzufügen.

```
function db_add(string $TABLE, array $DATA)
{
    include 'db.php';

    $INDEX = '';
    $VALUE = '';

    foreach ($DATA as $INDEXES => $VALUES) {

        if ($VALUES == end($DATA))
        {
            $INDEX .= $INDEXES."";
            $VALUE .= "".$VALUES."";
        }
        else
        {
            $INDEX .= $INDEXES.", ";
            $VALUE .= "".$VALUES."', ";
        }
    }

    $EXEC_DATA = "INSERT INTO $TABLE( $INDEX ) VALUES ( $VALUE )";
    $db -> exec($EXEC_DATA);
}
```

Werte aus der Datenbank in einer Variable speichern.

```
function db_get(string $TABLE, string $ID_NAME, string $ID_VALUE, $DATA)
{
    include 'db.php';

    if (is_array($DATA) && $DATA != 'all') // EINZELNE INDEX-EINTRÄGE
    {
        $NEW_STRING = '';
        $ARRAY = array();

        foreach ($DATA as $VALUE) { if ($VALUE == end($DATA)) { $NEW_STRING .= $VALUE; } else { $NEW_STRING .= $VALUE.", "; } }

        $QUERY = $db -> prepare("SELECT $NEW_STRING FROM $TABLE WHERE $ID_NAME = '$ID_VALUE'");
        $QUERY -> execute();
        $RESULT = $QUERY -> fetchAll();

        foreach ($RESULT as $VALUE) { foreach ($DATA as $INDEX) { $ARRAY += array($INDEX => $VALUE[$INDEX]); } }
    }
    else // ALLE INDEX-EINTRÄGE
    {
        $ARRAY = array();

        $QUERY1 = $db -> prepare("DESCRIBE $TABLE");
        $QUERY1 -> execute();
        $RESULT1 = $QUERY1 -> fetchAll(PDO::FETCH_COLUMN);

        $QUERY2 = $db -> prepare("SELECT * FROM $TABLE WHERE $ID_NAME = '$ID_VALUE'");
        $QUERY2 -> execute();
        $RESULT2 = $QUERY2 -> fetchAll();

        foreach ($RESULT1 as $INDEX) { foreach (array_values($RESULT2) as $VALUE) { $ARRAY += array($INDEX => $VALUE[$INDEX]); } }
    }

    return $ARRAY;
}
```

Werte in der Datenbank aktualisieren.

```
function db_update(string $TABLE, string $ID_NAME, string $ID_VALUE, array $DATA)
{
    include 'db.php';

    $VALUE = '';

    foreach ($DATA as $INDEX => $VALUES) { if ($VALUES == end($DATA)) { $VALUE .= $INDEX."=".$VALUES.""; } else { $VALUE .= $INDEX."=".$VALUES."", "; } }

    $UPDATE = $db -> prepare("UPDATE $TABLE SET $VALUE WHERE $ID_NAME = '$ID_VALUE'");
    $UPDATE -> execute();
}
```

Prüfen wie oft ein Wert in der Datenbank existiert.

```
function db_count(string $TABLE, string $INDEX, string $VALUE)
{
    include 'db.php';

    if ($VALUE != 'all')
    {
        $ZEILE = $db -> prepare("SELECT * FROM $TABLE WHERE $INDEX = '$VALUE'");
        $ZEILE -> execute();
        $RESULT = $ZEILE -> fetchAll();
        return count($RESULT);
    }
    else
    {
        $ZEILE = $db -> prepare("SELECT * FROM $TABLE WHERE $INDEX");
        $ZEILE -> execute();
        $RESULT = $ZEILE -> fetchAll();
        return count($RESULT);
    }
}
```

Eine Datei im Zielordner speichern und Umbenennen.

```
function save_file(string $PATH, $FILE, string $NEW_FILENAME, array $VALIDTYPES)
{
    $FILENAME = $FILE['name'];

    $PATHTOFILE = $PATH.$FILENAME;

    $FILETYPE = pathinfo($PATHTOFILE, PATHINFO_EXTENSION);
    $FILETYPE = strtolower($FILETYPE);

    if(in_array(strtolower($FILETYPE), $VALIDTYPES))
    {
        {
            if(move_uploaded_file($FILE['tmp_name'], $PATH.$NEW_FILENAME.'.'.$FILETYPE))
            {
                {
                    return $NEW_FILENAME.'.'.$FILETYPE;
                }
            }
        }
    }
}
```

**Anschrift Prüfling:**

Maximilian Merkl  
Halbergmooserstraße 17a  
85356 Freising

**Anschrift Ausbildungsbetrieb:**

Berufsbildungswerk St. Franziskus  
Abensberg  
Regensburger Straße 60  
93326 Abensberg

## Protokoll der durchgeführten Projektarbeit

**Prüf-Nr.:**

944073

**Ausbildungsberuf:**

Fachinformatiker für Anwendungsentwicklung

### 1. Arbeitszeit:

1.1 Die vom Prüfungsteilnehmer kalkulierte Zeit entspricht der betrieblichen Kalkulation.

☒ ja

☐ nein

Wenn nein: Sie ist um \_\_\_\_ % höher \_\_\_\_ % niedriger

1.2 Das Projekt wurde vom Prüfungsteilnehmer in der kalkulierten Zeit komplett fertiggestellt (einschließlich eventueller Nacharbeit):

☒ ja

☐ nein

Wenn nein: Um \_\_\_\_ Stunden früher fertig geworden,  
\_\_\_\_ Stunden länger gebraucht.

### 2. Ausführung:

2.1 Wurde das Projekt entsprechend dem eingereichten Konzept ausgeführt?

☒ ja

☐ nein

Wenn nein: Welche Änderungen ergaben sich?

---

---

---

bitte weiter auf der Rückseite!

**Prüfungsteilnehmer:**

Maximilian Merkl

BBW St. Franziskus Abensb

Name / Vorname

Firma

**2.2 Wurde das Projekt selbständig und ohne fremde Hilfe ausgeführt?**☒ ja☐ nein

Wenn nein: Begründung und Umfang der Hilfestellung.

---

---

---

**2.3 Das Projekt konnte ohne Nacharbeit in einem einwandfreien Zustand übergeben werden.**☒ ja☐ nein

Wenn nein: Begründung und Umfang der Nacharbeit.

---

---

---

**3. Dokumentation:****3.1 Die Dokumentation wurde vom Prüfungsteilnehmer selbständig und ohne fremde Hilfe erstellt.**☒ ja☐ nein

Wenn nein: Welche Hilfestellung wurde gegeben?

---

---

---

**3.2 Die Dokumentation entspricht den betrieblichen Anforderungen.**☒ ja☐ nein

Wenn nein: Worin besteht die Abweichung?

---

---

---

19.05.2022

Datum



Unterschrift Prüfungsteilnehmer



Unterschrift Projektverantwortlicher