



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



Ausbildungsbetrieb

Wilke Technology
Heider-Hof-Weg 23D
52080 Aachen

Ausbilder

Jürgen Wilke

Projektunterstützer

Tobias Esser

Inhalt

1	Einleitung.....	3
1.1	Projektbeschreibung	3
2	Ausgangssituation (IST-Analyse).....	3
2.1	Projektziel.....	Fehler! Textmarke nicht definiert.
2.2	Kundenwünsche.....	6
2.3	Abweichen zum Antrag	Fehler! Textmarke nicht definiert.
2.4	Projektumfeld.....	5
2.5	Prozessschnittstelle	6
3	Ressourcen und Ablaufplanung.....	6
3.1	Ressourcenplanung	6
3.2	Projektkosten	7
3.3	Ablaufplanung	7
4	Durchführung und Auftragsbearbeitung.....	7
4.1	Prozrssschritte	7
4.2	Vorgehensweise	8
4.3	Arbeitsschritte	8
4.4	Abweichungen, Anpassungen, entscheidungen	14
5	Projektergebnisse.....	15
5.1	Soll-Ist-Vergleich.....	15
5.2	Qualitätskontrolle.....	15
5.3	Abweichungen, Anpassungen, Entwicklungsmöglichkeiten	15
6	Dokumentation.....	16
6.1	Benutzerhandbuch	16
7	Anhang.....	19
8.1	Abkürzungsverzeichnis	19
8.2	Quellenverzeichnis	20
8.3	Code-Anhang.....	20

1 Einleitung

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

1.1 Projektbeschreibung

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


2 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:

Formular	Version:	erstellt/bearbeitet	freigegeben	
Erfassung von Fehlzeiten	005	TE	AW	
FO_Fehlzeiten.php		24.08.2017	16.02.2010	

08 November. 202

Vorname:
Nachname:
Grund der Abwesenheit:
 Krankheit: ☐
 Arztbesuch: ☐
 Sonstiges:
abwesend von:

bis voraussichtlich:

Aachen, den 08 November. 2022

Ort, Datum

Gemeldet durch (Kürzel)

Jetzt absenden!

2.1 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 liegt, 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.

2.2 Kundenwünsche

- Login mit mitarbeiterspezifischen Benutzernamen & Passwort
- Erstellung neuer Benutzer durch berechtigte Benutzer
- Fehlzeiterfassung der Benutzer
- API zum Lesen aktueller Fehlzeiten und Löschung veralteter
- Einfache Bedienung

2.3 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.

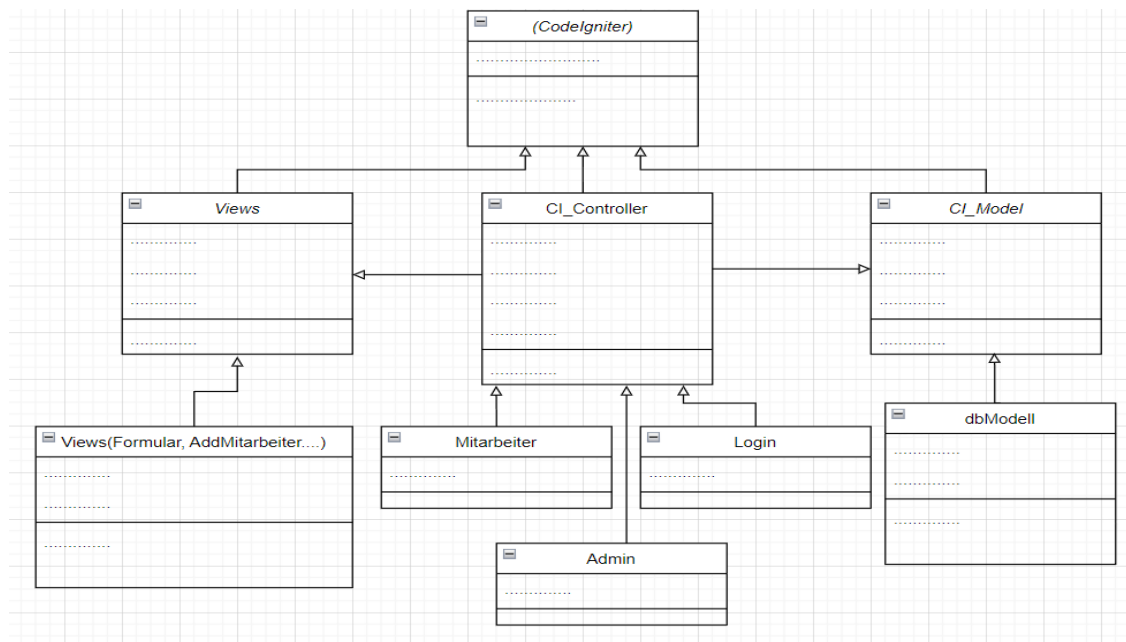
Das hat auch die Folge, dass es kein Auslöser mehr zur Verfügung steht, um veraltete Einträge aus der Datenbank zu löschen.

3 Projektumfeld

CodeIgniter

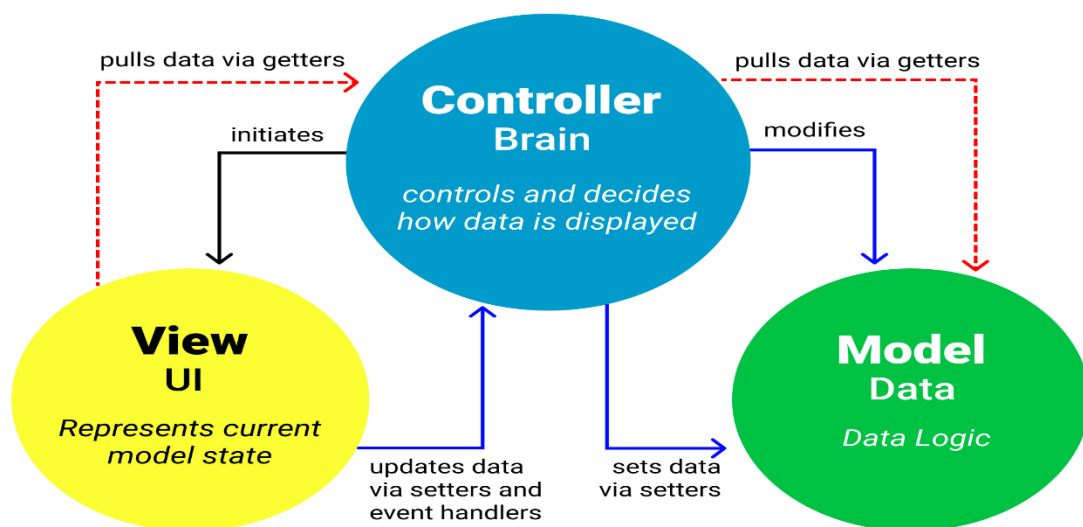
CodeIgniter ist ein seit 2014 erhältliches, in PHP geschriebenes, quelloffenes Webframework.

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:

MVC Architecture Pattern



Bootstrap

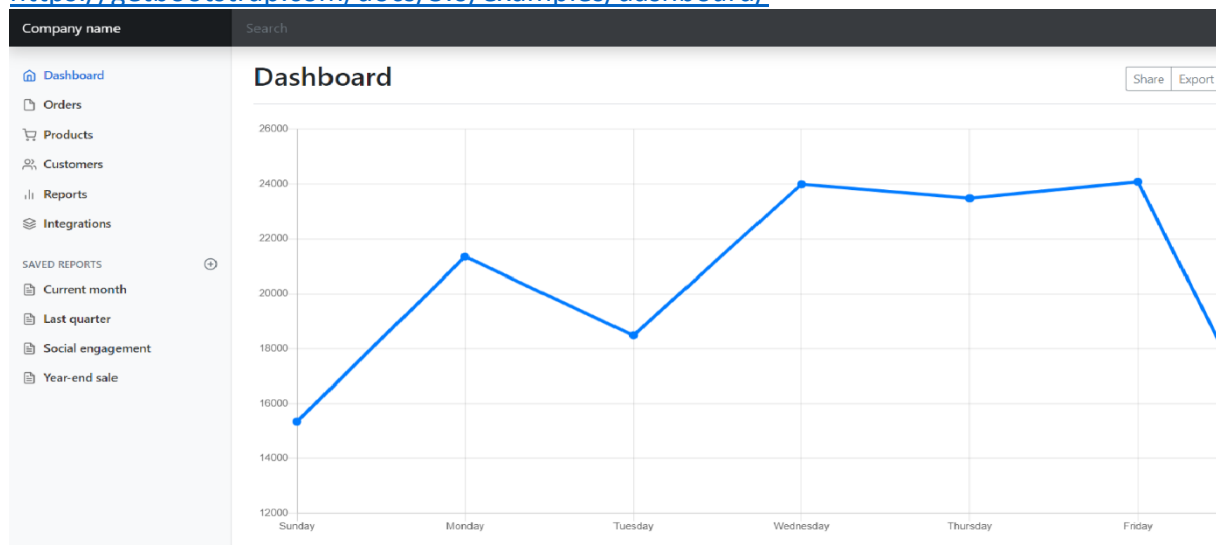
Bootstrap ist ein kostenloses und quelloffenes Frontend-Web-Framework zur Erstellung von Webseiten. Es dient nur der Front-End-Entwicklung und enthält HTML- und CSS-basierte Designvorlagen für Typografie, Formulare, Schaltflächen und Navigation.

3.1 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/>



4 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.

4.1 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.

4.2 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€ | $100\text{€} : 160\text{Std} = 0,625\text{€ pro Std.}$

Aus diesen Daten entsteht folgende Kalkulation:

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

4.3 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

5 Durchführung und Auftragsbearbeitung

5.1 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

5.2 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.

5.3 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.

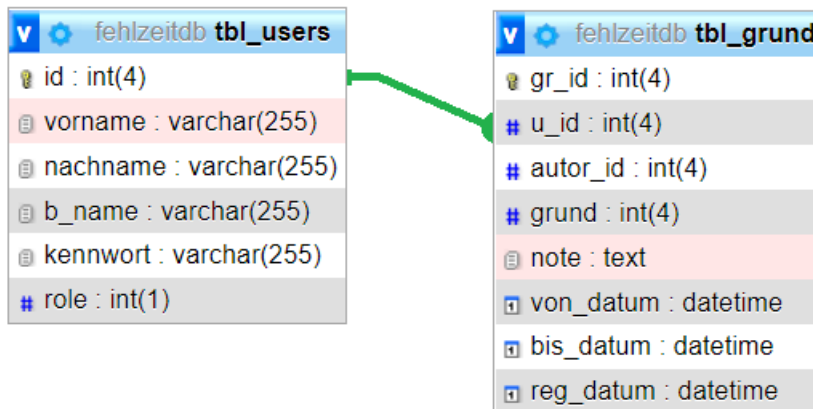
Als Frontend-Framework habe ich Bootstrap gewählt.

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.



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 ich 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 geschreiben
        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_benutzer)
{
    $this->db->select('*');
    $this->db->where('b_name', $mit_benutzer);
    $this->db->from('tbl_users');
    $query = $this->db->get();
}

```

```

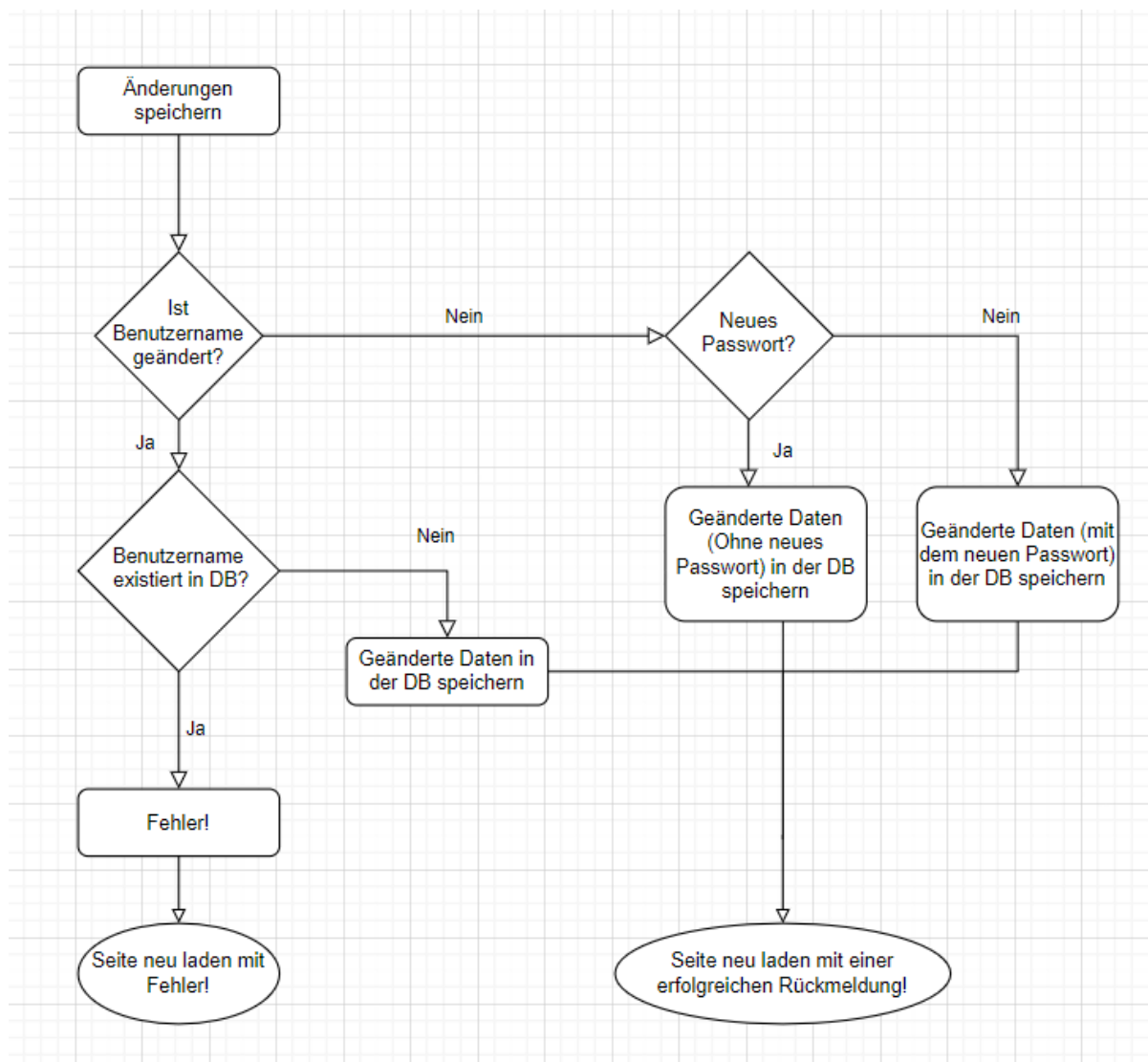
$result_array = $query->result_array();
if (!empty($result_array)) {
    return 1; // Benutzername existiert
}
else
{
    return 0; // Benutzername existiert nicht
}
}

```

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_benutzer)
```

Jetzt habe ich alle notwendigen Informationen, um meine Verzweigung in dem Controller 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_benutzer, $mit_password)
{
    ...
    if (empty($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 Controller die Verzweigung geschrieben werden:

```
public function con_add_mitarbeiter()
{
    ...
    $is_mit_added = $this->dbmodel->add_new_mitarbeiter($mit_name, $mit_nachname,
    $mit_benutzer, $key);
    if ($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
```

5.4 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 Webanwendung zur Administration von MySQL-Datenbanken und 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.

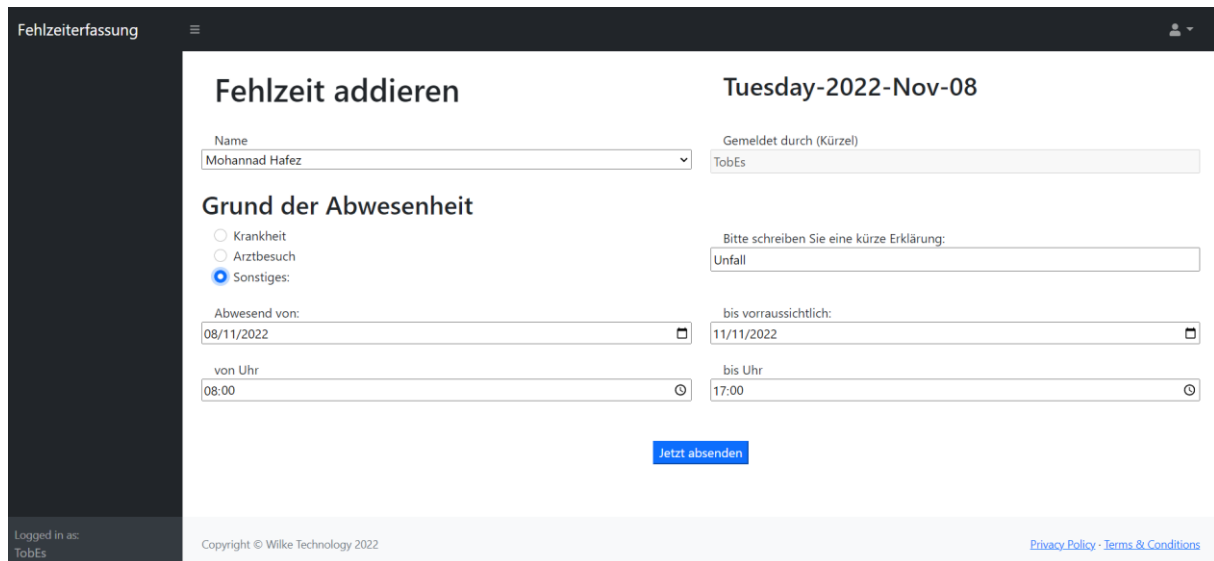
Deshalb müssen wir eine weitere Spalte in der Datenbank zur „tbl_grund“ Tabelle hinzufügen, um den Benutzer zu speichern, der diesen Eintrag hinzugefügt hat. Der Name dieser Spalte ist autor_id.

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.

6 Projektergebnisse

So sieht das neue Formular aus:



The screenshot shows a web application titled 'Fehlzeiterfassung'. The main heading is 'Fehlzeit addieren' for the date 'Tuesday-2022-Nov-08'. The form includes a 'Name' dropdown menu with 'Mohannad Hafez' selected, and a 'Gemeldet durch (Kürzel)' field with 'TobEs'. Under 'Grund der Abwesenheit', there are radio buttons for 'Krankheit', 'Arztbesuch', and 'Sonstiges:' (selected). A text field for 'Bitte schreiben Sie eine kurze Erklärung:' contains 'Unfall'. The 'Abwesend von:' section has date and time pickers: '08/11/2022' and '08:00' for the start, and '11/11/2022' and '17:00' for the end. A 'Jetzt absenden' button is at the bottom. The footer shows 'Logged in as: TobEs', 'Copyright © Wilke Technology 2022', and links for 'Privacy Policy' and 'Terms & Conditions'.

6.1 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.

6.2 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.

6.3 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.

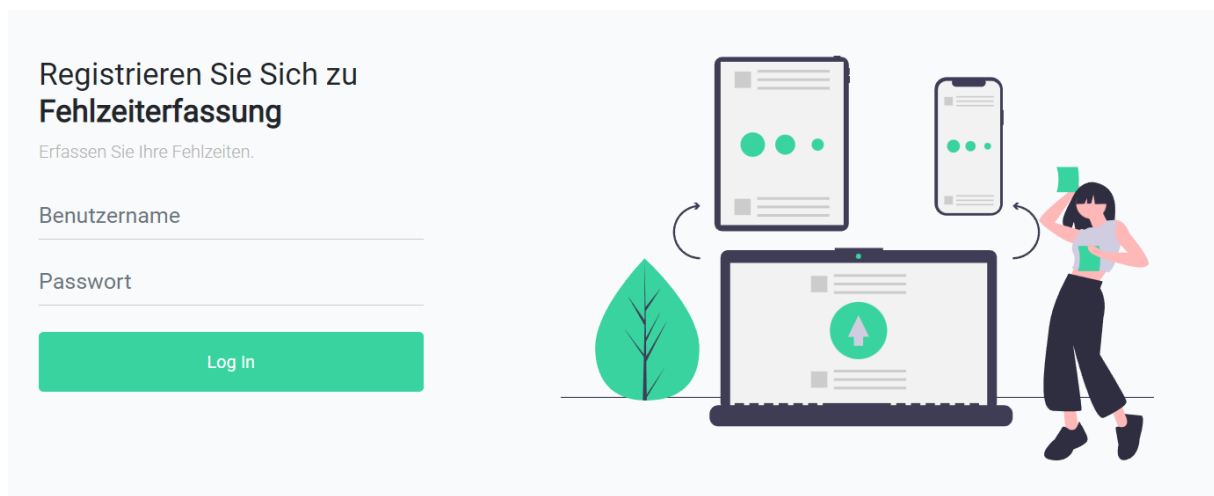
In diesem Fall können wir die Ansicht einsehen, dass dieser Mitarbeiter heute krank oder aus irgendeinem Grund nicht im Unternehmen ist.

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.

7 Dokumentation

7.1 Benutzerhandbuch

1. Anmeldung Layout



Registrieren Sie Sich zu
Fehlzeiterfassung

Erfassen Sie Ihre Fehlzeiten.

Benutzername

Passwort

Log In

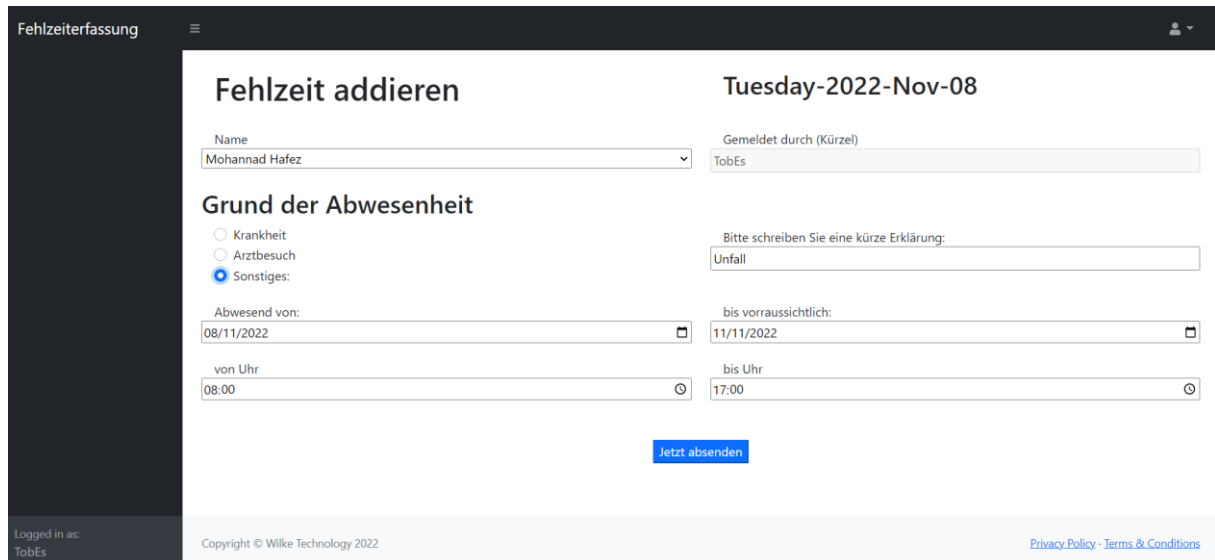
Hier soll der Benutzer sein Benutzername und Passwort eingeben, um anzumelden.

Admin hat der voreingestellte Benutzername (admin) und das voreingestellte Passwort

Online-Projektzeiterfassung

„admin1“ für der erste Anmeldung, auf Sicherheitsgründe, der Admin soll das Passwort als erstes ändern.

2. Benutzer Layout



Fehlzeit addieren Tuesday-2022-Nov-08

Name: Gemeldet durch (Kürzel):

Grund der Abwesenheit

☐ Krankheit ☐ Arztbesuch ☒ Sonstiges:

Bitte schreiben Sie eine kurze Erklärung:

Abwesend von: bis voraussichtlich:

von Uhr: bis Uhr:

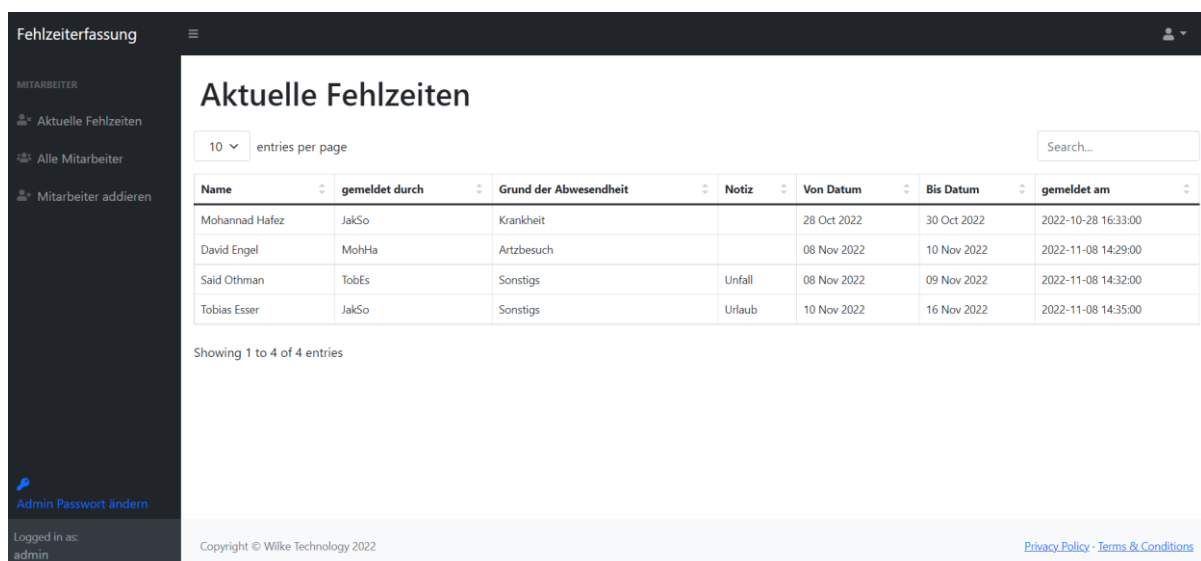
[Jetzt absenden](#)

Logged in as: TobEs Copyright © Wilke Technology 2022 [Privacy Policy](#) [Terms & Conditions](#)

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 wurde durch eine Grüne Rückmeldung „**Fehlzeit wurde hinzugefügt**“.

3. Admin Layout



Aktuelle Fehlzeiten

10 entries per page Search...

Name	gemeldet durch	Grund der Abwesenheit	Notiz	Von Datum	Bis Datum	gemeldet am
Mohannad Hafez	JakSo	Krankheit		28 Oct 2022	30 Oct 2022	2022-10-28 16:33:00
David Engel	MohiHa	Arztbesuch		08 Nov 2022	10 Nov 2022	2022-11-08 14:29:00
Said Othman	TobEs	Sonstigs	Unfall	08 Nov 2022	09 Nov 2022	2022-11-08 14:32:00
Tobias Esser	JakSo	Sonstigs	Urlaub	10 Nov 2022	16 Nov 2022	2022-11-08 14:35:00

Showing 1 to 4 of 4 entries

Admin Passwort ändern

Logged in as: admin Copyright © Wilke Technology 2022 [Privacy Policy](#) [Terms & Conditions](#)

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











4. Alle Mitarbeiter (Admin)

Fehlzeiterfassung

Alle Mitarbeiter

Fehlzeiten

10 entries per page Search...

Vorname	Nachname	Benutzername	Steuerung
Mohannad	Hafez	MohHa	 
David	Engel	DavEn	 
Jack	Soufi	JakSo	 
Said	Othman	SaiOt	 
Tobias	Esser	TobEs	 

Showing 1 to 5 of 5 entries

[Admin Passwort ändern](#)

Logged in as: admin

Copyright © Wilke Technology 2022 [Privacy Policy](#) [Terms & Conditions](#)

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.

Fehlzeiterfassung

Mitarbeiter bearbeiten

Verwaltung

Name: Nachname:

Benutzername:

Passwort: Passwort Bestätigung:

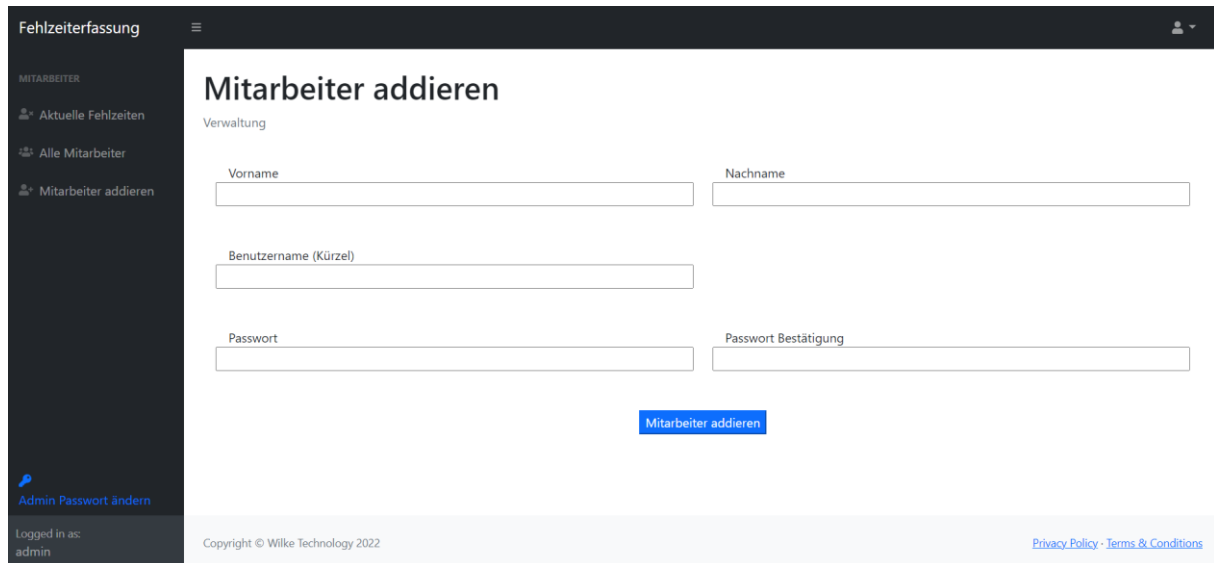
[Speichern](#)

[Admin Passwort ändern](#)

Logged in as: admin

Copyright © Wilke Technology 2022 [Privacy Policy](#) [Terms & Conditions](#)

5. Mitarbeiter addieren (Admin)



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

8 Anhang

8.1 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

8.2 Quellenverzeichnis

[https://en.wikipedia.org/wiki/Bootstrap_\(front-end_framework\)](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/>

8.3 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_benutzer  = $this->input->post('b_name');
    $mit_password  = $this->input->post('password');
    $key = $this->encryption->encrypt($mit_password);
    $is_mit_added = $this->dbmodel->
>add_new_mitarbeiter($mit_name, $mit_nachname, $mit_benutzer,
$key);
    if ($is_mit_added == 1) { // Benutzername existiert
nicht in der DB
        $this->session->set_userdata('success', 1);
    } else {
        $this->session->set_userdata('fail', 1);
    }
    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_benutzer_name = $this->input->post('b_name');
    $mit_password     = $this->input->post('password');
    $key = $this->encryption->encrypt($mit_password); //
Passwort verschlüsseln
    $name_not_changed = $this->dbmodel->
>check_is_same_Bname($mit_id, $mit_benutzer_name);
    $is_Benutzer_exsitiert = $this->dbmodel->
>check_is_Bname_existiert($mit_benutzer_name);
    if ($name_not_changed == 1) { //das selbe
Benutzerdaten
        if ($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_Benutzer_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_benutzer_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('password');
        $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/');
    }
}

```

2. 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);
    }
}

```

3. 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');
}
}
```

4. 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);
    }
}
```

5. 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_benutzer, $mit_password)
{
    $this->db->select('*');
    $this->db->where('b_name', $mit_benutzer);
    $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_benutzer,
            '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_benutzer)
{
    $this->db->select('*');
    $this->db->where('b_name', $mit_benutzer);
    $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_benutzer)
{
    $this->db->select('*');
    $this->db->where('id', $mit_id);
    $this->db->where('b_name', $mit_benutzer);
    $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_benutzer_name,
$mit_password)
{
    $data = array(
```

```

        'vorname' => $mit_vorname, 'nachname' =>
$mit_nachname, 'kennwort' => $mit_password, 'b_name' =>
$mit_benutzer_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;
}
}

```

6. 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;
}

```

7. Views -> Admin(addMitarbeiter.php)

```

<style>
    #error_mitarbeiter {
        display: none;
    }

    #success_added {
        display: none;
    }

    #fail_added {
        display: none;
    }
</style>
<?php
$mit_added_success = $this->session->userdata('success');
//controller hat die Session gesetzt anhand die DBmodell
Rückgabe
if (isset($mit_added_success)) {
    ?>
        <style>
            #success_added {
                display: block;
            }
        </style>
    <?php
    }
    $this->session->unset_userdata('success'); //controller hat
    die Session gesetzt 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>Benutzername 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 Zeichen?

```

```

        show_error('Passwort ist zu kurz!! Das
Passwort sollte mindestens 5 Zeichen sein ');
    } else if (mit_passwort != mit_passwort_w) { //
ist das Passwort mit seiner Wiederholung identisch?
        show_error('Die eingegebenen Passwörter
stimmen nicht überein');
    } else {
        $('#add_mitarbeiter_form').submit(); // Datein
an den Controller schicken POST
    }
}
</script>

```

8. 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>Benutzername</th>
                        <th>Steuerung</th>
                    </tr>
                </thead>
                <tfoot>
                    <tr>
                        <th>Vorname</th>
                        <th>Nachname</th>
                        <th>Benutzername</th>
                        <th>Steuerung</th>
                    </tr>
                </tfoot>
            </table>
        </div>
    </main>
</div>

```

```

        <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>

```

9. 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 Abwesenheit</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 Abwesenheit</th>
            <th>Notiz</th>
            <th>Von Datum</th>
            <th>Bis Datum</th>
            <th>gemeldet am</th>
        </tr>
    </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>

```

10. 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

```

```

ec
ho $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>

```

11. 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>Benutzername existiert schon!</p>
        </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>

```

12. 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('password')[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 Zeichen sein ');
        } else if (mit_passwort_w.length <= 5) {
            show_error('Passwort Bestätigung ist zu kürz!!
Das Passwort sollte mindestens 5 Zeichen 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>

```

13. 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) {
                        <?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>
        </div>
    </form>

```

```

<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 voraussichtlich:
</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 hinzugefügt</p>
    </div>
    <div id="error_formular" class="alert
alert-danger" role="alert">
        <p id="error_msg"></p>
    </div>

```

```

        </form>
      </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
geschrieben
            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>

```

14. 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
ec
ho $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>

```

15. 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&d
isplay=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">
```

```

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

```

16. assets -> css(styles.css)

```
.radiodiv{
  padding-left: 40px;
}
#label_id{
  padding-top: 50px;
}
```

17. 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';
```

18. 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';
```