

Abschlussprüfung 2022

Fachinformatiker für Anwendungsentwicklung

Dokumentation zur betrieblichen Projektarbeit

**Online- Fehlzeitenerfassen**

Durchführungszeitraum: 05.10.2022 - 09.11.2022

**Prüfungsbewerber**

Mohannad Hafez

Hans-Böckler-Allee 56

52074 Aachen

Logo

Description automatically generated

**Ausbildungsbetrieb**

Wilke Technology

Heider-Hof-Weg 23D

52080 Aachen

**Ausbilder**

Jürgen Wilke

**Projektunterstützer**

Tobias Esser

**Inhalt**

[1 Einleitung 3](#_Toc100836549)

[1.1 Projektbeschreibung 3](#_Toc100836550)

[2 Ausgangsituation (IST-Analyse) 3](#_Toc100836554)

[2.1 Projektziel 4](#_Toc100836555)

[2.2 Kundenwünche 4](#_Toc100836556)

[2.3 Abweichen zum Antrag 4](#_Toc100836557)

[2.4 Projektumfeld 5](#_Toc100836557)

[2.5 Prozessschnittstelle 6](#_Toc100836557)

[3 Ressourcen und Ablaufplanung 6](#_Toc100836558)

[3.1 Ressourcenplanung 6](#_Toc100836559)

[3.2 Projektkosten 7](#_Toc100836560)

[3.3 Ablaufplanung 7](#_Toc100836561)

[4 Durchführung und Auftragsbearbeitung 7](#_Toc100836564)

[4.1 Prozrssschritte 7](#_Toc100836565)

[4.2 Vorgehensweise 8](#_Toc100836565)

[4.3 Arbeitsschritte 8](#_Toc100836565)

[4.4 Abweichungen, Anpassungen, entscheidungen 14](#_Toc100836565)

[5 Projektergebnisse 15](#_Toc100836567)

[5.1 Soll-Ist-Vergleich 15](#_Toc100836568)

[5.2 Qualitätskontrolle 15](#_Toc100836569)

[5.3 Abweichungen, Anpassungen, Entwicklungsmöglichkeiten 15](#_Toc100836565)

[6 Dokumentation 16](#_Toc100836570)

[6.1 Benutzerhandbuch 16](#_Toc100836571)

[7 Anhang 19](#_Toc100836578)

[8.1 Abkürzungsverzeichnis 19](#_Toc100836579)

[8.2 Quellenverzeichnis 20](#_Toc100836580)

[8.3 Code-Anhang 20](#_Toc100836581)

# Einleitung

Die folgende Projektdokumentation beinhaltet den Ablauf des IHK-Abschlussprojektes, welches ich zum Ende meiner Ausbildung zum Fachinformatiker der Fachrichtung Anwendungsentwicklung durchgeführt habe.

## Projektbeschreibung

Entwicklung einer webbasierten Projektzeitverfassung mit zentralen Datenbanken.  
Zur Realisierung sollen die Webtechnologien und Programmiersprachen PHP, MySQL, HTML, CSS, JavaScript zum Einsatz kommen.

# Ausgangssituation (IST-ANALYSE)

# Die Mitarbeiter unseres Unternehmens melden sich über eine passwortgeschützte Webseite als abwesend. Hierbei gibt es nur einen Satz Login-Daten, die allen Mitarbeiter bekannt sind. Es handelt sich um ein simples Formular, bei dem Vorname, Nachname, Grund der Abwesenheit (Krankheit, Arztbesuch, Sonstiges), sowie Start- und Endzeitpunkt angegeben werden. Durch einen Button wird eine E-Mail mit den eingegebenen Daten erzeugt und an eine E-Mail-Adresse geschickt, die nur von berechtigten Personen eingesehen werden kann.

So sah diese Webseite bisher aus:



## Projektziel

Alle Angestellten sollen Ihre Fehlzeiten weiterhin per Web-Browser in ein zentrales System eintragen. Ein Administrator soll hier die Möglichkeit haben neue Benutzer anzulegen und Vorhandene zu ändern oder zu löschen.

Die Fehlzeitendaten sollen in einer Datenbank gespeichert werden.  
Die in der Datenbank gespeicherten Fehlzeiten sollen über eine interne API abrufbar sein.

Diese Schnittstelle dient der automatischen Eintragung in die Stundenzettel mithilfe einer anderen Applikation und Organisation der Vertretung eines abwesenden Mitarbeiters.

Die zeitnahe Löschung nicht mehr benötigter Datenbankeinträge soll ebenfalls über einen API-Aufruf ausgelöst werden. Hier werden automatisch alle Fehlzeiten deren Endzeitpunkt in der Vergangenheit liegen entfernt.

Da die Applikation bei unserem Webhoster in Betrieb genommen wird muss sie mit PHP 8.0 und MariaDB 10 funktionieren.

Zudem soll ein einwandfreier Betrieb mit den aktuellen Versionen von Google Chrome und Mozilla Firefox sichergestellt werden.

## Kundenwünsche

* Login mit mitarbeiterspezifischen Benutzername & Passwort
* Erstellung neuer Benutzer durch berechtigte Benutzer
* Fehlzeiterfassung der Benutzer
* API zum lesen aktueller Fehlzeiten und Löschung veralteter
* Einfache Bedienung

## Abweichen zum Antrag

Nach einer Besprechung mit dem Projektunterstützer und Genehmigung von Herrn Timo Beyel habe ich die Teilaufgabe der API zum Lesen aktueller Fehlzeiten und Löschung veralteter auf einen späteren Zeitpunkt verschoben. Dadurch ist diese Aufgabe nicht ein Teil meines Projektes.

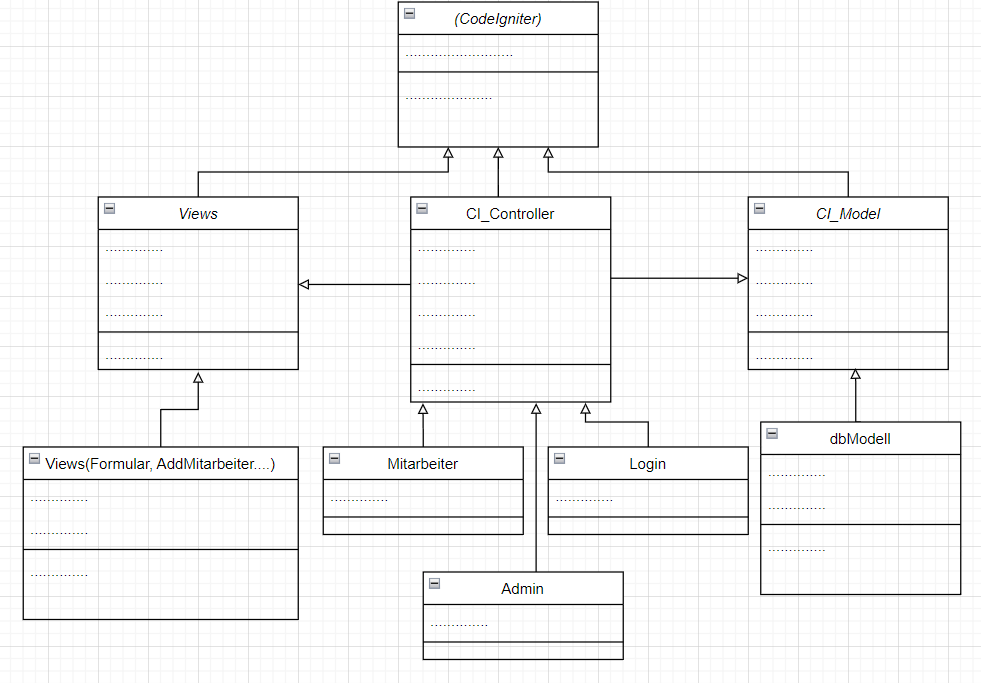
Das hat auch die Folge, dass es kein Auslöser mehr zur Verfügung steht, um veraltete Einträge aus der Datenbank zu löschen.

# Projektumfeld

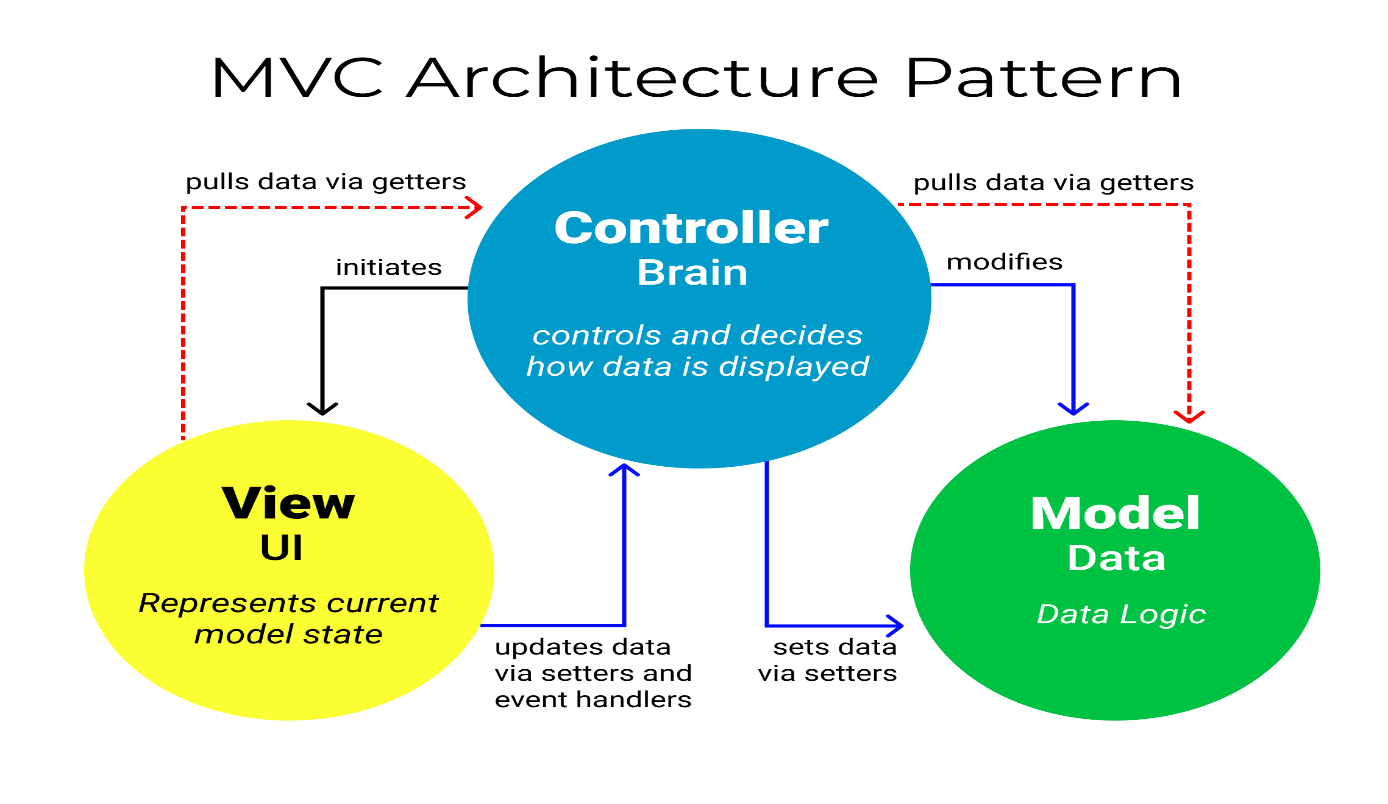
***CodeIgniter***

CodeIgniter ist ein seit 2014 erhältliches, in PHP geschriebenes, quelloffenes Webframework.

CodeIgniter ist schlank gehalten, wodurch eine hohe Performance erreicht wird und die Einarbeitungszeit im Vergleich zu anderen Frameworks kurz ist. Zentraler Bestandteil ist die Model View-Controller-Architektur (MVC). CodeIgniter enthält eine große Zahl an Bibliotheksklassen und Hilfsfunktionen, die mir Standardaufgaben abnehmen.



MVC-Architektur:



***Bootstrap***

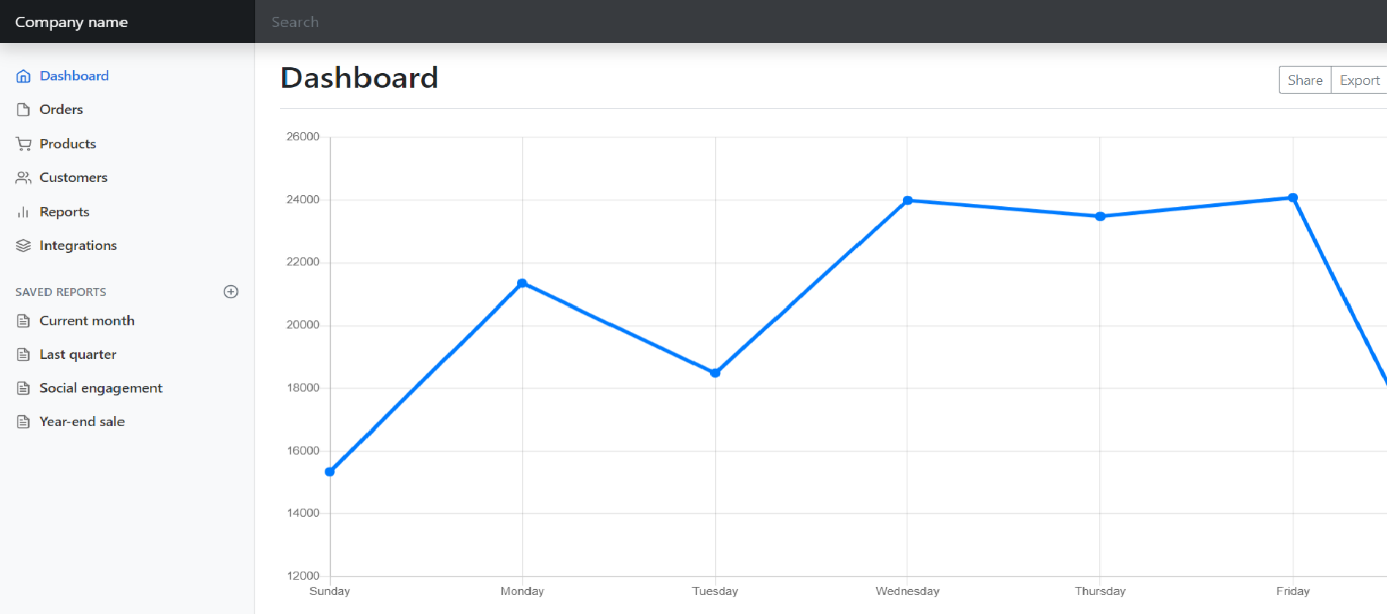
Bootstrap ist ein kostenloses und quelloffenes Frontend-Web-Framework zur Erstellung von Webseiten. Es dient nur der Front-End-Entwicklung und enthält HTML- und CSS-basierte Designvorlagen für Typografie, Formulare, Schaltflächen und Navigation.

## Prozessschnittstellen

Ich für Anzeigeschnittstelle habe ich ein Template von Bootstrap heruntergeladen, dann habe ich sie bearbeitet, da meine Anforderung anpasst.

Ansonsten hätte ich in der angegebenen Zeit nicht das Projekt schaffen könnte.

<https://getbootstrap.com/docs/5.0/examples/dashboard/>



# Ressourcen und Ablaufplanung

In der Projektplanung soll die notwendige Zeit und die benötigten Ressourcen in eine für die Entwicklung sinnvolle Struktur gebracht werden. Dazu wird der Ablauf des Projektes geplant und erfasst.

## Ressourcenplanung

Die Ressourcenplanung zeigt alle für das Projekt eingesetzten Ressourcen auf.

In meinem Fall sind die:

* Arbeitszeit
* Entwicklungsumgebung
* Programmierfähigkeiten
* Softwarelizenzen

Die im Projekt eingesetzten Softwarebibliotheken unterliegen der GPL V3. Die zur Entwicklung notwendigen Hard- und Softwareprodukte wurden vom Ausbildungsbetrieb zur Verfügung gestellt.

## Projektkosten

Im Folgenden werden die anfallenden Kosten für dieses Projekt kalkuliert.  
Der Kostensatz eines Auszubildenden liegt bei ca. 11,25 € und der eines Mitarbeiters 20 €. Für die Ressourcennutzung.

In meinem Fall kostet eine Arbeitsstunde den Betrieb nur 0,625€ gekostet, da ich nur mit 100€ pro Monat gerechnet werde.

160 Std monatlich:100€ | 100€:160Std = 0,625€ pro Std.

Aus diesen Daten entsteht folgende Kalkulation:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Aufgabe | Mitarbeiter | Zeit | Personalkosten | Ressourcenkosten |
| Entwicklung | Umschüler | 79h | 49,375€ | Kostenlos (Open-Source) |

## Ablaufplanung

Für das Projekt standen mir 79 Stunden zur Verfügung, welche von mir auf einzelne Phasen verteilt wurden. Diese Phasen lassen sich grob auf Analyse, Entwurf, Implementierung, Deployment und Dokumentation aufbrechen. Dadurch entstehen fünf Hauptphasen, welche wiederum in kleinere Unterphasen zerlegt werden, wodurch eine detaillierte Ansicht aller Aufgaben entsteht.

|  |  |
| --- | --- |
| Projektphase | Dauer in Stunden |
| Analyse | 4 |
| Entwurf | 13 |
| Implementierung | 40 |
| Deployment & Abnahme | 2 |
| Dokumentation | 20 |
| Summe | 79 Stunden |

# Durchführung und Auftragsbearbeitung

## Prozessschritte

Analyse

* Durchführung einer Ist-Analyse
* Erstellung eines Soll-Konzepts

Entwurf

* Erstellung einer GUI
* Datenbank Struktur erstellen

Implementierung

* Code erstellen
* Code cleanup

Abnahme

* Code-Review mit dem Ausbilder
* Funktionstest intern
* Gegebenenfalls Fehlerkorrekturen Dokumentation
* Erstellung der Projektdokumentation
* Erstellung des Benutzerhandbuch

## Vorgehensweise

Um eine flexible Umsetzung der Anforderungen zu ermöglichen, habe ich mich für den

SCRUM Entwicklungsprozess entschieden. SCRUM ist für Projekte geeignet, in denen viel Unklarheit herrscht. Häufig sind dies Vorhaben, die nach neuen Lösungen suchen und deren Kombination aus unvorhersehbaren Risiken der Implementierung und noch unklaren Anforderungen bestehen.

## Arbeitsschritte

Zu Beginn meiner Arbeit bat mich der Projektunterstützer ein neues Repository auf GitHub einzurichten, dass ihm die Möglichkeit gibt meinen Fortschritt zu beobachten. Zusätzlich bietet das Arbeiten mit einer Versionsverwaltung besten Überblick über gemachte Änderungen und ein Onlinespeicher wie GitHub liefert gleichzeitig noch Schutz vor Datenverlust.

Als Frontend-Framework habe ich Bootstrap gewählt.  
Bootstrap bietet gute Möglichkeiten für Webentwickler geräteübergreifende Websites verschiedenster Art zu erstellen. Zu diesem Zweck bietet das Open-Source-Projekt diverse Gestaltungsvorlagen, die auf HTML und CSS basieren, sowie optionale JavaScript-Erweiterungen. Und dann habe ich das Framework bearbeitet und auf meine Anforderungen umgestellt.

Ich habe zunächst mein DatenBank (fehlzeitdb) hergestellt, die aus 2 Tabellen besteht.

1. tbl\_users: [id(int)PK, vorname(varchar), nachname(varchar), b\_name(varchar), pass(varchar), role(int)].
2. tbl\_grund: [gr\_id(int)PK, u\_id(int)FK, grund(int), note(varchar), von\_datum(datetime), bis\_datum(datetime), reg\_datum(datetime)].

Das Bild zeigt der Primärschlüssel und der Sekundärschlüssel mit einer Beziehung von n zu m.

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

Als Entwicklungsumgebung habe ich Visual Studio Code gewählt und dort nach dem Download von CodeIgniter ein neues Projekt erstellt.   
Einige Dateien mussten bearbeitet werden, wie zum Beispiel die root URL der Webseite:  
application\config\config.php

$config['base\_url'] = 'http://localhost/FehlzeitPro/';;//der Name meines Projektverzeichnis.

Wichtig ist hier die Aktivierung der Fehlerprotokollierung, indem man Schwellenwert über Null festlegen. Der Schwellenwert bestimmt, was protokolliert wird.

$config['log\_threshold'] = 4;

4 = All Messages.

Diese Fehlermeldungen werden in dem Pfad: application\config gespeichert und erleichtern einem immens die Fehlersuche.

Da ist die Encryption-Klasse verwenden will, um Passwörter abzulegen, muss ich einen Encryption Key festlegen.

$config['encryption\_key'] = 'UVvM3gosO8yK9JeijFwasEXD8kHs2G1y';

Klassen initialisieren  
Dies sind die Klassen, die sich in system/libraries oder Ihrem application/libraries Verzeichnis befinden, mit dem Zusatz der Datenbank-Bibliothek.

application\config\autoload.php

$autoload['libraries'] = array('database','session','encryption');

Zusätzlich muss dbmodel zu den models hinzugefügt werden mit dessen Hilfe die Seite später mit der Datenbank kommunizieren kann.

$autoload['model'] = array('dbmodel');

application\config\database.php

...

$db['default'] = array(

...

    'database' => 'fehlzeitdb',

  ...

);

Dann habe ich die Anmeldeansicht erstellt, die ein Textfeld für den Benutzernamen und ein weiteres für das Passwort enthält, das der Benutzer ausfüllen muss, um sich anzumelden.

Als nächstes erstellte ich mein Benutzerformular, das der Benutzer ausfüllen sollte, um seine Fehlzeit hinzuzufügen. Es enthält einige Überschriften, Textfelder, Drop-down und ein Optionsfeld (select tag).

Ich habe JavaScript verwendet, um Notizen und ein Textfeld ein- und auszublenden. Wenn der Arbeitnehmer bei einem Arztbesuch oder krank ist, muss er eine Arbeitsunfähigkeitsbescheinigung nachweisen, bzw. einen Bescheid des Arztes über die Dauer des Termins.

Wenn es irgendein sonstiger Grund ist, muss dieser in kurzen Worten angegeben werden.

Danach habe ich eine neue Datei application\helpers\info\_helper.php erstellt und, weil dieses Verzeichnis außerhalb der Scope ist, muss ich eine Instanz von der Klasse CI\_Controller erzeugen, damit ich die Funktion in dbmodel.php erreichen kann.

   function get\_autor\_name\_from\_DB($id)

    {

        $this->db->select('\*');

        $this->db->from('tbl\_users');

        $this->db->where('id', $id);

        $query = $this->db->get();

        $row\_array = $query->row\_array();

        return $row\_array;

    }

Formular ausfüllen: Nach dem Absenden des Formulars muss geprüft werden, ob alle Felder korrekt ausgefüllt sind, und einer der drei möglichen Gründe ausgewählt wurde.

Zusätzlich muss, wenn Sonstiges gewählt wurde, eine kurze Erklärung dazu geschrieben werden.

 function check\_mitarbeiter\_data() {

...

            if (mit\_name.length == 0) { // ist ein Name geschrieben?

                show\_error('Mitarbeitername ist ungültig!!');

            } else if (grund) {  // ist ein Grund gegeben?

                show\_error('Bitte wählen Sie einen Grund aus!!');

            } else if (radiovar == 3 && note.length == 0) { // wenn Sonstiges ausgesucht ist, muss eine Erklärung auch geschieben

                show\_error('Bitte schreiben Sie eine Kürze Erklärung!');

            } else if (von\_date.length == 0) { // ist ein Anfangsdatum gegeben?

                show\_error('Ab welchem Tag sind Sie abwesend?');

            } else if (bis\_date.length == 0) { // ist ein Enddatum gegeben?

                show\_error('Bis zu welchem Tag sind Sie abwesend?');

            } else if (bis\_uhr.length == 0) { // ist ein Endzeit gegeben?

                show\_error('Bis wie viel Uhr sind Sie abwesend?');

            } else {

                $('#add\_fehlzeit\_form').submit();

            }

        }

Radiobutton programmieren (JavaScript):

Wenn eine von den zwei oberen Optionen (Krankheit, Arztbesuch) von den drei Optionen ausgesucht wird muss ein Label angezeigt werden, das den Nutzer darauf hinweist, dass ein Nachweis vorgelegt werden muss, ansonst muss Label und Eingabefeld für die Begründung angezeigt werden.

        var radiovar = 0;

        $(function() {

            $('input[name="radio"]').on('click', function() {

                $('#error\_formular').hide(); // Fehlermeldung löschen wenn auf Radiobutton gedrückt werden

                if ($(this).val() == '3') {

                    $('#label\_id').show();

                    $('#note\_id').show();

                    $('#label2\_id').hide();

                    radiovar = 3;

                } else if ($(this).val() == '1' || $(this).val() == '2') {

                    $('#label2\_id').show();

                    $('#label\_id').hide();

                    $('#note\_id').hide();

                    radiovar = 0;

                }

            });

        });

Beim „Mitarbeiter bearbeiten“ und „Mitarbeiter addieren“ muss geprüft werden, ob der neue Benutzername schon in der DB existiert ist. Dafür habe ich eine Funktion in dbmodel.php geschrieben, um das zu überprüfen.

    function check\_is\_Bname\_existiert($mit\_benuzter)

    {

        $this->db->select('\*');

        $this->db->where('b\_name', $mit\_benuzter);

        $this->db->from('tbl\_users');

        $query = $this->db->get();

        $result\_array = $query->result\_array();

        if (!empty($result\_array)) {

            return 1; // Benutzername existier

        }

        else

        {

            return 0; // Benutzername existiert nicht

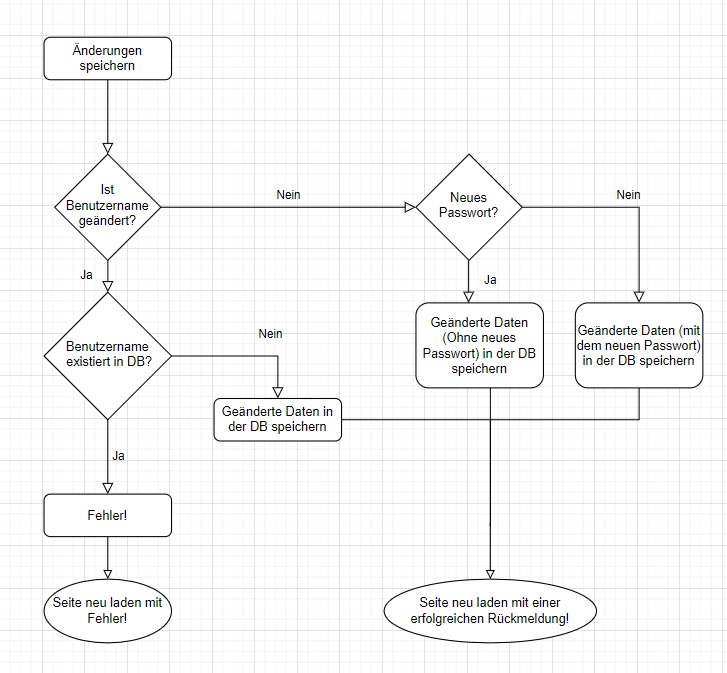
        }

    }

Beim Bearbeiten eines Benutzers war es einfacher zu prüfen, ob sich der Benutzername überhaupt geändert hat bevor geprüft wird, ob es diesen Namen schon gibt. Deswegen habe ich eine Funktion in dbmodel.php geschrieben, um das zu überprüfen.

function check\_is\_same\_Bname($mit\_id, $mit\_benuzter)

Jetzt habe ich alle notwendige Informationen, um meine Verzweigung in dem Kontroller schreiben zu können:



Im anderen Fall, bei Mitarbeiter addieren, muss nun geprüft werden, ob derselbe Benutzername schon in der DB existiert, aber ich habe das in anderen Weg gemacht.  
Ich habe in der dbmodell Funktion diese Verzweigung geschrieben.

    function add\_new\_mitarbeiter($mit\_name, $mit\_nachname, $mit\_benuzter, $mit\_password)

    {

      ...

        if (empty($result\_array)) { // check if the user name already exist in the DB

...

            $this->db->insert('tbl\_users', $data);

            return 1;

        } else {

             return 0; // Benutzername existiert schon in DB

        }

    }

Dann muss in dem Kontroller die Verzweigung geschrieben werden:

    public function con\_add\_mitarbeiter()

    {

...

        $is\_mit\_added = $this->dbmodel->add\_new\_mitarbeiter($mit\_name, $mit\_nachname, $mit\_benuzter, $key);

        if ($is\_mit\_added == 1) {// Benutzername existiert nicht in der DB

            $this->session->set\_userdata('success', 1);

        } else {

            $this->session->set\_userdata('fail', 1);

        }

        ...

    }

Für die Verschlüsselung habe ich die Verschlüsselungsbibliothek von CodeIgniter benutzt. Die Verschlüsselungsbibliothek bietet eine bidirektionale Datenverschlüsselung.

Standardmäßig verwendet die Verschlüsselungsbibliothek die AES-128-Verschlüsselung im CBC-Modus (Cipher Block Chaining Mode).

$key = $this->encryption->encrypt($mit\_password); // Passwort verschlüsseln

$inc\_password = $this->encryption->decrypt($result\_row\_array['kennwort']); // decrypt the DB password and the users password

## Abweichungen, Anpassungen, Entscheidungen

Das Projekt wurde mithilfe von PHP, JavaScript sowie dem CodeIgniter- und Bootstrap Framework entwickelt.

Für mein lokalen Webserver habe ich mich für XAMPP entschieden. XAMPP ist die Kombination des Apache Webservers, der Datenbank MySQL und der Skriptsprachen Perl und PHP.

Mit XAMPP ist es möglich, in wenigen Schritten einen eigenen Webserver mit all seinen Komponenten auf meinem lokalen Rechner zu installieren.  
  
In XAMPP ist der Ordner htdocs, in dem alle Projekte, die auf „localhost“ sich befinden, gespeichert werden müssen. Deswegen habe ich mein Repository in dem Ordner gemacht, und es befindet sich in demselben Ordner, in dem sich mein Projekt befindet.

Für die Implementierung und die Bearbeitung von mein PHP code habe ich Visual Studio code benutzt. Visual Studio Code ist ein kostenloser Quelltext-Editor von Microsoft zur Erstellung von Cross-Plattform-Applikationen mit JavaScript, HTML und CSS entwickelt.

Ich habe mich für CodeIgniter entschieden, CodeIgniter ist eines der besten PHP-Frameworks, das eine Open-Source-Entwicklungsplattform bietet. Es unterstützt PHP-Entwickler, indem es die einfachste Möglichkeit bietet, modulare Methoden zu verwenden und jede explizite Funktion zu erhalten. Aufgrund seiner benutzerfreundlichen URLs ist es einfach, große dynamische und sichere Webanwendungen zu erstellen.

Für die Datenbank-Aufgaben habe ich mich für phpMyAdmin entschieden. phpMyAdmin ist eine [freie](https://de.wikipedia.org/wiki/Freie_Software) [Webanwendung](https://de.wikipedia.org/wiki/Webanwendung) zur Administration von [MySQL](https://de.wikipedia.org/wiki/MySQL)-[Datenbanken](https://de.wikipedia.org/wiki/Datenbank) und [MariaDB](https://de.wikipedia.org/wiki/MariaDB" \o "MariaDB) und wird in XAMPP mitgeliefert.

In der Datenbank habe ich einen Standardbenutzer mit dem Benutzernamen (admin) und einem Passwort (admin) für die erste Verwendung der Software erstellt und ich werde in der Bedienungsanleitung deutlich machen, dass dieses Passwort nach der ersten Anmeldung ersetzt werden sollte.

Wenn sich der Benutzer als Administrator anmeldet, kann er alle Benutzer sehen, die bearbeitet oder entfernt werden können, und durch Klicken auf Benutzer hinzufügen kann er einen neuen Benutzer hinzufügen.

Nach einer Diskussion mit dem Projektunterstützer haben wir uns entschieden, das Namensfeld zu einer Dropdown-Liste zu machen in der alle angelegten Benutzer wählbar sind. Hier kann also auch ein Benutzer einen anderen als Abwesend melden.

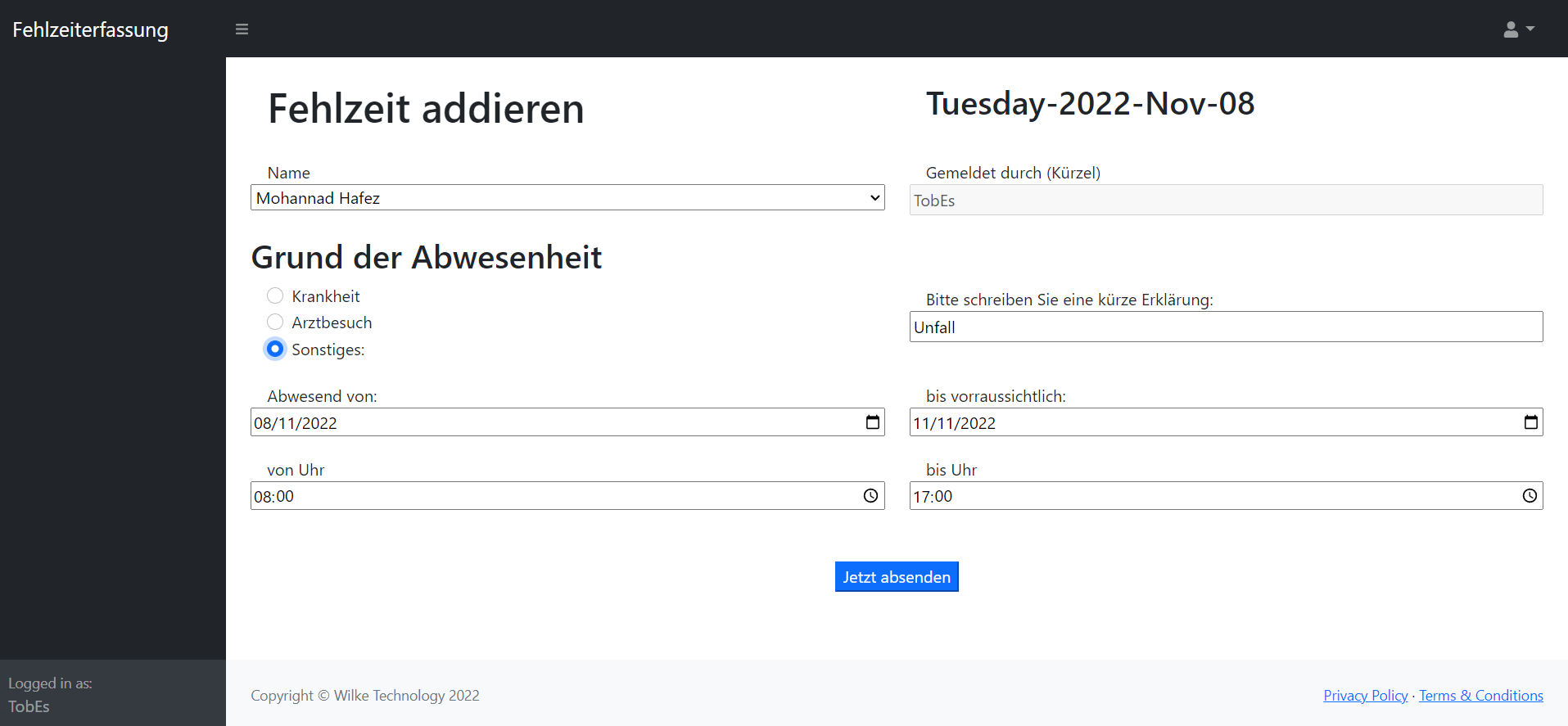
Deshalb müssen wir eine weitere Spalte in der Datenbank zur „tbl\_grund“ Tabelle hinzufügen, um den Benutzer zu speichern, der diesen Eintrag hinzugefügt hat. Der Name dieser Spalte ist autor\_id.

Damit brauchen wir auch den Benutzername des Autors von der DB, weswegen wir einen „Helper“ benutzen dessen Aufgabe darin besteht aus der AutorID einen Autorname zu machen.

Nach einer Diskussion mit dem Projektunterstützer haben wir uns entschieden, dass, wenn einen Benutzer gelöscht wird, auch alle seine Fehlzeiten von der DB auch gelöscht.

# Projektergebnisse

So sieht das neue Formular aus:



## Soll-Ist-Vergleich

Der tatsächliche Aufwand lag insgesamt bei 76 Arbeitsstunden.

Die gestellten Anforderungen wurden erfüllt. Die Software entspricht seinen Erwartungen und macht genau das, was sie soll, und der Kunde ist mit dem Ergebnis zufrieden.

## Qualitätskontrolle

Es viel auf, dass ein eingeloggter Benutzer theoretisch mit einer manuellen Manipulation der URL auf Views Zugriff erhalten kann, die nur für den Administrator gedacht sind.  
Dies wurde bei der Entwicklung nicht bedacht.

Lösung: Es muss bei jedem Admin View Aufruf überprüft werden, ob $user\_role immer noch 1 (admin) ist.

Das kann man mit den Session Daten einfach erreichen.

## Abweichungen, Anpassungen, Entwicklungsmöglichkeiten

Durch die Implementierung einer API wollen wir die DB-Daten als Jason oder XML zu anderer Software weitergeben. Damit kann man viele andere Anwendungen oder Funktionen machen.

Mit Blick auf die Zukunft können wir dieses Projekt in Verbindung mit unserem PSI-Programm entwickeln. PSI ist eine kostenlose Instant-Messaging-Anwendung, die für das XMPP-Netzwerk entwickelt wurde.

In diesem Fall können wir die Ansicht einsehen, dass dieser Mitarbeiter heute krank oder aus irgendeinem Grund nicht im Unternehmen ist.

In einem anderen Fall gibt es in unserem Unternehmen einen Küchendienst, den jeder Mitarbeiter (Auszubildende) erledigen sollte. Wenn der Mitarbeiter, der diesen Dienst leisten sollte, krank ist oder aus irgendeinem Grund nicht im Unternehmen ist, sollte sein Stellvertreter diese Verantwortung übernehmen. Dann kann der Stellvertreter über die Abwesenheit des Verantwortlichen informiert werden, um seine Aufgabe zu übernehmen.

Ein weiterer Gedanke war, ob wir die Möglichkeit zum Hinzufügen weiterer Administratoren benötigen. Hier kommt aber die Frage auf: Kann ein Admin anderen Admin löschen? Ich behaupte: Ja, aber es muss darauf geachtet werden, dass der Standardbenutzer nicht gelöscht werden darf, oder, dass immer mindestens einen Admin in der Datenbank bleibt.  
Das kann man erreichen entweder: Der erste Antrag in DB „Admin“ immer außerhalb die Löschung Querreys, oder es muss immer überprüft werden, ob dieser Admin der letzte Admin ist, bevor gelöscht wird.

# Dokumentation

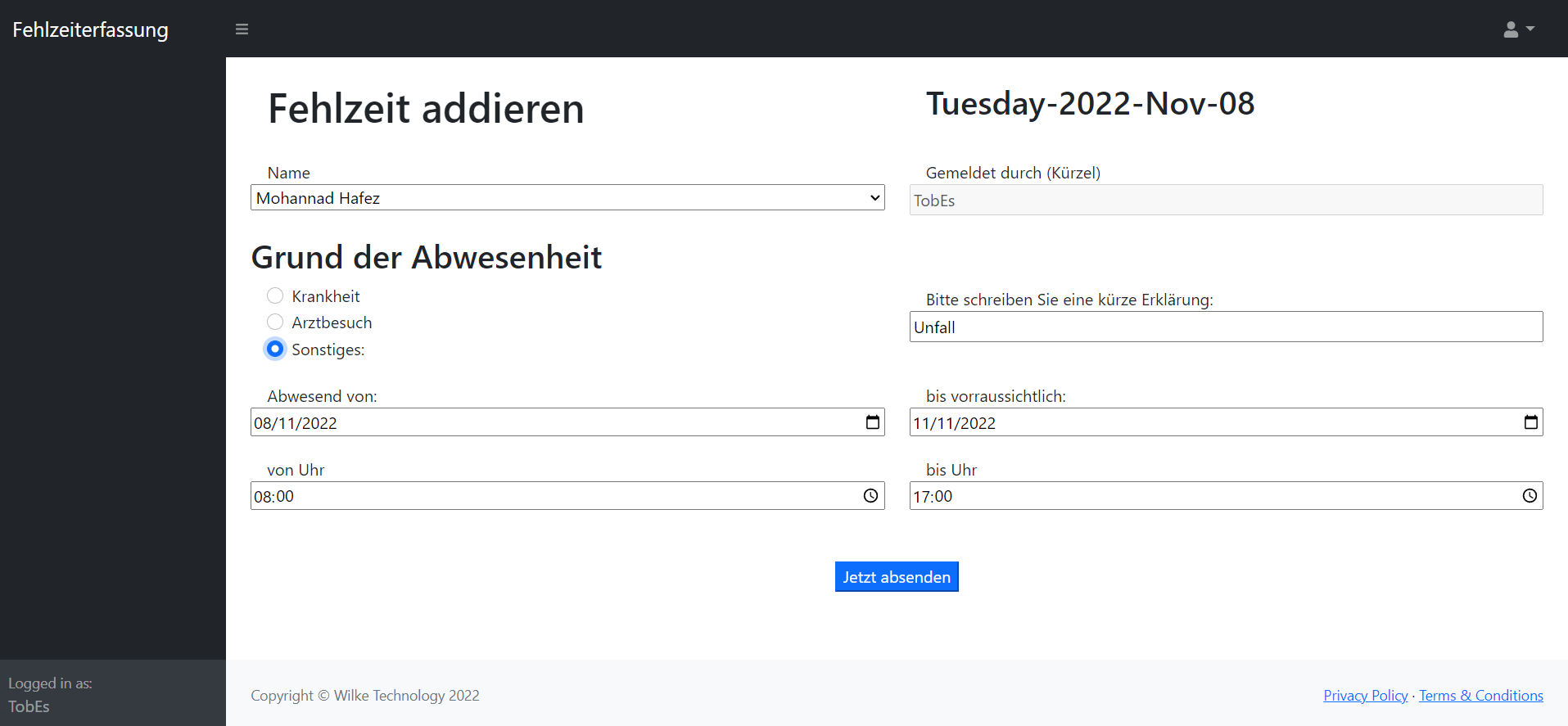
## Benutzerhandbuch

1. Anmeldung Layout



Hier soll der Benutzer sein Benutzername und Passwort eingeben, um anzumelden.  
  
Admin hat der voreingestellte Benutzername (admin) und das voreingestellte Passwort „admin1“ für der erste Anmeldung, auf Sicherheitsgründe, der Admin soll das Passwort als erstes ändern.

1. Benutzer Layout



Alle Felder müssen ausgefüllt, ansonsten schlägt ein Fehler für mit einer roten Rückmeldung, die der Benutzer erklärt, worum es geht.

Nach einer richtigen Eingabe würde der Benutzer mitgeteilt, dass seine Fehlzeiten erfolgreich gespeichert würde durch eine Grüne Rückmeldung „Fehlzeit wurde hinzufügt“.

.

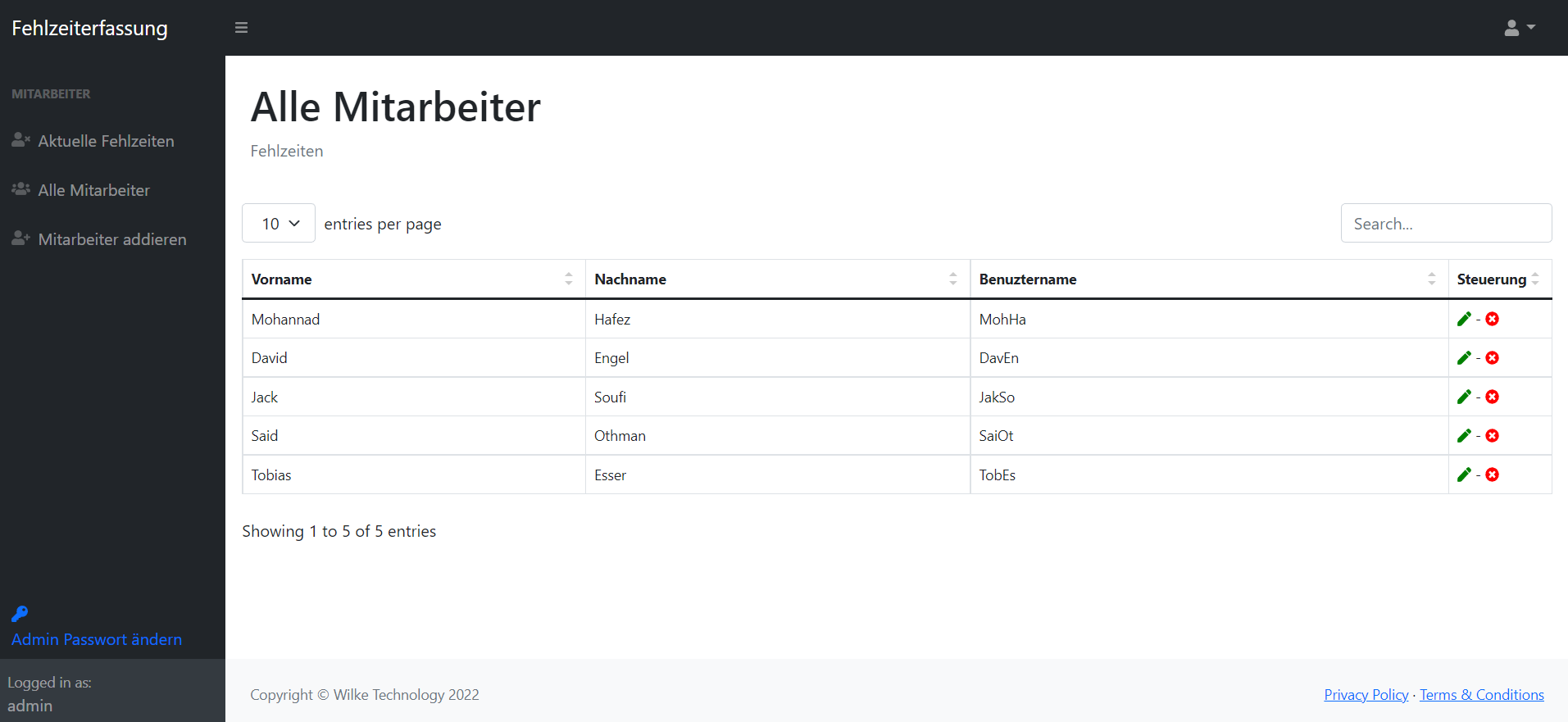
1. Admin Layout

Ein Bild, das Tisch enthält.

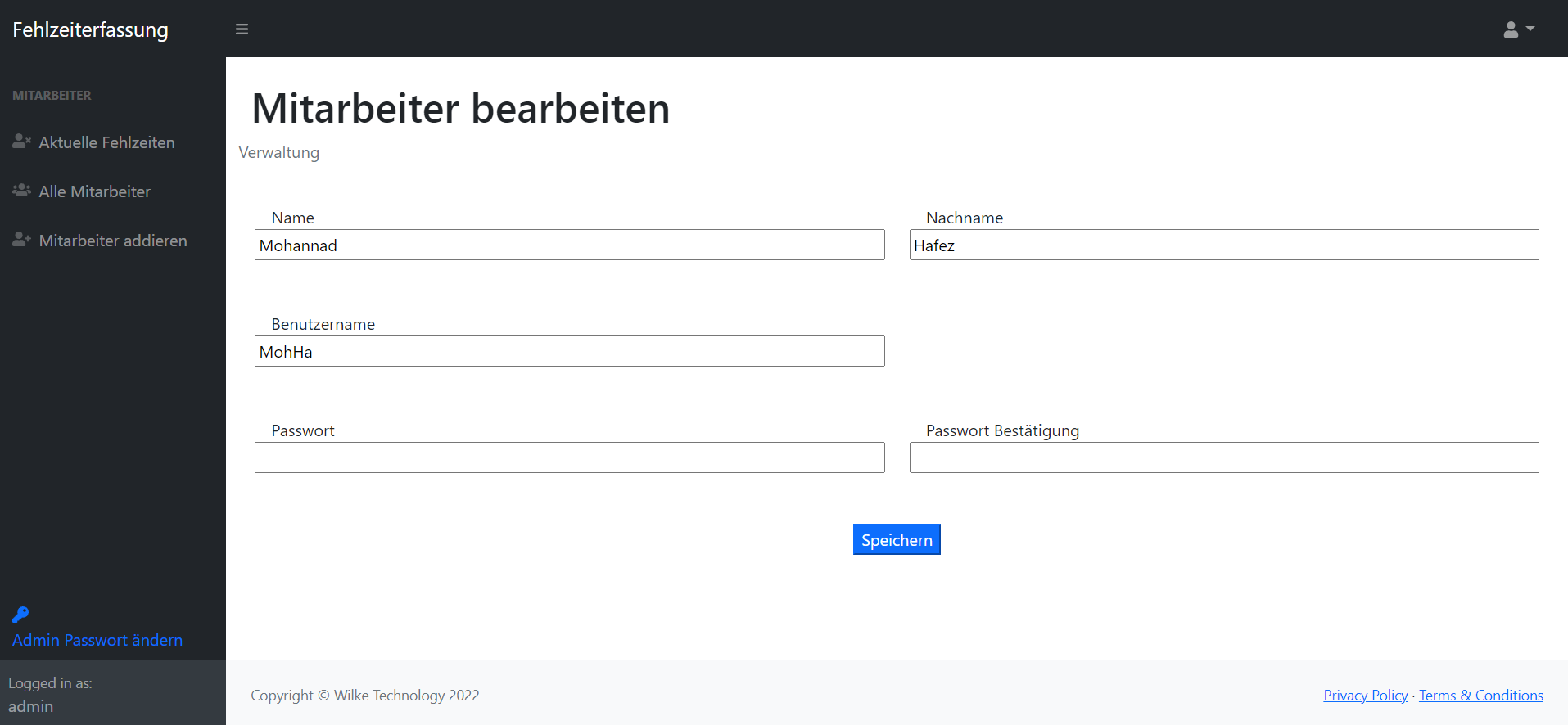
Automatisch generierte Beschreibung

Zuerst wurde der Admin die aktuellen Fehlzeiten sehen, und von dem Seitennavigator kann der Admin verschiedene Optionen aussuchen.

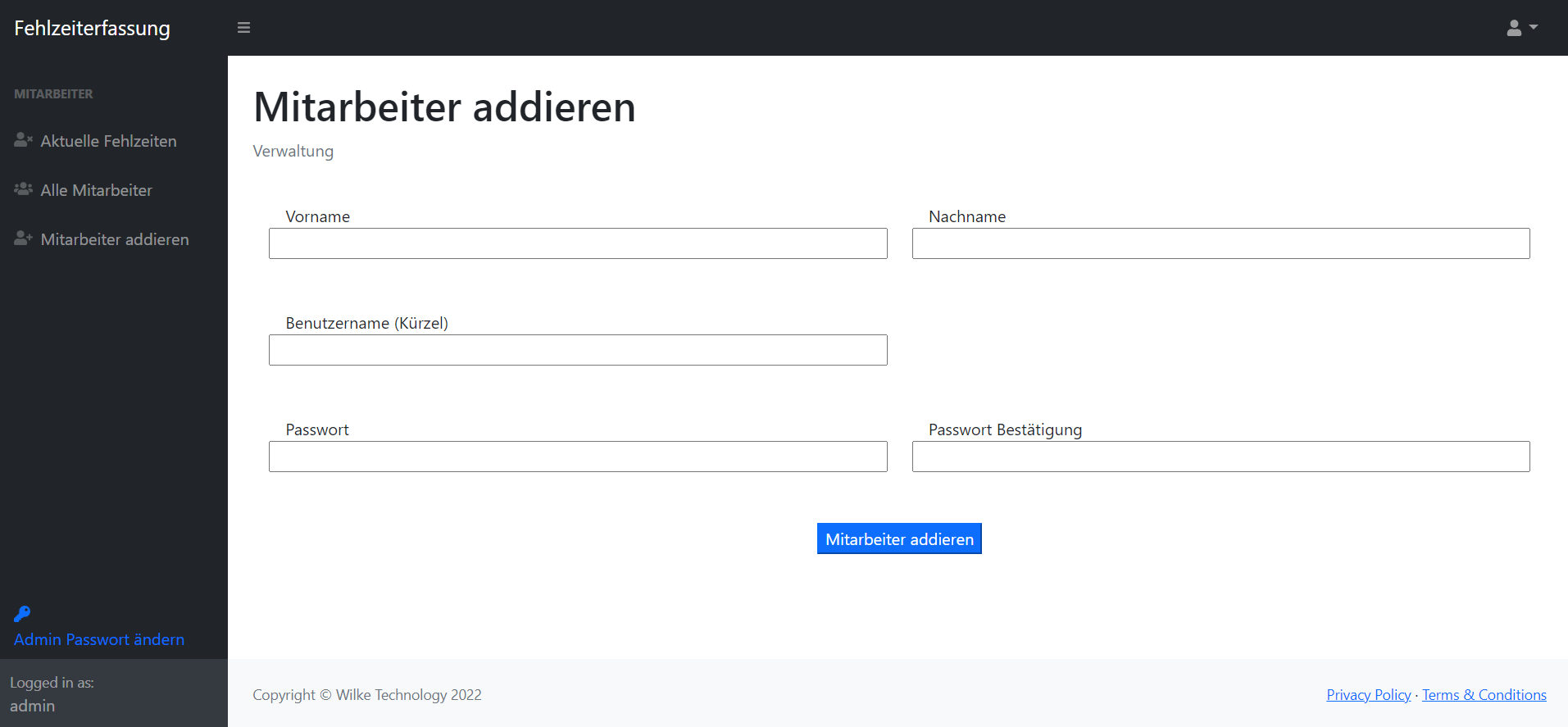
1. Alle Mitarbeiter (Admin)



Alle Mitarbeiter wurden angezeigt mit der Möglichkeit jeder Mitarbeiterdaten zu bearbeiten oder zu löschen.   
Bei Löschen wurde eine Bestätigung Frage angezeigt.  
Bei Bearbeiten wurde neues Layout gezeigt.



1. Mitarbeiter addieren (Admin)



Muss nur hier angepasst werden, weder verdoppelte Benutzernamen akzeptiert werden dürfen noch nicht identische Passwort und Passwort Bestätigung.

# Anhang

## Abkürzungsverzeichnis

|  |  |
| --- | --- |
| MVC | Model-View-Controller |
| CodeIgniter | Ist ein in PHP geschriebenes quelloffenes Webframework |
| Bootstrap | Ein freies Frontend-CSS-Framework |
| Responsive design | Es handelt sich um ein gestalterisches und technisches Paradigma zur Erstellung von Websites |
| CSS | Cascading Style Sheets |
| GPL | General Public License |
| CBC-Modus | Cipher Block Chaining Mode |
| UML | Unified Modeling Language |
| XMPP | Extensible Messaging and Presence Protocol |
| SQL | Structured Query Language |
| GUI | Graphical User Interface |
| PHP | Hypertext Preprocessor |
| HTML | Hypertext Markup Language |
| XML | eXtensible Markup Language |
| JSON | JavaScript Object Notation |
| DB | Databank |

## Quellenverzeichnis

<https://en.wikipedia.org/wiki/Bootstrap_(front-end_framework)>

https://getbootstrap.com/

<https://de.wikipedia.org/wiki/CodeIgniter>

https://codeigniter.com/userguide3/index.html

<https://de.wikipedia.org/wiki/Cipher_Block_Chaining_Mode>

<https://www.apachefriends.org/de/index.html>

https://www.phpmyadmin.net/

## Code-Anhang

1. Controllers (Admin.php):

<?php

defined('BASEPATH') or exit('No direct script access allowed');

class Admin extends CI\_Controller

{

    public function \_\_construct()

    {

        parent::\_\_construct();

        if (!$this->session->userdata('user\_role\_session')) {

            redirect('userlogin');

        }

    }

    public function all\_fehlzeiten()

    {

        $this->load->helper('info\_helper');

        $array['mitarbeiter\_from\_DB'] = $this->dbmodel->get\_all\_fehlzeiten();

        $this->load->view('layout/header');

        $this->load->view('admin/sidenav');

        $this->load->view('admin/fehlzeiten', $array);

        $this->load->view('layout/footer');

    }

    public function all\_mitarbeiter()

    {

        $array['mitarbeiter\_from\_DB'] = $this->dbmodel->get\_all\_users();

        $this->load->view('layout/header');

        $this->load->view('admin/sidenav');

        $this->load->view('admin/all\_mitarbeiter', $array);

        $this->load->view('layout/footer');

    }

    public function add\_mitarbeiter()

    {

        $this->load->view('layout/header');

        $this->load->view('admin/sidenav');

        $this->load->view('admin/add\_mitarbeiter');

        $this->load->view('layout/footer');

    }

    public function con\_add\_mitarbeiter()

    {

        $mit\_name      =   $this->input->post('m\_name');

        $mit\_nachname  =   $this->input->post('nach\_name');

        $mit\_benuzter  =   $this->input->post('b\_name');

        $mit\_password  =   $this->input->post('passwort');

        $key = $this->encryption->encrypt($mit\_password);

        $is\_mit\_added = $this->dbmodel->add\_new\_mitarbeiter($mit\_name, $mit\_nachname, $mit\_benuzter, $key);

        if ($is\_mit\_added == 1) {// Benuztername existiert nicht in der DB

            $this->session->set\_userdata('success', 1);

        } else {

            $this->session->set\_userdata('fail', 1);

        }

        redirect('add\_mitarbeiter\_ref');

    }

    public function update\_mit\_in\_DB()

    {

        $mit\_id              = $this->input->post('mit\_id');

        $mit\_vorname         = $this->input->post('m\_name');

        $mit\_nachname        = $this->input->post('nach\_name');

        $mit\_benuzter\_name   = $this->input->post('b\_name');

        $mit\_password        = $this->input->post('passwort');

        $key = $this->encryption->encrypt($mit\_password); // Passwort verschlüsseln

        $name\_not\_changed = $this->dbmodel->check\_is\_same\_Bname($mit\_id, $mit\_benuzter\_name);

        $is\_Benuzter\_exsitiert = $this->dbmodel->check\_is\_Bname\_existiert($mit\_benuzter\_name);

        if ($name\_not\_changed == 1) { //das selbe Benuzterdaten

            if ($mit\_password == "") {// das Passwortfeld ist leer

                $is\_mit\_updated = $this->dbmodel->update\_mitarbeiter\_same\_Bname($mit\_id, $mit\_vorname, $mit\_nachname);

                if ($is\_mit\_updated == 1) {

                    $this->session->set\_userdata('success\_updated', 1);

                }

            } else { // neues Passwort wurde angegeben, muss mitgespeichert werden

                $is\_mit\_updated = $this->dbmodel->update\_mitarbeiter\_with\_pass\_same\_Bname($mit\_id, $mit\_vorname, $mit\_nachname, $key);

                if ($is\_mit\_updated == 1) {

                    $this->session->set\_userdata('success\_updated', 1);

                }

            }

        } else { // neuer Benutzername wurde angegeben

            if ($is\_Benuzter\_exsitiert == 1) { // existiert der neue Benuztername in der DB

                $this->session->set\_userdata('fail\_name', 1); // Fehler Benuztername existiert

                redirect('update\_mitarbeiter/' . $mit\_id);

            } else { // der neue Benutzername existiert nicht in der DB, muss alle Daten gespeichert werden

                $is\_mit\_updated = $this->dbmodel->update\_mitarbeiter\_with\_pass($mit\_id, $mit\_vorname, $mit\_nachname, $mit\_benuzter\_name, $key);

                if ($is\_mit\_updated == 1) {

                    $this->session->set\_userdata('success\_updated', 1);

                }

            }

        }

        redirect('all\_mitarbeiter\_ref/');

    }

    public function update\_mit($id)

    {

        $array['get\_mit\_info\_from\_DB'] = $this->dbmodel->get\_mit($id);

        $this->load->view('layout/header');

        $this->load->view('admin/sidenav');

        $this->load->view('admin/update\_mit', $array);

        $this->load->view('layout/footer');

    }

    public function delete\_mitarbeiter($id)

    {

        $this->dbmodel->delete\_all\_fehlzeiten($id);

        $is\_mitarbeiter\_deleted = $this->dbmodel->delete\_mitarbeiter($id);

        if ($is\_mitarbeiter\_deleted == 1) {

            $this->session->set\_userdata('deleted', 1);

        }

        redirect('all\_mitarbeiter\_ref');

    }

    public function update\_admin\_pass()

    {

        $this->load->view('layout/header');

        $this->load->view('admin/sidenav');

        $this->load->view('admin/update\_pass');

        $this->load->view('layout/footer');

    }

    public function update\_admin\_pass\_in\_DB()

    {

        $admin\_password        = $this->input->post('passwort');

        $key = $this->encryption->encrypt($admin\_password);

        $is\_admin\_pass\_changed = $this->dbmodel->change\_admin\_pass\_in\_DB($key);

        if ($is\_admin\_pass\_changed == 1) {

            $this->session->set\_userdata('success\_admin\_pass\_updated', 1);

        }

        redirect('all\_mitarbeiter\_ref/');

    }

}

1. Controllers (Login.php)

<?php

defined('BASEPATH') or exit('No direct script access allowed');

class Login extends CI\_Controller

{

    public function register()

    {

        $username =  $this->input->post('username');

        $userpass =  $this->input->post('password');

        $check\_user\_type = $this->dbmodel->checkuser($username, $userpass);

        switch ($check\_user\_type) {

            case 1:                //  1 is the admin role

                $this->load->helper('info\_helper');

                $array['mitarbeiter\_from\_DB'] = $this->dbmodel->get\_all\_fehlzeiten();

                $this->load->view('layout/header');

                $this->load->view('admin/sidenav');

                $this->load->view('admin/fehlzeiten', $array);

                $this->load->view('layout/footer');

                $array['error'] = null;

                break;

            case 2:                // 2 is the client role

                $array['users\_array'] = $this->dbmodel->get\_all\_users();

                $this->load->view('layout/header');

                $this->load->view('client/sideNavClient');

                $this->load->view('client/formular', $array);

                $this->load->view('layout/footer');

                $array['error'] = null;

                break;

            case 0:                // 0 is wrong user name or password

                $array['error'] = 1;

                $this->load->view('login', $array);

                // Fehler msg!!

                break;

        }

    }

    public function logout()

    {

        $this->session->sess\_destroy();

        $array['error'] = null;

        $this->load->view('login', $array);

    }

}

1. Controllers (Mitarbeiter.php)

<?php

defined('BASEPATH') or exit('No direct script access allowed');

class Mitarbeiter extends CI\_Controller

{

    public function \_\_construct()

    {

        parent::\_\_construct();

        if (!$this->session->userdata('user\_role\_session')) {

            redirect('userlogin');

        }

    }

    public function formular\_con()

    {

        $array['users\_array'] = $this->dbmodel->get\_all\_users();

        $this->load->view('layout/header');

        $this->load->view('client/sideNavClient');

        $this->load->view('client/formular', $array);

        $this->load->view('layout/footer');

    }

    public function fun\_add\_fehlzeit()

    {

        $user\_id = $this->session->userdata('user\_id\_session');

        $krank\_id = $this->input->post('name');

        $grund = $this->input->post('radio');

        $notiz = $this->input->post('note');

        $von = $this->input->post('von\_date');

        $bis = $this->input->post('bis\_date');

        $von\_uhr = $this->input->post('von\_uhr');

        $bis\_uhr = $this->input->post('bis\_uhr');

        $reg\_datum = date('Y-m-d H:i');

        if ($von > $bis) {

            $temp = $von;

            $von = $bis;

            $bis = $temp;

        }

        $ist\_fehlzeit\_addiert = $this->dbmodel->add\_fehlzeit($user\_id, $krank\_id, $grund, $notiz, $von, $bis, $von\_uhr, $bis\_uhr, $reg\_datum);

        if ($ist\_fehlzeit\_addiert == 1) {

            $this->session->set\_userdata('time\_added', 1);

        }

        redirect('fehlzeiten\_view');

    }

}

1. Controllers (Welcome.php)

<?php

defined('BASEPATH') or exit('No direct script access allowed');

class Welcome extends CI\_Controller

{

    public function index()

    {

        $array['error'] = null;

        $this->load->view('login', $array);

    }

}

1. Models (dbmodell.php)

<?php

class dbmodel extends CI\_Model

{

    function checkuser($username, $password)

    {

        $this->db->where('b\_name', $username);

        $this->db->from('tbl\_users');

        $query = $this->db->get();

        $result\_row\_array = $query->row\_array();

        if (!empty($result\_row\_array)) {

            $inc\_password = $this->encryption->decrypt($result\_row\_array['kennwort']); // decrypt the DB password and the users password

            if ($inc\_password == $password) {

                $user\_role = $result\_row\_array['role'];

                $user\_id = $result\_row\_array['id'];

                $user\_name = $result\_row\_array['b\_name'];

                $vor\_name = $result\_row\_array['vorname'];

                $nach\_name = $result\_row\_array['nachname'];

                $this->session->set\_userdata('user\_role\_session', $user\_role);

                $this->session->set\_userdata('user\_id\_session', $user\_id);

                $this->session->set\_userdata('user\_name\_session', $user\_name);

                $this->session->set\_userdata('vorname\_session', $vor\_name);

                $this->session->set\_userdata('nachname\_session', $nach\_name);

                return $user\_role;

            }

        } else

            return 0;

    }

    function get\_all\_users()

    {

        $this->db->select('\*');

        $this->db->from('tbl\_users');

        $this->db->where('id !=', 1);

        $query = $this->db->get();

        $result\_array = $query->result\_array();

        return $result\_array;

    }

    function get\_all\_fehlzeiten()

    {

        $this->db->select('\*');

        $this->db->from('tbl\_grund');

        $this->db->join('tbl\_users', 'tbl\_users.id = tbl\_grund.u\_id');

        $this->db->group\_by('gr\_id');

        $query = $this->db->get();

        return $query->result\_array();

    }

    function get\_autor\_name\_from\_DB($id)

    {

        $this->db->select('\*');

        $this->db->from('tbl\_users');

        $this->db->where('id', $id);

        $query = $this->db->get();

        $row\_array = $query->row\_array();

        return $row\_array;

    }

    function add\_fehlzeit($user\_id, $krank\_id, $grund, $notiz, $von, $bis, $von\_uhr, $bis\_uhr, $reg\_datum)

    {

        $von\_date = $von . " " . $von\_uhr;

        $bis\_date = $bis . " " . $bis\_uhr;

        $data = array(

            'autor\_id' => $user\_id, 'u\_id' => $krank\_id, 'grund' => $grund,

            'note' => $notiz, 'von\_datum' => $von\_date, 'bis\_datum' => $bis\_date, 'reg\_datum' => $reg\_datum

        );

        $this->db->insert('tbl\_grund', $data);

        return 1;

    }

    function add\_new\_mitarbeiter($mit\_name, $mit\_nachname, $mit\_benuzter, $mit\_password)

    {

        $this->db->select('\*');

        $this->db->where('b\_name', $mit\_benuzter);

        $this->db->from('tbl\_users');

        $query = $this->db->get();

        $result\_array = $query->result\_array();

        if (empty($result\_array)) { // check if the user name already exist in the DB

            $data = array(

                'vorname' => $mit\_name, 'nachname' => $mit\_nachname, 'b\_name' => $mit\_benuzter,

                'kennwort' => $mit\_password, 'role' => 2

            );

            $this->db->insert('tbl\_users', $data);

            return 1;

        } else {

            return 0; // Benutzername existiert schon in DB

        }

    }

    function get\_mit($id) // getting the user info to show it in the update view

    {

        $this->db->from('tbl\_users');

        $this->db->where('id', $id);

        $query = $this->db->get();

        return $query->result\_array();

    }

    function check\_is\_Bname\_existiert($mit\_benuzter)

    {

        $this->db->select('\*');

        $this->db->where('b\_name', $mit\_benuzter);

        $this->db->from('tbl\_users');

        $query = $this->db->get();

        $result\_array = $query->result\_array();

        if (!empty($result\_array)) {

            return 1; // Benutzername  existier

        }

        else

        {

            return 0; // Benutzername  existiert nicht

        }

    }

    function check\_is\_same\_Bname($mit\_id, $mit\_benuzter)

    {

        $this->db->select('\*');

        $this->db->where('id', $mit\_id);

        $this->db->where('b\_name', $mit\_benuzter);

        $this->db->from('tbl\_users');

        $query = $this->db->get();

        $result\_array = $query->result\_array();

        if (!empty($result\_array)) {

            return 1; // Selbe Benutzer

        }

        else{

            return 0; // nicht der Selbe Benutzer

        }

    }

    function update\_mitarbeiter\_same\_Bname($mit\_id, $mit\_vorname, $mit\_nachname)

    {

        $data = array(

            'vorname' => $mit\_vorname, 'nachname' => $mit\_nachname

        );

        $this->db->where('id', $mit\_id);

        $this->db->update('tbl\_users', $data);

        return 1;

    }

    function update\_mitarbeiter\_with\_pass\_same\_Bname($mit\_id, $mit\_vorname, $mit\_nachname, $mit\_password)

    {

        $data = array(

            'vorname' => $mit\_vorname, 'nachname' => $mit\_nachname, 'kennwort' => $mit\_password

        );

        $this->db->where('id', $mit\_id);

        $this->db->update('tbl\_users', $data);

        return 1;

    }

    function update\_mitarbeiter\_with\_pass($mit\_id, $mit\_vorname, $mit\_nachname, $mit\_benuzter\_name, $mit\_password)

    {

        $data = array(

            'vorname' => $mit\_vorname, 'nachname' => $mit\_nachname, 'kennwort' => $mit\_password, 'b\_name' => $mit\_benuzter\_name

        );

        $this->db->where('id', $mit\_id);

        $this->db->update('tbl\_users', $data);

        return 1;

    }

    function delete\_mitarbeiter($id)

    {

        $this->db->where('id', $id);

        $this->db->delete('tbl\_users');

        return 1;

    }

    function delete\_all\_fehlzeiten($id)

    {

        $this->db->where('u\_id', $id);

        $this->db->delete('tbl\_grund');

    }

    function change\_admin\_pass\_in\_DB($admin\_password)

    {

        $this->db->set('kennwort', $admin\_password);

        $this->db->where('id', 1);

        $this->db->update('tbl\_users');

        return 1;

    }

}

1. Helpers (info\_helper.php)

<?php

function get\_autor\_name($id)

{

    $cv = &get\_instance();

    $row\_array= $cv->dbmodel->get\_autor\_name\_from\_DB($id);

    $name = $row\_array['b\_name'];

    return $name;

}

1. Views -> Admin(addMitarbeiter.php)

<style>

    #error\_mitarbeiter {

        display: none;

    }

    #success\_added {

        display: none;

    }

    #fail\_added {

        display: none;

    }

</style>

<?php

$mit\_added\_success = $this->session->userdata('success'); //controller hat die Session gesetted anhand die DBmodell Rückgabe

if (isset($mit\_added\_success)) {

?>

    <style>

        #success\_added {

            display: block;

        }

    </style>

<?php

}

$this->session->unset\_userdata('success'); //controller hat die Session gesetted anhand die DBmodell Rückgabe

$mit\_not\_added = $this->session->userdata('fail');

if (isset($mit\_not\_added)) {

?>

    <style>

        #fail\_added {

            display: block;

        }

    </style>

<?php

}

$this->session->unset\_userdata('fail');

?>

<div id="layoutSidenav\_content">

    <main>

        <div class="container-fluid px-4">

            <h1 class="mt-4">Mitarbeiter addieren</h1>

            <ol class="breadcrumb mb-4">

                <li class="breadcrumb-item active">Verwaltung</li>

            </ol>

            <div class="card-body">

                <form class="form-card" id="add\_mitarbeiter\_form" action="act\_add\_mitarbeiter" method="post">

                    <div class="row justify-content-between text-left">

                        <div class="form-group col-sm-6 flex-column d-flex"> <label class="form-control-label px-3">

                                Vorname</label> <input type="text" id="m\_name" name="m\_name"> </div>

                        <div class="form-group col-sm-6 flex-column d-flex"> <label class="form-control-label px-3">

                                Nachname</label> <input type="text" id="nach\_name" name="nach\_name"> </div>

                    </div>

                    <div class="row justify-content-between text-left" style="margin-top: 50px;">

                        <div class="form-group col-sm-6 flex-column d-flex"> <label class="form-control-label px-3" autocomplete="off">

                                Benutzername (Kürzel)</label> <input type="text" id="b\_name" name="b\_name"> </div>

                    </div>

                    <div class="row justify-content-between text-left" style="margin-top: 50px;">

                        <div class="form-group col-sm-6 flex-column d-flex"> <label class="form-control-label px-3">

                                Passwort</label> <input type="password" id="passwort" name="passwort" autocomplete="new-password"> </div>

                        <div class="form-group col-sm-6 flex-column d-flex"> <label class="form-control-label px-3">

                                Passwort Bestätigung</label> <input type="password" id="w\_passwort" name="w\_passwort"> </div>

                    </div>

                    <div class="d-flex justify-content-center" style="margin-top: 50px;">

                        <div class="form-group "> <button type="button" onclick="check\_mitarbeiter\_data()" class="btn-block btn-primary">

                                Mitarbeiter addieren</button> </div>

                    </div>

                </form>

                <br><br>

                <div id="error\_mitarbeiter" class="alert alert-danger" role="alert">

                    <p id="error\_msg"></p>

                </div>

                <div id="success\_added" class="alert alert-success" role="alert">

                    <p>Mitarbeiter wurde addiert</p>

                </div>

                <div id="fail\_added" class="alert alert-danger" role="alert">

                    <p>Benuzername existiert schon!</p>

                </div>

            </div>

    </main>

    <script>

        function show\_error(error) { // error Funktion

            $('#error\_mitarbeiter').show();

            $('#error\_msg').html(error);

            $('#success\_added').hide();

            $('#fail\_added').hide();

        }

        function check\_mitarbeiter\_data() {

            var mit\_name = document.getElementsByName('m\_name')[0].value;

            var mit\_nachname = document.getElementsByName('nach\_name')[0].value;

            var mit\_benutzer = document.getElementsByName('b\_name')[0].value;

            var mit\_passwort = document.getElementsByName('passwort')[0].value;

            var mit\_passwort\_w = document.getElementsByName('w\_passwort')[0].value;

            if (mit\_name.length == 0) { // ist ein Vorname geschrieben?

                show\_error('Vorname ist ungültig!!');

            } else if (mit\_nachname.length == 0) { // ist ein Nachname geschrieben?

                show\_error('Nachname ist ungültig!!');

            } else if (mit\_benutzer.length == 0) { // ist ein Benutzername geschrieben?

                show\_error('Benutzername ist ungültig!!');

            } else if (mit\_passwort.length <= 5) { // ist das Passwort größer als 5 Zeichnen?

                show\_error('Passwort ist zu kürz!! Das Passwort sollte mindestens 5 Zeichnen sein ');

            } else if (mit\_passwort != mit\_passwort\_w) { // ist das Passwort mit seiner Wederholung idäntisch?

                show\_error('Die eingegebenen Passwörter stimmen nicht überein');

            } else {

                $('#add\_mitarbeiter\_form').submit(); // Datein an den Kontroller schicken POST

            }

        }

    </script>

1. Views -> Admin(allMitarbeiter.php)

<style>

    #success\_delete {

        display: none;

    }

    #success\_edit {

        display: none;

    }

    #pass\_edit {

        display: none;

    }

</style>

<?php

$deleted = $this->session->userdata('deleted');

if (isset($deleted)) {

?>

    <style>

        #success\_delete {

            display: block;

        }

    </style>

<?php

}

$this->session->unset\_userdata('deleted');

$updated = $this->session->userdata('success\_updated');

if (isset($updated)) {

?>

    <style>

        #success\_edit {

            display: block;

        }

    </style>

<?php

}

$this->session->unset\_userdata('success\_updated');

$admin\_pass\_updated = $this->session->userdata('success\_admin\_pass\_updated');

if (isset($admin\_pass\_updated)) {

?>

    <style>

        #pass\_edit {

            display: block;

        }

    </style>

<?php

}

$this->session->unset\_userdata('failed');

?>

<div id="layoutSidenav\_content">

    <main>

        <div class="container-fluid px-4">

            <h1 class="mt-4">Alle Mitarbeiter</h1>

            <ol class="breadcrumb mb-4">

                <li class="breadcrumb-item active">Fehlzeiten</li>

            </ol>

        </div>

        <div class="card-body">

            <table id="datatablesSimple">

                <thead>

                    <tr>

                        <th>Vorname</th>

                        <th>Nachname</th>

                        <th>Benuztername</th>

                        <th>Steuerung</th>

                    </tr>

                </thead>

                <tfoot>

                    <tr>

                        <th>Vorname</th>

                        <th>Nachname</th>

                        <th>Benuztername</th>

                        <th>Steuerung</th>

                    </tr>

                </tfoot>

                <tbody>

                    <?php

                    foreach ($mitarbeiter\_from\_DB as $mit) {

                    ?>

                        <tr>

                            <td><?php echo $mit['vorname']    ?> </td>

                            <td><?php echo $mit['nachname']    ?> </td>

                            <td><?php echo $mit['b\_name'] ?> </td>

                            <td style="width: 60px">

                                <a href="<?php echo base\_url() ?>update\_mitarbeiter/<?php echo $mit['id'] ?>">

                                    <i class="fa fa-pen" style="color:green"></i></a>

                                -

                                <a href="#" onclick="ConfirmDelete(<?php echo $mit['id'] ?>)">

                                    <i class="fa fa-times-circle" style="color:red"></i> </a>

                            </td>

                        </tr>

                    <?php

                    }

                    ?>

                </tbody>

            </table>

            <br><br>

            <div id="success\_delete" class="alert alert-success" role="alert">

                <p>Mitarbeiter wurde gelöscht</p>

            </div>

            <div id="success\_edit" class="alert alert-success" role="alert">

                <p>Änderung wurden gespeichert</p>

            </div>

            <div id="pass\_edit" class="alert alert-success" role="alert">

                <p>Admin Passwort wurde erfolgreich gändert</p>

            </div>

        </div>

    </main>

    <script type="text/javascript">

        function ConfirmDelete($mit\_id) {

            if (confirm("Delete Mitarbeiter? Alle Fehlzeiten des Mitarbeiters werden auch gelöscht"))

                location.href = "deleteMitarbeiter/" + $mit\_id;

        }

    </script>

1. Views -> Admin(fehlzeiten.php)

<style>

    #last\_admin {

        display: none;

    }

    #success\_delete {

        display: none;

    }

</style>

<?php

$last\_admin = $this->session->userdata('last\_admin');

if (isset($last\_admin)) {

?>

    <style>

        #last\_admin {

            display: block;

        }

    </style>

<?php

}

$this->session->unset\_userdata('last\_admin');

$mit\_deleted = $this->session->userdata('mit\_deleted');

if (isset($mit\_deleted)) {

?>

    <style>

        #success\_delete {

            display: block;

        }

    </style>

<?php

}

$this->session->unset\_userdata('mit\_deleted');

?>

<div id="layoutSidenav\_content">

    <main>

        <div class="container-fluid px-4">

            <h1 class="mt-4">Aktuelle Fehlzeiten</h1>

        </div>

        <div class="card-body">

            <table id="datatablesSimple">

                <thead>

                    <tr>

                        <th>Name</th>

                        <th>gemeldet durch</th>

                        <th>Grund der Abwesendheit</th>

                        <th>Notiz</th>

                        <th>Von Datum</th>

                        <th>Bis Datum</th>

                        <th>gemeldet am</th>

                    </tr>

                </thead>

                <tfoot>

                    <tr>

                        <th>Name</th>

                        <th>gemeldet durch</th>

                        <th>Grund der Abwesendheit</th>

                        <th>Notiz</th>

                        <th>Von Datum</th>

                        <th>Bis Datum</th>

                        <th>gemeldet am</th>

                </tfoot>

                <tbody>

                    <?php

                    foreach ($mitarbeiter\_from\_DB as $mit) {

                    ?>

                        <tr>

                            <td><?php echo $mit['vorname'] . " " . $mit['nachname'] ?> </td>

                            <td><?php echo get\_autor\_name($mit['autor\_id']) ?> </td>

                            <td><?php echo GRUND[$mit['grund']] ?> </td>

                            <td><?php echo $mit['note'] ?> </td>

                            <td><?php echo date("d M Y", strtotime($mit['von\_datum']))  ?> </td>

                            <td><?php echo date("d M Y", strtotime($mit['bis\_datum']))  ?> </td>

                            <td><?php echo $mit['reg\_datum'] ?> </td>

                        </tr>

                    <?php

                    }

                    ?>

                </tbody>

            </table>

            <br><br>

            <div id="last\_admin" class="alert alert-danger" role="alert">

                <p>Letzter Admin darf nicht gelöscht werden!</p>

            </div>

            <div id="success\_delete" class="alert alert-success" role="alert">

                <p>Account wurde gelöscht</p>

            </div>

        </div>

    </main>

    <script type="text/javascript">

        function ConfirmDelete(id) {

            console.log(id);

            if (confirm("Delete Account?"))

                location.href = "<?php echo base\_url() ?>ref\_delete\_mit/" + id;

        }

    </script>

1. Views -> Admin(sideNav.php)

<body class="sb-nav-fixed">

    <nav class="sb-topnav navbar navbar-expand navbar-dark bg-dark">

        <!-- Navbar Brand-->

        <a class="navbar-brand ps-3" href="<?php echo base\_url() ?>admin\_aktuel\_fehlzeiten">Fehlzeiterfassung</a>

        <!-- Sidebar Toggle-->

        <button class="btn btn-link btn-sm order-1 order-lg-0 me-4 me-lg-0" id="sidebarToggle" href="#!"><i class="fas fa-bars"></i></button>

        <!-- Navbar Search-->

        <form class="d-none d-md-inline-block form-inline ms-auto me-0 me-md-3 my-2 my-md-0">

        </form>

        <!-- Navbar-->

        <ul class="navbar-nav ms-auto ms-md-0 me-3 me-lg-4">

            <li class="nav-item dropdown">

                <a class="nav-link dropdown-toggle" id="navbarDropdown" href="#" role="button" data-bs-toggle="dropdown" aria-expanded="false"><i class="fas fa-user fa-fw"></i></a>

                <ul class="dropdown-menu dropdown-menu-end" aria-labelledby="navbarDropdown">

                    <li><a class="dropdown-item" href="#!"> <?php

                                                            echo $this->session->userdata('user\_name\_session');

                                                            ?></a></li>

                    <li>

                        <hr class="dropdown-divider" />

                    </li>

                    <li><a class="dropdown-item" href="<?php echo base\_url() ?>ref\_logout">Logout</a></li>

                </ul>

            </li>

        </ul>

    </nav>

    <div id="layoutSidenav">

        <div id="layoutSidenav\_nav">

            <nav class="sb-sidenav accordion sb-sidenav-dark" id="sidenavAccordion">

                <div class="sb-sidenav-menu">

                    <div class="nav">

                        <div class="sb-sidenav-menu-heading"> Mitarbeiter </div>

                        <a class="nav-link" href="<?php echo base\_url() ?>admin\_aktuel\_fehlzeiten">

                            <div class="sb-nav-link-icon"><i class="fa fa-user-times"></i></div>

                            Aktuelle Fehlzeiten

                        </a>

                        <a class="nav-link" href="<?php echo base\_url() ?>all\_mitarbeiter\_ref">

                            <div class="sb-nav-link-icon"><i class="fa fa-users"></i></div>

                            Alle Mitarbeiter

                        </a>

                        <a class="nav-link" href="<?php echo base\_url() ?>add\_mitarbeiter\_ref">

                            <div class="sb-nav-link-icon"><i class="fas fa-user-plus"></i></div>

                            Mitarbeiter addieren

                        </a>

                    </div>

                </div>

                <a class="nav-link" href="<?php echo base\_url() ?>ref\_update\_admin\_pass">

                    <div class="sb-nav-link-icon"><i class="fa fa-key" aria-hidden="true"></i></div>

                    Admin Passwort ändern

                </a>

                <div class="sb-sidenav-footer">

                    <div class="small">Logged in as: <?= $this->session->userdata('user\_vorname\_session') . " " . $this->session->userdata('user\_nachname\_session'); ?>

                    </div>

                    <?php

                    echo $this->session->userdata('user\_name\_session');

                    ?>

                </div>

            </nav>

        </div>

1. Views -> Admin(update\_mit.php)

<style>

    #error\_mitarbeiter {

        display: none;

    }

    #error\_password {

        display: none;

    }

    #failed\_edited {

        display: none;

    }

</style>

<?php

$same\_name = $this->session->userdata('fail\_name');

if (isset($same\_name)) {

?>

    <style>

        #failed\_edited {

            display: block;

        }

    </style>

<?php

}

$this->session->unset\_userdata('fail\_name');

?>

<div id="layoutSidenav\_content">

    <main>

        <div class="container-fluid px-4">

            <div class="row">

                <h1 class="mt-4">Mitarbeiter bearbeiten</h1>

                <ol class="breadcrumb mb-4">

                    <li class="breadcrumb-item active">Verwaltung</li>

                </ol>

                <div class="card-body">

                    <form class="form-card" id="update\_mitarbeiter\_form" action="<?php echo base\_url() ?>act\_update\_mitarbeiter" method="post">

                        <div class="row justify-content-between text-left">

                            <div class="form-group col-sm-6 flex-column d-flex"> <label class="form-control-label px-3">

                                    Name</label> <input type="text" id="m\_name" name="m\_name" value="<?php echo $get\_mit\_info\_from\_DB[0]['vorname'] ?>">

                            </div>

                            <div class="form-group col-sm-6 flex-column d-flex"> <label class="form-control-label px-3">

                                    Nachname</label> <input type="text" id="nach\_name" name="nach\_name" value="<?php echo $get\_mit\_info\_from\_DB[0]['nachname'] ?>"></div>

                        </div>

                        <div class="row justify-content-between text-left" style="margin-top: 50px;">

                            <div class="form-group col-sm-6 flex-column d-flex"> <label class="form-control-label px-3" >

                                    Benutzername</label> <input type="text" id="b\_name" name="b\_name" value="<?php echo $get\_mit\_info\_from\_DB[0]['b\_name'] ?>"></div>

                        </div>

                        <div class="row justify-content-between text-left" style="margin-top: 50px;">

                            <div class="form-group col-sm-6 flex-column d-flex"> <label class="form-control-label px-3">

                                    Passwort</label> <input type="password" id="passwort" name="passwort" autocomplete="new-password"></div>

                            <div class="form-group col-sm-6 flex-column d-flex"> <label class="form-control-label px-3">

                                    Passwort Bestätigung</label> <input type="password" id="W\_passwort" name="w\_passwort"></div>

                        </div>

                        <input type="hidden" name="mit\_id" value="<?php echo $get\_mit\_info\_from\_DB[0]['id'] ?>">

                        <div class="d-flex justify-content-center" style="margin-top: 50px;">

                            <div class="form-group "> <button type="button" onclick="check\_mitarbeiter\_data()" class="btn-block btn-primary">

                                    Speichern</button> </div>

                        </div>

                    </form>

                    <br>

                    <div id="error\_mitarbeiter" class="alert alert-danger" role="alert">

                        <p id="error\_msg"></p>

                    </div>

                    <div id="failed\_edited" class="alert alert-danger" role="alert">

                        <p>Benuztername exsitiert schon!</p>

                    </div>

                </div>

            </div>

    </main>

    <script>

        function show\_error(error) {

            $('#error\_mitarbeiter').show();

            $('#error\_msg').html(error);

        }

        function check\_mitarbeiter\_data() {

            var mit\_name = document.getElementsByName('m\_name')[0].value;

            var mit\_nachname = document.getElementsByName('nach\_name')[0].value;

            var mit\_benutzer = document.getElementsByName('b\_name')[0].value;

            var mit\_passwort = document.getElementsByName('passwort')[0].value;

            var mit\_passwort\_w = document.getElementsByName('w\_passwort')[0].value;

            if (mit\_name.length == 0) {

                show\_error('Mitarbeitername ist ungültig!!');

            } else if (mit\_nachname.length == 0) {

                show\_error('Mitarbeiter Nachname ist ungültig!!');

            } else if (mit\_benutzer.length == 0) {

                show\_error('Benutzername ist ungültig!!');

            } else if (mit\_passwort != mit\_passwort\_w) {

                show\_error('Die eingegebenen Passwörter stimmen nicht überein');

                $('#error\_password').hide();

            } else {

                $('#update\_mitarbeiter\_form').submit();

            }

        }

    </script>

1. Views -> Admin(update\_pass.php)

<style>

    #error\_data {

        display: none;

    }

    #error\_password {

        display: none;

    }

</style>

<?php

$same\_name = $this->session->userdata('fail\_name');

if (isset($same\_name)) {

?>

<?php

}

$this->session->unset\_userdata('fail\_name');

?>

<div id="layoutSidenav\_content">

    <main>

        <div class="container-fluid px-4">

            <h1 class="mt-4">Admin Passwort ändern</h1>

            <ol class="breadcrumb mb-4">

                <li class="breadcrumb-item active">Verwaltung</li>

            </ol>

            <div class="card-body">

                <form class="form-card" id="update\_pass\_form" action="<?php echo base\_url() ?>act\_update\_admin\_pass" method="post">

                    <div class="row justify-content-between text-left" style="margin-top: 50px;">

                        <div class="form-group col-sm-6 flex-column d-flex"> <label class="form-control-label px-3">

                                Neues Passwort</label> <input type="password" id="passwort" name="passwort"></div>

                        <div class="form-group col-sm-6 flex-column d-flex"> <label class="form-control-label px-3">

                                Passwort Bestätigung</label> <input type="password" id="W\_passwort" name="w\_passwort"></div>

                    </div>

                    <div class="d-flex justify-content-center" style="margin-top: 50px;">

                        <div class="form-group "> <button type="button" onclick="check\_neu\_pass\_data()" class="btn-block btn-primary">

                                Speichern</button> </div>

                    </div>

                </form>

                <br>

                <div id="error\_data" class="alert alert-danger" role="alert">

                    <p id="error\_msg"></p>

                </div>

            </div>

    </main>

    <script>

        function show\_error(error) {

            $('#error\_data').show();

            $('#error\_msg').html(error);

        }

        function check\_neu\_pass\_data() {

            var mit\_passwort = document.getElementsByName('passwort')[0].value;

            var mit\_passwort\_w = document.getElementsByName('w\_passwort')[0].value;

            if (mit\_passwort.length <= 5) {

                show\_error('Passwort ist zu kürz!! Das Passwort sollte mindestens 5 Zeichnen sein ');

            } else if (mit\_passwort\_w.length <= 5) {

                show\_error('Passwort Bestätigung ist zu kürz!! Das Passwort sollte mindestens 5 Zeichnen sein');

            } else if (mit\_passwort != mit\_passwort\_w) {

                show\_error('Die eingegebenen Passwörter stimmen nicht überein');

                $('#error\_password').hide();

            } else {

                $('#update\_pass\_form').submit();

            }

        }

    </script>

1. Views -> Client(formular.php)

<style>

    #success\_added {

        display: none;

    }

    #error\_formular {

        display: none;

    }

</style>

<?php

$deleted = $this->session->userdata('time\_added');

if (isset($deleted)) {

?>

    <style>

        #success\_added {

            display: block;

        }

    </style>

<?php

}

$this->session->unset\_userdata('time\_added');

?>

<div id="layoutSidenav\_content">

    <main>

        <div class="container-fluid px-4">

            <div class="row">

                <div class="form-group col-6 flex-column d-flex">

                    <label class="form-control-label px-3">

                        <h1 class="mt-4">Fehlzeit addieren </h1>

                </div>

                <div class="form-group col-6 flex-column d-flex">

                    <label class="form-control-label px-3">

                        <h2 class="mt-4"><?= date("l-Y-M-d") ?></h2>

                </div>

            </div>

            <div class="row" style="margin-top: 20px;">

                <div class="col-12">

                    <form id="add\_fehlzeit\_form" method="POST" action="act\_add\_fehlzeit">

                        <div class="row">

                            <div class="form-group col-6 flex-column d-flex">

                                <label class="form-control-label px-3">

                                    Name</label>

                                <select name="name" id="name\_id">

                                    <?php

                                    foreach ($users\_array as $value) {

                                    ?>

                                        <option value="<?php echo $value['id'] ?>">

                                            <?php

                                            echo $value['vorname'] . " " . $value['nachname']

                                            ?>

                                        </option>

                                    <?php

                                    }

                                    ?>

                                </select>

                            </div>

                            <div class="form-group col-6 flex-column d-flex">

                                <label class="form-control-label px-3">

                                    Gemeldet durch (Kürzel)</label>

                                <input type="text" value="<?= $this->session->userdata('user\_name\_session') ?>" disabled>

                            </div>

                        </div>

                        <div class="row" style="margin-top: 20px;">

                            <div class="col-6">

                                <h2>Grund der Abwesenheit</h2>

                                <div class="form-check radiodiv">

                                    <input class="form-check-input" type="radio" name="radio" id="krank" value="1">

                                    <label class="form-check-label" for="krank">

                                        Krankheit

                                    </label>

                                </div>

                                <div class="form-check radiodiv">

                                    <input class="form-check-input" type="radio" name="radio" id="artz" value="2">

                                    <label class="form-check-label" for="artz">

                                        Arztbesuch

                                    </label>

                                </div>

                                <div class="form-check radiodiv">

                                    <input class="form-check-input" type="radio" name="radio" id="sonst" value="3">

                                    <label class="form-check-label" for="sonst">

                                        Sonstiges:

                                    </label>

                                </div>

                            </div>

                            <div class="form-group col-6 flex-column d-flex"> <label class="form-control-label px-3" id="label\_id" style="display: none;">

                                    Bitte schreiben Sie eine kürze Erklärung: </label> <input type="text" id="note\_id" name="note" style="display: none;">

                            </div>

                        </div>

                        <div class="form-group col-12 flex-column d-flex"> <label class="form-control-label px-3" id="label2\_id" style="display: none; color:tomato ; font-weight:1000 ">

                                Bitte unbedingt eine Bescheinigung einfordern und unverzüglich an AW schicken. </label>

                        </div>

                        <div class="row" style="margin-top: 20px;">

                            <div class="form-group col-6 flex-column d-flex"> <label class="form-control-label px-3">

                                    Abwesend von: </label> <input type="date" id="von\_date\_id" name="von\_date" value="<?php echo (new DateTime())->format('Y-m-d'); ?>">

                            </div>

                            <div class="form-group col-6 flex-column d-flex"> <label class="form-control-label px-3">

                                    bis vorraussichtlich: </label> <input type="date" id="bis\_date\_id" name="bis\_date">

                            </div>

                        </div>

                        <div class="row" style="margin-top: 20px;">

                            <div class="form-group col-sm-6 flex-column d-flex">

                                <label class="form-control-label px-3">

                                    von Uhr

                                </label>

                                <input type="time" id="von\_uhr\_id" name="von\_uhr" value="<?php echo date('H:i') ?>">

                            </div>

                            <div class="form-group col-sm-6 flex-column d-flex">

                                <label class="form-control-label px-3">

                                    bis Uhr

                                </label>

                                <input type="time" id="bis\_uhr\_id" name="bis\_uhr">

                            </div>

                        </div>

                        <div class="d-flex justify-content-center">

                            <div class="form-group">

                                <button type="button" onclick="check\_mitarbeiter\_data()" class="btn-block btn-primary" style="margin-top: 50px;">

                                    Jetzt absenden</button>

                            </div>

                        </div>

                        <br>

                        <div id="success\_added" class="alert alert-success" role="alert">

                            <p>Fehlzeit wurde hinzufügt</p>

                        </div>

                        <div id="error\_formular" class="alert alert-danger" role="alert">

                            <p id="error\_msg"></p>

                        </div>

                    </form>

                </div>

            </div>

        </div>

    </main>

    <script>

        var radiovar = 0;

        $(function() {

            $('input[name="radio"]').on('click', function() {

                $('#error\_formular').hide(); // Fehlermeldung löschen wenn auf Radiobutton gedrückt werden

                if ($(this).val() == '3') {

                    $('#label\_id').show();

                    $('#note\_id').show();

                    $('#label2\_id').hide();

                    radiovar = 3;

                } else if ($(this).val() == '1' || $(this).val() == '2') {

                    $('#label2\_id').show();

                    $('#label\_id').hide();

                    $('#note\_id').hide();

                    radiovar = 0;

                }

            });

        });

        function show\_error(error) {

            $('#error\_formular').show();

            $('#error\_msg').html(error);

            $('#success\_added').hide();

        }

        function check\_mitarbeiter\_data() {

            var mit\_name = document.getElementsByName('name')[0].value;

            var grund = document.querySelectorAll('input[type="radio"]:checked').length == 0;

            var note = document.getElementsByName('note')[0].value;

            var von\_date = document.getElementsByName('von\_date')[0].value;

            var bis\_date = document.getElementsByName('bis\_date')[0].value;

            var von\_uhr = document.getElementsByName('von\_uhr')[0].value;

            var bis\_uhr = document.getElementsByName('bis\_uhr')[0].value;

            console.log(grund);

            console.log(radiovar);

            if (mit\_name.length == 0) { // ist ein Name geschrieben?

                show\_error('Mitarbeitername ist ungültig!!');

            } else if (grund) {  // ist ein Grund gegeben?

                show\_error('Bitte wählen Sie einen Grund aus!!');

            } else if (radiovar == 3 && note.length == 0) { // wenn Sonstiges ausgesucht ist, muss eine Erklärung auch geschieben

                show\_error('Bitte schreiben Sie eine Kürze Erklärung!');

            } else if (von\_date.length == 0) { // ist ein Anfangsdatum gegeben?

                show\_error('Ab welchem Tag sind Sie abwesend?');

            } else if (bis\_date.length == 0) { // ist ein Enddatum gegeben?

                show\_error('Bis zu welchem Tag sind Sie abwesend?');

            } else if (bis\_uhr.length == 0) { // ist ein Endzeit gegeben?

                show\_error('Bis wie viel Uhr sind Sie abwesend?');

            } else {

                $('#add\_fehlzeit\_form').submit();

            }

        }

    </script>

1. Views -> Client(sideNavClient.php)

<body class="sb-nav-fixed">

    <nav class="sb-topnav navbar navbar-expand navbar-dark bg-dark">

        <!-- Navbar Brand-->

        <a class="navbar-brand ps-3" href="<?php echo base\_url() ?>admin\_all\_projects">Fehlzeiterfassung</a>

        <!-- Sidebar Toggle-->

        <button class="btn btn-link btn-sm order-1 order-lg-0 me-4 me-lg-0" id="sidebarToggle" href="#!"><i class="fas fa-bars"></i></button>

        <!-- Navbar Search-->

        <form class="d-none d-md-inline-block form-inline ms-auto me-0 me-md-3 my-2 my-md-0">

        </form>

        <!-- Navbar-->

        <ul class="navbar-nav ms-auto ms-md-0 me-3 me-lg-4">

            <li class="nav-item dropdown">

                <a class="nav-link dropdown-toggle" id="navbarDropdown" href="#" role="button" data-bs-toggle="dropdown" aria-expanded="false"><i class="fas fa-user fa-fw"></i></a>

                <ul class="dropdown-menu dropdown-menu-end" aria-labelledby="navbarDropdown">

                    <li><a class="dropdown-item" href="#!"> <?php

                                                            echo $this->session->userdata('user\_name\_session');

                                                            ?></a></li>

                    <li>

                        <hr class="dropdown-divider" />

                    </li>

                    <li><a class="dropdown-item" href="<?php echo base\_url() ?>ref\_logout">Logout</a></li>

                </ul>

            </li>

        </ul>

    </nav>

    <div id="layoutSidenav">

        <div id="layoutSidenav\_nav">

            <nav class="sb-sidenav accordion sb-sidenav-dark" id="sidenavAccordion">

                <div class="sb-sidenav-menu">

                    <div class="nav">

                    </div>

                </div>

                <div class="sb-sidenav-footer">

                    <div class="small">Logged in as: <?= $this->session->userdata('user\_vorname\_session') . " " . $this->session->userdata('user\_nachname\_session'); ?>

                    </div>

                    <?php

                    echo $this->session->userdata('user\_name\_session');

                    ?>

                </div>

            </nav>

        </div>

1. Views -> layout(login.php)

<?php

if (isset($error)) {

?>

  <style>

    #login\_fail {

      display: block !important;

    }

  </style>

<?php

}

?>

<!doctype html>

<html lang="en">

<head>

  <!-- Required meta tags -->

  <meta charset="utf-8">

  <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">

  <link href="https://fonts.googleapis.com/css?family=Roboto:300,400&display=swap" rel="stylesheet">

  <link rel="stylesheet" href="<?php echo base\_url() ?>assets/login/fonts/icomoon/style.css">

  <link rel="stylesheet" href="<?php echo base\_url() ?>assets/login/css/owl.carousel.min.css">

  <!-- Bootstrap CSS -->

  <link rel="stylesheet" href="<?php echo base\_url() ?>assets/login/css/bootstrap.min.css">

  <!-- Style -->

  <link rel="stylesheet" href="<?php echo base\_url() ?>assets/login/css/style.css">

  <title>Login #8</title>

</head>

<body>

  <div class="content">

    <div class="container">

      <div class="row">

        <div class="col-md-6 order-md-2">

          <img src="<?php echo base\_url() ?>assets/login/images/undraw\_file\_sync\_ot38.svg" alt="Image" class="img-fluid">

        </div>

        <div class="col-md-6 contents">

          <div class="row justify-content-center">

            <div class="col-md-8">

              <div class="mb-4">

                <h3>Registrieren Sie Sich zu <strong>Fehlzeiterfassung</strong></h3>

                <p class="mb-4">Erfassen Sie Ihre Fehlzeiten.</p>

              </div>

              <form action="userlogin" method="post">

                <div class="form-group first">

                  <label for="Benutzername"></label>

                  <input type="text" class="form-control" id="username" name="username" placeholder="Benutzername">

                </div>

                <div class="form-group last mb-4">

                  <label for="Passwort"></label>

                  <input type="password" class="form-control" id="password" name="password" placeholder="Passwort" autocomplete="new-password">

                </div>

                <input type="submit" value="Log In" class="btn text-white btn-block btn-primary">

              </form>

              <br><br>

              <div id="login\_fail" class="alert alert-danger" role="alert" style="display: none;">

                <p>Benuzername oder Passwort ist falsch!</p>

              </div>

            </div>

          </div>

        </div>

      </div>

    </div>

  </div>

  <script src="js/jquery-3.3.1.min.js"></script>

  <script src="js/popper.min.js"></script>

  <script src="js/bootstrap.min.js"></script>

  <script src="js/main.js"></script>

</body>

</html>

1. assets -> css(styles.css)

.radiodiv{

  padding-left: 40px;

}

#label\_id{

  padding-top: 50px;

}

1. config (routes.php)

defined('BASEPATH') or exit('No direct script access allowed');

$route['default\_controller'] = 'welcome';

$route['404\_override'] = '';

$route['translate\_uri\_dashes'] = FALSE;

$route['ref\_logout'] = 'Login/logout';

$route['userlogin'] = 'Login/register';

$route['fehlzeiten\_view'] = 'Mitarbeiter/formular\_con';

$route['act\_add\_fehlzeit'] = 'Mitarbeiter/fun\_add\_fehlzeit';

$route['admin\_aktuel\_fehlzeiten'] = 'Admin/all\_fehlzeiten';

$route['all\_mitarbeiter\_ref'] = 'Admin/all\_mitarbeiter';

$route['add\_mitarbeiter\_ref'] = 'Admin/add\_mitarbeiter';

$route['act\_add\_mitarbeiter'] = 'Admin/con\_add\_mitarbeiter';

$route['act\_update\_mitarbeiter'] = 'Admin/update\_mit\_in\_DB';

$route['update\_mitarbeiter/(:any)'] = 'Admin/update\_mit/$1';

$route['deleteMitarbeiter/(:any)'] = 'Admin/delete\_mitarbeiter/$1';

$route['ref\_update\_admin\_pass'] = 'Admin/update\_admin\_pass';

$route['act\_update\_admin\_pass'] = 'Admin/update\_admin\_pass\_in\_DB';

1. config (config.php)

$config['base\_url'] = 'http://localhost/FehlzeitPro/';;//der Name meines Projektverzeichnis.

$config['charset'] = 'UTF-8';

$config['log\_threshold'] = 4;

$config['log\_date\_format'] = 'Y-m-d H:i:s';

$config['encryption\_key'] = 'UVvM3gosO8yK9JeijFwasEXD8kHs2G1y';

$config['time\_reference'] = 'local';