



# DIPLOMARBEIT

## CareCRM

Webbasiertes CRM-System für den Einsatz in Agenturen zur Vermittlung von Pflegekräften in der Hauskrankenpflege

**Ausgeführt im Schuljahr 2019/20 von:**

Mario Praxmarer      6CBKIF

Bernhard Bonengl      6CBKIF

**Betreuer:**

Mag. Hannes Grundschober

Mag. Hannes Grunschober

Wien, im Juni 2020



## EIDESSTATTLICHE ERKLÄRUNG

Wir erklären an Eides statt, dass wir die vorliegende Diplomarbeit selbstständig und ohne fremde Hilfe verfasst, andere als die angegebenen Quellen und Hilfsmittel nicht benutzt und die den benutzten Quellen wörtlich und inhaltlich entnommenen Stellen als solche kenntlich gemacht haben.

Wien, am 25.06.2020

.....

(Mario Praxmarer)

.....

(Bernhard Bonengl)



## KURZFASSUNG

Gegenstand der Diplomarbeit war die Entwicklung eines Customer Relations Management System als Webapplikation.

Folgende Neuerungen wurden in dieser Arbeit realisiert: Bisher gibt es kein CRM-System auf dem deutschsprachigen Markt, welches explizit für die Verwendung in der Administration der Hauskrankenpflege bestimmt ist bzw. vermarktet wird. Im Zuge dieses Diplomprojekts wurde so ein System entwickelt und dabei auf die speziellen Bedürfnisse und nötigen Funktionen dieser Branche eingegangen.

Folgende Teilaufgaben bzw. Entwicklungen waren dabei zu lösen: Die Auswahl der zu benötigten Entwicklungsumgebungen, Programmiersprachen und Frameworks für das User Interface, das Backend, die Datenbank und die Serverstruktur.

Erstellung Wireframe-Konzept und Geschäftsprozesse

Datenbank inkl. Backupkonzept

User Interface Design

Implementierung Webinterface

User Acceptance Test/Quality Assurance

Zusätzliches Augenmerk wurde auf effiziente Datenstruktur und einfache Usability gelegt.



## DIPLOMARBEIT

### DOKUMENTATION

Namen der Verfasser/innen	Mario Praxmarer Bernhard Bonengl
Jahrgang/ Klasse Schuljahr	6CBKIF 2019/20
Thema der Diplomarbeit	CRM als Webapplikation
Kooperationspartner	-

Aufgabenstellung	<p>Entwicklung eines Webbasiertes CRM (Customer Relations Management) Systems mit Schwerpunkt auf dem Einsatz im privaten, medizinischen Dienstleistungssektor, explizit: Eine Agentur für die Vermittlung von Pflegekräften für den Einsatz in der Hauskrankenpflege (24h oder stundenweise Betreuung von Patienten).</p> <p>Die wichtigsten Funktionen beinhalten die Verwaltung der Pflegekräfte, der Patienten sowie deren Sachwalter und/oder Auftraggeber. Pflegepläne, Kalender für Hausbesuche zur Qualitätssicherung, Faktura, Export von Listen für die Buchführung und den Rechnungsdruck, sowie Backupfunktionalität und Ausfallsicherung für die Datenbank.</p> <p>User acceptance Tests aufgrund der Userstories</p> <p>Behebung etwaiger während der Tests erschienenen Mängel.</p>
------------------	--

Realisierung	Es wurden folgende logische Arbeitsbereiche erstellt, welche unabhängig voneinander geplant und realisiert werden konnten: Serveraufbau, Datenbank, User Interface, Backend-Logik, Tests.
--------------	--



Realisierung	<p>Die Serverlösung, welche die Applikation zur Verfügung stellen soll, wurde als Linux CentOS Server Instanz entworfen. Auf diesem wurden via Docker eine Clusterlösung zur Ausfallsicherheit der Datenbank realisiert.</p> <p>Als Datenbankumgebung wurde eine MariaDB installation auf MySQL-Basis verwirklicht.</p> <p>Für das User Interface wurde das HTML &amp; CSS Framework Bootstrap ausgewählt und das UI zusätzlich mit individuellem CSS und Javascript benutzerfreundlicher gestaltet.</p> <p>Die Backend-Logik als Brücke zur Datenbank wurde ausschließlich mithilfe von PHP 7 realisiert.</p> <p>Tests wurden laufend während der Implementierung der Logik durchgeführt, um die Funktionsfähigkeit der teilweise aufeinander aufbauenden Module sicher zu stellen.</p>
Ergebnisse	<p>Fertigstellung sämtlicher Module der Applikation zur logischen Abbildung der Geschäftsfälle sowie einer effizienten Datenbankstruktur und eines ansprechenden und benutzerfreundlichen User Interfaces.</p>



Applikation im Desktop-Modus

The screenshot shows the Care CRM application running in desktop mode. The main window displays a list of customers under the heading "Kunden → Alle Kunden". The list includes columns for Anrede (Title), Nachname (Last Name), Vorname (First Name), PLZ (Zip Code), Adresse (Address), Telefon (Phone), and Details. The data is as follows:

Anrede	Nachname	Vorname	PLZ	Adresse	Telefon	Details
Herr	Musterman	Mario	1130	Anton-Langer 66/3	+43 660 5634	Details
Herr	Kleinmann	Peter	1110	Luise Montag Gasse 1	01 3434	Details
Herr	Sedlacek	Markus	1110	Grillgasse 66/2/6	01 567 89 01	Details
Herr	Österreicher	Markus	1010	Zweigeltstraße 12/7	01 234 3456	Details
Frau	Kraft	Elena	1010	Markusplatz 1	01 987 34 44	Details
Herr	Berger	Willibald	1030	Hansgasse 12/2/7	01 234 55 66	Details
Herr	Zimmer	Peter	1200	Mariusplatz 12/1/2/4	01 34 3445	Details
Herr	Egger	Helmut	1200	Vogelgasse 12/1/2/4	01 56 444 5	Details
Herr	Schindler	Herbert	8020	Kleingasse 12		Details
Herr	Müller	Bernhard	1110	Hausbrandstraße 12		Details

Eingeloggt als: **Mario Praxmarer** [Ausloggen](#)

**Abb. 1.1.1 - Applikation im Desktop-Modus**

Teilnahme an Wettbewerben: Auszeichnungen	Keine
--	-------

Approbation (Datum / Unterschrift)	Prüfer/in	Abteilungsvorstand / Direktion DI Andreas Heinbach
---------------------------------------	-----------	---



## DIPLOMA THESIS

### DOCUMENTATION

Authors	Mario Praxmarer Bernhard Bonengl
From Academic year	6CBKIF 2019/20
Topic	CRM System as a web application
Co-operation partners	-

Assignment of tasks	Development of a web-based CRM (Customer Relations Management) system for use in the private medical service sector, meaning an agency specialized in arranging caregivers and nursing staff for round-the-clock or hour-wise caretaking of patients.  The main functions include management of staff, patients, and their relatives and custodians. Schedules for caretaking activities and quality assurance visits, billing, export of lists for accounting and invoice printing, and backup function of the database.  User acceptance testing based upon user stories and rectifying flaws accordingly.
---------------------	--

Realization	Logical tasks have been created which were planned and realized independently from each other: Server structure, database, user interface, backend logic, testing.
-------------	--



Realization	<p>The server from which the application is deployed has been set up in form of as a Linux CentOS Server instance.</p> <p>Backup functionality implemented as a cluster solution has been realized via docker on the server.</p> <p>The database functions are carried out by a MariaDB installation, based on MySQL.</p> <p>Bootstrap, an HTML &amp; CSS framework, has been used to implement the user interface. The usability has been further improved by individual CSS and Javascript.</p> <p>PHP 7 exclusively has been used for implementation of the backend logic</p> <p>Tests have been applied continuously during programming to make sure the modules of which some are logically based on each other were working correctly</p>
Results	<p>Completion of all modules of the application to logically model of the business processes. Creation of an efficient database structure and a user-friendly interface.</p>



<p><b>Desktop version of the application</b></p>	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>Anrede</th><th>Nachname</th><th>Vorname</th><th>PLZ</th><th>Adresse</th><th>Telefon</th><th>Details</th></tr> </thead> <tbody> <tr><td>Herr</td><td>Musterman</td><td>Mario</td><td>1130</td><td>Anton-Langer Gasse 1</td><td>+43 660 5634</td><td><a href="#">Details</a></td></tr> <tr><td>Herr</td><td>Kleinmann</td><td>Peter</td><td>1110</td><td>Luise Montag Gasse 1</td><td>01 3434</td><td><a href="#">Details</a></td></tr> <tr><td>Herr</td><td>Sedlacek</td><td>Markus</td><td>1110</td><td>Grillgasse 66/2/6</td><td>01 567 89 01</td><td><a href="#">Details</a></td></tr> <tr><td>Herr</td><td>Österreicher</td><td>Markus</td><td>1010</td><td>Zweigeltstraße 12/7</td><td>01 234 3456</td><td><a href="#">Details</a></td></tr> <tr><td>Frau</td><td>Kraft</td><td>Elena</td><td>1010</td><td>Markusplatz 1</td><td>01 987 34 44</td><td><a href="#">Details</a></td></tr> <tr><td>Herr</td><td>Berger</td><td>Willibald</td><td>1030</td><td>Hansgasse 12/2/7</td><td>01 234 55 66</td><td><a href="#">Details</a></td></tr> <tr><td>Herr</td><td>Zimmer</td><td>Peter</td><td>1200</td><td>Mariusplatz 12/1/2/4</td><td>01 34 3445</td><td><a href="#">Details</a></td></tr> <tr><td>Herr</td><td>Egger</td><td>Helmut</td><td>1200</td><td>Vogelgasse 12/1/2/4</td><td>01 56 444 5</td><td><a href="#">Details</a></td></tr> <tr><td>Herr</td><td>Schindler</td><td>Herbert</td><td>8020</td><td>Kleingasse 12</td><td></td><td><a href="#">Details</a></td></tr> <tr><td>Herr</td><td>Müller</td><td>Bernhard</td><td>1110</td><td>Hausbrandstraße 12</td><td></td><td><a href="#">Details</a></td></tr> </tbody> </table>	Anrede	Nachname	Vorname	PLZ	Adresse	Telefon	Details	Herr	Musterman	Mario	1130	Anton-Langer Gasse 1	+43 660 5634	<a href="#">Details</a>	Herr	Kleinmann	Peter	1110	Luise Montag Gasse 1	01 3434	<a href="#">Details</a>	Herr	Sedlacek	Markus	1110	Grillgasse 66/2/6	01 567 89 01	<a href="#">Details</a>	Herr	Österreicher	Markus	1010	Zweigeltstraße 12/7	01 234 3456	<a href="#">Details</a>	Frau	Kraft	Elena	1010	Markusplatz 1	01 987 34 44	<a href="#">Details</a>	Herr	Berger	Willibald	1030	Hansgasse 12/2/7	01 234 55 66	<a href="#">Details</a>	Herr	Zimmer	Peter	1200	Mariusplatz 12/1/2/4	01 34 3445	<a href="#">Details</a>	Herr	Egger	Helmut	1200	Vogelgasse 12/1/2/4	01 56 444 5	<a href="#">Details</a>	Herr	Schindler	Herbert	8020	Kleingasse 12		<a href="#">Details</a>	Herr	Müller	Bernhard	1110	Hausbrandstraße 12		<a href="#">Details</a>
Anrede	Nachname	Vorname	PLZ	Adresse	Telefon	Details																																																																								
Herr	Musterman	Mario	1130	Anton-Langer Gasse 1	+43 660 5634	<a href="#">Details</a>																																																																								
Herr	Kleinmann	Peter	1110	Luise Montag Gasse 1	01 3434	<a href="#">Details</a>																																																																								
Herr	Sedlacek	Markus	1110	Grillgasse 66/2/6	01 567 89 01	<a href="#">Details</a>																																																																								
Herr	Österreicher	Markus	1010	Zweigeltstraße 12/7	01 234 3456	<a href="#">Details</a>																																																																								
Frau	Kraft	Elena	1010	Markusplatz 1	01 987 34 44	<a href="#">Details</a>																																																																								
Herr	Berger	Willibald	1030	Hansgasse 12/2/7	01 234 55 66	<a href="#">Details</a>																																																																								
Herr	Zimmer	Peter	1200	Mariusplatz 12/1/2/4	01 34 3445	<a href="#">Details</a>																																																																								
Herr	Egger	Helmut	1200	Vogelgasse 12/1/2/4	01 56 444 5	<a href="#">Details</a>																																																																								
Herr	Schindler	Herbert	8020	Kleingasse 12		<a href="#">Details</a>																																																																								
Herr	Müller	Bernhard	1110	Hausbrandstraße 12		<a href="#">Details</a>																																																																								

**Abb. 1.1.1 - desktop version of the application**

Participation in competitions Awards	none
--------------------------------------	------

Approval (Date/Sign)	Examiner	Head of Department/ College  DI Andreas Heinbach
----------------------	----------	---



## ABSTRACT

Development of a customer relations management system realized as web-based application.

The following innovative changes have been incorporated: There are no CRM Systems readily available in the German speaking regions which are exclusively made for use in home care management. In the course of this diploma project such a system has been created.

In this context, the following tasks respectively innovations had to be implemented: Selection of the necessary development environment, programming languages, and frameworks for the user interface, the backend logic, the database, and the server structure.

Creating of a wireframe concept to model the business processes

Database including backup functionality

User interface design

Implementation of the web interface

User acceptance tests and quality assurance

Special emphasis has been put in efficient data structure and easy-to-understand usability.



## INHALTSVERZEICHNIS

EIDESSTATTLICHE ERKLÄRUNG .....	2
KURZFASSUNG.....	3
DIPLOMARBEIT .....	4
DIPLOMA THESIS.....	7
ABSTRACT .....	10
INHALTSVERZEICHNIS.....	11
<b>1. EINLEITUNG - PRAXMARER.....</b>	<b>16</b>
1.1. VORWORT.....	16
1.2. ZUSTÄNDIGKEITEN .....	17
1.2.1. MARIO PRAXMARER.....	18
1.2.2. BERNHARD BONENGL .....	19
1.3. ANFORDERUNGEN/LASTENHEFT.....	20
1.4. USERSTORIES/EPICS .....	22
1.5. NICHT-ZIELE .....	25
<b>2. DATENBANKKONZEPT – BONENGL.....</b>	<b>28</b>
2.1. TABELLEN .....	28
2.2. RELATIONENSCHEMA .....	32
2.3. HERAUSFORDERUNGEN UND LÖSUNGEN .....	35
2.4. REALISIERUNG DER DATENBANK .....	37
<b>3. SERVERAUFBAU/-KONFIGURATION – BONENGL .....</b>	<b>38</b>
3.1. AUSWAHL SERVERUMGEBUNG .....	38



3.2. AUFSETZEN DES SERVERS .....	38
3.3. VIRTUALISIERUNGSSOFTWARE.....	39
3.4. GRUNDLEGENDE KONFIGURATION DER SERVER-IMAGES .....	39
3.4.1. NETZWERKSCHITTSTELLE .....	39
3.4.2. FIREWALL.....	40
<b>4. AUSFALLSICHERHEIT/REDUNDANZ – BONENGL.....</b>	<b>41</b>
4.1. CLUSTERLÖSUNG .....	41
4.2. INDIVIDUELLE SERVERKONFIGURATION.....	42
4.2.1. MANAGEMENT-SERVER .....	42
4.2.2. DATA-SERVER .....	42
4.2.3. DATENBANK-SERVER .....	43
4.2.4. APACHE-WEBSERVER.....	43
<b>5. USER INTERFACE UND DESIGN - PRAXMARER .....</b>	<b>44</b>
5.1. DESIGNENTScheidungen .....	44
5.1.1. FARBEN.....	45
5.1.2. LOGO .....	46
5.2. AUSWAHLPROZESS FRAMEWORK.....	47
5.3. ARBEITSBEREICH.....	49
5.4. NAVIGATION.....	50
5.5. CSS.....	52
5.6. JAVASCRIPT-LÖSUNGEN.....	54
<b>6. FRONTEND-LOGIK - PRAXMARER .....</b>	<b>59</b>
6.1. LOGISCHE UNTERTEILUNG DES SYSTEMS .....	59



6.2. GRUNDAUFBAU JEDER DATEI.....	62
6.3. FUNKTIONEN .....	63
6.4. FORMULARHANDLING UND SICHERHEIT.....	66
6.4.1. SICHERHEIT DER DATENBANKVERBINDUNG.....	67
6.4.2. ZUGRIFFSKONTROLLE .....	69
6.5. PERSONENMANAGEMENT .....	72
6.5.1. XXALLE.PHP - ANZEIGE ALLER PERSONEN.....	73
6.5.2. XXDETAILS.PHP - ANZEIGE DER DETAILS DES AKTS .....	74
6.5.3. XXDETAILS2.PHP - ANZEIGE DER VERMERKE.....	75
6.5.4. XXDETAILS3.PHP - ANZEIGE DES PFLEGEPLANS .....	76
6.5.5. KDDTAILS4.PHP - ANZEIGE DER TARIFE DES KUNDEN.....	78
6.5.6. KDDTAILS5.PHP - ANZEIGE DER RECHNUNGEN DES KUNDEN .....	78
6.5.7. XXNEU.PHP - NEUANLAGE EINES AKTS .....	81
6.5.8. XXAENDERN.PHP - ÄNDERN DER DATEN DES AKTS .....	81
6.5.9. XXSUCHE.PHP - SUCHEN EINES KUNDEN .....	83
6.6. LEISTUNGEN .....	84
6.6.1. LGALLE.PHP – ANZEIGE ALLER LEISTUNGEN.....	84
6.6.2. LGNEU.PHP – NEUE LEISTUNG HINZUFÜGEN .....	85
6.6.3. LGINDEX.PHP – INDEXIERUNG BESTEHENDER LEISTUNGEN.....	86
6.7. VERRECHNUNG .....	87
6.7.1. RGALLE.PHP – ANZEIGE ALLER RECHNUNGEN.....	88
6.7.2. RGOFFEN.PHP – ANZEIGE DER OFFENEN RECHNUNGEN.....	89
6.7.3. RGEXPORT.PHP – EXPORT DER RECHNUNGEN ALS CSV-DATEI.....	90



6.7.4. RGTODO.PHP – NOCH NICHT VERRECHNETE LEISTUNGEN ANZEIGEN .....	91
6.7.5. RGNEUKD.PHP – ERSTELLUNG DER RECHNUNG FÜR EINEN KUNDEN .....	92
6.7.6. RGNEUMAN.PHP – MANUELLE VERRECHNUNG .....	98
6.7.7. RGSTORNO.PHP – RECHNUNGSDETAILS ANSEHEN .....	99
6.7.8. RECHNUNG.PHP – EXPORT DER RECHNUNG ALS PDF .....	100
6.8. PFLEGEPLAN .....	101
6.8.1. PPHB.PHP – ANZEIGE DER HAUSBESUCHE .....	102
6.8.2. PPDETAILS.PHP – DETAILANSICHT DER TÄTIGKEIT .....	103
6.8.3. PPNEU.PHP – ANLAGE EINER NEUEN TÄTIGKEIT .....	104
6.8.4. PPEXPORT BIS PPEXPORT3.PHP – EXPORT DER DATEN ALS CSV .....	105
6.9. USERMANAGEMENT .....	106
6.9.1. USALLE.PHP – ANZEIGE ALLER USER DES SYSTEMS .....	107
6.9.2. USNEU.PHP – NEUANLAGE EINES BENUTZERS .....	107
6.9.3. USAENDERN.PHP – BENUTZER BEARBEITEN .....	108
<b>7. TESTUNG – BONENGL .....</b>	<b>110</b>
7.1. TESTSzenarien anhand der Userstories .....	110
7.2. INJECTIONS .....	110
7.3. UMGEHEN USERLOGINS .....	112
<b>8. VERWEISE/LITERATURVERZEICHNIS .....</b>	<b>113</b>
<b>9. ABBILDUNGSVERZEICHNIS .....</b>	<b>114</b>
<b>10. ANHANG .....</b>	<b>115</b>
10.1. ZEITAUFZEICHNUNG PRAXMARER .....	115
10.2. ZEITAUFZEICHNUNG BONENGL .....	117



10.3. TESTPROTOKOLL – BONENGL .....	119
10.3.1. FUNKTIONSTEST ANHAND USERSTORIES .....	119
10.3.1.1. KUNDENMODUL.....	120
10.3.1.2. PFLEGER.....	143
10.3.1.3. SACHWALTER UND ANGEHÖRIGE.....	158
10.3.1.4. PFLEGEPLAN .....	170
10.3.1.5. VERRECHNUNG.....	183
10.3.1.6. LEISTUNGEN.....	198
10.3.1.7. USER UND SICHERHEIT.....	213
10.3.2. UMGEHEN USERLOGINS .....	223
10.3.3. INJECTION-TESTS.....	234



## 1. Einleitung - Praxmarer

### 1.1. Vorwort

Die ehemalige Arbeitgeberin von Mario Praxmarer - eine in Niederösterreich niedergelassene eingetragene Unternehmerin im medizinischen Dienstleistungssektor - spezialisierte sich auf die Vermittlung von Pflegekräften an bedürftige Personen. Die Hauptgeschäftszweige waren 24h-Betreuung und die Qualitätssicherung dieser, sowie stundenweise Betreuung von Patienten.

Da die Kundenanzahl in den vorhergehenden Jahren derart gewachsen war, dass die Übersicht über die laufenden Geschäftsprozesse unmöglich weiterhin in Papierform gewährleistet werden konnte, wurde eine EDV-Lösung notwendig.

Diese Prozesse beinhalteten insbesondere das Management der Pflegepläne (welcher Pfleger betreut welchen Patienten?), die Verrechnung der Leistungen, sowie die Aufbereitung für die externe Buchhaltung,

Da eine Webapplikation für alle Mitarbeiter einfach, geräteunabhängig und ohne lokale Installation oder der Notwendigkeit der Kontrolle durch eine IT-Abteilung zugänglich ist, wurde entschieden, CareCRM über diesen Weg zu realisieren.

Als Arbeitnehmer konnte Herr Praxmarer Einblicke in die Geschäftsfälle dieser Branche werfen. Aufgrund der laufenden Ausbildung an der HTL Wien West war auch die praktische Umsetzung des CRMs eine durchführbare Aufgabe. Aufgrund des Umfangs der abzubildenden Prozesse wurde entschlossen, die Entwicklung dieser Applikation als Diplomprojekt einzureichen.



Es war geplant, ein für solche Aufträge übliches Pflichtenheft zu erstellen, um sämtliche Arbeiten im Einvernehmen mit dem Auftraggeber ohne spätere Missverständnisse verwirklichen zu können. Anlässlich persönlicher Gründe wurde die Zusammenarbeit zwischen Herrn Praxmarer und der Unternehmerin jedoch bereits in der Planungsphase des Diplomprojekts beendet, wodurch das Pflichtenheft nicht mehr notwendig für die Realisierung des Systems war.

Ebenso wurden durch die abrupte Beendigung der Kommunikation fehlende Details zu Geschäftsprozessen nachträglich von den Verfassern der Arbeit selbstständig vervollständigt und erweitert, um eine logisch zusammenhängende Prozesslandschaft zu kreieren.

Die in dieser Arbeit abgebildeten Geschäftsfälle basieren demnach auf reellen Prozessen eines existierenden Unternehmens, welche zum Zwecke der Fertigstellung logisch vervollständigt wurden.

## 1.2. Zuständigkeiten

Um Abhängigkeiten zwischen den Verfassern und damit mögliche Verzögerungen zu vermeiden, wurde versucht, die Arbeitsaufträge in geschlossene Pakete zu verpacken.

Abseits der Planungsphase konnte die zum größten Teil autarke Handlungsfähigkeit der Mitarbeiter gewährleistet werden. Regelmäßige Planungs- und Organisationsmeetings halfen dabei, den Überblick zu bewahren und die unmittelbar nächsten Ziele zu definieren und zu verwirklichen.



Während der Migration des Systems auf den Server sowie bei logischen Änderungen am Datenbanksystem zeigte sich die Zusammenarbeit der Projektpartner reibungslos. Die rege Kommunikation während der Testphase sorgte ebenso für einen schnellen Ablauf dieses Arbeitspaketes.

Folgende Aufgaben wurden den jeweiligen Mitarbeitern zugeordnet.

### **1.2.1. Mario Praxmarer**

Die Aufgaben von Herrn Praxmarer beinhalteten

- die Erstellung des Pflichtenhefts aufgrund des vorhandenen Lastenhefts in Zusammenarbeit mit dem Auftraggeber. Dies beinhaltete ein Wireframe-Konzept mit Mockup-Screenshots, um dem Auftraggeber ein Bild des Endprodukts zu vermitteln. Ebenso sollte sichergestellt werden, welche Funktionen nicht im Umfang der Lösung enthalten sind.
- die Erstellung des User Interface Designs. Beginnend bei Planung des Aufbaus, der Farbauswahl und der Entwicklung eines Logos über die Auswahl der zu verwendenden Werkzeuge/Frameworks bis hin zur Realisierung.
- die Programmierung der Logik, welche das Frontend mit der Datenbank verbindet. Hier waren die Planung der Module und deren Schnittstellen ein wichtiger Bestandteil, sowie die Auswahl von Bibliotheken, um wichtige Funktionen zu verwirklichen. Zuletzt die Implementierung aller Funktionen.



### 1.2.2. Bernhard Bonengl

Die Aufgaben von Herrn Bonengl beinhalteten

- die Planung des Datenbankkonzepts inkl. Diagramme und Schemas sowie die genaue Implementierung, um spätere Komplikationen bei den Schnittstellen zu vermeiden. Dies beinhaltete ebenfalls die laufende Pflege und Aktualisierung der Datenbank, wo dies notwendig war.
- Planung und Implementierung eines Backupkonzepts für die Datenbank auf einem eigenen Server, sodass diese auch bei einem teilweisen Ausfall weiterhin für den Endbenutzer zur Verfügung steht. Dies beinhaltete ebenfalls die Planung, die Auswahl der zu verwendenden Systeme sowie Lösungen und letztendlich deren Anwendung.
- Quality Assurance aufgrund der erstellten Userstories/Epics am Ende des Entwicklungsprozesses, um etwaige Fehler aufzuspüren und beseitigen zu können. Dies beinhaltete ebenfalls die Prüfung des Systems auf seine sicherheitsrelevanten Aspekte.



### 1.3. Anforderungen/Lastenheft

Die Anforderungen des Systems waren wie folgt:

Ein CRM-System, das von Mitarbeitern bedient werden kann, welche nicht viel Erfahrung mit ähnlichen Applikationen haben. Das Vermeiden von User-Fehlern hatte hohe Priorität.

Detaillierte Abbildungen der folgenden Geschäftsprozesse in Module.

#### Kunden

Kunden sind Patienten. Kundendaten müssen gespeichert und verwaltet werden können. Die notwendigen Personendaten sollten leicht einsehbar sein.

Kunden haben oftmals unterschiedliche Rechnungs- und Aufenthaltsadressen.

Wichtiger Kundenkontakt soll dem Akt in Form von schriftlichen Vermerken beigelegt und eingesehen werden können.

#### Pfleger

Pfleger sind Mitarbeiter. Der Sinn der Pfleger in diesem System ist die Abbildung der Relation zu den Patienten (Welcher Kunde wird von welchem Pfleger betreut). Daten, Adressen & Vermerke sollten ebenso wie bei Kunden aussehen.

#### Angehörige/Sachwalter

Kunden sind oftmals beschwaltet oder haben Angehörige, welche Entscheidungen für sie treffen. Diese Beziehungen müssen ebenfalls abgebildet sein. Die Daten der Sachwalter und Angehörigen sind gleich aufgebaut wie die der Kunden.



## Pflegeplan

Es soll ersichtlich sein, welcher Pfleger bei welchem Patienten tätig ist, war, oder sein wird. Diese Daten beinhalten wann der Pfleger begonnen hat, den Kunden zu betreuen und wann die Betreuung beendet wurde. Die Abbildung reicht tageweise, da die Pfleger i.d.R. mehrere Wochen tätig sind.

Ebenso soll abgebildet werden, bei welchen Kunden im aktuellen Monat noch ein Besuch zur Qualitätssicherung notwendig ist.

## Verrechnung

Leistungen wie Qualitätssicherung, 24h-Betreuung und stundenweise Betreuung sollen monatlich verrechnet werden können.

Die Rechnungen sollen aus dem System gedruckt bzw. als PDF exportiert werden können.

Ebenso sollen für die externe Buchhaltung CSV-Listen mit allen Rechnungsdaten exportiert werden können.

## Sicherheit

Da es sich um sensible Kundendaten, sowie um Daten des Unternehmens handelt, soll kein unbefugter Zugriff auf das System möglich sein.



## 1.4. Userstories/Epics

Userstories sind Abbildungen des Tagesgeschäfts in einzelnen Arbeitsaufträgen. Anhand dieser wird das System auf seine Sinnhaftigkeit überprüft.

### Kunden

- Ein Kunde wird angelegt und hat eine Rechnungsadresse, welche von der Betreuungsadresse abweicht.
- Ein Kunde wird angelegt. Es soll ein bestehender Sachwalter hinzugefügt werden.
- Dem Kunden wird ein Sachwalter hinzugefügt. Der Kunde hat ebenso einen Angehörigen.
- Der Kunde soll ab 18 Uhr nicht mehr angerufen werden. Dies soll im Kundenakt sofort ersichtlich sein.
- Der Kunde ist verstorben. Es muss hinterlegt werden, wann dies passiert ist.

### Pfleger

- Ein Pfleger wird angelegt.
- Ein Pfleger ändert seine Telefonnummer.
- Ein Pfleger wechselt von stundenweiser Betreuung in den Pool der 24h-Betreuer.
- Ein Pfleger beendet die Zusammenarbeit. Er soll nicht mehr in Pflegeplänen auftauchen.



## Sachwalter/Angehörige

- Ein Sachwalter betreut mehrere Patienten.
- Ein Sachwalter wird angelegt und einem Patienten zugeordnet.
- Ein Angehöriger ändert seine Adresse.
- Ein Angehöriger wird mit Namen und Adresse gesucht - und gefunden.

## Pflegeplan

- Es wird für einen Kunden ein zusätzlicher Hausbesuch zur Qualitätssicherung angelegt.
- Es wird eine Tätigkeit eines 24h-Pflegers über 3 Monate angelegt. Diese Leistung soll beim Kunden in allen 3 Monaten als aktiv aufscheinen.
- Ein angelegter Hausbesuch wurde fehlerhaft erstellt. Dieser soll gelöscht werden.
- Der Pflegeplan eines Kunden des aktuellen Monats wird angezeigt. Es sollen alle Pflegekräfte, welche dem Kunden zugeordnet und diesen Monat dort aktiv sind, angezeigt werden.

## Leistungen:

- Es sollen 5 Leistungen für das Jahr 2019 angelegt werden.
- Die Leistungen sollen einem Kunden zugeordnet werden.
- Diese Leistungen (nur die neu angelegten) sollen indexiert werden mit einem Index von 1,62% (= Faktor 1,0162) für 2020.
- Die neuen Leistungen sollen nun in 2020 aufscheinen mit den um 1,62% erhöhten Preisen.



- Beim Kunden sollen automatisch die neu indexierten Leistungen erscheinen.

## Verrechnung

- Es soll eine Rechnung für einen Kunden erstellt werden, mit welcher sämtliche in diesem Monat noch nicht verrechneten Leistungen verrechnet werden.
- Diese Leistungen sollten bei der nächsten Verrechnung als "bereits verrechnet bis [Datum]" angezeigt werden, um eine Doppelverrechnung zu verhindern.
- Eine Rechnung wurde fehlerhaft erstellt. Sie soll mit einem Klick gutgeschrieben werden.
- Es soll eine außerordentliche Leistung außerhalb des Katalogs für einen Kunden erstellt werden, welcher sich nicht im System befindet
- Eine Liste an Rechnungen für die Buchhaltung soll für diesen Monat exportiert werden und sämtliche Rechnungen inkl. Gutschriften enthalten

## User/Sicherheit

- Es soll ein neuer User angelegt werden (ohne Administratorrechte).
- Der User soll sich mit dem generierten Passwort erfolgreich anmelden können.
- Der User soll sich 3 Mal mit falschem Passwort anmelden und automatisch gesperrt werden.
- Nachdem ein Administrator ihn wieder freigeschaltet hat, soll der User sich wieder anmelden können.



- Der User soll versuchen, auf die Administratorfunktionen (lgneu.php, lgalle.php, lgindex.php, usalle.php, usneu.php, usaendern.php) zuzugreifen und einen Fehler erhalten.
- Nachdem ein Admin dem User Administratorrechte erteilt hat, soll der User versuchen, auf die Seiten zuzugreifen (sollte funktionieren).
- User soll dann von einem anderen Admin manuell gesperrt werden und keinen Zugang mehr zum System haben.
- Nachdem der User wieder aktiviert wurde, soll für ihn ein neues Passwort generiert werden, mit dem er sich einloggen kann.

## 1.5. Nicht-Ziele

Nicht-Ziele sind notwendig, um Missverständnissen bei der Auslieferung vorzubeugen.

Definitive Nicht-Ziele lauten wie folgt:

- Personalmanagement wie Verträge, Löhne, oder HR-Angelegenheiten sind nicht Bestandteil dieses Systems.
- Bezuglich Dienstplanung für Pfleger, Verfügbarkeiten, Kapazitätsplanung wird festgehalten, dass Pfleger mit diesem System nicht arbeiten sollen. Es dient lediglich der Erleichterung der Arbeit der Büroangestellten.



- Doppelte Buchführung ist nicht Teil der Funktionalität. Die Faktura erfolgt laut Vorgaben des Gesetzgebers, es soll jedoch keine Buchhaltung stattfinden. Die exportierten Listen beinhalten lediglich die Daten der Rechnungen, um diese der externen Buchhaltung zur Verfügung zu stellen.
- Ein großer Leistungskatalog bzw. Produktmanagement ist nicht Ziel der Anwendung. Der Leistungskatalog basiert auf Pauschalen. Diese können zwar in beliebiger Anzahl angelegt werden, das Management der Leistungen außerhalb der automatischen Indexierung findet jedoch nicht statt.  
Es gibt keine Kategorien oder Unterteilungen. Leistungen werden anhand der Preise und Bezeichnungen unterschieden. Sonderfälle müssen manuell gehandhabt werden.
- Umfangreiches User-Management ist ebenso nicht vorgesehen. Das System wird von einer Handvoll Mitarbeiter genutzt. Es gibt *User* und *Administratoren*. Außerhalb dieser Gruppen gibt es keine Möglichkeit der besonderen Rechtevergabe.
- DSGVO Konformität kann nicht garantiert werden. Die Vorschriften der DSGVO werden möglicherweise nicht eingehalten. Dies betrifft vor allem die automatische Löschung und Anonymisierung. Ebenso ist das Löschen von Daten nicht möglich. Dies kann nur über den direkten Zugang zur Datenbank erfolgen und ist mit enormen Risiken für die Funktionsfähigkeit des Systems verbunden.



- Langzeitige Nutzung des Systems ohne Wartung kann ebenfalls nicht gewährleistet werden. Das System muss regelmäßig gewartet werden, um Sicherheit und Sinnhaftigkeit zu gewährleisten. Das System unterstützt keine Änderungen der festgelegten Geschäftsprozesse. Die Administration beschränkt sich auf die Verwaltung von Usern und den Leistungskatalog.



## 2. Datenbankkonzept – Bonengl

In der Datenbank werden die Daten des CareCRM hinterlegt. Zudem können diese Daten ausgelesen, verändert und gelöscht werden. Als Technologie wurde MySQL verwendet. Die Datenbank setzt sich aus den Tabellen *Kunde*, *Sachwalter*, *Leistung*, *User*, *Vermerke*, *Adressen*, *Angehoerige*, *Rechnung*, *Pfleger*, *Pflegeplan*, *Bezug* und *Positionen* zusammen. Die Zusammensetzung dieser Tabellen können im folgenden Realtionenshema eingesehen werden.

### 2.1. Tabellen

#### Kunde

In der Tabelle Kunde werden Informationen über die Kunden abgespeichert. Diese Inforamtionen sind: der Vorname, der Nachname, die Anrede, der Titel, das Geburtsdatum, die Festnetznummer, die Mobilnummer und die Email-Adresse der Kunden. Weiters wird vermekert ob ein Kunde aktiv oder inaktiv ist, auf welche Art und Weise dieser seine Rechnung erhalten möchte und ob dieser verstorben ist.

Die Spalte *kd\_id* wird in der Tablle Kunde als Primärschlüssel verwendet und ist in den Tabellen *Adressen*, *Angehoerige*, *Bezug*, *Pflegeplan*, *Rechnung* und *Vermerke* als Fremdschlüssel hinterlegt. Dadurch lassen sich die Kunden den Datensätzen der jeweiligen Tabellen zuordnen.



## Sachwalter

In der Tabelle Sachwalter werden die Informationen über die Sachwalter bzw. die Angehörigen abgespeichert. Die Tabelle enthält die Spalten Anrede, den Titel, den Vor- und Nachnamen, das Geburtsdatum, die Festnetznummer, die Mobilnummer, die E-Mail-Adresse, die Art wie ein Sachwalter die Rechnung erhält und ob es sich bei einem Eintrag um einen Angehörigen oder einen Sachwalter handelt.

Der Primärschlüssel der Tabelle Sachwalter liegt in der Spalte *sw\_id* vor und ist in den Tabellen *Adressen*, *Angehoerige* und *Vermerke* als Fremdschlüssel vorhanden um die Datensätze der Tabelle Sachwalter den jeweiligen Tabellen zuordnen zu können.

## Angehoerige

Die Tabelle Angehoerige dient dazu die m:n-Beziehung von Kunde und Sachwalter abzubilden. In der Tabelle sind die Primärschlüssel der Tabellen *Kunde* und *Sachwalter* als Fremdschlüssel hinterlegt. Zusätzlich besitzt sie einen Statuswert, welcher angibt, ob ein Angehöriger oder Sachwalter aktiv ist.

## User

Unter der Tabelle User werden die Zugangsdaten für die Benutzer und deren Rechte im CareCRM-System gespeichert. Diese Daten bestehen aus dem Nutzer-oder Anmeldenamen, dem errechneten Hash-Wertes des Passworts, der E-Mail-Adresse, dem Vor-und Nachnamen, ob das Konto des Users aktiv ist und ob dieser Administrator-Rechte besitzt. Der Primärschlüssel *us\_id* liegt in den Tabellen *Vermerke* und *Rechnungen* als Fremdschlüssel damit diese nachvollziehbar sind.



## Leistung

Um Leistungen verwalten zu können, wurde die Tabelle *Leistung* erstellt. In dieser werden die folgenden Werte erfasst: Leistungsbezeichnungen, Leistungseinheiten, Rabttgrenzen, Einzeltarife, Mengentarife und das entsprechende Jahr, in dem die Leistung zu tragen kommt. Die Leistungseinheiten sind ENUM-Werte, welche die folgende Werte akzeptiert: *h* (Stunde), *d* (Tag), *w* (Woche), *m* (Monat), *s* (Stück), *b* (Besuch) und *e* (Einheit).

Der Primärschlüssel *lg\_id* ist in den Tabellen *Bezug* und *Positionen* als Fremdschlüssel hinterlegt um eine Verbindung zwischen Kunden bzw. Rechnungen mit der Tabelle *Leistung* zu ermöglichen.

## Adressen

Um die Adressen der Kunden, Sachwalter und Pfleger abzuspeichern, wurde die Tabelle *Adressen* erstellt. Die Tabelle *Adressen* besteht aus Statuswert, Typ, einer Zeichenkette für die Straße, einer Straßennummer, einer Stiegennummer, einem Wert für das Stockwerk, einer Türnummer und der Postleitzahl.

Die Primärschlüssel der Tabellen *Kunden*, *Pfleger* und *Sachwalter* sind in der Tabelle als Fremdschlüssel enthalten. Somit können die gespeicherten Adressen dem jeweiligen Kunden, Pflegekräften und Sachwaltern zugeordnet werden.

Hinzuzufügen ist, dass der Typ der Adresse ein ENUM-Wert ist, welcher die folgenden Werte zulässt: *b* (Betreuungsadresse), *r* (Rechnungsadresse) und *rb* (Rechnungs- und Betreuungsadresse).

## PLZ

Die Postleitzahlen wurden in einer eigenen Tabelle abgespeichert



## Rechnungen

Damit Rechnungen erstellt werden können, wurde eine Tabelle *Rechnung* angefertigt. In dieser Tabelle sind die folgenden Werte enthalten: das Rechnungsjahr, die Rechnungsnummer, die Rechnungsadresse, der Beginn und das Ende des Rechnungszeitraums, das Ausstellungsdatum der Rechnung, Art der Rechnung, Rechnungsstatus und Status des Rechnungsdrucks. Der Primärschlüssel *rg\_id* liegt in den Tabllen *Positionen* und *Vermerke* vor und die Primärschlüssel der Tabellen *Kunde*, *Pfleger*, *Adressen* und *User* sind in der Table *Rechnung* als Fremdschlüssel enthalten um diese miteinander zu verbinden.

Die Art der Rechnung und der Status des Rechnungsdrucks sind ENUM-Werte.

In der Spalte *rg\_art* sind folgende Werte zulässig: *r* (Rechnung), *m1* (1. Mahnung), *m2* (2. Mahnung) und *g* (Gutschrift).

Die Werte *m1* und *m2* sind zwar in der Datenstruktur vorhanden, der Mahnlauf ist aber kein Bestandteil des endgültigen Systems.

Der Status des Rechnungsdrucks lässt folgende Werte zu: *n* (nicht gedruckt), *p* (printed = gedruckt), *s* (sent = bereits versandt).

## Pflegeplan

In der Tabelle Pflegeplan finden sich die folgenden Spalten: der Beginn einer Pflegeleistung, das Ende einer Pflegeleistung und für welchen Kunden und Pfleger diese Tätigkeit gilt. In dieser Tabelle sind die Primärschlüssel der Tabellen *Kunde* und *Pfleger* als Fremdschlüssel angelegt um die Tabelle Pflegeplan mit diesen Tabellen verbinden zu können.



## Pfleger

Die Tabelle Pfleger enthält die Informationen zu den Pflegern. Diese setzen sich aus: Anrede, Vornamen, Nachnamen, Festnetznummer, Mobilnummer, E-Mail-Adresse, Sozialversicherungsnummer, Verwendungszweck des Mitarbeiters, Mädchennamen, Geburtsort, Geburtsstaat, Familienstand, Vornamen der Mutter, Vornamen des Vaters, Nationsangehörigkeit, Passnummer, Passausstellungsdatum, Behörde an welcher der Pass ausgestellt worden war, Staat der Passbehörde, Status des Innlandsaufenthalt, Namen für den Bankkontoinhaber, dem IBAN des Bankkontos, dem BIC des Bankkontos, Statuswert ob der Pfleger arbeitet, Datum des Arbeitsbeginns und Datum des Arbeitsendes. Der Primärschlüssel *pg\_id* ist in den Tabellen *Adressen*, *Pflegeplan*, *Rechnung* und *Vermerke* als Fremdschlüssel enthalten um diese miteinander in Verbindung setzen zu können.

Der Zweck des Pflegers ist ein ENUM-Wert welcher die folgenden Werte zulässt: *sw* (Stundenweise Betreuung), *24h* (24h-Betreuung), *dgks* (Diplomierte Gesundheits- und Krankenschwester bzw. Pfleger)

## 2.2. Relationenschema

### Kunde

```
(kd_id (int 6); kd_anrede(enum("herr", "frau", "Damen & Herren"));  
kd_titel(enum("Ing.", 'Dipl.-Ing.', 'Mag.', 'Dr.')); kd_vorname(varchar 64);  
kd_nachname(varchar 64); kd_gebdatum(date); kd_tel1(varchar 32);  
kd_tel2(varchar 32); kd_mail(varchar 64); kd_rg_art(enum("Post", "Email")); kd_aktiv  
(tinyint 1); kd_beginn(date); kd_ende(date); kd_ableben(tinyint 1))
```



## Sachwalter

(sw\_id (int 6); sw\_anrede(enum("Frau','Herr','Damen und Herren"));  
sw\_titel(enum("Ing.','Dipl.-Ing.','Mag.','Dr.")); sw\_vorname(varchar 64);  
sw\_nachname(varchar 64); sw\_gebdatum (date); sw\_tel1(varchar 32);  
sw\_tel2(varchar 32); sw\_mail (varchar 64); sw\_rg\_art (enum("Post" "Email"));  
typ(enum "sw", "ag") )

## Angehoerige

(sw\_id (int 6); kd\_id(int 6); ag\_aktiv(int 1))

Die Tabelle Angehoerige dient dazu die m:n-eziehung zwischen *Kunde* und *Sachwalter* zu abzubilden.

## User

(us\_id (int 6); us\_name(varchar 8); us\_pw (varchar 40); us\_mail (varchar 64);  
us\_vorname (varchar 64); us\_nachname (varchar 64); us\_status (tinyint 1);  
us\_admin(tinyint 1))

## Leistung

(lg\_id (int 6); lg\_bezeichnung (varchar 64); lg\_einheit (enum ("h', 'd', 'w', 'm', 's', 'b', 'e' ")); lg\_rabattgrenze (int 4); lg\_einzeltarif (float 12.2); lg\_mengentarif (float 12.2);  
lg\_jahr (year 4))

## Bezug

(kd\_id (int 6); lg\_id (int 6))

Dient dazu die m:n-Beziehungen zwischen *Kunde* und *Leistung* darzustellen.

## Adressen

(ad\_id (int 6); kd\_id (int 6); sw\_id (int 6); pg\_id (int 6); ad\_aktiv (tinyint 1); ad\_typ  
(enum(" 'r', 'b', 'rb' ")); ad\_strasse (varchar 128); ad\_nr (varchar 4); ad\_stiege  
(varchar 4); ad\_stock (varchar 3); ad\_tuer (var 4); ad\_plz (int 6))



## PLZ

(plz (int 4); ort (varchar 128))

## Pflegeplan

(pp\_id (int 6); kd\_id (int 6); pg\_id (int 6); pp\_beginn (date); pp\_ende (date); pp\_verrechnung (tinyint 1); pp\_verrechnet (tinyint 1))

## Vermerke

(ve\_id (int 6); kd\_id (int 6); sw\_id (int 6); pg\_id (int 6); rg\_id (int 6); us\_id (int 6) ve\_art (enum "Information", "Hausbesuch", "Besorgung", "Telefonat", "Datei", "System", "Aenderung"); ve\_flag (enum "w", "n", "i", "1"); ve\_text (longtext); ve\_datum (date); ve\_datei (varchar 64))

## Rechnung

(rg\_id (int 6); kd\_id (int 6); pg\_id (int 6); ad\_id (int 6); us\_id (int 6); rg\_jahr (year 4); rg\_nr (int 6); rg\_anschrift (text); rg\_zeitraum\_start (date); rg\_zeitraum\_ende (date); rg\_datum (date); rg\_art (enum "r", "m1", "m2"); rg\_bezahlt (tinyint 1); rg\_druck (enum "n", "p", "s"))

## Pfleger

(pg\_id (int 6); pg\_anrede (enum "herr", "frau"); pg\_vorname (varchar 64); pg\_nachname (varchar 64); pg\_gebdatum (date); pg\_tel1 (varchar 32); pg\_tel2 (varchar 32); pg\_mail (varchar 64); pg\_svnr (int 10); pg\_art (enum "sw", "24h", "dgks"); pg\_zmr (int 12); pg\_mname (varchar 64); pg\_gebort (varchar 64); pg\_gebstaat (varchar 64); pg\_famstand (enum ("Ledig", "Verheiratet", "Partnerschaft", "Geschieden", "Verwittwert", "aufgelöste Partnerschaft", "hinterbliebener Partner")); pg\_vnmutter (varchar 64); pg\_vnvater (varchar 64); pg\_nation (varchar 64); pg\_pass\_nr (varchar 32); pg\_pass\_datum (date); pg\_pass\_behoerde (varchar 64); pg\_pass\_behstaat (varchar 64); pg\_aufenthalt (tinyint 1); pg\_kto\_inhaber (varchar 128); pg\_kto\_ibanc (varchar 20); pg\_kto\_bic (varchar 12); pg\_aktiv (tinyint 1); pg\_beginn (date); pg\_ende (date);)



## Positionen

(po\_id (int 6); lg\_id (int 6); rg\_id (int 6); po\_nr (int 2); po\_anzahl(int 4))

Diese Tabelle bildet die Grundlage der m:n-Beziehung zwischen *Rechnung* und *Leistung* sowie für die Berechnung der Rechnungsbeträge.

## 2.3. Herausforderungen und Lösungen

Es mussten sehr früh einige Entscheidungen zum Aufbau der Datenbankstruktur getroffen werden, welche während der Entwicklung nicht mehr geändert werden konnten, da dies einen enormen Arbeitsaufwand bedeutet hätte.

Eine Herausforderung war, die korrekte Atomarität der Tabellen herzustellen, sodass gleichzeitig die effiziente Ablage der Daten sichergestellt aber auch Konflikte verhindert werden konnten.

Ein Beispiel hierfür sind die Tabellen *Adressen* und *Rechnung*. In *Adressen* werden sämtliche Adressen der Kunden, Pfleger und Sachwalter bzw. Angehörigen dargestellt. Da jedoch Adressen grundsätzlich als änderbare Tupel eingerichtet wurden, wurde entschieden, die Rechnungsadressen als eigenes Feld in die *Rechnung*-Tabelle zu erstellen. Die Datensätze kommen somit doppelt in der Datenbank vor, da sich die Rechnungsdaten jedoch nicht ändern dürfen, musste dies so gehandhabt werden.



Ebenso musste entschieden werden, ob Angehörige und Sachwalter auf Datenbankebene in getrennten oder in derselben Tabelle gehandhabt werden. Da die Daten dieser Tupel grundsätzlich gleich waren, fiel die Entscheidung, keine separate Tabelle anzulegen, sondern die Art des Sachwalters in der Spalte *sw\_typ* mit dem ENUM-Wert *sw* (Sachwaler) oder *ag* (Angehöriger) festzuhalten.

Eine große Herausforderung war ebenfalls die Verbindung von Rechnung zu Leistungen. Der erste Gedanke, pro Rechnung *n* Einträge in *Positionen* einzurichten, in welchen die Leistungen 1:1 übernommen wurden, wurde verworfen. Da Leistungen nicht änder- oder lösbar sind, stand der klassischen Implementierung einer m:n-Beziehung nichts im Wege. In den Positionen musste dann jedoch die Anzahl sowie die Nummer der Position auf der Rechnung hinzugefügt werden. Hier war ebenso zu beachten, wie ein Rechnungsbetrag errechnet werden soll. Es wurde entschieden, keine Preise oder Summen in der Tabelle *Rechnung* abzubilden. Der Gesamtpreis von einzelnen Positionen und damit der gesamten Rechnung wird anhand der Anzahl der Leistungen in der Tabelle *Positionen* und der Preise der entsprechenden Leistung in *Leistung* durch das Frontend errechnet.

Eine weitere Herausforderung waren die zum Zeitpunkt der Beendigung der Zusammenarbeit mit dem Kunden noch nicht ausgearbeiteten Geschäftsfälle und -prozesse. Das Modell der Datenbank enthält z.B. in der Tabelle *Pfleger* viele Spalten, welche in der endgültigen Version nicht zum Einsatz kommen. Der Grund hierfür ist, dass Daten wie z.B. eine ZMR-Zahl (Zentrales Melderegister) für den Funktionsumfang der Applikation keine Verwendung finden und grundsätzlich nicht sollten, da dies mit dem Nicht-Ziel der HR-Angelegenheiten interferiert. Die gegebenen Spalten wurden in den Tabellen belassen, jedoch mit *NULL* als Standardwert implementiert.



## 2.4. Realisierung der Datenbank

Die Datenbank wurde mittels *phpmyAdmin* anhand des Entity-Relationship-Diagramms und des Relationenschemas realisiert und bearbeitet. Jedoch mussten die Fremdschlüssel und Constraints am Server direkt manuell über die Befehlszeile angelegt werden, da es nicht möglich war die Tabellen zu verändern, weil diese durch Fremdschlüssel und Constraints verbunden waren. Dies war nötig um die *Storage-Engine* der Tabellen auf die *NDB Cluster-Engine* zu ändern. Wie hier verdeutlicht:

```
ALTER TABLE plz ENGINE = NDBCLUSTER;
```



### 3. Serveraufbau/-konfiguration – Bonengl

Um dem Endnutzer Datenbanken und das Webinterface zur Verfügung stellen zu können, wurde ein Rechnernetzwerk aus Servern notwendig. Diese werden mit der Software CentOS 7 betrieben.

#### 3.1. Auswahl Serverumgebung

Der Entscheidung CentOS 7 als Betriebssystem zu nutzen, lag zu Grunde, dass es sich bei CentOS um ein Enterprise-Betriebssystem handelt, welches eine hohe Stabilität aufweist und dennoch kostenlos zu erhalten ist. Außerdem erhält das System regelmäßige Aktualisierungen durch seine Entwickler und die Community.

Eine Kopie der ISO-Datei wurde heruntergeladen und die Sha256 Checksumme verglichen um sicherzustellen, dass die heruntergeladene Datei nicht korrumptiert wurde. Die Datei ermöglichte es, einen virtuellen Server in einer Virtualisierungssoftware zu erstellen.

#### 3.2. Aufsetzen des Servers

Zum Bearbeiten des Server-Betriebssystems wurde eine virtuelle Maschine unter der Virtualisierungssoftware *Oracle VM Virtual Box* erstellt und die standardmäßige Installation des Images durchgeführt. Der dadurch generierte virtuelle Server wurde grundlegender Konfiguration unterzogen.



### 3.3. Virtualisierungssoftware

Als Virtualisierungssoftware wurde *Oracle VM Virtual Box* verwendet. Diese ermöglicht es, einen virtuellen Server zu generieren und macht somit die Anschaffung von zusätzlicher Hardware obsolet. Zudem können mehrere Maschinen simultan betrieben werden, was das Konfigurieren und Testen der Systeme erheblich vereinfacht.

## 3.4. Grundlegende Konfiguration der Server-Images

### 3.4.1. Netzwerkschnittstelle

Im ersten Schritt - damit eine Netzwerkverbindung aufgebaut werden konnte - wurde die Konfigurationsdatei des Netzwerkinterfaces *enp0s3* mit dem Kommando *vi /etc/sysconfig/network-scripts/ifcfg-enp0s3* verändert. Das besagte Dokument wurde so verändert, dass das System neugestartet und das Netzwerkinterface nach dem Neustart aktiv ist. Damit die gesetzten Netzwerkeinstellungen überprüfen werden konnten, musste erst das Paket *net-tools* heruntergeladen und installiert werden. Weitere Änderungen der Netzwerkschnittstellen wurden individuell an den Servern vorgenommen und werden in deren Erläuterung behandelt.



### 3.4.2. Firewall

Um Komplikationen mit aktiven Firewalls zu vermeiden, wurden diese zur Gänze deaktiviert. Die Eingabe `vi /etc/selinux/config` ermöglichte Zugriff auf das Konfigurationsdokument der Selinux Firewall, dort mussten die Zeilen `SELINUX=enforcing` zu `SELINUX=disabled` geändert werden, um die Firewall zu deaktivieren. Um das Sicherungssystem `firewalld` auszuschalten, wurden die Kommandos `systemctl stop firewalld`, `systemctl disable firewalld` und `systemctl mask --now firewalld` ausgeführt.

Mit der Eingabe `systemctl stop firewalld` wurde die aktive Firewall ausgeschaltet und mit `systemctl disable firewalld` permanent deaktiviert. Durch `systemctl mask --now firewalld` wurde verhindert, dass die Firewall nicht durch andere Programme erneut gestartet werden konnte. Danach wurde das System mit dem Kommando `init 6` neu gestartet und überprüft, ob diese Änderungen am System erfolgreich waren. Mit `sestatus` wurde der Status von `Selinux` überprüft und um den Status des Sicherheitssystems `firewalld` zu prüfen wurde das Kommando `firewall-cmd –state` ausgeführt.

Die weitere Konfiguration der Server wird in den individuellen Beschreibungen dieser behandelt.



## 4. Ausfallsicherheit/Redundanz – Bonengl

Um das System gegen einen Serverausfall der Datenbankserver abzusichern und dem Endnutzer die Funktionen der Datenbank des CareCRM weiterhin zur Verfügung zu stellen, wurde ein sogenannter *MySQL-NDB-Cluster* implementiert. Dieser besteht aus einem Management-Server, zwei Dataservern und drei Datenbank-Servern. Die Datenbank-Server sind mit einem Apache-Webserver verbunden, von welchem die Webapplikation zur Verfügung stellt wird.

### 4.1. Clusterlösung

Der MySQL NDB Cluster ist ein Zusammenschluss von mehreren Datenbankservern, welche ihre Daten untereinander austauschen um eine gewisse Ausfallsicherheit zu gewährleisten. Da MySQL NDB Cluster leicht erweitert werden können und noch viele weitere Vorteile bieten, wurde entschieden, diese Lösung zu implementieren.

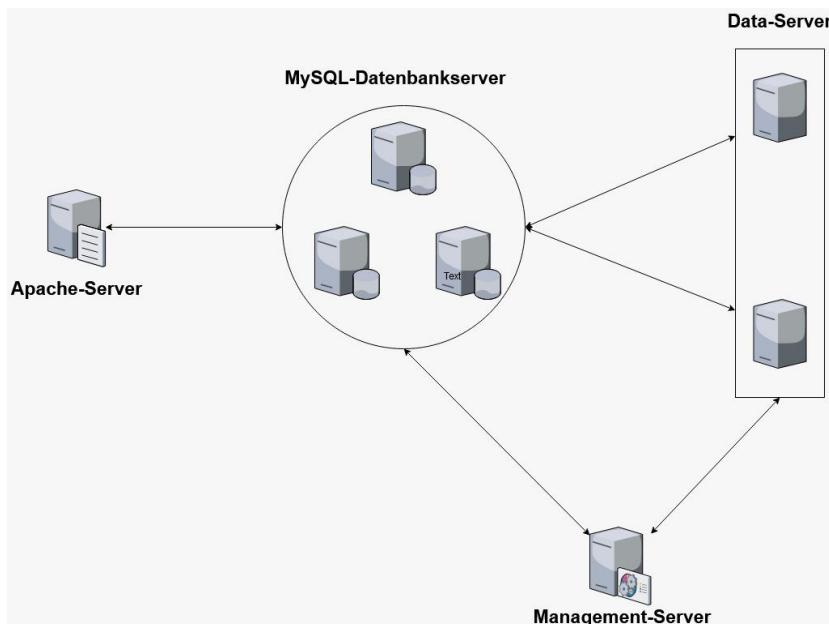


Abb. 4.1.1 - Konzept Clusterlösung



## 4.2. Individuelle Serverkonfiguration

### 4.2.1. Management-Server

Der Management-Server des Clusters dient dazu, den MySQL NDB Cluster zu konfigurieren, zu starten und zur Überwachung des gesamten Systems. Das Dokument *ifcfg-enp0s3.cnf* unter dem Pfad */etc/sysconfig/network-scripts/* wurde so konfiguriert, dass der Server die statische IP-Adresse 192.168.0.190 erhielt. Um den Cluster konfigurieren zu können, musste zuerst das Konfigurations-Dokument *config.ini* erstellt werden. Dieses Konfigurations-Dokument des Management-Servers wurde unter dem Pfad */var/lib/mysql-cluster* am Management-Server erzeugt.

Dieses Dokument beinhaltet unter anderem die IP-Adressen der zum Cluster gehörenden Server.

Mit dem Kommando *ndb\_mgmd --config-file=/var/lib/mysql-cluster/config.ini* kann der Management-Server gestartet werden.

### 4.2.2. Data-Server

Die Data-Server ermöglichen das Austauschen von Daten zwischen den Datenbank-Servern und halten diese somit synchron. Das Dokument *ifcfg-enp0s3.cnf* unter dem Pfad */etc/sysconfig/network-scripts/* wurde so konfiguriert, dass die Server die statischen IP-Adressen 192.168.0.200 (data\_nd1) und 192.168.0.126(data\_nd2) erhielten. Unter dem Pfad */etc/my.cnf* liegt das Konfigurations-Dokument des jeweiligen Servers auf. In diesem wird die IP-Adresse des Management-Server angeben um eine Kommunikation zu ermöglichen. Damit die Data-Server gestartet werden konnten, wurde der Befehl *ndbd* auf allen Data-Servern ausgeführt.



#### 4.2.3. Datenbank-Server

Auf den Datenbank-Servern liegen die Datenbanken, welche zum Abspeichern und Bereitstellen der Daten des CareCRM-Systems verantwortlich sind.

Das Konfigurations-Dokument der Datenbank-Server befindet sich jeweils unter `/etc/my.cnf`. In diesem Dokument wurde die IP-Adresse des Management-Servers angegeben und das Standardverwaltungssystem für die MySQL-Datenbank eingetragen. Die Netzwerkschnittstelle `enp0s3` wurde auch auf jedem der Datenbankserver im Dokument `ifcfg-enp0s3.cnf` unter dem Pfad `/etc/sysconfig/network-scripts` so verändert, dass die statischen IP-Adressen der einzelnen Datenbankserver wie folgt lauten:

Mysql\_DB\_ND1: IPADDR=192.168.0.148

Mysql\_DB\_ND2: IPADDR=192.168.0.128 .

Mysql\_DB\_ND3: IPADDR=192.168.0.158

#### 4.2.4. Apache-Webserver

Der Apache-Webserver stellt die Schnittstelle zwischen Applikation und Datenbank dar. Der Server wurde so konfiguriert, dass *Apache2* auf diesem läuft.

Das Dokument `ifcfg-enp0s3` unter dem Pfad `/etc/sysconfig/network-scripts/` wurde so konfiguriert, dass der Server die statische IP-Adresse 192.168.0.167 erhielt.

Auf dem Webserver liegen die PHP-Dateien der Applikation auf und ermöglichen die Funktion des CareCRM.



## 5. User Interface und Design - Praxmarer

Das User Interface oder Webinterface ist die grafische Benutzeroberfläche. Das Design beinhaltet Farben, das Logo und das allgemeine Layout der Applikation bis hin zur Anordnung einzelner Elemente.

### 5.1. Designentscheidungen

Da das System ausschließlich für die Benutzung innerhalb eines Unternehmens bestimmt ist, wurde das Hauptaugenmerk auf effiziente Verwendbarkeit des Systems und der Vermeidung von Verwirrung durch einen unerfahrenen Benutzer gelegt.



### 5.1.1. Farben

Als Thema wurden Blautöne gewählt, da diese Farbe angenehm für die Augen, sämtliche Texte darauf gut lesbar sind und es als kühle Farbe die Aufmerksamkeit des Mitarbeiters erhält.

Es wurde eine kurze Farbstudie mit drei Farbschemen mit je drei Farben durchgeführt: Eine Hintergrundfarbe, eine Primär- und Textfarbe, und eine Sekundärfarbe. Aufgrund persönlicher Präferenzen wurde aus dieser Auswahl das zu verwendende Farbschema gewählt.

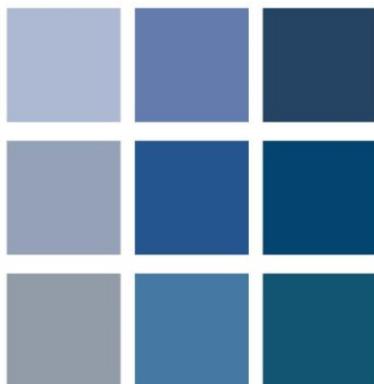


Abb. 5.1.1 - Farbstudie



Abb. 5.1.2 – Ausgewähltes Farbschema



### 5.1.2. Logo

Das Logo des Systems wurde in Anbetracht der möglichen öffentlichen Verbreitung entwickelt. Es sollte die Wortmarke *CareCRM* darstellen sowie die Verbindung mit dem medizinischen Sektor zeigen.

Auch hier wurden einige Entwürfe unter Verwendung der bestehenden Farbschemen kreiert und eine endgültige Auswahl getroffen.

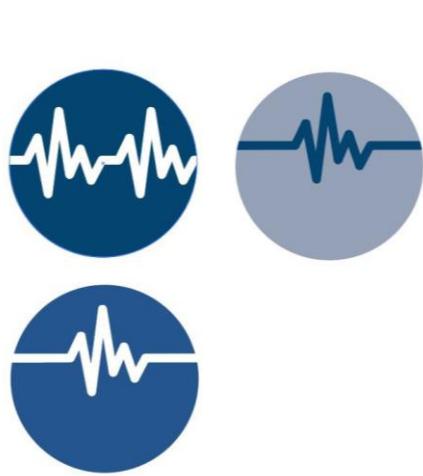


Abb. 5.1.3 - Logoentwürfe



Abb. 5.1.4 – endgültiges CareCRM Logo



## 5.2. Auswahlprozess Framework

Vor allem in Bezug auf die Anforderung der gleichzeitigen Nutzung durch mobile Geräte und Desktop- bzw. Laptopcomputer wurde die Entscheidung gegen den immensen Aufwand der Implementierung eigener CSS-Klassen und Javascript-Libraries getroffen. Es wurde daher beschlossen, Frameworks, d.h. Sammlungen bestehender Bibliotheken aus dritter Hand, zu integrieren

Da eine gewisse Flexibilität gewünscht und bereits Erfahrung damit vorhanden war, fiel die Entscheidung auf Bootstrap.

Bootstrap ist eine Sammlung an CSS-Klassen sowie einigen Javascript-Funktionalitäten, um die Usability des Systems zu erhöhen. Sie können einfach implementiert und auf die eigenen Anforderungen zugeschnitten werden.

Die Eigenschaften diverser Elemente werden definiert, indem die CSS-Klasse über das `class`-Attribut definiert wird. Im folgenden Beispiel findet sich ein `div`-Element, welches auf kleinen Bildschirmen als Spalte mit einer Breite von 6 (von 12) und auf großen Bildschirmen mit einer Breite von 4 angezeigt wird.

```
<div class="col-sm-6 col-lg-4">
```

Dies dient dazu, dass auf kleineren und mobilen Bildschirmen ein Umbruch stattfindet und nicht horizontal gescrollt werden muss.



Da HTML 5 bereits viele Möglichkeiten der Formularüberprüfung beinhaltet und diese Funktionen ausreichend sind, um den Benutzer auf Fehlende Eingaben aufmerksam zu machen, wurde auf weitere Javascript-Frameworks wie JQuery oder VUE verzichtet.

Allgemeine Daten

Anrede:	<input type="text" value="Anrede"/>
Titel:	<input type="text" value="Titel"/>
Vorname	<input type="text" value="Vorname"/>
Nachname	<input type="text"/> ! Füllen Sie dieses Feld aus.
Geburtsdatum	<input type="text" value="tt.mm.jjjj"/> <input type="button" value=""/>

Abb. 5.2.1 - HTML 5 Form Validierung in Chrome



### 5.3. Arbeitsbereich

Im Arbeitsbereich wurde großer Wert darauf gelegt, Verwirrung zu vermeiden und u.a. Tabellenreihen mit abwechselnden Farben zu gestalten, außerordentliche Einträge hervorzuheben, oder aufrufbare Funktionen mit leicht sichtbaren und farblich gekennzeichneten Buttons auszuführen.

Anzeigen: <input type="checkbox"/> beendete <input checked="" type="checkbox"/> verstorbene						
Anrede	Nachname	Vorname	PLZ	Adresse	Telefon	Details
Herr	Musterman	Mario	1130	Anton-Langer 66/3	+43 660 5634	Details
Herr	Kleinmann	Peter	1110	Luise Montag Gasse 1	01 3434	Details
Herr	Rüdiger	Richard	1110	Teststraße 69/2	01 3434	Details
Herr	Sedlacek	Markus	1110	Grillgasse 66/2/6	01 567 89 01	Details
Herr	Österreicher	Markus	1010	Zweigeltstraße 12/7	01 234 3456	Details
Frau	Kraft	Elena	1010	Markusplatz 1	01 987 34 44	Details
Herr	Berger	Willibald	1030	Hansgasse 12/2/7	01 234 55 66	Details
Herr	Zimmer	Peter	1200	Mariusplatz 12/1/2/4	01 34 3445	Details
Herr	Egger	Helmut	1200	Vogelgasse 12/1/2/4	01 56 444 5	Details
Herr	Schindler	Herbert	8020	Kleingasse 12		Details
Herr	Müller	Bernhard	1110	Hausbrandstraße 12		Details

Abb. 5.3.1 - Tabelle "Kunden", verstorbene werden rot unterlegt

#### Tätigkeit Nr. 7

Kunde:  zum Kundenakt

Pfleger:  zum Pflegerakt

Beginn:

Ende:  offen

Abb. 5.3.2 - Buttons mit Funktionszuweisung



## 5.4. Navigation

Für die Navigations- und Informationsleiste wurde ein minimalistisches Design gewählt, um eine Reizüberflutung bei der Nutzung zu vermeiden. Die Navigation wurde klassisch am linken Rand des Fensters platziert, da dies für ähnliche Systeme der Standard ist. Für die mobile Version des Systems wurde die übliche Dropdown-Navigation realisiert.



Abb. 5.4.1 - Navigation im Desktop-Modus



Abb. 5.4.2 - Navigation im Mobil-Modus



In bestimmten Bereichen wurde überdies eine zusätzliche Navigation in Form von Reitern implementiert, um die zusammenhängenden Funktionen klar von der allgemeinen Navigation abzuheben.

A screenshot of a web-based application interface for managing customer accounts. At the top, there is a horizontal navigation bar with five tabs: "Allgemein" (highlighted in light blue), "Vermerke", "Pflegeplan", "Tarife", and "Rechnungen". Below this, a section titled "Allgemeine Daten" is displayed. Under this section, there are two rows of information. The first row shows "Anrede:" followed by "Herr" and "Besachw". The second row shows "Titel:" followed by "Dr." and "Sachwal".

Allgemein	Vermerke	Pflegeplan	Tarife	Rechnungen
Allgemeine Daten				
Anrede:	Herr	Besachw		
Titel:	Dr.	Sachwal		

Abb. 5.4.3 - Zusätzliche Reiter, um die Funktionen des Kundenakts aufzuteilen

## 5.5. CSS

Da Bootstrap nicht auf sämtliche der gewünschten Funktionalitäten des Layouts eingeht, war die Implementierung von eigenem CSS notwendig. Im Speziellen die Anzeige der richtigen Schriftgröße auf diversen Bildschirmgrößen war hierfür ein Grund.



Hierzu wurden die Eigenschaften der verwendeten Klassen überschrieben bzw. modifiziert und an die jeweilige Bildschirmbreite angepasst, wie im folgenden Beispiel für Bildschirme ab 567 Pixel breite zu sehen ist.

```
@media (min-width: 567px) {  
  .container {  
    max-width: 0px;  
    width: 100%;  
    font-size: 4vw;  
  }  
  
  .container .col-form-label {  
    max-width: 1400px;  
    width: 500px;  
    font-size: 4vw;  
  }  
  
  .container .prxm {  
    font-size: 5vw;  
  }  
  
  .container .btn {  
    font-size: 4vw;  
  }  
  
  .container input {  
    font-size: 3vw;  
  }  
  
  .container select {  
    font-size: 3vw;  
  }  
  
  .navbar-light .navbar-toggler-icon{  
    padding: 8vw !important;  
  }  
}
```



## 5.6. Javascript-Lösungen

Javascript ist eine Programmiersprache, welche vom Browser des Endnutzers interpretiert und ausgeführt wird, ohne dass vom Server neue Daten angefordert werden müssen. Mit Javascript sind daher Echtzeit-Änderungen an der Ansicht des Systems während der Nutzung möglich.

Obwohl die Entscheidung bewusst gegen visuelle Effekte getroffen wurde, musste für die Realisierung bestimmter Teile des User Interfaces Javascript genutzt werden. Diese Funktionen erhöhen die Benutzerfreundlichkeit der Oberfläche und verringern damit die Fehleranfälligkeit durch den Benutzer.

Sämtliche Javascript-Funktionen wurden für bessere Übersicht in einer eigenen Bibliothek namens *functions.js* zusammengefasst. Nachfolgend finden sich Auszüge und Beschreibungen aus dieser Sammlung.



- Ein bestimmter Bereich eines Formulars wird erst dann angezeigt, wenn eine bestimmte Auswahl getroffen wurde. Dies ermöglicht die Informationsflut zu minimieren, wenn der gegenständliche Bereich für den aktuellen Arbeitsschritt nicht notwendig ist.

Beispiel: Kundendaten werden bearbeitet aber die Adresseingabe bleibt verborgen, solange der Benutzer die Adresse nicht ändern möchte.

```
function toggleField(input, id) {
    var x = document.getElementById(id);
    var y = document.getElementById(input);

    if(y.type == "checkbox")
    {
        if (y.checked)
        {
            x.style.display = "";
        }
        else
        {
            x.style.display = "none";
        }
    }

    if(y.type == "select-one")
    {
        if (y.options[y.selectedIndex].value == 0)
        {
            x.style.display = "none";
        }
        else
        {
            x.style.display = "";
        }
    }
}
```



- Die Ausgabe des Titels der aktuellen Seite ist wichtig, da dieser für die sogenannten Breadcrumbs notwendig ist. Der User weiß dadurch, wo er sich befindet, während der Reiter der Seite jedoch sofort als CareCRM sichtbar ist. Diese Funktion wurde jedoch stillgelegt, da eine Lösung via PHP verwirklicht wurde.

```
function writeTitle()
{
    var seite = document.title;
    var titel = document.title.replace(" - CareCRM", "");
    document.writeln(titel);
}
```

- Bestimmte Eingaben erfolgen über einen Schieberegler am effizientesten. Damit der derzeit ausgewählte Wert des Reglers auch in Echtzeit angezeigt werden kann, wurde hier eine Javascript-Lösung implementiert. Diese Lösung wurde stillgelegt, da aufgrund deren Ungenauigkeit später die Entscheidung gegen Schieberegler getroffen wurde.

```
$(document).ready(
function()
{
    const $valueSpan = $('.valueSpan');
    const $value = $('#kd_tel2');
    $valueSpan.html("0" + $value.val());
    $value.on('input change', () =>
    {
        $valueSpan.html("0" + $value.val());
    });
});
```



- Nachdem eine Funktion ausgeführt und dem Benutzer eine Erfolgsmeldung angezeigt wurde, wird er weitergeleitet, um den Workflow einfacher zu gestalten. Das Laden einer neuen Funktion nach x Sekunden wurde mit Javascript realisiert.

```
function redirect(site, time)
{
    window.setTimeout(function()
    {
        window.location.href = site;
    }, time)
```

- In bestimmten Situationen ist es hilfreich, Text in die Zwischenablage zu kopieren. Hierfür wurde ebenfalls eine Javascript-Funktion implementiert, welche dieses mit einem Klick ermöglicht.

```
function copytext(id)
{
    var copyText = document.getElementById(id);

    copyText.select();
    //für mobile:
    copyText.setSelectionRange(0, 99999);

    document.execCommand("copy");
}
```



- Die Vermerke werden in einem Textfeld angezeigt, welches nach einem Klick auf das entsprechende Listenelement mit dem Vermerktext gefüllt wird.

The screenshot shows a CRM application interface. On the left, there is a table with a list of notes. The notes are as follows:

20.05.2020	Hausbesuch	mpraxm
20.05.2020	Information	mpraxm
04.05.2020	Information	SYSTEM
04.05.2020	Telefonat	⚠ SYSTEM
04.05.2020	Information	SYSTEM
29.04.2020	System	SYSTEM

On the right, a modal window is open, displaying the details of the selected note (the one with the yellow warning icon). The modal contains the following text:

Vermerk:  
Wichtiges Telefonat!

Abb. 5.6.1 – Vermerkauswahl

```
function display(x)
{
    var vermerktable =
document.getElementById("vermerktable");
    var y = "1";
    while (y <
vermerktable.getElementsByTagName("tr").length )
    {
        document.getElementById("tr"+y).className = "";
        y++;
    }

    document.getElementById("tr"+x).className =
"ausgewaehlt table-active";

    var text = document.getElementById("div"+x).value;
    document.getElementById("vermerk").value =
"Vermerk:\n"+text;
}
```



## 6. Frontend-Logik - Praxmarer

Da es sich um eine Webapplikation handelt, wurde der größte Teil der Logik der Datenverarbeitung mit Hilfe von PHP realisiert. *PHP: Hypertext Preprocessor* ist eine Programmiersprache, welche auf einem Webserver interpretiert wird, Verbindungen mit Datenbanken herstellen, Dateien auf dem Server erstellen, lesen und bearbeiten und die Ergebnisse über den Browser ausgeben kann. Im Gegensatz zu Javascript ist ein direktes Eingreifen in den Code durch den Endnutzer nicht möglich.

### 6.1. Logische Unterteilung des Systems

Das System ist in folgende Module unterteilt:

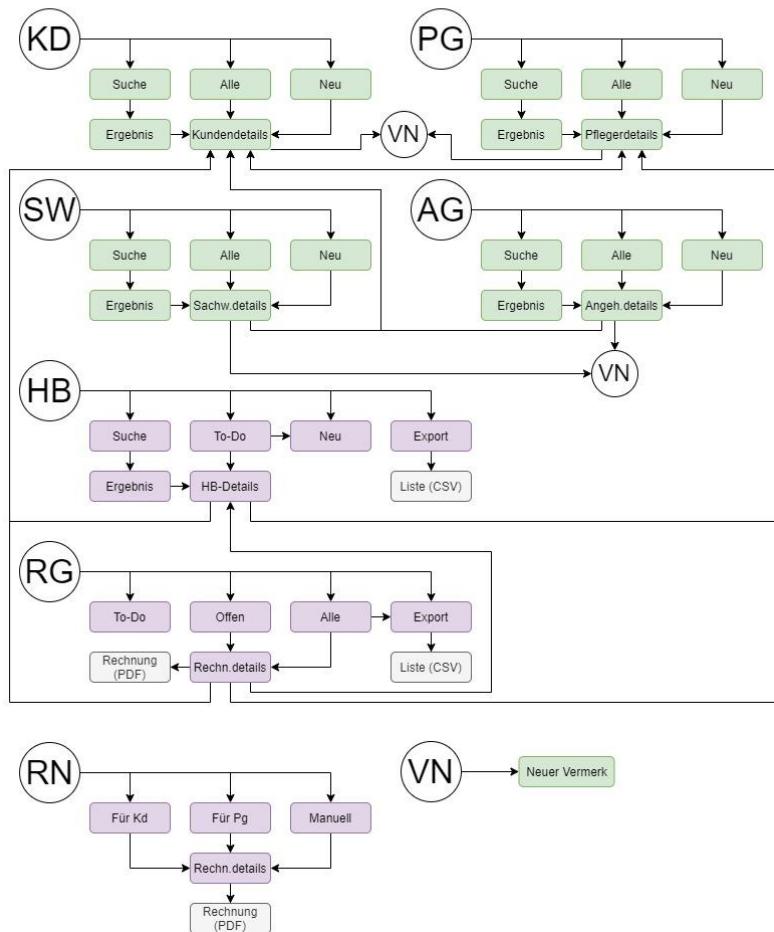
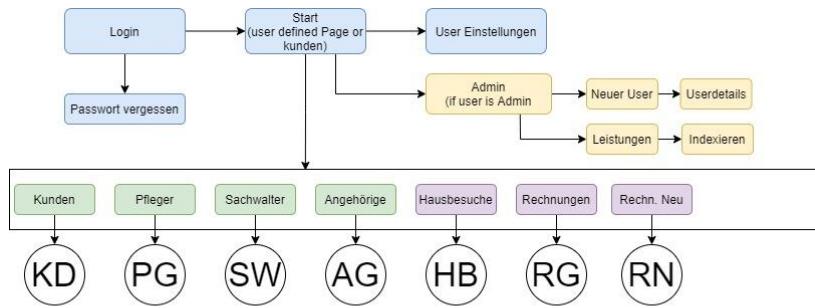
- Kundenmodul  
Anlage, Anzeige, Bearbeitung und Management der Kundenakte.
- Pflegermodul  
Anlage, Anzeige, Bearbeitung und Management der Pflegerakte.
- Angehörigen- und Sachwaltermodul  
Angehörige und Sachwalter sind im Backend, d.h. der Datenbank, nicht logisch getrennt. Um welche Art von Person es sich handelt, wird mit einer Flagge in der MySQL Tabelle bestimmt (die Werte *sw* oder *ag* in der Spalte *sw\_typ*).
- Pflegeplan  
Management der Besuche bei Kunden sowie der Zuordnungen Pfleger zu Kunde.



- Rechnungen  
Faktura an Kunden inkl. Listenexport für Druck und Buchhaltung sowie PDF-Erzeugung.
- Administration  
Anlage, Anzeige, Bearbeitung und Management der Benutzer, welche auf das System zugreifen, sowie die Verwaltung des Leistungskatalogs inkl. Anlage und Indexierung.



Folgende Grafik zeigt den groben Aufbau der Module sowie einige deren Funktionen.



**Abb. 6.1.1 - Flowchart Frontend**



## 6.2. Grundaufbau jeder Datei

Jeder Teil des Layouts wurde nur einmal implementiert. Es gibt jeweils nur eine Datei für Header, Navigation und Footer, welche in jedem Arbeitsbereich importiert werden.

```
include_once('header.php');  
include_once('nav.php');
```

Dies hat zum Vorteil, dass die jeweiligen Bereiche bei Aktualisierung nicht in jeder vorhandenen Datei verändert werden mussten. Wenn eine Aktualisierung des Headers notwendig war, wurde die Datei *header.php* bearbeitet und alle Module und Seiten übernahmen diese Änderung.

Jede Seite besitzt einen Titel, welcher in der Titelleiste des Browsers, im Tab-Titel, sowie in den Breadcrumbs ausgegeben wird. Die Navigation nutzt diese Variablen ebenso.

```
$siteTitle = "Neuer Kunde";  
$siteCategory = "Kunden";
```

Der Header bezieht einmalig die Dateien *functions.php* und *connectvars.php*, in welchen häufig verwendete Programmteile und die Konstanten für den Aufbau der Datenbankverbindung inklusive der Logik für die Clusterlösung gespeichert sind.

```
include_once('functions.php');  
include_once('connectvars.php');
```



Eine Vorlage, in welche Funktionen implementiert werden können, sieht demnach wie folgt aus.

```
<?php
$siteTitle = "Alle Kunden";
$siteCategory = "Kunden";

include_once('header.php');
include_once('nav.php');
?>
<!-- MAIN START -->
<?php

/*
 *   In diesem Bereich
 *       wird
 *       die Logik der
 *   gewünschten Funktion
 *       implementiert
 */
?>
<!-- MAIN END -->
<?php
include_once('footer.php');
?>
```

### 6.3. Funktionen

Häufig angesprochene Funktionen, welche vor allem innerhalb HTML-Tags Benutzung finden, wurden - um die Lesbarkeit des Codes und die Fehleranalyse zu vereinfachen - in einer eigenen Bibliothek *functions.php* zusammengefasst.

Ein Beispiel für den Gebrauch der Funktion *navlinkhighlight()* innerhalb eines [Tags lautet wie folgt.](#)

```
<li><a href="kdsuche.php" class="nav-link mr-3 pt-0 pb-0 mb-1<?php navCtrlhighlight("kdsuche"); ?>">Suchen</a>
</li>
```



Folgende Auszüge aus dieser Datei sind die wichtigsten Funktionen, welche auf diese Art und Weise genutzt werden.

- Um dem Benutzer die Navigation zu erleichtern, wird die aktuelle Seite, auf welcher er sich befindet, in der Navigation hervorgehoben, indem das Element die notwendige CSS-Klasse erhält.

```
function navlinkhighlight($sitename)
{
    if (basename($_SERVER['SCRIPT_FILENAME'], '.php') == $sitename) echo " bg-primary text-white";
```

- Die Unterpunkte des Moduls, in welchem sich der Benutzer momentan befindet, bleiben aufgeklappt.

```
function navcategoryshow($category)
{
    global $siteCategory;
    if($siteCategory == $category) echo " show";
```

- Adressen müssen zumindest eine Hausnummer besitzen, können jedoch auch eine Stiege, einen Stock und eine Türnummer beinhalten. Damit es nicht zur Anzeige irrelevanter Schrägstriche kommt, wurde diese Funktion implementiert.

```
function addressseperation($number)
{
    if ((is_int($number) && $number != 0) || (is_string($number) && $number != ""))
    {
        return "/".$number;
    }
}
```



- Bei Fehlern während der Verbindung zur Datenbank werden aussagekräftige Fehlermeldungen für die Serveradministratoren erstellt, welche in Textdateien auf dem Server abgespeichert und als E-Mail verschickt werden können. Diese Funktion wurde aus Sicherheitsgründen in der Endversion des Systems zum Teil deaktiviert, um einem Benutzer keine internen Informationen über die Datenbank zur Verfügung zu stellen.

```
function errorlog($dbc, $query)
{
    //global $dbc;

    //Fehler im Verzeichnis /errorlogs in ein eigenes
    Textfile mit Datum, Uhrzeit und zufallszahl schreiben:
    $file = "errorlogs/error-" . date("Y-m-d-H-i-s-") .
    mt_rand() . ".txt";
    $handle = fopen($file, "w");
    $fehler = "mysqli_error: " . mysqli_error($dbc) .
    "\nmysqli_errno: " . mysqli_errno($dbc) .
    "\nmysqli_connect_error: " . mysqli_connect_error() .
    "\nmysqli_connect_errno: " . mysqli_connect_errno() .
    "\nQuery: " . $query;
    $inhalt = "Datum: " . date("d.m.Y") . "\nUhrzeit: " .
    date("H:i:s") . "\nSeite: " . $_SERVER["PHP_SELF"] .
    "\nUser: " . $_SESSION["username"] . "\n\nFehler:\n" .
    $fehler;
    fwrite($handle,$inhalt);
    fclose($handle);

    //Fehler alert anzeigen:
    echo '<div class="alert alert-danger" role="alert">';
    echo str_replace("\n","<br/>",$inhalt);
    echo '</div>';

    //Mail an Admin schicken:
/*
    $nachricht = "Genaue angaben zum aufgetretenem Fehler
    sind im Errorlog unter " . $_SERVER["DOCUMENT_ROOT"] .
    "/" . $file . " zu finden.";
    $nachricht = wordwrap($nachricht, 70,"\n");
    mail(ADMIN_MAIL, 'Neuer Fehler', $nachricht);
*/
}
```



- Für neue User werden Passwörter generiert, da dies sicherer ist, als diese vom User selbst wählen zu lassen. Passwörter haben eine Länge von 16 Zeichen und werden zufällig aus einer Zeichensammlung, bestehend aus Groß-, Kleinbuchstaben, Zahlen und Sonderzeichen, generiert.

```
function passwordgen()
{
    $laenge = 16;
    $zeichen =
'0123456789abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ
UVWXYZ!=%$;&/+-_.:?"\}{/*';

    $str = '';
    $max = strlen($zeichen) - 1;

    for ($i = 0; $i < $laenge; ++$i)
    {
        $str .= $zeichen[random_int(0, $max)];
    }

    return $str;
}
```

## 6.4. Formularhandling und Sicherheit

Im Gegensatz zu Javascript ist ein direkter Eingriff durch den Benutzer in die PHP-Logik nicht möglich. Aus diesem Grund wurde großes Augenmerk darauf gelegt, Formulareingaben nicht nur im Browser, sondern auch via PHP zu validieren. Überdies wurde via PHP sichergestellt, dass nur berechtigte Benutzer Zugang zum System und speziellen Funktionen erhalten.



#### 6.4.1. Sicherheit der Datenbankverbindung

Obwohl das System nur innerhalb einer Organisation von einer überschaubaren Anzahl an Benutzern verwendet wird, kann es trotzdem zu Angriffsversuchen durch einen Mitarbeiter kommen, welcher die dafür notwendigen Kenntnisse besitzt. Angriffe könnten sogar unbeabsichtigt durch fehlerhafte Eingaben durch einen Nutzer entstehen, wenn auch sehr unwahrscheinlich.

Eine sogenannte *Injection* tritt dann auf, wenn über ein Formular Zeichenketten an das Backend übergeben werden, welche von der Datenbank nicht als Daten, sondern als Anweisungen betrachtet werden. Dies kann katastrophale und irreversible Auswirkungen auf das System haben, da durch derartige Anweisungen ganze Datenbanken unwiderruflich gelöscht werden könnten.

A screenshot of a web form titled 'Rechnungsaadresse'. The 'Straße:' field contains the value ': DROP TABLE kunde'. The entire input field is highlighted with a light blue border, indicating it is the focus of the injection attempt.

Abb. 6.4.1 - Beispiel einer "Injection"

Aus diesem Grund wurde sichergestellt, dass sämtliche vom Benutzer übergebene Daten vor der Übermittlung an die Datenbank kontrolliert und etwaige Zeichen, welche von der Datenbank falsch interpretiert werden könnten, umformatiert werden.

Hierzu wurden die Methoden *mysqli\_real\_escape\_string()* sowie *trim()* verwendet.

```
$kd_anrede_neu = mysqli_real_escape_string($dbc,  
trim($_POST['kd_anrede']));
```



`mysqli_real_escape_string()` formatiert Zeichen, welche innerhalb der MySQL-Datenbank Funktionalität innehaben, so um, dass diese von der Datenbank als Text und nicht als Funktionszeichen angenommen werden. Diese Methode wird als *escaping* bezeichnet. Ein einfaches Beispiel hierfür ist der *Backslash* (\) mit welchem man die meisten Zeichen in vielen Entwicklungsumgebungen als Teil eines Strings definiert.

`trim()` löscht nicht benötigte Leerzeichen aus den übergebenen Werten. Dies ist notwendig, da zwischen bestimmten MySQL-Funktionen zwingend Leerzeichen notwendig sind und einem Angreifer hierdurch auch diese Möglichkeit genommen wird.

Formulareingaben werden stets der eigenen Datei übergeben. Dies hat nicht nur zum Vorteil, dass die Anzahl der vorhandenen Dateien reduziert und das System damit für die Entwickler übersichtlicher gestaltet wurde, sondern auch, dass bei Verschiebung der Dateien keine neuen Ziele für die Formulare gesetzt werden müssen, da das Ziel des Formulars von PHP gepasst wird.

```
<form action="php echo $_SERVER["PHP_SELF"] ?&gt;?kd_id=&lt;?php<br/echo $kd_id; ?>" method="POST">
```

Damit das Formularhandling nicht bereits beim ersten Aufruf einer Seite ohne Formulareingaben erfolgt, wird mit einer einfachen Abfrage geprüft, ob der Button des Formulars betätigt wurde:

```
if (isset($_POST["submit"]))  
{  
    //form handling  
}
```



Für die Verbindung zur Datenbank wurde die PHP eigene Klasse *mysqli* genutzt. Da die Datenbank auf MySQL beruht und es keinen Grund gab, eine Änderung der Umgebung in Betracht zu ziehen, wurde die Entscheidung gegen die PDO Klasse getroffen.

Die Programmierung der Logik erfolgte mit einer einzigen Ausnahme im prozeduralen Stil. Da bei der Verwendung der objektorientierten Implementierung für dieses Projekt weder Vererbungen, noch Polymorphismen zum Einsatz kamen, wären die größten Vorteile des objektorientierten Stils nicht genutzt worden.

#### 6.4.2. Zugriffskontrolle

Der Zugriff auf das System erfolgt über eine Login-Funktion. User können von Administratoren angelegt und ihnen Rechte gewährt werden. Der Unterschied zwischen Administrator und regulärem User ist der Zugriff auf den Bereich *Administration* und dessen Funktionen (Management der Benutzer und der Leistungen).



Es wurde großes Augenmerk darauf gelegt, die Sicherheit der Daten zu gewährleisten und unbefugten Zugriff zu verhindern. Hierzu wurde die Login-Logik nach folgendem Schema implementiert.

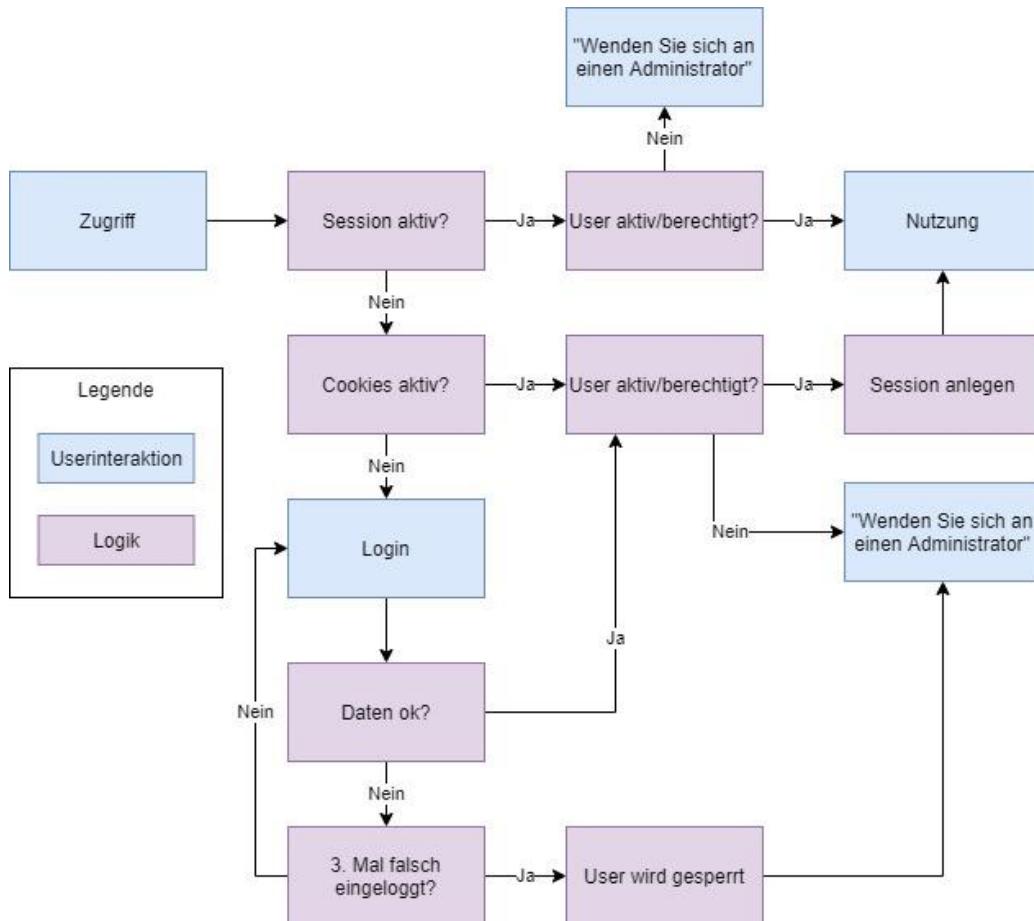


Abb. 6.4.2 - Sicherheitsschema Login

Sogenannte Brute-Force Attacken sind Versuche, sich unbefugt Zutritt zu einem System zu verschaffen, indem wiederholt versucht wird, ein Passwort zu erraten. Dies geschieht im Regelfall mithilfe von Algorithmen, welche mehrere Tausend solcher Versuche pro Sekunde durchführen können.



Obwohl die generierten Passwörter aus 16 Zeichen bestehen, welche Groß-, Kleinbuchstaben, Zahlen und Sonderzeichen beinhalten und damit ein Passwort zu erraten praktisch unmöglich ist, wurde trotzdem ein maximaler Fehllogin-Zähler implementiert.

Sobald sich ein User 3-mal hintereinander mit fehlerhaftem Passwort anmeldet, wird der User gesperrt und das Konto kann erst durch einen Administrator wieder aktiviert werden.

Der Zähler wird zurückgesetzt, sobald sich der User mit dem Korrekten Passwort einloggt, sofern das Konto nicht schon gesperrt wurde.

Ob ein User aktiv ist oder nicht, wird auf jeder Seite überprüft, um zu verhindern, dass ein von einem Administrator gesperrter User weiterhin Zugriff auf das System hat, solange die Session nicht unterbrochen wird (was theoretisch nie passiert, solange der User ständig im System agiert).



## 6.5. Personenmanagement

Folgende Punkte beziehen sich auf die Eigenheiten der personenbezogenen Module *Kunden*, *Pfleger*, *Sachwalter* und *Angehörige*, welche grundlegend ähnlich aufgebaut sind. XX bezeichnet hier den Beginn des jeweiligen Dateinamens:

kd – Kunden

pg – Pfleger

sw – Sachwalter

ag - Angehörige



### 6.5.1. XXalle.php - Anzeige aller Personen

Eine einfache Abfrage aus der Datenbank zeigt sämtliche Personen inkl. Adresse an. Der User kann sich zusätzlich zu den aktiven auch die inaktiven und/oder verstorbene Kunden ansehen. Die Logik, um die Datenbankabfrage nicht für alle Instanzen einzeln angeben zu müssen, basiert auf leeren Variablen, wenn die Filterkriterien aktiviert sind:

```
$endequery = " AND (kd_ende IS NULL OR kd_ende >=
DATE(NOW()))";
if($kd_ende == 1)
{
    $endequery = "";
}

$totquery = " AND kd_ableben = '0'";
if($kd_tot == 1)
{
    $totquery = "";
}

//Daten holen
$query = "SELECT kunde.kd_id, kunde.kd_ableben, kunde.kd_ende,
kunde.kd_anrede, kunde.kd_vorname, kunde.kd_nachname,
kunde.kd_tell, adressen.ad_strasse, adressen.ad_nr,
adressen.ad_stiege, adressen.ad_stock, adressen.ad_tuer,
adressen.ad_plz FROM kunde, adressen WHERE kunde.kd_id =
adressen.kd_id AND adressen.ad_aktiv = 1 AND adressen.ad_typ
!= 'r'" . $endequery . $totquery;
$data = mysqli_query($dbc, $query)
    or die(errorlog($dbc, $query));
```



### 6.5.2. XXdetails.php - Anzeige der Details des Akts

XXdetails.php zeigt sämtliche Daten der Person inkl. aller zugeordneten Adressen sowie Verbindungen zwischen Sachwaltern und Kunden an.

Die *ID* des Akts wird via GET an die Logik übergeben, da dies leicht über die Navigation zu bewerkstelligen war. Da dieser Wert über die Adresszeile durch den Benutzer manipuliert werden kann, wurde sichergestellt, dass dem Benutzer im Falle eines ungültigen Wertes ein entsprechender Fehler ausgegeben wird. Die Tabelle mit den Personendaten wird nur angezeigt, wenn die Variable \$error 0 oder nicht gesetzt ist. Hier am Beispiel der Kundennummer \$kd\_id:

```
//KdNr definieren
if (isset($_GET["kd_id"]))
{
    $kd_id = mysqli_real_escape_string($dbc,
trim($_GET['kd_id']));
}
else
{
    $kd_id = 0;
}

//Wenn kein GET gesetzt ist, kd_id auf 0 setzen und error anzeigen (unten im file)
if ($kd_id == "")
{
    $kd_id = 0;
    $error = 1;
}
```

```
//wenn kd_nr nicht existiert
if (empty(mysqli_fetch_array($data)))
{
    $error = 1;
    $kd_exist = 0;
}
```



```
if(isset($error) && $error == 1)
{
    //keine Kundennummer in GET
    if ($kd_id == 0)
    {
        ?>
        <div class="alert alert-warning" role="alert">
            Ungültiger Link. Sie werden zur Übersicht
weitergeleitet.
        </div>
        <script>
            window.setTimeout(function(){
                window.location.href = "kdalle.php";
            }, 5000);
        </script>
        <?php
    }

    //kdnr existiert nicht
    if (isset($kd_exist) && $kd_exist == 0 && $kd_id != 0)
    {
        ?>
        <div class="alert alert-warning" role="alert">
            Es existiert kein Kunde mit der Nummer "<?php
echo $kd_id ?>".
        </div>
        <?php
    }
}
```

### 6.5.3. XXdetails2.php - Anzeige der Vermerke

*XXdetails2.php* Zeigt sämtliche bei der Person hinterlegten Vermerke an. Diese Seite beinhaltet eine URL auf *veneu.php*, über welche die ID des Aktes und die Zuordnung (kd, pg, ag, sw) via GET weitergegeben werden und ein neuer Vermerk für die Person hinzugefügt werden kann.



Vermerke werden mit Datum und Art in einer scrollbaren Tabelle angezeigt, womit eine einfache und übersichtliche Navigation durch die einzelnen Vermerke möglich ist. Der Text wird je Zeile in ein unsichtbares `<div>` geladen, welcher via Javascript aufgerufen und in das Textfeld zur Anzeige kopiert wird.

```
echo '<input id="div' . $x . '" type="hidden" value="" .  
str_replace('\\"', '-', $row['ve_text']) . '" />';
```

#### 6.5.4. XXdetails3.php - Anzeige des Pflegeplans

Diese Ansicht bezieht sich lediglich auf Kunden und Pfleger. Sachwalter und Angehörige sind nicht Teil des Pflegeplans.

Beinhaltet eine URL auf `ppneu.php`, über welche die ID des Akts sowie deren Zuordnung (kd, pg) via GET weitergegeben werden und dem Kunden ein neuer Pfleger zugeordnet werden kann.

Die Anzeige erfolgt via Auswahl von Jahr/Monat. Bei Kunden werden sämtliche Pfleger angezeigt, welche zu diesem Zeitpunkt für einen Kunden aktiv sind sowie Start- und Enddaten der Zuordnung und um welche Art von Pflege es sich handelt.

Bei Pflegern werden die Kunden angezeigt, für welche der Pfleger im angegebenen Zeitraum aktiv ist.



Die Weitergabe der Benutzeroauswahl für Jahr und Monat an die Logik erfolgt ebenso via GET. Wenn kein GET-Wert vorhanden ist, wird das aktuelle Jahr/Monat für die Ansicht genutzt.

```
if(isset($_GET["pp_jahr"]))
{
    $pp_jahr = mysqli_real_escape_string($dbc,
trim($_GET['pp_jahr']));
}
else
{
    $pp_jahr = date("Y", time());
}

if(isset($_GET["pp_monat"]))
{
    $pp_monat = mysqli_real_escape_string($dbc,
trim($_GET['pp_monat']));
}
else
{
    $pp_monat = date("m", time());
}
```



### 6.5.5. kddetails4.php - Anzeige der Tarife des Kunden

Diese Detailansicht betrifft lediglich Kunden.

Tarife des Kunden können hier über 2 Tabellen angepasst werden. Aus der Tabelle der verfügbaren Leistungen können diese mit einem Klick zugeordnet bzw. aus der Zuordnung entfernt werden. Die hinzugefügten bzw. entfernten Leistungen werden ebenso via GET an dieselbe Datei übergeben und sofort in der Datenbank bearbeitet.

```
if(isset($_GET["lg_delete_id"]))
{
    $lg_delete_id = mysqli_real_escape_string($dbc,
trim($_GET['lg_delete_id']));
    $query = "DELETE FROM bezug WHERE kd_id = '$kd_id' AND
lg_id = '" . $lg_delete_id . "'";
    mysqli_query($dbc, $query)
        or die(errorlog($dbc,$query));
}

if(isset($_GET["lg_add_id"]))
{
    $lg_add_id = mysqli_real_escape_string($dbc,
trim($_GET['lg_add_id']));
    $query = "INSERT INTO bezug(kd_id, lg_id) VALUES('$kd_id',
'$lg_add_id')";
    mysqli_query($dbc, $query)
        or die(errorlog($dbc,$query));
}
```

### 6.5.6. kdDetails5.php - Anzeige der Rechnungen des Kunden

Auf dieser Seite werden die Rechnungen des ausgewählten Kunden nach Jahren angezeigt. Die Seite beinhaltet den URL zu *rgneukd.php*, über welchen eine neue Rechnung für diesen Kunden erstellt werden kann.



Der Wert des Jahres wird wie die Kundennummer via GET weitergegeben.  
Standardmäßig wird das letzte Jahr angezeigt, in dem für den Kunden eine Rechnung erstellt wurde.

```
//aus der db die min und max werte der vorhandenen Jahre holen  
$scopequery = "SELECT MAX(rg_jahr) as rg_jahr_max,  
MIN(rg_jahr) as rg_jahr_min FROM rechnung WHERE kd_id =  
'$kd_id' ORDER BY rg_jahr";
```

Das Select zur Jahresauswahl wird mit den verfügbaren Daten basierend auf vorhandenen Rechnungen gefüllt.

```
$selectquery = "SELECT DISTINCT rg_jahr FROM rechnung WHERE  
kd_id = '$kd_id' ORDER BY rg_jahr";  
$selectdata = mysqli_query($dbc, $selectquery)  
or die(errorlog($dbc, $selectquery));  
<select class="form-control form-control-sm" name="rg_jahr"  
id="rg_jahr"  
onchange="document.getElementById('rg_jahr_select').submit()">  
    <?php  
    while($selectrow = mysqli_fetch_array($selectdata))  
    {  
        echo '<option ';  
        if($selectrow["rg_jahr"] == $rg_jahr) echo 'selected';  
        echo '>' . $selectrow["rg_jahr"];  
        echo '</option>';  
    }  
    ?>  
</select>
```



Die Rechnungen werden mithilfe von zwei Anfragen aus der Datenbank gelesen. Die erste Anfrage ermittelt die Rechnungsdaten, die zweite Anfrage liest die in dieser Rechnung enthaltenen Positionen inkl. Anzahl aus und iteriert durch die Einträge, um die Gesamtsumme bilden zu können.

```
$query = "SELECT *, DATE_FORMAT(rg_datum, '%d.%m.%Y') AS rg_datum1 FROM rechnung WHERE kd_id = '$kd_id' AND rg_jahr = '$rg_jahr' ORDER BY rg_nr DESC, rg_art DESC";
$data = mysqli_query($dbc, $query)
or die(errorlog($dbc, $query));
```

```
while($row = mysqli_fetch_array($data))
{
    $query2 = "SELECT * FROM leistung, positionen WHERE
positionen.rg_id = '" . $row["rg_id"] . "' AND
positionen.lg_id = leistung.lg_id ORDER BY positionen.po_nr";
    $data2 = mysqli_query($dbc, $query2)
    or die(errorlog($dbc, $query2));
```

```
while($row2 = mysqli_fetch_array($data2))
{
    $positionen = $row2["po_nr"];
    $lg_rabattgrenze = $row2["lg_rabattgrenze"];
    $lg_einzeltarif = $row2["lg_einzeltarif"];
    $lg_mengentarif = $row2["lg_mengentarif"];
    $po_anzahl = $row2["po_anzahl"];

    $preis = $lg_einzeltarif;
    if($po_anzahl >= $lg_rabattgrenze &&
!empty($lg_rabattgrenze)) $preis = $lg_mengentarif;
    $gesamt = round($preis*$po_anzahl,2);
    $summe = $summe + $gesamt;
}
```



### 6.5.7. XXneu.php - Neuanlage eines Akts

Ein neuer Kunde, Pfleger oder Sachwalter/Angehöriger wird über diese Seite angelegt. Sie besteht aus einem Formular, welches Daten nach diversen Überprüfungen an die Datenbank übergibt. Überprüft wird u.a., ob alle notwendigen Felder ausgefüllt wurden

```
if (!empty($kd_anrede) && !empty($kd_vorname) &&
!empty($kd_nachname) && !empty($kd_strasse) && !empty($kd_nr)
&& !empty($kd_plz) &&
((isset($kd_b_strasse) && !empty($kd_b_strasse) &&
!empty($kd_b_nr) && !empty($kd_b_plz)) ||
(!isset($kd_b_strasse)))
```

, ob der Akt nicht schon existiert

```
$query = "SELECT kd_id FROM kunde WHERE kd_vorname =
'$kd_vorname' AND kd_nachname = '$kd_nachname' AND kd_gebdatum
= '$kd_gebdatum' LIMIT 1";
$data = mysqli_query($dbc, $query)
    or die(errorlog($dbc,$query));

//Wenn Kd nicht existiert:
if (mysqli_num_rows($data) == 0)
{
```

und dass die E-Mail-Adresse das korrekte Format hat

```
if (($kd_mail != "" && filter_var($kd_mail,
FILTER_VALIDATE_EMAIL)) || $kd_mail == "")
```

### 6.5.8. XXaendern.php - Ändern der Daten des Akts

Dies ist ein Formular ähnlich deren in *XXneu.php* mit einigen, zusätzlichen Auswahlmöglichkeiten. Die Daten der Person werden in den Formularfeldern angezeigt bzw. die notwendigen Dropdown-Auswahlen vorausgewählt und die Checkboxen angehakt.



Für Textfelder wird die implementierte Funktion `keeppostvalues()` genutzt, um die Änderungen beizubehalten, sollte ein Benutzerfehler vorliegen (z.B. bei Vergessen eines notwendigen Feldes oder einem ungültigem Wert).

Die Logik der Selects bzw. Checkboxen wurde inline implementiert.

```
<option <?php if((isset($_POST["submit"])) && $kd_anrede_neu == "Herr") || (!isset($_POST["submit"])) && $kd_anrede == "Herr") echo 'selected="selected"';?>Herr</option>
```

```
<input type="checkbox" <?php if((isset($_POST["submit"])) && $kd_ableben_neu == 1) || (!isset($_POST["submit"])) && $kd_ableben == 1)) echo ' checked '?> name="kd_ableben" id="kd_ableben" />
```



### 6.5.9. XXsuche.php - Suchen eines Kunden

Personen können anhand von Vornamen, Nachnamen und Adresse gesucht werden. Die Suche erfolgt mit einer einfachen *LIKE* Abfrage der Tabellen inkl. einem *Join*. Die einzelnen Suchbedingungen werden in einem Array gespeichert, welches als Abfrage an die Datenbank mit dem Teiler *OR* übergeben wird. Hier am Beispiel der Kundensuche:

```
$suchbegriffe = explode(" ", $suchbegriff);
$query_start = "SELECT kunde.kd_id, kunde.kd_vorname,
kunde.kd_nachname, adressen.ad_strasse, adressen.ad_plz FROM
kunde, adressen";
$query_where_list = array();

{
    foreach ($suchbegriffe as $wort)
    {
        if (isset($_POST["suche_vn"])) $query_where_list[] =
"(kunde.kd_vorname LIKE '%$wort%' AND adressen.kd_id =
kunde.kd_id AND adressen.ad_aktiv = 1 AND adressen.ad_typ
IN('rb','r'))";
        if (isset($_POST["suche_nn"])) $query_where_list[] =
"(kunde.kd_nachname LIKE '%$wort%' AND adressen.kd_id =
kunde.kd_id AND adressen.ad_aktiv = 1 AND adressen.ad_typ
IN('rb','r'))";
        if (isset($_POST["suche_ad"])) $query_where_list[] =
"(adressen.ad_strasse LIKE '%$wort%' AND adressen.kd_id =
kunde.kd_id)";
    }
}

$query_where = implode(" OR ", $query_where_list);

$query = "$query_start WHERE $query_where";
$data = mysqli_query($dbc, $query)
or die(errorlog($dbc, $query));
```



## 6.6. Leistungen

Leistungen bilden die Grundlage der Faktura. Diese werden aufgrund bestehender Verträge angelegt und Kunden zugeordnet. Da lediglich die Führungskräfte Einfluss auf die Vertragsbestandteile haben, ist dieses Modul nur von Administratoren nutzbar.

### 6.6.1. Igalle.php – Anzeige aller Leistungen

Hier werden die Leistungen eines bestimmten Jahres angezeigt. Die Auswahl des Jahres erfolgt über ein Select, in welchem nur die Jahre zur Auswahl stehen, für welche auch Leistungen existieren. Das Jahr 0000 wird bewusst nicht inkludiert, da diese Leistungen nur einmalig als Sonderfall angelegt wurden (siehe Kapitel 6.7.6).

```
$selectquery = "SELECT DISTINCT lg_jahr FROM leistung WHERE  
lg_jahr NOT IN('0000') ORDER BY lg_jahr";  
$selectdata = mysqli_query($dbc, $selectquery)  
or die(errorlog($dbc, $selectquery));
```



Die Einheiten werden in der MySQL Tabelle als verkürzte Version dargestellt. Um dies für den Benutzer auch lesbar anzuzeigen, wurde ein Switch Statement genutzt.

```
switch($row["lg_einheit"])
{
    case "h":
        $lg_einheit = "Stunde";
        break;
    case "m":
        $lg_einheit = "Monat";
        break;
    case "d":
        $lg_einheit = "Tag";
        break;
    case "w":
        $lg_einheit = "Woche";
        break;
    case "b":
        $lg_einheit = "Besuch";
        break;
    case "s":
        $lg_einheit = "Stück";
        break;
    case "e":
        $lg_einheit = "Einheit";
        break;
    default:
        $lg_einheit = "Einheit";
}
```

### 6.6.2. Igneu.php – Neue Leistung hinzufügen

Auf dieser Seite kann eine neue Leistung hinzugefügt werden. Leistungen haben eine Bezeichnung, ein Jahr als Preisbasis, einen Einzeltarif und optional einen Mengentarif sowie eine Grenze, ab wann der Mengentarif gilt.

Die Logik unterscheidet sich nicht zu anderen Neuanlagen in diesem System.



### 6.6.3. Igindex.php – Indexierung bestehender Leistungen

Regelmäßige Teuerung - die Inflation - ist ein Bestandteil der Preiskultur jedes Unternehmens. Tarife müssen angepasst werden, um einen Ausgleich zur Inflation herzustellen. Die Indexierung erfolgt auf Jahresbasis.

Die Auswahl und Anzeige der Leistungen erfolgt analog zu *Igalle.php* mit dem Unterschied, dass für jede Leistung eine Checkbox generiert wird, womit man sie für die Indexierung markieren kann.

```
echo '<input type="checkbox" name="lg_index_id_' . $i . '"  
id="lg_index_id_' . $i . '" value="' . $lg_id . '"';  
if ((isset($_POST["lg_index_id_" . $i])) &&  
isset($_POST["submit"])) || !isset($_POST["submit"])) echo '  
checked';  
echo '/>';
```

Ebenso befinden sich zur Auswahl des Jahres zur Anzeige auch zwei Felder auf dieser Seite, über welche der Faktor, mit dem indexiert werden soll, sowie das neue Jahr für die Preisbasis eingegeben werden können.



Nach dem Klick auf *indexieren* werden sämtliche ausgewählten Leistungen kopiert, die Tarife mit dem angegebenen Faktor multipliziert und mit dem neuen Jahr als Basis angelegt. Zusätzlich wird jedem Kunden, welchem die ursprüngliche Leistung zugeordnet war, die neue Leistung zugeordnet und die alte Zuweisung entfernt.

```
mysqli_begin_transaction($dbc);

foreach($lg_index_ids as $lg_id)
{
    $query = "INSERT INTO `leistung`(`lg_bezeichnung`,
`lg_einheit`, `lg_rabattgrenze`, `lg_einzeltarif`,
`lg_mengentarif`, `lg_jahr`) "
        . "SELECT `lg_bezeichnung`, `lg_einheit`,
`lg_rabattgrenze`, `lg_einzeltarif`*, '$lg_index .',
`lg_mengentarif`* . $lg_index . ", " . $lg_index_jahr . "
FROM `leistung` WHERE lg_id = '$lg_id'";
    mysqli_query($dbc, $query)
        or die(errorlog($dbc, $query));
    $lg_id_neu = mysqli_insert_id($dbc);

    $query = "INSERT INTO bezug(kd_id, lg_id) "
        . "SELECT kd_id, $lg_id_neu FROM bezug WHERE lg_id
= $lg_id";
    mysqli_query($dbc, $query)
        or die(errorlog($dbc, $query));

    $query = "DELETE FROM bezug WHERE lg_id = $lg_id";
    mysqli_query($dbc, $query)
        or die(errorlog($dbc, $query));
}

mysqli_commit($dbc)
```

## 6.7. Verrechnung

Die Faktura ist ein wichtiger und kritischer Teil des Systems. Da Fehler in diesem Modul im schlimmsten Fall zu fehlerhafter Buchhaltung und damit zu Strafen für das Unternehmen führen können, hatte der korrekte Ablauf dieser Funktionen oberste Priorität.



Rechnungsnummern dürfen nur einmal existieren und die Summen und Preise der verrechneten Positionen müssen bei jedem Aufruf, bei der PDF-Erstellung, sowie beim Export exakt gleich berechnet und ausgegeben werden. Ebenso dürfen Rechnungen in keinem Fall geändert werden können.

### 6.7.1. rgalle.php – Anzeige aller Rechnungen

Hier werden sämtliche Rechnungen angezeigt. Die Auswahl des Jahres und das Abrufen der Daten sowie die Berechnung erfolgt analog zu *kddetails5.php*. Der Unterschied besteht darin, dass die Rechnungen aller Kunden angezeigt, Rechnungen als bezahlt bzw. offen markiert werden können und offene Rechnungen mit einem Blick ersichtlich sind.

Offene Rechnungen werden in der Tabelle mit roter Farbe unterlegt.

```
echo "<tr>";
if($row["rg_bezahlt"] != 1) echo " class='table-danger' ";
echo ">";
```

Rechnungen erhalten eine Checkbox, über welche sie als bezahlt bzw. nicht bezahlt markiert werden können. Der Name der Checkbox ist der Zähler der Schleife, welche durch die Rechnungen iteriert, der Wert ist die jeweilige Rechnungs-ID.

```
//bezahlt
echo "<td>";
?>
<input type="checkbox" <?php if($row["rg_bezahlt"] == 1) echo
" checked " ?> name="<?php echo $i; ?>" id="<?php echo $i; ?>"
value="<?php echo $row["rg_id"]; ?>">
<label for="<?php echo $i; ?>" class="col-form-label-
sm">bezahlt</label>
<?php
echo "</td>";
```



Wenn Rechnungen markiert und das Formular abgeschickt wird, iteriert die Logik durch die Rechnungsnummern bis zum Maximum-Wert, welcher mithilfe unsichtbarer Felder ebenso weitergegeben wird, und setzt die jeweiligen Rechnungen auf bezahlt bzw. nicht bezahlt.

```
for($i=1;$i<=$rg_max;$i++)
{
    $komma = ",";
    if($i == 1 || $rg_bezahlt== "") $komma = "";

    if(isset($_POST["$i"])) $rg_bezahlt = $rg_bezahlt . $komma
. $_POST["$i"];
}

if(!empty($rg_alle))
{
    $query = "UPDATE rechnung SET rg_bezahlt = 0 WHERE rg_id
IN($rg_alle)";
    mysqli_query($dbc,$query)
    or die(errorlog($dbc,$query));
}

if(!empty($rg_bezahlt))
{
    $query = "UPDATE rechnung SET rg_bezahlt = 1 WHERE rg_id
IN($rg_bezahlt)";
    mysqli_query($dbc,$query)
    or die(errorlog($dbc,$query));
}
```

### 6.7.2. rgoffen.php – Anzeige der offenen Rechnungen

Diese Seite ist ähnlich aufgebaut wie *rgalle.php* mit dem Unterschied, dass hier lediglich die offenen Rechnungen des ausgewählten Zeitraums aufscheinen und diese auch nicht farblich hervorgehoben sind. Sobald diese als bezahlt markiert werden, verschwinden die Rechnungen aus der Ansicht. Daher ist hier auch keine Logik für das Setzen auf nicht bezahlt vorhanden.



### 6.7.3. rgexport.php – Export der Rechnungen als CSV-Datei

*rgexport.php* ermöglicht dem Benutzer, sämtliche Rechnungsdaten als CSV-Datei aus dem System zu exportieren. Diese Datei enthält die Positionen mit Zuordnung via Rechnungsnummern.

Die Auswahl der zu exportierenden Daten erfolgt anhand von Datumswerten. Die Daten der Rechnungen und der Positionen werden wie in den vorhergehenden Implementierungen aus der Datenbank gelesen und durch die Tupel iteriert, sämtliche Daten in die Variable *\$file\_inhalt* geschrieben, an *rg\_export.php* übergeben, in eine CSV-Datei geschrieben und letztendlich als Download ausgegeben.

```
$file_inhalt =
"Rechnungsnummer;Rechnungsdatum;Kundennummer;Anschrift;Leistung Start;Leistung Ende;Pos Nr;Pos Bezeichnung;Pos Anzahl;Pos Einheit;Pos Einzel;Pos Gesamt;\n";

$file_inhalt = $file_inhalt .
"\$rg_nr;\$rg_datum;\$kd_id;\$rg_anschrift;\$rg_zeitraum_start;\$rg_zeitraum_ende;\$po_nr;\$po_bezeichnung;\$po_anzahl;\$po_einheit;\$po_einzel;\$po_gesamt;\n";

$file_inhalt = $_POST["file_inhalt"];
$file = "csv/" . $_POST["von"] . "-" . $_POST["bis"] .
"_RECHNUNGEN" . "_" . mt_rand() . ".csv";
$handle = fopen($file, "w");

// fwrite($handle,$file_inhalt);

file_put_contents($file, "\xEF\xBB\xBF". $file_inhalt);
fclose($handle);
//readfile($file);

header("Location: $file");
```



#### 6.7.4. rgtodo.php – Noch nicht verrechnete Leistungen anzeigen

Auf dieser Seite werden Kunden angezeigt, welche für den ausgewählten Monat keine Rechnung erhalten haben, welche als Leistungsdatum das Ende des Monats einschließt. D.h. auch wenn nur die ersten zwei Wochen des Monats verrechnet wurden, wird der Kunde angezeigt, da noch zwei nicht verrechnete Wochen aufscheinen.

Das auszuwählende Monat und Jahr werden wie in den vorhergehenden Seiten via GET übergeben und über Selects ausgewählt.

Nachdem die aktiven Kunden aus der Datenbank ausgelesen wurden, werden die dazugehörigen Rechnungen gesucht und die entsprechenden Daten in der Spalte *verrechnet bis* angezeigt.

```
$query2 = "SELECT rg_zeitraum_ende,
DATE_FORMAT(rg_zeitraum_ende, '%d.%m.%Y') AS rg_zeitraum_bis
FROM rechnung WHERE kd_id = " . $row["kd_id"] . " ORDER BY
rg_zeitraum_ende DESC LIMIT 1";
$data2 = mysqli_query($dbc, $query2)
or die(errorlog($dbc, $query2));

$offene = 0;
$row2 = mysqli_fetch_array($data2);

if($row2["rg_zeitraum_ende"] < $rg_datum_ende || 
empty($row2["rg_zeitraum_ende"]))
{
    $offene = 1;
}
else
{
    $offene = 0;
}

if($offene == 1)
{
//In die Tabelle eintragen
```



In jeder Spalte werden URLs zu *kddetails.php* sowie zu *rgneukd.php* generiert, um den Kundenakt ansehen bzw. eine neue Rechnung für den Kunden erstellen zu können.

### 6.7.5. **rgneukd.php – Erstellung der Rechnung für einen Kunden**

Auf dieser Seite wird im ersten Schritt ein Select mit allen aktiven Kunden generiert.

Der Zweite Schritt erhält die Kundennummer via POST, falls diese über das Formular eingegeben wurde oder über GET, falls der Benutzer über einen der URLs in *kddetails5.php* oder *rgtodo.php* auf diese Seite navigiert.

```
if(isset($_POST["submit"]) || isset($error_step2) ||
isset($_GET["kd_id"]))
{
    if(isset($_GET["kd_id"])) $kd_id =
mysqli_real_escape_string($dbc, trim($_GET['kd_id']));
    if(isset($_POST["kd_id"])) $kd_id =
mysqli_real_escape_string($dbc, trim($_POST['kd_id']));
```



Im zweiten Schritt kann die Rechnungsadresse (Kunde oder Sachwalter, falls vorhanden) ausgewählt werden. Diese Abfrage wurde auf zwei Queries aufgeteilt.

```
$kd_query = "SELECT * FROM kunde, adressen, plz WHERE
kunde.kd_id = '$kd_id' AND adressen.kd_id = '$kd_id' AND
adressen.ad_aktiv = 1 AND adressen.ad_typ IN ('rb', 'r') AND
plz.plz = adressen.ad_plz";
$kd_data = mysqli_query($dbc, $kd_query)
    or die(errorlog($dbc, $kd_query));
$kd_row = mysqli_fetch_array($kd_data);
$sw_query = "SELECT * FROM sachwalter, angehoerige, adressen,
plz WHERE angehoerige.ag_aktiv = 1 AND angehoerige.kd_id =
'$kd_id' AND sachwalter.sw_id = angehoerige.sw_id AND
sachwalter.sw_typ = 'sw' AND adressen.ad_aktiv = 1 AND
adressen.ad_typ IN ('rb', 'r') AND adressen.sw_id =
sachwalter.sw_id AND adressen.sw_id = angehoerige.sw_id AND
plz.plz = adressen.ad_plz";
$sw_data = mysqli_query($dbc, $sw_query)
    or die(errorlog($dbc, $sw_query));
$kd_sw = 1;
if(mysqli_num_rows($sw_data) == 0) $kd_sw = 0;
$sw_row = mysqli_fetch_array($sw_data);
```

Die Daten der Adressen werden als Optionen ausgegeben und die jeweiligen Werte für Kundennummer, Name und Adresse in ein unsichtbares Textfeld geschrieben, da diese Daten so in die Tabelle *Rechnung* gespeichert werden. Constraints sind hier nicht sinnvoll, da Namen und Adressen geändert werden können und Rechnungen unverändert bleiben müssen.

```
<div class="col-auto mr-5 ml-4">
    <input class="form-check-input" type="radio" name="rg_ad"
id="rg_ad_kd" value="kd" checked>
    <input type="number" name="rg_ad_id_kd" id="rg_ad_id_kd"
value="<?php echo $kd_row["ad_id"]; ?>" hidden />
    <textarea name="rg_ad_kd" hidden><?php echo $kd_name . .
"\n" . $kd_ad ?></textarea>
    <label for="rg_ad_kd">
        (Kunde)<br> <?php echo $kd_name . "<br>" .
str_replace("\n", "<br>", $kd_ad) ?>
    </label>
</div>
```



Die diesem Kunden zugeordneten Leistungen können darunter zur Verrechnung ausgewählt und die jeweilige Anzahl eingegeben werden. Es erscheint ebenso das Datum der letzten Verrechnung der Leistung, um doppelte Verrechnung zu vermeiden. Da es jedoch häufig zu Nachverrechnungen kommt, ist diese Prüfung nicht in die Logik implementiert, sondern muss vom Mitarbeiter selbst überprüft werden. Auch dies wurde auf zwei Anfragen aufgeteilt.

```
$lg_query = "SELECT * FROM leistung, bezug WHERE  
leistung.lg_id = bezug.lg_id AND bezug.kd_id = '$kd_id'";  
$lg_data = mysqli_query($dbc, $lg_query)  
    or die(errorlog($dbc, $lg_query));  
while($lg_row = mysqli_fetch_array($lg_data))  
{  
    $po_query = "SELECT  
        max(DATE_FORMAT(rg_zeitraum_ende, '%d.%m.%Y')) AS datum FROM  
        rechnung, positionen WHERE positionen.lg_id = '" .  
        $lg_row["lg_id"] . "' AND rechnung.rg_id = positionen.rg_id  
        AND rechnung.rg_art = 'r' AND rechnung.kd_id = '$kd_id'";  
    $po_data = mysqli_query($dbc, $po_query)  
        or die(errorlog($dbc, $po_query));  
    $po_row = mysqli_fetch_array($po_data);
```

Die Leistungen können mit einer Checkbox zur Verrechnung ausgewählt werden. Sobald dies passiert, wird ein entsprechendes Script ausgeführt, welches das Anzahl-Feld für diese Leistung auf *required* setzt.

```
<input type="checkbox" checked id="php echo $i; ?"  
name="php echo $i; ?" value="php echo $lg_row["lg_id"];<br/?>" onchange="toggleRequired('php echo $i; ?', 'anzahl_php<br/echo $i; ?>')"/>
```



Wenn keine Leistungen für diesen Kunden hinterlegt sind, wird der Benutzer mit einem URL auf `kddetails4.php` darauf hingewiesen, zuerst Leistungen zu hinterlegen.

```
if(mysqli_num_rows($lg_data) == 0)
{
    $no_leistungen = 1;
    ?>
    <div class="alert alert-danger">
        Keine Leistungen für diesen Kunden vorhanden!<br>
        Bitte <a href="kddetails4.php?kd_id=<?php echo $kd_id;
?>">fügen Sie Leistungen hinzu</a>.
    </div>
    <?php
}
```

Im dritten Schritt, nach einem Klick auf *Weiter* (submit2), wird nun auch via PHP überprüft, ob die zu verrechnenden Leistungen auch wirklich eine Anzahl ungleich 0 aufweisen, zumindest eine Leistung und eine Adresse ausgewählt wurden sowie das Ende des Leistungszeitraums nicht vor dem Beginn liegt.

```
if($_POST["rg_ad"] == "kd")
{
    $rg_ad = mysqli_real_escape_string($dbc,
trim($_POST['rg_ad_kd']));
    $rg_ad_id = mysqli_real_escape_string($dbc,
trim($_POST['rg_ad_id_kd']));
}
elseif($_POST["rg_ad"] == "sw")
{
    $rg_ad = mysqli_real_escape_string($dbc,
trim($_POST['rg_ad_sw']));
    $rg_ad_id = mysqli_real_escape_string($dbc,
trim($_POST['rg_ad_id_sw']));
}
else
{
    ?>
    <div class="alert alert-danger">
        Bitte eine Anschrift auswählen!
    </div>
    <?php
    $error_step2 = 1;
}
```



```
$rg_leistungen = array();
for($i=1; $i <= $lg_gesamt; $i++)
{
    if(isset($_POST["$i"]))
    {
        $rg_leistungen[$_POST["$i"]] = $_POST["anzahl_$i"];
        if(!isset($_POST["anzahl_$i"])) || $_POST["anzahl_$i"]
== "" || $_POST["anzahl_$i"] == 0) $rg_leistung_leer = 1;
    }
}

if(sizeof($rg_leistungen) <= 0)
{
    ?>
<div class="alert alert-danger">
    Bitte mindestens eine Leistung zur Verrechnung auswählen!
</div>
<?php
$error_step2 = 1;
}
```

Sollten die Prüfungen nicht erfolgreich sein und die Variable `$error_step2` gesetzt sein, wird die entsprechende Fehlermeldung mit dem zweiten Schritt angezeigt und die Eingaben müssen angepasst werden.



Bei erfolgreicher Prüfung wird eine neue Rechnung angelegt, indem die notwendigen Daten in die Tabelle *Rechnung* eingefügt sowie die Rechnung mit den Leistungen über entsprechende Einträge in die Tabelle *Positionen* verbunden.

```
mysqli_begin_transaction($dbc);
$query = "INSERT INTO rechnung" .
    "(kd_id, us_id, ad_id, rg_jahr, rg_nr, rg_anschrift,
rg_zeitraum_start, rg_zeitraum_ende, rg_datum, rg_art)" .
    "SELECT '$kd_id', '' . $_SESSION['userid'] . ,
'$rg_ad_id' . ", '$rg_jahr', case when min(abs(rg_nr)) IS
null then 1 else max(rg_nr)+1 end, '$rg_ad',
'$rg_zeitraum_start', '$rg_zeitraum_ende', '$rg_datum',
'$rg_art' FROM rechnung WHERE rg_jahr = '$rg_jahr'";
mysqli_query($dbc, $query)
    or die(errorlog($dbc, $query));

$rg_id = mysqli_insert_id($dbc);

$i = 1;
foreach($rg_leistungen as $lg_id => $po_anzahl)
{
    $query = "INSERT INTO positionen" .
        "($lg_id, rg_id, po_nr, po_anzahl) " .
        "VALUES('$lg_id', '$rg_id', '$i', '$po_anzahl')";
    mysqli_query($dbc, $query)
        or die(errorlog($dbc, $query));

    $i++;
}

mysqli_commit($dbc);
```

Dem Benutzer wird eine Erfolgsmeldung angezeigt sowie URLs zu *rgstorno.php* und *rechnung.php*.



### 6.7.6. *rgneuman.php* – Manuelle Verrechnung

Um auch Leistungen außerhalb des regulären Katalogs verrechnen zu können, ohne diese zuerst durch einen Administrator anlegen und dem Kunden zuordnen zu müssen, können Rechnungen auch manuell erstellt werden. Ebenso kann über diese Seite auch an Nicht-Kunden verrechnet werden, d.h. an Personen, welche sich nicht im System befinden.

Der Unterschied zu *rgneukd.php* ist, dass zusätzlich zur Auswahl des Kunden im ersten Schritt auch die Möglichkeit besteht, Empfänger und Anschrift manuell einzugeben.

Ein weiterer Unterschied ist, dass Leistungen nicht zur Auswahl aus dem Katalog geladen werden, sondern via Inputs manuell eingetragen werden können. Diese Leistungen werden in die Datenbank mit dem Jahr *0000* eingetragen, damit sie in den anderen Teilen des Systems nicht aufscheinen.

Die Erfolgsmeldung beinhaltet ebenfalls URLs zu *rgstorno.php* und *rechnung.php*.



### 6.7.7. rgstorno.php – Rechnungsdetails ansehen

Auf dieser Seite können Rechnungsdetails eingesehen werden und Rechnungen storniert werden. Die Logik der Anzeige unterscheidet sich kaum von den bereits erwähnten Modulen und Seiten.

Für das Storno einer Rechnung werden die Rechnungsdaten inkl. Positionen kopiert, jedoch mit negativen Anzahlen in die Datenbank eingefügt, wodurch eine negative Rechnung, d.h. eine Gutschrift, erstellt wird.

```
mysqli_begin_transaction($dbc);

//Rg auf bezahlt setzen
$query = "UPDATE rechnung SET rg_bezahlt = 1 WHERE rg_id =
$rg_id";
mysqli_query($dbc, $query)
or die(errorlog($dbc, $query));

//Rechnung kopieren
$query = "INSERT INTO rechnung (kd_id, pg_id, ad_id, us_id,
rg_jahr, rg_nr, rg_anschrift, rg_zeitraum_start,
rg_zeitraum_ende, rg_datum, rg_art, rg_bezahlt, rg_druck)
SELECT kd_id, pg_id, ad_id, us_id, rg_jahr, rg_nr,
rg_anschrift, rg_zeitraum_start, rg_zeitraum_ende,
CURRENT_DATE(), 'g', rg_bezahlt, rg_druck FROM rechnung WHERE
rg_id = $rg_id";
mysqli_query($dbc, $query)
or die(errorlog($dbc, $query));

$rg_id_neu = mysqli_insert_id($dbc);

//Positionen kopieren mit anzahl negativ:
$query = "INSERT INTO positionen(lg_id, rg_id, po_nr,
po_anzahl)
SELECT lg_id, $rg_id_neu, po_nr, (-1
* po_anzahl) FROM positionen WHERE rg_id = '$rg_id'";
mysqli_query($dbc, $query)
or die(errorlog($dbc, $query));

mysqli_commit($dbc);
```

Auch hier wird eine Erfolgsmeldung angezeigt und der Benutzer kann die Rechnung sowie die eben erstellte Gutschrift als PDF betrachten.



### 6.7.8. rechnung.php – Export der Rechnung als PDF

Auf dieser Seite werden die Daten der Rechnungen in eine PDF-Datei exportiert und angezeigt. Dies ist der einzige Teil des Systems, in welchem der objektorientierte Programmier-Stil zum Einsatz kommt, da eine Klasse erstellt wird, welche eine bestehende aus dem Framework *TCPDF* erweitert.

```
//Klasse und Methode für Footer
class MYPDF extends TCPDF {

    public function Footer() {
        // Schriftart
        $this->SetFont('helvetica', '', 10);
        // Seitennummer
        $this->SetY(-35);
        //Individueller Footer (Firmenname, Bank, usw)...
```



Nachdem das Objekt dieser Klasse erstellt und diverse Attribute gesetzt wurden, wird das HTML generiert, ebenso an das Objekt übergeben und letztendlich als PDF ausgegeben.

```
$pdf = new MYPDF(PDF_PAGE_ORIENTATION, PDF_UNIT,  
PDF_PAGE_FORMAT, true, 'UTF-8', false);
```

```
$html = "  
<br><br><br>  
<table border=\"0\" cellpadding=\"20\">  
  <tr>  
    <td>An<br>$rg_anschrift_html  
    </td>  
    <td align="right">  
      <table border="0">  
        <tr>  
          <td colspan="2">wien, am  
$rg_datum<br></td>  
        </tr>  
        <tr>  
          <td>Rechnungsnummer:</td>  
          <td>$rg_nr" . "-" . "$rg_jahr</td>  
        </tr>  
        <tr>  
          <td>Kundennummer:</td>  
          <td>$kd_id</td>  
        </tr>  
      </table>" ;
```

```
$pdf->writeHTML($html, true, false, true, false, '' );  
$pdf->lastPage();  
  
$pdf->Output('rechnung.pdf', 'I');
```

## 6.8. Pflegeplan

Im Pflegeplan befinden sich die Funktionen, Pfleger und Kunden zu verbinden, Pläne zu verändern, To-dos der Qualitätssicherung anzuzeigen und diverse Pläne im CSV-Format zu exportieren.



### 6.8.1. pphb.php – Anzeige der Hausbesuche

Um die Übersicht darüber zu behalten, bei welchen Kunden im ausgewählten Monat noch Besuche zur Qualitätssicherung stattfinden müssen, wurde diese Seite erstellt. Kunden werden in Tabellenform angezeigt und rot unterlegt, wenn noch keine Hausbesuche erfolgt sind.

Die Tabelle enthält URLs zum Kundenakt via *kddetails.php*, falls vorhanden zur Detailansicht der bereits eingetragenen Hausbesuche auf *ppdetails.php* sowie zu *ppneu.php*, um einen neuen Hausbesuch anlegen zu können.

Jahr und Monat der Auswahl werden wie zuvor via GET übergeben und die Anzeige erfolgt ähnlich zu *rgalle.php*.



### 6.8.2. ppdetails.php – Detailansicht der Tätigkeit

Auf dieser Seite werden die Details des jeweiligen Hausbesuchs bzw. der Tätigkeit angezeigt. Die einzigen Felder, welche nicht geändert werden können, sind Kunde und Pfleger, da dies keinen Sinn macht. Falls ein Pflegeplan falsch angelegt wurde, kann der Eintrag hier ebenso gelöscht werden. Dies ist die einzige Anwendung einer solchen Funktion im gesamten System. Beim Klick auf *Löschen* muss in einem zweiten Schritt die Löschung erneut bestätigt werden.

```
if (isset($_GET["delete"]) && !isset($error))
{
    if(!isset($_GET["delete2"]))
    {
        echo '<div class="alert alert-warning" role="alert">';
        echo 'Wollen Sie diesen Eintrag wirklich';
        echo 'Löschen?<br>';
        echo '<a href="ppdetails.php?pp_id=' . $pp_id . '&delete=1&delete2=1" class="btn btn-danger">Löschen</a>';
        echo ' <a href="ppdetails.php?pp_id=' . $pp_id . '"';
        echo ' class="btn btn-primary">Abbrechen</a>';
        echo '</div>';
    }
    else
    {
        $query = "DELETE FROM pflegeplan WHERE pp_id =
'$pp_id'";
        mysqli_query($dbc,$query)
        or die(errorlog($dbc,$query));

        echo '<div class="alert alert-success" role="alert">';
        echo 'Der Eintrag wurde erfolgreich gelöscht.';
        echo '</div>';
        exit();
    }
}
```



### 6.8.3. ppneu.php – Anlage einer neuen Tätigkeit

ppneu.php ermöglicht das Anlegen einer neuen Tätigkeit. Hier sind Kunde und Pfleger wählbar sowie Start- und Enddatum des Einsatzes. Das Enddatum kann offen sein, dann wird diese Tätigkeit ab dem Beginndatum im entsprechenden Kunden- bzw. Pflegerakt immer angezeigt.

Wenn der Benutzer über den entsprechenden URL von *pphp.php* auf diese Seite gelangt, sind im Select der Pfleger lediglich die diplomierten Kräfte wählbar, da diese als einzige diese Leistung durchführen dürfen bzw. können. Dies wurde mit der Variable \$hb realisiert.

```
if (isset($_GET["hb"]))
{
    $hb = 1;
}
else
{
    $hb = 0;
}
```

```
<select name="pg_id" id="pg_id" class="form-control form-control-sm">
    <?php
        if($hb == 0)
        {
    ?>
        <optgroup label="24h Pfleger:>
            <?php
                $query = "SELECT pg_id, pg_vorname, pg_nachname FROM
                    pfleger WHERE pg_art = '24h' ORDER BY pg_nachname";
                $data = mysqli_query($dbc, $query)
                or die(errorlog($dbc,$query));
                while($row = mysqli_fetch_array($data))
                {
                    echo '<option value="' . $row["pg_id"] . '"';
                    if($pg_id == $row["pg_id"]) echo ' selected';
                    echo '>' . $row["pg_nachname"] . ' ' .
                        $row["pg_vorname"] . '</option>';
                }
    ?>
        </optgroup>
        <optgroup label="Stundenweise:>
```



```
<?php
    $query = "SELECT pg_id, pg_vorname, pg_nachname FROM
pfleger WHERE pg_art = 'sw' ORDER BY pg_nachname";
    $data = mysqli_query($dbc, $query)
    or die(errorlog($dbc,$query));
    while($row = mysqli_fetch_array($data))
    {
        echo '<option value="' . $row["pg_id"] . '"';
        if($pg_id == $row["pg_id"]) echo ' selected';
        echo '>' . $row["pg_nachname"] . ' ' .
$row["pg_vorname"] . '</option>';
    }
    ?>
</optgroup>
<?php
}
?>
<optgroup label="DGKS/P:">
    <?php
        $query = "SELECT pg_id, pg_vorname, pg_nachname FROM
pfleger WHERE pg_art = 'dgks' ORDER BY pg_nachname";
        $data = mysqli_query($dbc, $query)
        or die(errorlog($dbc,$query));
        while($row = mysqli_fetch_array($data))
        {
            echo '<option value="' . $row["pg_id"] . '"';
            if($pg_id == $row["pg_id"]) echo ' selected';
            echo '>' . $row["pg_nachname"] . ' ' .
$row["pg_vorname"] . '</option>';
        }
    ?>
</optgroup>
</select>
```

#### 6.8.4. ppexport bis ppexport3.php – Export der Daten als CSV

Der Export ist ähnlich zu den Kundendetails via Reiter organisiert. Hier kann der Benutzer zwischen Export der Pläne von Kunden, Pflegern oder den noch offenen Hausbesuchen wählen.



Beim Export der Pläne von Kunden oder Pflegern wird die Person, für welche der Plan exportiert werden soll, via eines Selects ausgewählt und der zu exportierende Zeitraum analog zu *rgexport.php* gehandhabt. Die Logik der Erstellung der CSV-Datei unterscheidet sich hier ebenso kaum.

Für Hausbesuche wird eine Liste sowie die Auswahl Monat/Jahr analog zu *pphb.php* generiert, mit dem Unterschied, dass hier lediglich Kunden aufscheinen, bei welchen noch keine Hausbesuche im ausgewählten Monat stattgefunden haben.

## 6.9. Usermanagement

Dieses Modul dient zum Management der Benutzer, welche das System verwenden. Das User-Modul kann nur von Benutzern mit Administratorrechten eingesehen und genutzt werden.

Auf den folgenden Seiten können Benutzer angelegt, verändert, gesperrt und deren Berechtigungen verwaltet werden.



### 6.9.1. usalle.php – Anzeige aller User des Systems

User werden analog zu anderen Personen in einer Tabelle angezeigt. Die Art sowie der Status des jeweiligen Benutzers werden farbig gedruckt.

```
if($row["us_admin"] == 1)
{
    echo '<span class="text-success">Admin</span>';
}
else
{
    echo '<span class="text-info">User</span>';
}
echo '</td>';
echo '<td>';
if($row["us_status"] == 1)
{
    echo '<span class="text-success">aktiv</span>';
}
else
{
    echo '<span class="text-danger">inaktiv</span>';
}
```

### 6.9.2. usneu.php – Neuanlage eines Benutzers

Die Anlage eines Users unterscheidet sich kaum zur Anlage eines Kunden oder anderer Personen. Der Unterschied besteht darin, dass für den Benutzer ein Name sowie Passwort generiert wird.



Der Username wird aus dem ersten Buchstaben des Vornamens und den ersten fünf Buchstaben des Nachnamens generiert. Wenn dieser Benutzername bereits existiert, wird die nächste ID aus der Tabelle *user* als Präfix an den Benutzernamen angefügt.

```
$us_name = substr(strtolower($us_vorname), 0, 1) .  
substr(strtolower($us_nachname), 0, 5);
```

```
// Da es den User mit Vor- und Nachnamen bereits gibt, dem  
usernamen die nächste ID zuweisen:  
$query2 = "SELECT Auto_increment FROM  
information_schema.tables WHERE table_name='user' LIMIT 1, 1";  
$data2 = mysqli_query($dbc, $query2)  
or die(errorlog($dbc, $query2));  
$row2 = mysqli_fetch_array($data2);  
  
$us_name = $us_name . $row2["Auto_increment"];
```

Für den Eintrag des Passworts in die Datenbank wird die MySQL-Funktion *SHA()* genutzt, welche einen Hash für das Passwort weitergibt. Passwörter werden daher weder im Klartext, noch verschlüsselt gespeichert.

```
$query = "INSERT INTO `user` "  
        "(us_name, us_pw, us_mail, us_vorname, us_nachname,  
us_status, us_admin)" .  
        "VALUES" .  
        "('$us_name', SHA('$us_pw'), '$us_mail',  
'$us_vorname', '$us_nachname', '1', '$us_admin')";
```

### 6.9.3. usaendern.php – Benutzer bearbeiten

usaendern.php ermöglicht es, nicht nur die Personendaten zu ändern, sondern auch die Rechte des Benutzers zu bearbeiten, den User zu deaktivieren bzw. bei Sperrung durch zu viele fehlerhafte Loginversuche wieder zu aktivieren und ein neues Passwort zu generieren.



Beim Ändern des Namens ändert sich der Benutzername jedoch nicht. Dies ist gewollt und dient der Nachvollziehbarkeit, da Änderungen und Vermerke anhand des Usernamens für andere Benutzer nachvollziehbar bleiben sollten.



## 7. Testung – Bonengl

Im Rahmen der Qualitätssicherung wurden diverse Tests durchgeführt. Diese Tests basieren auf den Userstories und sollten nicht nur die einwandfreie Funktion des Systems testen, sondern auch mögliche Probleme der Usability aufzeigen, wie zB Fehlende Links oder Brüche des Workflows.

### 7.1. Testszenarien anhand der Userstories

Grundlage für die Funktionstests waren die vorliegenden Userstories. Für jede Story wurden fünf Testfälle durchgeführt.

Eine detailliertere Erfassung der Tests wurde im Anhang (10.3.1) festgehalten.

Zusammenfassend kann jedoch erwähnt werden, dass sämtliche Funktionen ohne Komplikationen erfolgreich ausgeführt werden konnten.

### 7.2. Injections

Die Injection-Tests wurden auf jeweils einer Seite jedes Moduls durchgeführt, um zu testen, ob ungewünschte Eingriffe in die Datenbank durchführbar sind.

Es wurde für die Tests der POST-Formulare in der Datenbank eine Tabelle *test* erstellt, welche versucht wurde, zu löschen, indem der MySQL Ausdruck *DROP TABLE test* weitergegeben wurde.



Für die GET-Formulare wurde versucht, sämtliche Daten aus der Datenbank anzuzeigen, indem an das Ende des Parameters in der Adresszeile der Ausdruck **OR 1=1** angefügt wurde.

Eine detailliertere Erfassung der Tests wurde ebenfalls im Anhang festgehalten (10.3.3). Zusammenfassend kann jedoch auch hier informiert werden, dass kein Versuch, eine Injection zu verursachen, erfolgreich an die Datenbank übergeben wurde.



### 7.3. Umgehen Userlogins

Ebenfalls wurde überprüft, ob ein Benutzer auf die Seiten, Daten oder Funktionen des CareCRM zugreifen kann, ohne eingeloggt zu sein.

Zu diesem Zweck wurde versucht, die PHP-Dateien ohne bestehende Anmeldung aufzurufen.

Das Ergebnis war, dass auf keine der Seiten zugegriffen werden und keine Daten ausgelesen werden konnten. Das System gab auf allen Seiten, inklusive beim Versuch, eine PDF-Rechnung abzurufen, folgenden Hinweis aus.

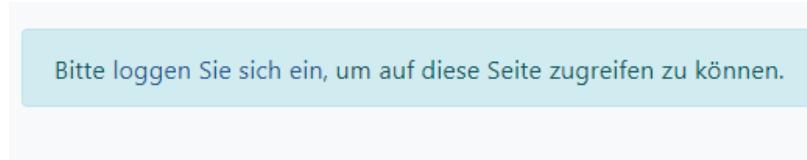


Abb. 7.3.1 - Meldung an den User, dass kein Login vorliegt

Eine detailliertere Erfassung der Tests wurde ebenfalls im Anhang festgehalten (10.3.2).



## 8. Verweise/Literaturverzeichnis

Adobe Systems Software Ireland Limited. (2020). *Adobe Color*. Von <https://color.adobe.com> abgerufen

Arntzen, T. C., Bakken, S., Caraveo, S., Gutmans, A., Lerdorf, R., Ruby, S., . . . Zmievski, A. (2020). *PHP Documentation*. Von <https://www.php.net/> abgerufen

DigitalOcean, LLC. (2020). *DigitalOcean - How To Install the Apache Web Server on CentOS 7*. Von <https://www.digitalocean.com/community/tutorials/how-to-install-the-apache-web-server-on-centos-7> abgerufen

Oracle Corporation. (2020). *MySQL 8.0 Reference Manual*. Von <https://dev.mysql.com/doc/refman/8.0/en/> abgerufen

Otto, M. (., & fat. (2020). *Bootstrap Documentation*. Von <https://getbootstrap.com/docs/> abgerufen

projektfarm GmbH. (2020). *HowtoForge*. Von <https://www.howtoforge.com> abgerufen

Refsnes Data. (2020). *W3Schools*. Von <https://www.w3schools.com/> abgerufen

Stack Exchange, Inc. (2020). *Stack Overflow*. Von <https://stackoverflow.com/> abgerufen

The CentOS Project. (2020). *CentOS Image Mirrors*. Von [http://isoredirect.centos.org/centos/7/isos/x86\\_64/](http://isoredirect.centos.org/centos/7/isos/x86_64/) abgerufen



## 9. Abbildungsverzeichnis

<b>Abb. 1.1.1 - Applikation im Desktop-Modus.....</b>	<b>6</b>
<b>Abb. 1.1.1 - desktop version of the application .....</b>	<b>9</b>
<b>Abb. 4.1.1 - Konzept Clusterlösung.....</b>	<b>41</b>
<b>Abb. 5.1.1 - Farbstudie.....</b>	<b>45</b>
<b>Abb. 5.1.2 – Ausgewähltes Farbschema .....</b>	<b>45</b>
<b>Abb. 5.1.3 - Logoentwürfe .....</b>	<b>46</b>
<b>Abb. 5.1.4 – endgültiges CareCRM Logo .....</b>	<b>46</b>
<b>Abb. 5.2.1 - HTML 5 Form Validierung in Chrome .....</b>	<b>48</b>
<b>Abb. 5.3.1 - Tabelle "Kunden", verstorbene werden rot unterlegt .....</b>	<b>49</b>
<b>Abb. 5.3.2 - Buttons mit Funktionszuweisung.....</b>	<b>49</b>
<b>Abb. 5.4.1 - Navigation im Desktop-Modus.....</b>	<b>50</b>
<b>Abb. 5.4.2 - Navigation im Mobil-Modus.....</b>	<b>51</b>
<b>Abb. 5.4.3 - Zusätzliche Reiter, um die Funktionen des Kundenakts aufzutei...</b>	<b>52</b>
<b>Abb. 5.6.1 – Vermerkauswahl.....</b>	<b>58</b>
<b>Abb. 6.1.1 - Flowchart Frontend.....</b>	<b>61</b>
<b>Abb. 6.4.1 - Beispiel einer "Injection".....</b>	<b>67</b>
<b>Abb. 6.4.2 - Sicherheitsschema Login.....</b>	<b>70</b>
<b>Abb. 7.3.1 - Meldung an den User, dass kein Login vorliegt .....</b>	<b>112</b>



## 10. Anhang

### 10.1. Zeitaufzeichnung Praxmarer

Datum	Stunden	Bereich	Detail
12.09.2019	2	Antrag DA	
17.09.2019	1	Antrag DA	
25.09.2019	5	Planung	User Stories + Wireframe
30.09.2019	4,5	Planung	Wireframe
24.10.2019	0,5	Antrag DA	Abgabe
28.10.2019	4	Planung	Datenbank
11.11.2019	4	Planung	Datenbank
14.11.2019	2	Test	CSV Export
14.11.2019	2,5	Test	PDF Export
20.11.2019	3	Planung	Userflow
21.11.2019	2,5	Planung	Mockup
25.11.2019	2	Planung	Mockup
04.01.2020	6	UI	Desktop
05.01.2020	3,5	UI	Desktop
06.01.2020	4	UI	Desktop
07.01.2020	4	Nav	Desktop
09.01.2020	4	Kunde	neu
12.01.2020	2	Kunde	details1
12.01.2020	1,5	Kunde	alle
13.01.2020	3	Kunde	Suche
14.01.2020	3	Präsentation DA	Vorbereitung
15.01.2020	0,5	Präsentation DA	Präsentation
18.01.2020	2	Vermerke	UI
19.01.2020	1	Datebank	Planung angepasst
23.01.2020	2,5	Vermerke	Logik
23.01.2020	1	Kunde	details2
01.02.2020	3	Kunde	ändern
04.02.2020	2	User	neu
04.02.2020	1	User	ändern
04.02.2020	0,5	User	alle
07.02.2020	1,5	User	login
07.02.2020	2,5	User	startsession
09.02.2020	1,5	User	logout



Datum	Stunden	Bereich	Detail
20.04.2020	1	Planung	Organisationsbesprechung
04.05.2020	1	Planung	Organisationsbesprechung
04.05.2020	1	Präsentation DA	Zwischenpräsentation Google Meetup
06.05.2020	2	Pfleger	neu
06.05.2020	2	Pfleger	details
06.05.2020	0,5	Pfleger	details2
07.05.2020	1	Pfleger	ändern
07.05.2020	1	Pfleger	suche + alle
09.05.2020	2	Angehörige	neu
09.05.2020	0,5	Datenbank	Planung angepasst
09.05.2020	2	Angehörige	details
09.05.2020	0,5	Angehörige	details2
10.05.2020	1	Angehörige	ändern
11.05.2020	1	Angehörige	suche + alle
11.05.2020	2	Sachwalter	gesamtes Modul
11.05.2020	3	Pflegeplan	neu
15.05.2020	3,5	Kunde+Pfleger	details3
15.05.2020	2,5	Pflegeplan	Hausbesuche
15.05.2020	1	Pflegeplan	export CSV
17.05.2020	2,5	UI	Mobile
18.05.2020	1,5	Nav	Mobile
18.05.2020	2	UI	Mobile
21.05.2020	3	Tarife	neu
22.05.2020	2,5	DA Buch	Entwurf
23.05.2020	1,5	Leistungen	alle
23.05.2020	3	Leistungen	Index
24.05.2020	1,5	Kunde	details4
25.05.2020	3,5	DA Buch	Entwurf
25.05.2020	1	Präsentation DA	Zwischenpräsentation Google Meetup
27.05.2020	3,5	Rechnung	neu
27.05.2020	1,5	Rechnung	PDF Export
28.05.2020	2	Rechnung	PDF Export
28.05.2020	2	Rechnung	manuell
29.05.2020	1,5	Rechnung	alle
02.06.2020	2,5	Rechnung	offene
02.06.2020	2	Kunde	details5
03.06.2020	1,5	Rechnung	export CSV



Datum	Stunden	Bereich	Detail
06.06.2020	1,5	UI	Mobile
08.06.2020	1	Datenbankverbindung	Connectvars (Anschluss Clusterlösung)
12.06.2020	2,5	Präsentation DA	Vorbereitung
15.06.2020	0,5	Planung	Organisationsbesprechung
15.06.2020	0,5	Sachwalter	details Fehlerbehebung
15.06.2020	1,5	Pflegeplan	neu Fehlerbehebung
15.06.2020	1	Leistungen	neu Fehlerbehebung
15.06.2020	0,5	Kunde	neu Fehlerbehebung
16.06.2020	0,5	Rechnung	neu Fehlerbehebung
16.06.2020	1	Leistungen	index Fehlerbehebung
16.06.2020	0,5	Nav	Fehlerbehebung
16.06.2020	0,5	Präsentation DA	Aufnahme Video
17.06.2020	2,5	Präsentation DA	Videobearbeitung + Abgabe

## 10.2. Zeitaufzeichnung Bonengl

Datum	Stunden	Bereich	Detail
12.09.2019	2	Antrag DA	Planung
17.09.2019	1	Antrag DA	Planung
25.09.2019	5	Recherche	Backend
30.09.2019	4,5	Recherche	Backend
24.10.2019	0,5	Antrag DA	Abgabge
28.10.2019	4	Planung	Datenbank
11.11.2019	4	Planung	Datenbank
14.11.2019	2	Test	CSV Export
14.11.2019	2,5	Test	PDF Export
20.11.2019	3	Planung	Userflow
21.11.2019	0,5	Recherche	Serveraufbau
25.11.2019	2	Testing	Serveraufbau
04.01.2020	3,5	Erstellung	Datenbank
05.01.2020	7,5	Testing	Serveraufbau
06.01.2020	1	Recherche	Ausfallsicherung & Serveraufbau
07.01.2020	4	Testing	Ausfallsicherung & Serveraufbau
09.01.2020	2	Recherche	Ausfallsicherung & Serveraufbau
12.01.2020	2	Testing	Ausfallsicherung & Serveraufbau



Datum	Stunden	Bereich	Detail
12.01.2020	1,5	Recherche	Ausfallsicherung & Serveraufbau
13.01.2020	3	Testing	Ausfallsicherung & Serveraufbau
14.01.2020	3	Präsentation DA	Vorbereitung
15.01.2020	0,5	Präsentation DA	Präsentation
19.01.2020	1	Datebank	Planung angepasst
20.04.2020	1	Planung	Organisationsbesprechung
04.05.2020	1	Planung	Organisationsbesprechung
04.05.2020	1	Präsentation DA	Zwischenpräsentation Google Meetup
09.05.2020	0,5	Datenbank	Planung angepasst
25.05.2020	2	Serverkonfiguration	Ausfallsicherung & Serveraufbau
27.05.2020	4	Testing	Ausfallsicherung & Serveraufbau
27.05.2020	1,5	Serverkonfiguration	Ausfallsicherung & Serveraufbau
28.05.2020	1,5	Datenbank	Anpassung
28.05.2020	3	Serverkonfiguration	Ausfallsicherung
02.06.2020	1,5	Recherche	Ausfallsicherung & Serveraufbau
02.06.2020	5,5	Serverkonfiguration	Ausfallsicherung & Serveraufbau
03.06.2020	5	Serverkonfiguration	Ausfallsicherung
06.06.2020	7	Datenbank	Anpassung
08.06.2020	6	Serverkonfiguration	Ausfallsicherung
12.06.2020	6	Testing	Testseznarios Userstories
15.06.2020	3	Testing	Umgehen Userlogin
15.06.2020	5,5	Serverkonfiguration	Ausfallsicherung
15.06.2020	2	Datenbank	Anpassung
15.06.2020	0,5	Recherche	Backupkonzept
15.06.2020	0,2	Implementierung	Backupkonzept
15.06.2020	0,5	Planung	Organisationsbesprechung
16.06.2020	9	Testing	Testseznarios Userstories
16.06.2020	5	Testing	Injectiontesting
16.06.2020	1	Präsentation DA	Präsentations Erstellung
17.06.2020	3	Präsentation DA	Aufnahme Video



## **10.3. Testprotokoll – Bonengl**

### **10.3.1. Funktionstest anhand Userstories**

Folgendes Protokoll zeigt die durchgeführten Tests anhand der Userstories inklusive Screenshots der Eingaben und der Ergebnisse (siehe 7.1).



### 10.3.1.1. Kundenmodul

**Ein Kunde wird angelegt und hat eine Rechnungsadresse, welche von der Betreuungssadresse abweicht**

Ergenix: erfolgreich

Details:

Kunden → Neuer Kunde

Eingeloggt als: Bernhard Bonengl  
[Ausloggen](#)

Anrede: Herr      Sachwalter: Keiner  
Titel: Titel      Festnetz: 01111111  
Vorname: Werner      Mobil: 0660 11111111  
Nachname: Eins      E-Mail: werner@eins.at  
Geburtsdatum: 11.02.1940      Rechnung: E-Mail

**Angehörige**  
Angehöriger 1: keiner  Notfallkontakt

**Rechnungsadresse**  
Straße: Wernerstraße      Nr.: 1      Stg.: 1      Stock: 1      Tür: 1      PLZ: 1100

**Betreuungssadresse (wenn von Rechnungsadresse abweichend)**  
 Betreuungssadresse abweichend von Rechnungsadresse  
Straße: Einerstraße      Nr.: 2      Stg.: 2      Stock: 2      Tür: 2      PLZ: 1220

**Wichtige Anmerkung**  
Anmerkung:  
Anmerkung

**Datensatz anlegen**

Der neue Kunde wurde erfolgreich angelegt

[Kundendetails](#)

[Neuen Kunden anlegen](#)



Allgemeine Daten

Anrede:	Frau	Sachwalter:	Altfrau Gertrud
Titel:	Dr.	Festnetz:	012222222
Vorname	Alex	Mobil:	06602222222
Nachname	Zwei	E-Mail:	alex@zwei.at
Geburtsdatum	04.05.1954	Rechnung:	Post

Angehörige

Angehöriger 1: keiner  Notfallkontakt

Rechnungsadresse

Straße:	Nr:	Stg:	Stock:	Tür:	PLZ:
zweistraße	2	2	2	2	1220

Betreuungsadresse (wenn von Rechnungsadresse abweichend)

Betreuungsadresse abweichend von Rechnungsadresse

Straße:	Nr:	Stg:	Stock:	Tür:	PLZ:
Einerstraße	1	2	2	2	1200

Wichtige Anmerkung

Anmerkung:  
Anmerkung

Der neue Kunde wurde erfolgreich angelegt

Kundendetails

[Neuen Kunden anlegen](#)



Allgemeine Daten

Anrede:	Herr	Sachwalter:	Barista Bernhard
Titel:	Mag.	Festnetz:	013333333
Vorname	Lenard	Mobil:	+43 660 234567
Nachname	Drei	E-Mail:	lenard@drei.at
Geburtsdatum	30.04.1945	Rechnung:	E-Mail

Angehörige

Angehöriger 1:	keiner	<input type="checkbox"/> Notfallkontakt
----------------	--------	---

Rechnungsadresse

Straße:	dreierstraße	Nr:	3	Stg:	3	Stock:	3	Tür:	3	PLZ:	1230
---------	--------------	-----	---	------	---	--------	---	------	---	------	------

Betreuungsadresse (wenn von Rechnungsadresse abweichend)

Betreuungsadresse abweichend von Rechnungsadresse

Straße:	zweierstraße	Nr:	2	Stg:	2	Stock:	1	Tür:	2	PLZ:	1230
---------	--------------	-----	---	------	---	--------	---	------	---	------	------

Wichtige Anmerkung

Anmerkung:

Anmerkung

Der neue Kunde wurde erfolgreich angelegt

Kundendetails

[Neuen Kunden anlegen](#)



Allgemeine Daten

Anrede:	Frau	Sachwalter:	Barista Bernha
Titel:	Titel	Festnetz:	014444444
Vorname	Anna	Mobil:	+43 660 234567
Nachname	Vier	E-Mail:	anna@vier.at
Geburtsdatum	04.05.1959	Rechnung:	Post

Angehörige

Angehöriger 1: keiner   Notfallkontakt

Rechnungsadresse

Straße:	Nr:	Stg:	Stock:	Tür:	PLZ:
Viererstraße	4	4	4	4	1140

Betreuungsadresse (wenn von Rechnungsadresse abweichend)

Betreuungsadresse abweichend von Rechnungsadresse

Straße:	Nr:	Stg:	Stock:	Tür:	PLZ:
zweierstraße	24	2	4	2	1140

Wichtige Anmerkung

Anmerkung:  
Anmerkung

Der neue Kunde wurde erfolgreich angelegt

Kundendetails

[Neuen Kunden anlegen](#)



Allgemeine Daten

Anrede:	Herr	Sachwalter:	Maria Eva
Titel:	Dr.	Festnetz:	015555555
Vorname	karl	Mobil:	+43 660 234567
Nachname	fünf	E-Mail:	karl@fuenf.at
Geburtsdatum	25.05.1955	Rechnung:	E-Mail

Angehörige

Angehöriger 1: keiner  Notfallkontakt

Rechnungsadresse

Straße:	Nr:	Stg:	Stock:	Tür:	PLZ:
fünferstraße	5	5	5	5	1150

Betreuungsadresse (wenn von Rechnungsadresse abweichend)

Betreuungsadresse abweichend von Rechnungsadresse

Straße:	Nr:	Stg:	Stock:	Tür:	PLZ:
dreierstraße	3	2	3	5	1150

Wichtige Anmerkung

Anmerkung:

Anmerkung

Der neue Kunde wurde erfolgreich angelegt

Kundendetails

[Neuen Kunden anlegen](#)



## Ein Kunde wird angelegt. Es soll ein bestehender Sachwalter hinzugefügt werden

Ergenix: erfolgreich

Details:

Allgemeine Daten

Anrede:	Frau	Sachwalter:	Altfrau Gertrud
Titel:	Dr.	Festnetz:	016666666
Vorname	Eva	Mobil:	+43 660 234567
Nachname	Sechs	E-Mail:	eva@sechs.at
Geburtsdatum	06.07.1966	Rechnung:	E-Mail

Angehörige

Angehöriger 1:	keiner	<input type="checkbox"/> Notfallkontakt
----------------	--------	---

Rechnungsadresse

Straße:	sechserstraße	Nr:	6	Stg:	6	Stock:	6	Tür:	6	PLZ:	1160
---------	---------------	-----	---	------	---	--------	---	------	---	------	------

Betreuungsadresse (wenn von Rechnungsadresse abweichend)

Betreuungsadresse abweichend von Rechnungsadresse

Wichtige Anmerkung

Anmerkung:

Anmerkung

**Datensatz anlegen**

Der neue Kunde wurde erfolgreich angelegt

Kundendetails

[Neuen Kunden anlegen](#)



Allgemeine Daten

Anrede:	Frau	Sachwalter:	Barista Bernha
Titel:	Mag.	Festnetz:	017777777
Vorname	Maria	Mobil:	+43 660 234567
Nachname	sieben	E-Mail:	maria@sieben.at
Geburtsdatum	07.07.1957	Rechnung:	Post

Angehörige

Angehöriger 1: keiner  Notfallkontakt

Rechnungsadresse

Straße:	Nr:	Stg:	Stock:	Tür:	PLZ:
siebnerstraße	7	7	7	7	1170

Betreuungsadresse (wenn von Rechnungsadresse abweichend)

Betreuungsadresse abweichend von Rechnungsadresse

Wichtige Anmerkung

Anmerkung:  
Anmerkung

**Datensatz anlegen**

Der neue Kunde wurde erfolgreich angelegt

[Kundendetails](#)

[Neuen Kunden anlegen](#)



Allgemeine Daten

Anrede:	Herr	Sachwalter:	Hundertwasser
Titel:	Titel	Festnetz:	018888888
Vorname	Manuel	Mobil:	+43 660 234567
Nachname	Acht	E-Mail:	manuel@acht.at
Geburtsdatum	08.08.1958	Rechnung:	Post

Angehörige

Angehöriger 1: keiner  Notfallkontakt

Rechnungsadresse

Straße:	Nr:	Stg:	Stock:	Tür:	PLZ:
Achterstraße	8	8	8	8	1180

Betreuungsadresse (wenn von Rechnungsadresse abweichend)

Betreuungsadresse abweichend von Rechnungsadresse

Wichtige Anmerkung

Anmerkung:  
Anmerkung

**Datensatz anlegen**

Der neue Kunde wurde erfolgreich angelegt

Kundendetails

[Neuen Kunden anlegen](#)



Allgemeine Daten

Anrede:	Frau	Sachwalter:	Maria Eva
Titel:	Titel	Festnetz:	019999999
Vorname	viktoria	Mobil:	+43 660 234567
Nachname	neun	E-Mail:	/iktoria@neun.at
Geburtsdatum	09.09.1959	Rechnung:	E-Mail

Angehörige

Angehöriger 1: keiner  Notfallkontakt

Rechnungsadresse

Straße:	Nr:	Stg:	Stock:	Tür:	PLZ:
neuernstraße	9	9	9	9	1190

Betreuungsadresse (wenn von Rechnungsadresse abweichend)

Betreuungsadresse abweichend von Rechnungsadresse

Wichtige Anmerkung

Anmerkung:  
Anmerkung

**Datensatz anlegen**

Der neue Kunde wurde erfolgreich angelegt

Kundendetails

[Neuen Kunden anlegen](#)



Allgemeine Daten

Anrede:	Frau	Sachwalter:	Altfrau Gertrud
Titel:	Dr.	Festnetz:	011010100
Vorname	Cesar	Mobil:	+43 660 234567
Nachname	Zehn	E-Mail:	cesar@zehn.at
Geburtsdatum	10.10.1950	Rechnung:	E-Mail

Angehörige

Angehöriger 1: keiner  Notfallkontakt

Rechnungsadresse

Straße: Nr: Stg: Stock: Tür: PLZ:

zehnerstraße	10	10	10	1000
--------------	----	----	----	------

Betreuungsadresse (wenn von Rechnungsadresse abweichend)

Betreuungsadresse abweichend von Rechnungsadresse

Wichtige Anmerkung

Anmerkung:

Anmerkung

**Datensatz anlegen**

Der neue Kunde wurde erfolgreich angelegt

Kundendetails

[Neuen Kunden anlegen](#)



**Dem Kunden wird ein Sachwalter hinzugefügt. Der Kunde hat ebenso einen Angehörigen.**

Ergebnis: erfolgreich

Details:

Allgemeine Daten

Anrede:	Herr	Sachwalter:	Altfrau Gertrude
Titel:	Titel	Festnetz:	017895456
Vorname	Max	Mobil:	+43 660 2345678
Nachname	Gernod	E-Mail:	max@gernod.at
Birthdate	15.07.1950	Rechnung:	E-Mail

Vertragsdaten

Aktiv:	<input type="checkbox"/>
Beginn	TT . MM . JJJJ
Ende	TT . MM . JJJJ
Verstorben	<input type="checkbox"/>

Angehörige

Angehöriger 1:	Stratosphäre Stefan
Angehöriger 2:	keiner
Angehöriger 3:	keiner

Rechnungsadresse

Rechnungsadresse ändern

Betreuungsadresse

Abweichende Betreuungsadresse

Wichtige Anmerkung

Anmerkung:  
Anmerkung

**Datensatz ändern** **Verwerfen**

Der Kunde wurde erfolgreich geändert

Kundendetails



Allgemeine Daten

Anrede:	Frau	Sachwalter:	Altfrau Gertrude
Titel:	Mag.	Festnetz:	017777777
Vorname	Maria	Mobil:	+43 660 2345678
Nachname	sieben	E-Mail:	maria@sieben.at
Geburtsdatum	07.07.1957	Rechnung:	Post

Vertragsdaten

Aktiv:	<input type="checkbox"/>
Beginn	TT . MM . JJJJ
Ende	TT . MM . JJJJ
Verstorben	<input type="checkbox"/>

Angehörige

Angehöriger 1:	Stratosphäre Stefan
Angehöriger 2:	keiner
Angehöriger 3:	keiner

Rechnungsadresse

Rechnungsadresse ändern

Betreuungsadresse

Abweichende Betreuungsadresse

Wichtige Anmerkung

Anmerkung:  
Anmerkung

**Datensatz ändern** **Verwerfen**

Der Kunde wurde erfolgreich geändert

Kundendetails



Allgemeine Daten

Anrede:	Herr	Sachwalter:	Hundertwasser Clara
Titel:	Titel	Festnetz:	014785963
Vorname	Gustav	Mobil:	+43 660 2345678
Nachname	Gans	E-Mail:	gustav@gans.at
Geburtsdatum	15. 06. 1950	Rechnung:	E-Mail

Vertragsdaten

Aktiv:	<input type="checkbox"/>
Beginn	TT . MM . JJJJ
Ende	TT . MM . JJJJ
Verstorben	<input type="checkbox"/>

Angehörige

Angehöriger 1:	Stratosphäre Stefan
Angehöriger 2:	keiner
Angehöriger 3:	keiner

Rechnungsadresse

Rechnungsadresse ändern

Betreuungsadresse

Abweichende Betreuungsadresse

Wichtige Anmerkung

Anmerkung:  
Anmerkung

**Datensatz ändern** **Verwerfen**

Der Kunde wurde erfolgreich geändert

Kundendetails



Allgemeine Daten

Anrede:	Herr	Sachwalter:	Hundertwasser Clara
Titel:	Mag.	Festnetz:	013333333
Vorname	Lenard	Mobil:	+43 660 2345678
Nachname	Drei	E-Mail:	lenard@drei.at
Geburtsdatum	30. 04. 1945	Rechnung:	E-Mail

Vertragsdaten

Aktiv:	<input type="checkbox"/>
Beginn	TT . MM . JJJJ
Ende	TT . MM . JJJJ
Verstorben	<input type="checkbox"/>

Angehörige

Angehöriger 1:	Stratosphäre Stefan
Angehöriger 2:	keiner
Angehöriger 3:	keiner

Rechnungsadresse

Rechnungsadresse ändern

Betreuungsadresse

Abweichende Betreuungsadresse

Wichtige Anmerkung

Anmerkung:  
Anmerkung

**Datensatz ändern** **Verwerfen**

Der Kunde wurde erfolgreich geändert

Kundendetails



Allgemeine Daten

Anrede:	Herr	Sachwalter:	Altfrau Gertrude
Titel:	Titel	Festnetz:	01 3434
Vorname	Peter	Mobil:	33434
Nachname	Großmann	E-Mail:	test@test.at
Geburtsdatum	14. 05. 1986	Rechnung:	E-Mail

Vertragsdaten

Aktiv:	<input checked="" type="checkbox"/>
Beginn	03. 02. 2020
Ende	TT. MM. JJJJ
Verstorben	<input type="checkbox"/>

Angehörige

Angehöriger 1:	Stratosphäre Stefan
Angehöriger 2:	keiner
Angehöriger 3:	keiner

Rechnungsadresse

Rechnungsadresse ändern

Betreuungsadresse

Abweichende Betreuungsadresse

Wichtige Anmerkung

Anmerkung:

Darf nach 18:00 nicht mehr angerufen werden

**Datensatz ändern** **Verwerfen**

Der Kunde wurde erfolgreich geändert

Kundendetails



**Der Kunde soll ab 18 Uhr nicht mehr angerufen werden. Dies soll im Kundenakt sofort ersichtlich sein.**

Ergebnis: erfolgreich

Details:

Allgemeine Daten

Anrede:	Herr	Sachwalter:	Altfrau Gertrude
Titel:	Titel	Festnetz:	01 3434
Vorname	Peter	Mobil:	33434
Nachname	Großmann	E-Mail:	test@test.at
Geburtsdatum	14. 05. 1986	Rechnung:	E-Mail

Vertragsdaten

Aktiv:	<input checked="" type="checkbox"/>
Beginn	03. 02. 2020
Ende	TT. MM. JJJJ
Verstorben	<input type="checkbox"/>

Angehörige

Angehöriger 1:	Stratosphäre Stefan
Angehöriger 2:	keiner
Angehöriger 3:	keiner

Rechnungsadresse

Rechnungsadresse ändern

Betreuungsadresse

Abweichende Betreuungsadresse

Wichtige Anmerkung

Anmerkung:  
Darf nicht nach 18:00 angerufen werden

**Datensatz ändern** **Verwerfen**

Der Kunde wurde erfolgreich geändert

Kundendetails



Allgemeine Daten

Anrede:	Frau	Sachwalter:	Barista Bernhard
Titel:	Titel	Festnetz:	014444444
Vorname	Anna	Mobil:	+43 660 2345678
Nachname	Vier	E-Mail:	anna@vier.at
Geburtsdatum	04.05.1959	Rechnung:	Post

Vertragsdaten

Aktiv:	<input type="checkbox"/>
Beginn	TT.MM.JJJJ
Ende	TT.MM.JJJJ
Verstorben	<input type="checkbox"/>

Angehörige

Angehöriger 1:	keiner
Angehöriger 2:	keiner
Angehöriger 3:	keiner

Rechnungsadresse

Rechnungsadresse ändern

Betreuungsadresse

Abweichende Betreuungsadresse

Wichtige Anmerkung

Anmerkung:

Darf nicht nach 18:00 angerufen werden!

**Datensatz ändern** **Verwerfen**

Der Kunde wurde erfolgreich geändert

Kundendetails



Allgemeine Daten

Anrede:	Herr	Sachwalter:	Altfrau Gertrude
Titel:	Titel	Festnetz:	017895456
Vorname	Max	Mobil:	+43 660 2345678
Nachname	Gernod	E-Mail:	max@gernod.at
Geburtsdatum	15.07.1950	Rechnung:	E-Mail

Vertragsdaten

Aktiv:	<input type="checkbox"/>
Beginn	TT . MM . JJJJ
Ende	TT . MM . JJJJ
Verstorben	<input type="checkbox"/>

Angehörige

Angehöriger 1:	Stratosphäre Stefan
Angehöriger 2:	keiner
Angehöriger 3:	keiner

Rechnungsadresse

Rechnungsadresse ändern

Betreuungsadresse

Abweichende Betreuungsadresse

Wichtige Anmerkung

Anmerkung:

Darf nicht nach 18:00 angerufen werden!

**Datensatz ändern** **Verwerfen**

Der Kunde wurde erfolgreich geändert

Kundendetails



Allgemeine Daten

Anrede:	Herr	Sachwalter:	Hundertwasser Clara
Titel:	Dr.	Festnetz:	014545367
Vorname	Harald	Mobil:	+43 660 2345678
Nachname	Potti	E-Mail:	harry@potti.com
Geburtsdatum	15.02.1984	Rechnung:	E-Mail

Vertragsdaten

Aktiv:	<input type="checkbox"/>
Beginn	TT . MM . JJJJ
Ende	TT . MM . JJJJ
Verstorben	<input type="checkbox"/>

Angehörige

Angehöriger 1:	Stratosphäre Stefan
Angehöriger 2:	keiner
Angehöriger 3:	keiner

Rechnungsadresse

Rechnungsadresse ändern

Betreuungsadresse

Abweichende Betreuungsadresse

Wichtige Anmerkung

Anmerkung:

Darf nicht nach 18:00 angerufen werden

**Datensatz ändern** **Verwerfen**

Der Kunde wurde erfolgreich geändert

Kundendetails



Allgemeine Daten

Anrede:	Frau	Sachwalter:	Altfrau Gertrude
Titel:	Dr.	Festnetz:	012222222
Vorname	Alex	Mobil:	0660222222
Nachname	Zwei	E-Mail:	alex@zwei.at
Geburtsdatum	04.05.1960	Rechnung:	Post

Vertragsdaten

Aktiv:	<input type="checkbox"/>
Beginn	TT . MM . JJJJ
Ende	TT . MM . JJJJ
Verstorben	<input type="checkbox"/>

Angehörige

Angehöriger 1:	keiner
Angehöriger 2:	keiner
Angehöriger 3:	keiner

Rechnungsadresse

Rechnungsadresse ändern

Betreuungsadresse

Abweichende Betreuungsadresse

Wichtige Anmerkung

Anmerkung:

Der Kunde wurde erfolgreich geändert

Datensatz ändern Verwerfen

Kundendetails



**Der Kunde ist verstorben. Es muss hinterlegt werden, wann dies passiert ist.**

Ergebnis: erfolgreich

Details:

Allgemeine Daten

Anrede:	Herr	Sachwalter:	Keiner
Titel:	Titel	Festnetz:	01111111
Vorname	Werner	Mobil:	0660 11111111
Nachname	Eins	E-Mail:	werner@eins.at
Geburtsdatum	11. 02. 1940	Rechnung:	E-Mail

Vertragsdaten

Aktiv:	<input type="checkbox"/>
Beginn	TT . MM . JJJJ
Ende	TT . MM . JJJJ
Verstorben	<input checked="" type="checkbox"/>

Der Kunde wurde erfolgreich geändert

Kundendetails

Allgemeine Daten

Anrede:	Frau	Sachwalter:	Maria Eva
Titel:	Titel	Festnetz:	019999999
Vorname	viktoria	Mobil:	+43 660 2345678
Nachname	neun	E-Mail:	viktoria@neun.at
Geburtsdatum	09. 09. 1959	Rechnung:	E-Mail

Vertragsdaten

Aktiv:	<input type="checkbox"/>
Beginn	TT . MM . JJJJ
Ende	TT . MM . JJJJ
Verstorben	<input checked="" type="checkbox"/>

Der Kunde wurde erfolgreich geändert

Kundendetails



Allgemeine Daten

Anrede:	Herr	Sachwalter:	Altfrau Gertrude
Titel:	Titel	Festnetz:	01 3434
Vorname	Peter	Mobil:	33434
Nachname	Großmann	E-Mail:	test@test.at
Geburtsdatum	14. 05. 1986	Rechnung:	E-Mail

Vertragsdaten

Aktiv:	<input checked="" type="checkbox"/>
Beginn	03. 02. 2020
Ende	TT. MM. JJJJ
Verstorben	<input checked="" type="checkbox"/>

Der Kunde wurde erfolgreich geändert

Kundendetails

Allgemeine Daten

Anrede:	Frau	Sachwalter:	Barista Bernhard
Titel:	Titel	Festnetz:	014444444
Vorname	Anna	Mobil:	+43 660 2345678
Nachname	Vier	E-Mail:	anna@vier.at
Geburtsdatum	04. 05. 1959	Rechnung:	Post

Vertragsdaten

Aktiv:	<input type="checkbox"/>
Beginn	TT. MM. JJJJ
Ende	TT. MM. JJJJ
Verstorben	<input checked="" type="checkbox"/>

Der Kunde wurde erfolgreich geändert

Kundendetails



Allgemeine Daten

Anrede:	Herr	Sachwalter:	Altfrau Gertrude
Titel:	Titel	Festnetz:	017895456
Vorname	Max	Mobil:	+43 660 2345678
Nachname	Gernod	E-Mail:	max@gernod.at
Geburtsdatum	15.07.1950	Rechnung:	E-Mail

Vertragsdaten

Aktiv:	<input type="checkbox"/>
Beginn	TT . MM . JJJJ
Ende	TT . MM . JJJJ
Verstorben	<input checked="" type="checkbox"/>

Der Kunde wurde erfolgreich geändert

Kundendetails



## 10.3.1.2. Pfleger

### Ein Pfleger wird angelegt

Ergebnis: erfolgreich

Details:

Allgemeine Daten					
Anrede:	Herr	Festnetz:	010111111		
Vorname	Clemens	Mobil:	+43 660 2345678		
Nachname	Eins	E-Mail:	clemens@eins.at		
Geburtsdatum	01 . 01 . 1951	Art:	<input checked="" type="radio"/> 24h Pflege <input type="radio"/> Stundenweise <input type="radio"/> DGKS/P		
Adresse					
Straße:	einserstraße	Nr:	1	Stg:	1
				Stock:	1
				Tür:	1
				PLZ:	1100
<a href="#">Datensatz anlegen</a>					

Der neue Pfleger wurde erfolgreich angelegt

Pflegerdetails

[Neuen Pfleger anlegen](#)



Allgemeine Daten

Anrede:	Frau	Festnetz:	010122222
Vorname	Linda	Mobil:	+43 660 2345678
Nachname	zwei	E-Mail:	linda@zwei.at
Geburtsdatum	02.02.1952	Art:	<input checked="" type="radio"/> 24h Pflege <input type="radio"/> Stundenweise <input type="radio"/> DGKS/P

Adresse

Straße:	zweistraße	Nr:	2	Stg:	2	Stock:	2	Tür:	2	PLZ:	1120
---------	------------	-----	---	------	---	--------	---	------	---	------	------

[Datensatz anlegen](#)

Der neue Pfleger wurde erfolgreich angelegt

Pflegerdetails

[Neuen Pfleger anlegen](#)

Allgemeine Daten

Anrede:	Herr	Festnetz:	010133333
Vorname	Alex	Mobil:	+43 660 2345678
Nachname	drei	E-Mail:	alex@drei.at
Geburtsdatum	03.03.1953	Art:	<input checked="" type="radio"/> 24h Pflege <input type="radio"/> Stundenweise <input type="radio"/> DGKS/P

Adresse

Straße:	dreistraße	Nr:	3	Stg:	3	Stock:	3	Tür:	3	PLZ:	1130
---------	------------	-----	---	------	---	--------	---	------	---	------	------

[Datensatz anlegen](#)

Der neue Pfleger wurde erfolgreich angelegt

Pflegerdetails

[Neuen Pfleger anlegen](#)



Allgemeine Daten

Anrede:	Frau	Festnetz:	0101444444
Vorname	Anja	Mobil:	+43 660 2345678
Nachname	vier	E-Mail:	anja@vier.at
Geburtsdatum	04. 04. 1954	Art:	<input checked="" type="radio"/> 24h Pflege <input type="radio"/> Stundenweise <input type="radio"/> DGKS/P

Adresse

Straße:	vierstraße	Nr:	4	Stg:	4	Stock:	4	Tür:	4	PLZ:	1140
---------	------------	-----	---	------	---	--------	---	------	---	------	------

[Datensatz anlegen](#)

Der neue Pfleger wurde erfolgreich angelegt

Pflegerdetails

[Neuen Pfleger anlegen](#)

Bitte die E-Mail Adresse im Format "name@beispiel.at" angeben!

Allgemeine Daten

Anrede:	Herr	Festnetz:	010155555
Vorname	karl	Mobil:	+43 660 2345678
Nachname	fünf	E-Mail:	karl@fuenf.at
Geburtsdatum	05. 05. 1955	Art:	<input checked="" type="radio"/> 24h Pflege <input type="radio"/> Stundenweise <input type="radio"/> DGKS/P

Adresse

Straße:	fünferstraße	Nr:	5	Stg:	5	Stock:	5	Tür:	5	PLZ:	1150
---------	--------------	-----	---	------	---	--------	---	------	---	------	------

[Datensatz anlegen](#)

Der neue Pfleger wurde erfolgreich angelegt

Pflegerdetails

[Neuen Pfleger anlegen](#)



## Ein Pfleger ändert seine Telefonnummer

Ergebnis: erfolgreich

Details:

Allgemeine Daten

Anrede:	Frau	Festnetz:	0123445678
Vorname	Maria	Mobil:	+43 660 2345678
Nachname	Duck	E-Mail:	maria@duck.at
Geburtsdatum	14.12.1948	Art:	<input checked="" type="radio"/> 24h Pflege <input type="radio"/> Stundenweise <input type="radio"/> DGKS/P
Beginn	TT . MM . JJJJ		
Ende	TT . MM . JJJJ		
aktiv	<input checked="" type="checkbox"/>		

Adresse

Rechnungsadresse ändern

**Datensatz ändern** **Verwerfen**

Der Pfleger wurde erfolgreich geändert

Pflegerdetails



Allgemeine Daten

Anrede:	Frau	Festnetz:	010155555
Vorname	Maria	Mobil:	+43 660 2345678
Nachname	Duck	E-Mail:	maria@duck.at
Geburtsdatum	14.12.1948	Art:	<input checked="" type="radio"/> 24h Pflege <input type="radio"/> Stundenweise <input type="radio"/> DGKS/P
Beginn	TT.MM.JJJJ		
Ende	TT.MM.JJJJ		
aktiv	<input checked="" type="checkbox"/>		

Adresse

Rechnungsadresse ändern

[Datensatz ändern](#) [Verwerfen](#)

Der Pfleger wurde erfolgreich geändert

Pflegerdetails

Allgemeine Daten

Anrede:	Herr	Festnetz:	0101444444
Vorname	abc	Mobil:	06604545676
Nachname	cda	E-Mail:	asd@asd.at
Geburtsdatum	16.06.2020	Art:	<input checked="" type="radio"/> 24h Pflege <input type="radio"/> Stundenweise <input type="radio"/> DGKS/P
Beginn	TT.MM.JJJJ		
Ende	TT.MM.JJJJ		
aktiv	<input checked="" type="checkbox"/>		

Adresse

Rechnungsadresse ändern

[Datensatz ändern](#) [Verwerfen](#)

Der Pfleger wurde erfolgreich geändert

Pflegerdetails



Allgemeine Daten

Anrede:	Herr	Festnetz:	010122222
Vorname	Clemens	Mobil:	+43 660 2345678
Nachname	Eins	E-Mail:	clemens@eins.at
Birthdate	01 . 01 . 1951	Art:	<input checked="" type="radio"/> 24h Pflege <input type="radio"/> Stundenweise <input type="radio"/> DGKS/P
Beginn	TT . MM . JJJJ		
Ende	TT . MM . JJJJ		
aktiv	<input checked="" type="checkbox"/>		

Adresse

Rechnungsadresse ändern

[Datensatz ändern](#) [Verwerfen](#)

Der Pfleger wurde erfolgreich geändert

Pflegerdetails

Allgemeine Daten

Anrede:	Herr	Festnetz:	0101444444
Vorname	Abraham	Mobil:	+43 660 2345678
Nachname	Allens	E-Mail:	aa@gmail.com
Birthdate	03 . 04 . 1977	Art:	<input checked="" type="radio"/> 24h Pflege <input type="radio"/> Stundenweise <input type="radio"/> DGKS/P
Beginn	TT . MM . JJJJ		
Ende	TT . MM . JJJJ		
aktiv	<input checked="" type="checkbox"/>		

Adresse

Rechnungsadresse ändern

[Datensatz ändern](#) [Verwerfen](#)

Der Pfleger wurde erfolgreich geändert

Pflegerdetails



Allgemeine Daten

Anrede:	Herr	Festnetz:	011055555
Vorname	Gernot	Mobil:	+43 660 2345678
Nachname	Grün	E-Mail:	gg@gmail.com
Geburtsdatum	03 . 04 . 1977	Art:	<input type="radio"/> 24h Pflege <input type="radio"/> Stundenweise <input checked="" type="radio"/> DGKS/p
Beginn	TT . MM . JJJJ		
Ende	TT . MM . JJJJ		
aktiv	<input checked="" type="checkbox"/>		

Adresse

Rechnungsadresse ändern

[Datensatz ändern](#) [Verwerfen](#)

Der Pfleger wurde erfolgreich geändert

[Pflegerdetails](#)



## Ein Pfleger wechselt von stundenweise Betreuung in den Pool der 24h-Betreuer

Ergebnis: erfolgreich

Details:

Allgemeine Daten

Anrede:	Frau	Festnetz:	010155555
Vorname	Maria	Mobil:	+43 660 2345678
Nachname	Duck	E-Mail:	maria@duck.at
Birthdate	14 . 12 . 1948	Art:	<input type="radio"/> 24h Pflege <input checked="" type="radio"/> Stundenweise <input type="radio"/> DGKS/P
Beginn	TT . MM . JJJJ		
Ende	TT . MM . JJJJ		
aktiv	<input checked="" type="checkbox"/>		

Adresse

Rechnungsadresse ändern

**Datensatz ändern** **Verwerfen**

Allgemeine Daten

Anrede:	Frau	Festnetz:	010155555
Vorname	Maria	Mobil:	+43 660 2345678
Nachname	Duck	E-Mail:	maria@duck.at
Birthdate	14 . 12 . 1948	Art:	<input checked="" type="radio"/> 24h Pflege <input type="radio"/> Stundenweise <input type="radio"/> DGKS/P
Beginn	TT . MM . JJJJ		
Ende	TT . MM . JJJJ		
aktiv	<input checked="" type="checkbox"/>		

Adresse

Rechnungsadresse ändern

**Datensatz ändern** **Verwerfen**



Der Pfleger wurde erfolgreich geändert

Pflegerdetails

Allgemeine Daten

Anrede:	Herr	Festnetz:	0101444444
Vorname	abc	Mobil:	06604545676
Nachname	cda	E-Mail:	asd@asd.at
Geburtsdatum	16 . 06 . 2020	Art:	<input checked="" type="radio"/> 24h Pflege <input type="radio"/> Stundenweise <input type="radio"/> DGKS/P
Beginn	TT . MM . JJJJ		
Ende	TT . MM . JJJJ		
aktiv	<input checked="" type="checkbox"/>		

Adresse

Rechnungsadresse ändern

[Datensatz ändern](#) [Verwerfen](#)

Der Pfleger wurde erfolgreich geändert

Pflegerdetails



Allgemeine Daten

Anrede:	Herr	Festnetz:	010122222
Vorname	Clemens	Mobil:	+43 660 2345678
Nachname	Eins	E-Mail:	clemens@eins.at
Geburtsdatum	01 . 01 . 1951	Art:	<input checked="" type="radio"/> 24h Pflege <input type="radio"/> Stundenweise <input type="radio"/> DGKS/P
Beginn	TT . MM . JJJJ		
Ende	TT . MM . JJJJ		
aktiv	<input checked="" type="checkbox"/>		

Adresse

Rechnungsadresse ändern

**Datensatz ändern** **Verwerfen**

Der Pfleger wurde erfolgreich geändert

Pflegerdetails

Allgemeine Daten

Anrede:	Herr	Festnetz:	0101444444
Vorname	Abraham	Mobil:	+43 660 2345678
Nachname	Allens	E-Mail:	aa@gmail.com
Geburtsdatum	03 . 04 . 1977	Art:	<input checked="" type="radio"/> 24h Pflege <input type="radio"/> Stundenweise <input type="radio"/> DGKS/P
Beginn	TT . MM . JJJJ		
Ende	TT . MM . JJJJ		
aktiv	<input checked="" type="checkbox"/>		

Adresse

Rechnungsadresse ändern

**Datensatz ändern** **Verwerfen**



Der Pfleger wurde erfolgreich geändert

Pflegerdetails

Allgemeine Daten

Anrede:	Herr	Festnetz:	01105555
Vorname	Gernot	Mobil:	+43 660 2345678
Nachname	Grün	E-Mail:	gg@gmail.com
Geburtsdatum	03 . 04 . 1977	Art:	<input checked="" type="radio"/> 24h Pflege <input type="radio"/> Stundenweise <input type="radio"/> DGKS/P
Beginn	TT . MM . JJJJ		
Ende	TT . MM . JJJJ		
aktiv	<input checked="" type="checkbox"/>		

Adresse

Rechnungsadresse ändern

[Datensatz ändern](#) [Verwerfen](#)

Der Pfleger wurde erfolgreich geändert

Pflegerdetails



## Ein Pfleger beendet die Zusammenarbeit. Er soll nicht mehr in Pflegeplänen aufscheinen.

Ergebnis: erfolgreich

Details:

Allgemeine Daten

Anrede:	Frau	Festnetz:	010155555
Vorname	Maria	Mobil:	+43 660 2345678
Nachname	Duck	E-Mail:	maria@duck.at
Geburtsdatum	14.12.1948	Art:	<input checked="" type="radio"/> 24h Pflege <input type="radio"/> Stundenweise <input type="radio"/> DGKS/P
Beginn	TT.MM.JJJJ		
Ende	TT.MM.JJJJ		
aktiv	<input checked="" type="checkbox"/>		

Adresse

Rechnungsadresse ändern

**Datensatz ändern** **Verwerfen**

Allgemeine Daten

Anrede:	Frau	Festnetz:	010155555
Vorname	Maria	Mobil:	+43 660 2345678
Nachname	Duck	E-Mail:	maria@duck.at
Geburtsdatum	14.12.1948	Art:	<input checked="" type="radio"/> 24h Pflege <input type="radio"/> Stundenweise <input type="radio"/> DGKS/P
Beginn	TT.MM.JJJJ		
Ende	TT.MM.JJJJ		
aktiv	<input type="checkbox"/>		

Adresse

Rechnungsadresse ändern

**Datensatz ändern** **Verwerfen**



Der Pfleger wurde erfolgreich geändert

Pflegerdetails

Allgemeine Daten

Anrede:	Herr	Festnetz:	0101444444
Vorname	abc	Mobil:	06604545676
Nachname	cda	E-Mail:	asd@asd.at
Geburtsdatum	16.06.2020	Art:	<input checked="" type="radio"/> 24h Pflege <input type="radio"/> Stundenweise <input type="radio"/> DGKS/P
Beginn	TT.MM.JJJJ		
Ende	TT.MM.JJJJ		
aktiv	<input type="checkbox"/>		

Adresse

Rechnungsadresse ändern

[Datensatz ändern](#) [Verwerfen](#)

Der Pfleger wurde erfolgreich geändert

Pflegerdetails



Allgemeine Daten

Anrede:	Herr	Festnetz:	010122222
Vorname	Clemens	Mobil:	+43 660 2345678
Nachname	Eins	E-Mail:	clemens@eins.at
Geburtsdatum	01 . 01 . 1951	Art:	<input checked="" type="radio"/> 24h Pflege <input type="radio"/> Stundenweise <input type="radio"/> DGKS/P
Beginn	TT . MM . JJJJ		
Ende	TT . MM . JJJJ		
aktiv	<input type="checkbox"/>		

Adresse

Rechnungsadresse ändern

**Datensatz ändern** **Verwerfen**

Der Pfleger wurde erfolgreich geändert

Pflegerdetails

Allgemeine Daten

Anrede:	Herr	Festnetz:	0101444444
Vorname	Abraham	Mobil:	+43 660 2345678
Nachname	Allens	E-Mail:	aa@gmail.com
Geburtsdatum	03 . 04 . 1977	Art:	<input checked="" type="radio"/> 24h Pflege <input type="radio"/> Stundenweise <input type="radio"/> DGKS/P
Beginn	TT . MM . JJJJ		
Ende	TT . MM . JJJJ		
aktiv	<input type="checkbox"/>		

Adresse

Rechnungsadresse ändern

**Datensatz ändern** **Verwerfen**



Der Pfleger wurde erfolgreich geändert

Pflegerdetails

Allgemeine Daten

Anrede:	Herr	Festnetz:	011055555
Vorname	Gernot	Mobil:	+43 660 2345678
Nachname	Grün	E-Mail:	gg@gmail.com
Geburtsdatum	03 . 04 . 1977	Art:	<input checked="" type="radio"/> 24h Pflege <input type="radio"/> Stundenweise <input type="radio"/> DGKS/P
Beginn	TT . MM . JJJJ		
Ende	TT . MM . JJJJ		
aktiv	<input type="checkbox"/>		

Adresse

Rechnungsaadresse ändern

[Datensatz ändern](#)

[Verwerfen](#)

Der Pfleger wurde erfolgreich geändert

Pflegerdetails



### 10.3.1.3. Sachwalter und Angehörige

#### Ein Sachwalter betreut mehrere Patienten

Ergebnis: erfolgreich

Details:

Allgemein	Vermerke	Pflegeplan	Tarife	Rechnungen
Allgemeine Daten				
Anrede:	<b>Herr</b>		Besachwaltet:	<b>ja</b>
Titel:	<b>Dr.</b>		Sachwalter:	<b>Frau Gertrude Altfrau</b>
Vorname:	<b>Harald</b>		Festnetz:	<b>014545367</b>
Nachname:	<b>Potti</b>		Mobil:	
Geburtsdatum:	<b>15.02.1984</b>		E-Mail:	<b>harry@potti.com</b>
Verstorben:	<b>nein</b>		Rechnung:	<b>Mail</b>
Allgemein	Vermerke	Pflegeplan	Tarife	Rechnungen
Allgemeine Daten				
Anrede:	<b>Herr</b>		Besachwaltet:	<b>ja</b>
Titel:			Sachwalter:	<b>Frau Gertrude Altfrau</b>
Vorname:	<b>Terry</b>		Festnetz:	<b>014556789</b>
Nachname:	<b>Scary</b>		Mobil:	
Geburtsdatum:	<b>04.05.1995</b>		E-Mail:	<b>terry@test.com</b>
Verstorben:	<b>nein</b>		Rechnung:	<b>Mail</b>
Allgemein	Vermerke	Pflegeplan	Tarife	Rechnungen
Allgemeine Daten				
Anrede:	<b>Frau</b>		Besachwaltet:	<b>ja</b>
Titel:	<b>Mag.</b>		Sachwalter:	<b>Frau Gertrude Altfrau</b>
Vorname:	<b>Maria</b>		Festnetz:	<b>017777777</b>
Nachname:	<b>sieben</b>		Mobil:	
Geburtsdatum:	<b>07.07.1957</b>		E-Mail:	<b>maria@sieben.at</b>
Verstorben:	<b>nein</b>		Rechnung:	<b>Post</b>



Allgemein   Vermerke   Pflegeplan   Tarife   Rechnungen

Allgemeine Daten

Anrede:	<b>Frau</b>	Beschwaltet:	<b>ja</b>
Titel:	<b>Dr.</b>	Sachwalter:	<b>Frau Gertrude Altfrau</b>
Vorname:	<b>Cesar</b>	Festnetz:	<b>011010100</b>
Nachname:	<b>Zehn</b>	Mobil:	
Geburtsdatum:	<b>10.10.1950</b>	E-Mail:	<b>cesar@zehn.at</b>
Verstorben:	<b>nein</b>	Rechnung:	<b>Mail</b>

Allgemein   Vermerke   Pflegeplan   Tarife   Rechnungen

Allgemeine Daten

Anrede:	<b>Herr</b>	Beschwaltet:	<b>ja</b>
Titel:		Sachwalter:	<b>Frau Gertrude Altfrau</b>
Vorname:	<b>Bernhard</b>	Festnetz:	<b>017895456</b>
Nachname:	<b>Sanders</b>	Mobil:	
Geburtsdatum:	<b>29.05.1995</b>	E-Mail:	<b>test@probe.at</b>
Verstorben:	<b>nein</b>	Rechnung:	<b>Mail</b>



## Ein Sachwalter wird angelegt und einem Patienten zugeordnet

Ergebnis: erfolgreich

Details:

Allgemeine Daten

Anrede:	Herr	Festnetz:	010101111
Titel:	Titel	Mobil:	+43 660 2345678
Vorname	Daniel	E-Mail:	daniel@eins.at
Nachname	Eins	Rechnung:	Post
Geburtsdatum	01.11.1951		

Adresse

Straße:	einerstraße	Nr:	1	Stg:	1	Stock:	1	Tür:	1	PLZ:	1100
---------	-------------	-----	---	------	---	--------	---	------	---	------	------

Kunde

Herr Doom Viktor
------------------

[Datensatz anlegen](#)

Der neue Pfleger wurde erfolgreich angelegt

Sachwalterdetails

[Neuen Sachwalter anlegen](#)



Allgemeine Daten

Anrede:	Frau	Festnetz:	010102222
Titel:	Dr.	Mobil:	+43 660 2345678
Vorname	Emy	E-Mail:	emy@zwei.at
Nachname	Zwei	Rechnung:	E-Mail
Geburtsdatum	02.02.1952		

Adresse

Straße:	zweitestraße	Nr:	2	Stg:	2	Stock:	2	Tür:	2	PLZ:	1120
---------	--------------	-----	---	------	---	--------	---	------	---	------	------

Kunde

Herr Acht Manuel
------------------

[Datensatz anlegen](#)

Der neue Pfleger wurde erfolgreich angelegt

Sachwalterdetails

[Neuen Sachwalter anlegen](#)

Allgemeine Daten

Anrede:	Herr	Festnetz:	010103333
Titel:	Mag.	Mobil:	+43 660 2345678
Vorname	Hanes	E-Mail:	hanes@drei.at
Nachname	Drei	Rechnung:	E-Mail
Geburtsdatum	03.03.1953		

Adresse

Straße:	dreierstraße	Nr:	3	Stg:	3	Stock:	3	Tür:	3	PLZ:	1130
---------	--------------	-----	---	------	---	--------	---	------	---	------	------

Kunde

Herr Drei Lenard
------------------

[Datensatz anlegen](#)



Der neue Pfleger wurde erfolgreich angelegt

Sachwalterdetails

[Neuen Sachwalter anlegen](#)

Allgemeine Daten

Anrede:	Frau	Festnetz:	010104444
Titel:	Titel	Mobil:	+43 660 2345678
Vorname	Lisa	E-Mail:	lisa@vier.at
Nachname	Vier	Rechnung:	E-Mail
Geburtsdatum	04.04.1954		

Adresse

Straße:	viererstraße	Nr:	4	Stg:	4	Stock:	4	Tür:	4	PLZ:	1140
---------	--------------	-----	---	------	---	--------	---	------	---	------	------

Kunde

Herr Frank Josef

[Datensatz anlegen](#)

Der neue Pfleger wurde erfolgreich angelegt

Sachwalterdetails

[Neuen Sachwalter anlegen](#)



Allgemeine Daten

Anrede:	Herr	Festnetz:	01010555
Titel:	Titel	Mobil:	+43 660 2345678
Vorname	Nils	E-Mail:	nils@fuenf.at
Nachname	fünf	Rechnung:	E-Mail
Geburtsdatum	05.05.1955		

Adresse

Straße:	fünferstraße	Nr:	5	Stg:	5	Stock:	5	Tür:	5	PLZ:	1150
---------	--------------	-----	---	------	---	--------	---	------	---	------	------

Kunde

Herr Gans Gustav
------------------

[Datensatz anlegen](#)

Der neue Pfleger wurde erfolgreich angelegt

[Sachwalterdetails](#)

[Neuen Sachwalter anlegen](#)



## Ein Angehöriger ändert seine Adresse

Ergebnis: erfolgreich

Details:

Allgemeine Daten

Anrede:	Herr	Festnetz:	010101111
Titel:	Titel	Mobil:	+43 660 2345678
Vorname	Markus	E-Mail:	markus@eins.at
Nachname	Eins	Rechnung:	E-Mail
Geburtsdatum	01.01.1951		

Adresse

Rechnungsadresse ändern

Straße:	einserstraße	Nr:	1	Stg:	1	Stock:	1	Tür:	1	PLZ:	1100
---------	--------------	-----	---	------	---	--------	---	------	---	------	------

Allgemeine Daten

Anrede:	Herr	Festnetz:	010101111
Titel:	Titel	Mobil:	+43 660 2345678
Vorname	Markus	E-Mail:	markus@eins.at
Nachname	Eins	Rechnung:	E-Mail
Geburtsdatum	01.01.1951		

Adresse

Rechnungsadresse ändern

Straße:	zweierstr	Nr:	1	Stg:	1	Stock:	1	Tür:	1	PLZ:	1100
---------	-----------	-----	---	------	---	--------	---	------	---	------	------

Angehöriger Kunde

Kundenzuordnung bitte bei den einzelnen Kunden ändern.

[Datensatz ändern](#) [Verwerfen](#)

Der Sachwalter wurde erfolgreich geändert

[Sachwalterdetails](#)



Allgemeine Daten

Anrede:	Herr	Festnetz:	010105555
Titel:	Titel	Mobil:	+43 660 2345678
Vorname	Ken	E-Mail:	ken@fuenf.at
Nachname	fünf	Rechnung:	E-Mail
Geburtsdatum	05 . 05 . 1955		

Adresse

Rechnungsadresse ändern

Straße:	viererstraße	Nr:	5	Stg:	5	Stock:	5	Tür:	5	PLZ:	1150
---------	--------------	-----	---	------	---	--------	---	------	---	------	------

Angehöriger Kunde

Kundenzuordnung bitte bei den einzelnen Kunden ändern.

[Datensatz ändern](#) [Verwerfen](#)

Der Sachwalter wurde erfolgreich geändert

[Sachwalterdetails](#)



Allgemeine Daten

Anrede:	Herr	Festnetz:	010120222
Titel:	Titel	Mobil:	+43 660 2345678
Vorname	ben	E-Mail:	ben@zwei.at
Nachname	zwei	Rechnung:	E-Mail
Geburtsdatum	02.02.1952		

Adresse

Rechnungsadresse ändern

Straße:	Manngasse	Nr:	2	Stg:	2	Stock:	1	Tür:	2	PLZ:	1120
---------	-----------	-----	---	------	---	--------	---	------	---	------	------

Angehöriger Kunde

Kundenzuordnung bitte bei den einzelnen Kunden ändern.

[Datensatz ändern](#) [Verwerfen](#)

Der Sachwalter wurde erfolgreich geändert

Sachwalterdetails



## Ein Angehöriger wird mit Namen und Adresse gesucht - und gefunden

Ergebnis: erfolgreich

Details:

Suchbegriff: andi

Suchen in  Nachname  
 Vorname  
 Adresse

**suchen**

Es wurden folgende Datensätze gefunden:

- Andi drei, dreistraße, 1130

[Neue Suche](#)

Es wurden folgende Datensätze gefunden:

- Betti Vier, vierstraße, 1140

[Neue Suche](#)

Es wurden folgende Datensätze gefunden:

- Clara Hundertwasser, Mariahilfer Straße, 1150

[Neue Suche](#)



Es wurden folgende Datensätze gefunden:

- Gertrude Altfrau, Megastraße, 1110

[Neue Suche](#)

Es wurden folgende Datensätze gefunden:

- Andi drei, dreistraße, 1130

[Neue Suche](#)

Keine Treffer für "einserstraße". in Sachwalter

Suchbegriff:

einserstraße

Suchen in

- Nachname  
 Vorname  
 Adresse

[suchen](#)

Es wurden folgende Datensätze gefunden:

- Daniel Eins, einerstraße, 1100

[Neue Suche](#)



Es wurden folgende Datensätze gefunden:

- Emy Zwei, zweitestraße, 1120

[Neue Suche](#)

Es wurden folgende Datensätze gefunden:

- Hanes Drei, dreierstraße, 1130

[Neue Suche](#)

Es wurden folgende Datensätze gefunden:

- Betti Vier, vierstraße, 1140

[Neue Suche](#)

Es wurden folgende Datensätze gefunden:

- Nils fünf, fünferstraße, 1150

[Neue Suche](#)



#### 10.3.1.4. Pflegeplan

**Es wird für einen Kunden ein zusätzlicher Hausbesuch zur Qualitätssicherung angelegt.**

Ergebnis: erfolgreich

Details:

Screenshot of the care CRM software interface showing the creation of a house visit for a customer named Scary Terry.

The top section shows a list of customers with their names, addresses, and the dates of their last three house visits. A green plus sign icon is next to each row, indicating the option to add a new visit.

Kunde	Hausbesuche	Neuer HB
Peter Großmann, Luise Montag Gasse 1, 1110	01.06.2020 15.06.2020 18.06.2020	+ (green)
Terry Scary, Scarystr 2/3/4/8, 1100	keine	+ (green)
Harald Potti, Winkelgasse 123/4/5/6, 1234	17.06.2020	+ (green)
Cesar Zehn, zehnerstraße 10/10/10/10, 1100	keine	+ (green)
Bernhard Mustermann, Beispielstr 12/3/2/13, 1100	keine	+ (green)
Viktor Doom, Winkelgasse 45/3/2/4, 1100	17.06.2020 17.06.2020 17.06.2020	+ (green)
Bernie Sanders, wienerbergstraße 123/4/3/2, 1100	keine	+ (green)
Mario Musterman, Anton-Langer 66/3, 1130	08.06.2020 15.06.2020 16.06.2020	+ (green)

The middle section is a form titled "Hausbesuch" (House Visit) with fields for "Kunde:" (Customer: Scary Terry), "DGKS/P" (Dienstleister: Duck Daisy), and "Datum:" (Date: 26.06.2020). A large blue button at the bottom of the form says "Hausbesuch anlegen" (Create House Visit).

The bottom section displays a success message: "Die neue Tätigkeit wurde erfolgreich angelegt" (The new activity was successfully created). It also includes links to "Zum Kundenakt" (Customer Activity) and "Zum Pflegerakt" (Caregiver Activity), and a link to "Neue Tätigkeit anlegen" (Create New Activity).



Hausbesuch

Kunde:	Zehn Cesar
DGKS/P:	Duck Maria
Datum:	21.10.2020 <input type="button" value="X"/>

[Hausbesuch anlegen](#)

Die neue Tätigkeit wurde erfolgreich angelegt

[Zum Kundenakt](#)

[Zum Pflegerakt](#)

[Neue Tätigkeit anlegen](#)

Hausbesuch

Kunde:	Zehn Cesar
DGKS/P:	Duck Donald
Datum:	13.04.2021 <input type="button" value="X"/>

[Hausbesuch anlegen](#)



Die neue Tätigkeit wurde erfolgreich angelegt

[Zum Kundenakt](#)

[Zum Pflegerakt](#)

[Neue Tätigkeit anlegen](#)

#### Hausbesuch

Kunde:

Sanders Bernie ▾

DGKS/P

Eins Clemens ▾

Datum:

13 . 11 . 2020

[Hausbesuch anlegen](#)

Die neue Tätigkeit wurde erfolgreich angelegt

[Zum Kundenakt](#)

[Zum Pflegerakt](#)

[Neue Tätigkeit anlegen](#)



**Es wird eine Tätigkeit eines 24h Pflegers über 3 Monate angelegt. Diese Leistung soll beim Kunden in allen 3 Monaten als aktiv aufscheinen.**

Ergebnis: erfolgreich

Details:

Tätigkeit	
Kunde:	Doom Viktor
Pfleger	Duck Daisy
Beginn:	17.06.2020 <input type="button" value="X"/>
Ende:	<input type="checkbox"/> offen 17.09.2020 <input type="button" value="X"/>

**Tätigkeit anlegen**

Allgemein	Vermerke	Pflegeplan	Tarife	Rechnungen
Jahr/Monat:	2020 <input type="button" value="▼"/> 06 <input type="button" value="▼"/>			
Start	Ende	Pfleger	Art	Details
Mi 17.06.2020	Do 17.09.2020	Daisy Duck	24h Betreuung	<a href="#">Details</a>
Mi 17.06.2020	Do 17.09.2020	Daisy Duck	24h Betreuung	<a href="#">Details</a>

**Neue Tätigkeit hinzufügen**

Die neue Tätigkeit wurde erfolgreich angelegt

[Zum Kundenakt](#)

[Zum Pflegerakt](#)

[Neue Tätigkeit anlegen](#)



**Tätigkeit**

Kunde:	Großmann Peter
Pfleger	Allens Abraham
Beginn:	17.06.2020
Ende:	<input type="checkbox"/> offen 17.09.2020

**Tätigkeit anlegen**

Allgemein Vermerke Pflegeplan Tarife Rechnungen

Jahr/Monat: 2020 06

Start	Ende	Pfleger	Art	Details
Mo 01.06.2020	Mo 01.06.2020	Gernot Grün	24h Betreuung	Details
Mo 15.06.2020	Mo 15.06.2020	Gernot Grün	24h Betreuung	Details
Mi 17.06.2020	Do 17.09.2020	Abraham Allens	24h Betreuung	Details

**Neue Tätigkeit hinzufügen**

Die neue Tätigkeit wurde erfolgreich angelegt

[Zum Kundenakt](#)

[Zum Pflegerakt](#)

[Neue Tätigkeit anlegen](#)



**Tätigkeit**

Kunde:	Zehn Cesar
Pfleger:	Allens Abraham
Beginn:	17.06.2020 <span style="color: red;">×</span>
Ende:	<input type="checkbox"/> offen 17.09.2020 <span style="color: red;">×</span>

**Tätigkeit anlegen**

Allgemein Vermerke Pflegeplan Tarife Rechnungen

Jahr/Monat: 2020 06

Start	Ende	Pfleger	Art	Details
Mi 17.06.2020		Max Mann	24h Betreuung	<a href="#">Details</a>
Mi 17.06.2020	Do 17.09.2020	Abraham Allens	24h Betreuung	<a href="#">Details</a>

[Neue Tätigkeit hinzufügen](#)

**Die neue Tätigkeit wurde erfolgreich angelegt**

[Zum Kundenakt](#)

[Zum Pflegerakt](#)

[Neue Tätigkeit anlegen](#)



Tätigkeit

Kunde:	Sanders Bernie
Pfleger	Allens Abraham
Beginn:	17.06.2020 <span style="color: red;">×</span>
Ende:	<input type="checkbox"/> offen 17.09.2020 <span style="color: red;">×</span>

[Tätigkeit anlegen](#)

Allgemein	Vermerke	Pflegeplan	Tarife	Rechnungen
Jahr/Monat:	2020 <span style="border: 1px solid #ccc; padding: 0 5px;"> </span> 06 <span style="border: 1px solid #ccc; padding: 0 5px;"> </span>			
Start	Ende	Pfleger	Art	Details
Mi 17.06.2020	Do 17.09.2020	Abraham Allens	24h Betreuung	<a href="#">Details</a>

[Neue Tätigkeit hinzufügen](#)

Die neue Tätigkeit wurde erfolgreich angelegt

[Zum Kundenakt](#)

[Zum Pflegerakt](#)

[Neue Tätigkeit anlegen](#)



Tätigkeit

Kunde:	Trude Gerti
Pfleger	Allens Abraham
Beginn:	17.06.2020
Ende:	<input type="checkbox"/> offen 17.09.2020

**Tätigkeit anlegen**

Allgemein	Vermerke	Pflegeplan	Tarife	Rechnungen
Jahr/Monat:	2020  06			
<b>Start</b>	<b>Ende</b>	<b>Pfleger</b>	<b>Art</b>	<b>Details</b>
Mi 17.06.2020	Do 17.09.2020	Abraham Allens	24h Betreuung	<a href="#">Details</a>
<a href="#">Neue Tätigkeit hinzufügen</a>				



**Ein angelegter Hausbesuch wurde fehlerhaft erstellt. Dieser soll gelöscht werden.**

Ergebnis: erfolgreich

Details:

Tätigkeit Nr. 19

Kunde:	Doom Viktor	<a href="#">zum Kundenakt</a>
Pfleger:	Duck Daisy	<a href="#">zum Pflegerakt</a>
Beginn:	17.06.2020 <span style="color: red;">×</span>	
Ende:	<input type="checkbox"/> offen 17.09.2020 <span style="color: red;">×</span>	

[Ändern](#) [Löschen](#)

Der Eintrag wurde erfolgreich gelöscht.

Tätigkeit Nr. 8

Kunde:	Musterman Mario	<a href="#">zum Kundenakt</a>
Pfleger:	Grün Gernot	<a href="#">zum Pflegerakt</a>
Beginn:	15.06.2020 <span style="color: red;">×</span>	
Ende:	<input type="checkbox"/> offen 15.06.2020 <span style="color: red;">×</span>	

[Ändern](#) [Löschen](#)

Der Eintrag wurde erfolgreich gelöscht.



Tätigkeit Nr. 27

Kunde:	Scary Terry	<input type="button" value="zum Kundenakt"/>
Pfleger:	Duck Daisy	<input type="button" value="zum Pflegerakt"/>
Beginn:	26.06.2020 <input type="button" value="X"/>	
Ende:	<input type="checkbox"/> offen	26.06.2020 <input type="button" value="X"/>

Der Eintrag wurde erfolgreich gelöscht.

Tätigkeit Nr. 27

Kunde:	Scary Terry	<input type="button" value="zum Kundenakt"/>
Pfleger:	Duck Daisy	<input type="button" value="zum Pflegerakt"/>
Beginn:	26.06.2020 <input type="button" value="X"/>	
Ende:	<input type="checkbox"/> offen	26.06.2020 <input type="button" value="X"/>

Der Eintrag wurde erfolgreich gelöscht.



Tätigkeit Nr. 24

Kunde:	Doom Viktor	<a href="#">zum Kundenakt</a>
Pfleger:	Duck Daisy	<a href="#">zum Pflegerakt</a>
Beginn:	17.06.2020 <span style="border: 1px solid black; padding: 0 2px;">×</span>	
Ende:	<input type="checkbox"/> offen	17.06.2020 <span style="border: 1px solid black; padding: 0 2px;">×</span>

[Ändern](#) [Löschen](#)

Der Eintrag wurde erfolgreich gelöscht.



**Der Pflegeplan eines Kunden des aktuellen Monats wird angezeigt. Es sollen alle Pflegekräfte, welche dem Kunden zugeordnet sind, angezeigt werden, die diesen Monat dort aktiv sind.**

Ergebnis: erfolgreich

Details:

Screenshot of a software interface showing a Pflegeplan (Care Plan) for June 2020. The interface includes tabs for Allgemein, Vermerke, Pflegeplan (selected), Tarife, and Rechnungen. A date selector shows 2020-06. The main table displays two entries:

Start	Ende	Pfleger	Art	Details
Mo 01.06.2020	Mo 01.06.2020	Gernot Grün	24h Betreuung	<a href="#">Details</a>
Mo 15.06.2020	Mo 15.06.2020	Gernot Grün	24h Betreuung	<a href="#">Details</a>

A blue button at the bottom left says "Neue Tätigkeit hinzufügen" (Add new activity).

Screenshot of a software interface showing a Pflegeplan (Care Plan) for July 2020. The interface includes tabs for Allgemein, Vermerke, Pflegeplan (selected), Tarife, and Rechnungen. A date selector shows 2020-07. The main table displays one entry:

Start	Ende	Pfleger	Art	Details
Mi 01.07.2020		Abraham Allens	24h Betreuung	<a href="#">Details</a>

A blue button at the bottom left says "Neue Tätigkeit hinzufügen" (Add new activity).

Screenshot of a software interface showing a Pflegeplan (Care Plan) for June 2020. The interface includes tabs for Allgemein, Vermerke, Pflegeplan (selected), Tarife, and Rechnungen. A date selector shows 2020-06. The main table displays four entries:

Start	Ende	Pfleger	Art	Details
Mo 08.06.2020	Mo 08.06.2020	Gernot Grün	24h Betreuung	<a href="#">Details</a>
Mo 15.06.2020		Abraham Allens	24h Betreuung	<a href="#">Details</a>
Mo 15.06.2020	Mo 15.06.2020	Gernot Grün	24h Betreuung	<a href="#">Details</a>
Di 16.06.2020	Di 16.06.2020	Gernot Grün	24h Betreuung	<a href="#">Details</a>

A blue button at the bottom left says "Neue Tätigkeit hinzufügen" (Add new activity).



Allgemein	Vermerke	Pflegeplan	Tarife	Rechnungen
Jahr/Monat: 2020 07				
Start	Ende	Pfleger	Art	Details
Mo 15.06.2020		Abraham Allens	24h Betreuung	<a href="#">Details</a>
<a href="#">Neue Tätigkeit hinzufügen</a>				
Allgemein	Vermerke	Pflegeplan	Tarife	Rechnungen
Jahr/Monat: 2020 06				
Start	Ende	Pfleger	Art	Details
Mi 17.06.2020	Do 17.09.2020	Daisy Duck	24h Betreuung	<a href="#">Details</a>
<a href="#">Neue Tätigkeit hinzufügen</a>				



### 10.3.1.5. Verrechnung

**Es soll eine Rechnung für einen Kunden erstellt werden, mit welcher sämtliche diesen Monat noch nicht verrechneten Leistungen verrechnet werden**

Ergebnis: erfolgreich

Details:

Kunde: Doom Viktor, Winkelgasse 45/3/2/4, 1100 ▼

Weiter

Kundendaten  
Herr Viktor Doom

Rechnungsadresse

(Kunde) Herr Viktor Doom  
Winkelgasse 45/3/2/4  
1100 Wien

(Sachwalter) Herr Viktor Doom  
p.A. Herr Daniel Eins  
einerstraße 1/1/1/1  
1100 Wien

Leistung

Bezeichnung	Einheit	Preis/Einh.	zuletzt verrechnet	verrechnen	Anzahl
Leistung1 (2020)	Stunde	40.65 €	31.05.2020	<input checked="" type="checkbox"/>	1

Leistungszeitraum  
01 . 05 . 2020 bis 31 . 05 . 2020

Rechnung erstellen

Die Rechnung wurde erfolgreich erstellt.

Details  
als PDF öffnen



Kunde: Großmann Peter, Luise Montag Gasse 1, 1110

Weiter

Kundendaten  
Herr Peter Großmann

Rechnungsadresse  
 (Kunde) Herr Peter Großmann  
Luise Montag Gasse 1  
1110 Wien  
 (Sachwalter) Herr Peter Großmann  
p.A. Mag. Gertrude Altfrau  
Megastraße 123/2  
1110 Wien

Leistung

Bezeichnung	Einheit	Preis/Einh.	zuletzt verrechnet	verrechnen	Anzahl
Leistung3 (2020)	Woche	152.43 €	nie	<input checked="" type="checkbox"/>	1
Leistung1 (2020)	Stunde	40.65 €	nie	<input checked="" type="checkbox"/>	1
Leistung4 (2020)	Monat	254.05 €	nie	<input checked="" type="checkbox"/>	1
Leistung2 (2020)	Tag	100.60 €	nie	<input checked="" type="checkbox"/>	1
Leistung5 (2020)	Stück	30.49 €	nie	<input checked="" type="checkbox"/>	1

Leistungszeitraum  
01.05.2020 bis 31.05.2020

Rechnung erstellen

Die Rechnung wurde erfolgreich erstellt.

Details  
als PDF öffnen



Kunde: Musterman Mario, Anton-Langer 66/3, 1130 ▼

[Weiter](#)

Die Rechnung wurde erfolgreich erstellt.

[Details](#)  
[als PDF öffnen](#)

Kunde: Mustermann Bernhard, Beispielstr 12/3/2/13, 1100 ▼

[Weiter](#)

Die Rechnung wurde erfolgreich erstellt.

[Details](#)  
[als PDF öffnen](#)



Kunde: Potti Harald, Winkelgasse 123/4/5/6, 1234 ▼

**Weiter**

**Kundendaten**

**Rechnungsadresse**  
 (Kunde)  (Sachwalter)

p.A. Mag. Gertrude Altfrau  
Megastraße 123/2  
1110 Wien

**Leistung**

Bezeichnung	Einheit	Preis/Einh.	zuletzt verrechnet	verrechnen	Anzahl
Leistung14 (2020)	Stunde	79.26 €	nie	<input checked="" type="checkbox"/>	2 <span style="border: 1px solid #ccc; padding: 2px;">▼</span>
Leistung15 (2020)	Monat	1422.68 €	nie	<input checked="" type="checkbox"/>	1 <span style="border: 1px solid #ccc; padding: 2px;">▼</span>
Leistung12 (2020)	Monat	304.86 €	nie	<input checked="" type="checkbox"/>	1 <span style="border: 1px solid #ccc; padding: 2px;">▼</span>
Leistung13 (2020)	Stück	81.30 €	nie	<input checked="" type="checkbox"/>	3 <span style="border: 1px solid #ccc; padding: 2px;">▼</span>
Leistung11 (2020)	Woche	152.43 €	nie	<input checked="" type="checkbox"/>	1 <span style="border: 1px solid #ccc; padding: 2px;">▼</span>

**Leistungszeitraum**  
01.05.2020 bis 31.05.2020

**Rechnung erstellen**

**Die Rechnung wurde erfolgreich erstellt.**

[Details als PDF öffnen](#)



**Diese Leistungen sollten bei der nächsten Verrechnung als "bereits verrechnet bis [Datum]" angezeigt werden, damit diese nicht doppelt verrechnet werden.**

Ergebnis: erfolgreich

Details:

Offene Rechnungen:						
Nr	Datum	Pos	Summe (€)	PDF	Details	bezahlt
0019-2020	18.06.2020	0	0,00	<a href="#">PDF</a>	<a href="#">details</a>	<input checked="" type="checkbox"/> bezahlt
0018-2020	18.06.2020	0	0,00	<a href="#">PDF</a>	<a href="#">details</a>	<input checked="" type="checkbox"/> bezahlt
0017-2020	18.06.2020	0	0,00	<a href="#">PDF</a>	<a href="#">details</a>	<input checked="" type="checkbox"/> bezahlt
0016-2020	18.06.2020	1	150,00	<a href="#">PDF</a>	<a href="#">details</a>	<input checked="" type="checkbox"/> bezahlt
0015-2020	18.06.2020	5	1.883,02	<a href="#">PDF</a>	<a href="#">details</a>	<input checked="" type="checkbox"/> bezahlt

Offene Rechnungen:						
Nr	Datum	Pos	Summe (€)	PDF	Details	bezahlt
0019-2020	18.06.2020	0	0,00	<a href="#">PDF</a>	<a href="#">details</a>	<input checked="" type="checkbox"/> bezahlt
0018-2020	18.06.2020	0	0,00	<a href="#">PDF</a>	<a href="#">details</a>	<input checked="" type="checkbox"/> bezahlt
0017-2020	18.06.2020	0	0,00	<a href="#">PDF</a>	<a href="#">details</a>	<input checked="" type="checkbox"/> bezahlt
0016-2020	18.06.2020	1	150,00	<a href="#">PDF</a>	<a href="#">details</a>	<input checked="" type="checkbox"/> bezahlt
0015-2020	18.06.2020	5	1.883,02	<a href="#">PDF</a>	<a href="#">details</a>	<input checked="" type="checkbox"/> bezahlt



**Eine Rechnung für einen Kunden wurde fehlerhaft erstellt. Sie soll per Knopfdruck gutgeschrieben werden.**

Ergebnis: erfolgreich

Details:

Jahr:	2020	<input type="button" value="▼"/>				
Offene Rechnungen:						
Nr	Datum	Pos	Summe (€)	PDF	Details	bezahlt
0014-2020	18.06.2020	5	1.883,02	<a href="#">PDF</a>	<a href="#">details</a>	<input type="checkbox"/> bezahlt

Rechnungsdaten:				
Rechnungsnummer: 0014-2020				
Rechnungsdatum: 18.06.2020				
Leistungszeitraum: 01.05.2020 bis 31.05.2020				
Kunde:				
Kundennummer: 8				
Rechnungsanschrift: Dr. Mario Musterman				
Luise Montag 1				
1110 Wien				
Positionen:				
Bezeichnung	Anzahl	Einheit	Einzelpreis (€)	Gesamt (€)
Leistung25	1	h	25,41	25,41
Leistung23	1	h	79,26	79,26
Leistung24	1	m	1.473,49	1.473,49
Leistung21	1	s	152,43	152,43
Leistung22	1	e	152,43	152,43
			<b>Gesamt:</b>	<b>1.883,02</b>
<a href="#">PDF öffnen</a> <a href="#">Rechnung stornieren</a>				

Die Gutschrift wurde erstellt.

[Rechnung ansehen \(PDF\)](#) [Gutschrift ansehen \(PDF\)](#)

GS-0016-2020	18.06.2020	1	-150,00	PDF	details	<input checked="" type="checkbox"/> bezahlt
0016-2020	18.06.2020	1	150,00	PDF	details	<input checked="" type="checkbox"/> bezahlt



Rechnungsdaten:				
Rechnungsnummer: 0016-2020				
Rechnungsdatum: 18.06.2020				
Leistungszeitraum: 01.05.2020 bis 31.05.2020				
Kunde:				
Kundennummer: 13				
Rechnungsanschrift: Herr Bernie Sanders 1600 Pennsylvania Ave NW 1600/1/E/1 2500 Baden				
Positionen:				
Bezeichnung	Anzahl	Einheit	Einzelpreis (€)	Gesamt (€)
Qualitätssicherung der 24h Betreuung	1	b	150,00	150,00
			<b>Gesamt:</b>	<b>150,00</b>
<a href="#">PDF öffnen</a> <a href="#">Rechnung stornieren</a>				

Die Gutschrift wurde erstellt.

[Rechnung ansehen \(PDF\)](#) [Gutschrift ansehen \(PDF\)](#)

GS-0014-2020	18.06.2020	5	-1.883,02	<a href="#">PDF</a>	<a href="#">details</a>	<input checked="" type="checkbox"/> bezahlt
0014-2020	18.06.2020	5	1.883,02	<a href="#">PDF</a>	<a href="#">details</a>	<input checked="" type="checkbox"/> bezahlt



Rechnungsdaten:				
Rechnungsnummer:	0007-2020			
Rechnungsdatum:	15.06.2020			
Leistungszeitraum:	01.05.2020 bis 31.05.2020			
Kunde:				
Kundennummer:	8			
Rechnungsanschrift:	Dr. Mario Musterman Luise Montag 1 1110 Wien			
Positionen:				
Bezeichnung	Anzahl	Einheit	Einzelpreis (€)	Gesamt (€)
Qualitätssicherung der 24h Betreuung	1	b	225,00	225,00
Qualitätssicherung der 24h Betreuung	1	b	225,00	225,00
Stundenweise Betreuung	1	h	85,20	85,20
			<b>Gesamt:</b>	<b>535,20</b>
<a href="#">PDF öffnen</a> <a href="#">Rechnung stornieren</a>				

Die Gutschrift wurde erstellt.

[Rechnung ansehen \(PDF\)](#) [Gutschrift ansehen \(PDF\)](#)

GS-0007-2020	18.06.2020	3	-535,20	<a href="#">PDF</a>	<a href="#">details</a>	<input checked="" type="checkbox"/> bezahlt
0007-2020	15.06.2020	3	535,20	<a href="#">PDF</a>	<a href="#">details</a>	<input checked="" type="checkbox"/> bezahlt



Rechnungsdaten:				
Rechnungsnummer:	0002-2020			
Rechnungsdatum:	10.06.2020			
Leistungszeitraum:	01.05.2020 bis 31.05.2020			
Kunde:				
Kundennummer:	8			
Rechnungsanschrift:	Dr. Mario Musterman p.A. Mag. Gertrude Altfrau Megastraße 123/2 1110 Wien			
Positionen:				
Bezeichnung	Anzahl	Einheit	Einzelpreis (€)	Gesamt (€)
Qualitätssicherung der 24h Betreuung	3	b	225,00	675,00
			<b>Gesamt:</b>	<b>675,00</b>
			<a href="#">PDF öffnen</a>	<a href="#">Rechnung stornieren</a>

Die Gutschrift wurde erstellt.

[Rechnung ansehen \(PDF\)](#) [Gutschrift ansehen \(PDF\)](#)

GS-0004-2020	18.06.2020	2	-6.554,80	<a href="#">PDF</a>	<a href="#">details</a>	<input checked="" type="checkbox"/> bezahlt
0004-2020	12.06.2020	2	6.554,80	<a href="#">PDF</a>	<a href="#">details</a>	<input checked="" type="checkbox"/> bezahlt



Rechnungsdaten:				
Rechnungsnummer: 0001-2020				
Rechnungsdatum: 10.06.2020				
Leistungszeitraum: 01.05.2020 bis 31.05.2020				
Kunde:				
Kundennummer: 8				
Rechnungsanschrift: Dr. Mario Mustermann Luise Montag 1 1110 Wien				
Positionen:				
Bezeichnung	Anzahl	Einheit	Einzelpreis (€)	Gesamt (€)
Qualitätssicherung der 24h Betreuung	1	b	225,00	225,00
Stundenweise Betreuung	5	h	73,50	367,50
			<b>Gesamt:</b>	<b>592,50</b>

[PDF öffnen](#) [Rechnung stornieren](#)

Die Gutschrift wurde erstellt.

[Rechnung ansehen \(PDF\)](#) [Gutschrift ansehen \(PDF\)](#)

GS-0003-2020	12.06.2020	3	-99,80	<a href="#">PDF</a>	<a href="#">details</a>	<input type="checkbox"/> bezahlt
0003-2020	12.06.2020	3	99,80	<a href="#">PDF</a>	<a href="#">details</a>	<input checked="" type="checkbox"/> bezahlt



**Es soll eine außerordentliche Leistung außerhalb des Katalogs für einen Kunden erstellt werden, welcher sich nicht im System befindet**

Ergebnis: erfolgreich

Details:

Screenshot of the care CRM software interface showing the process of creating an invoice for a new customer.

The left sidebar shows navigation links: Kunden, Pfleger, Sachwalter, Angehörige, Pflegeplan, Rechnungen, Neue Rechnung, Für Kd, manuell (which is selected), and Administration.

The main form is titled "Name und Anschrift Rechnungsempfänger". It contains fields for "Kunde:" (with a dropdown menu "Daten manuell eingeben") and "Anschrift:" (with a text input field containing "Bernd Tester", "Probenstraße 1", and "1100 Wien"). A "Weiter" button is located below these fields.

The next section contains fields for "Kundendaten" (Bernd Tester) and "Rechnungsadresse" (Probenstraße 1, 1100 Wien). It also includes a "Positionen" table with two rows:

1	außerordentliche Leistung 1	1	15,00
2	Bezeichnung (optional)	Anzahl (optional)	Einzelpreis (optional)

Below this is a "Leistungszeitraum" section with date inputs: "01.06.2020" to "30.06.2020". A "Rechnung erstellen" button is at the bottom of this section.

The final step shows a green message box stating "Die Rechnung wurde erfolgreich erstellt." (The invoice was successfully created.) Below it are "Details" and "als PDF öffnen" buttons.



Name und Anschrift Rechnungsempfänger

Kunde: Daten manuell eingeben

Anschrift: Fr. Amy Übung  
Übungstraße 1  
1220 Wien

Weiter

Kundendaten  
Fr. Amy Übung

Rechnungsadresse  
Übungstraße 1  
1220 Wien

Positionen

1	Außerordentliche Leistung 2	1	20,00
2	Bezeichnung (optional)	Anzahl (optional)	Einzelpreis (optional)

Leistungszeitraum  
15.07.2020 bis 31.07.2020

Rechnung erstellen

Die Rechnung wurde erfolgreich erstellt.

Details  
als PDF öffnen



Name und Anschrift Rechnungsempfänger

Kunde: Daten manuell eingeben

Anschrift: Dr. Maria Feinmann  
Runde-Erde-Str 1  
1230 Wien

Weiter

Kundendaten  
Dr. Maria Feinmann

Rechnungsadresse  
Runde-Erde-Str 1  
1230 Wien

Positionen

1	außerordentliche Leistung 3	1	1500,00
2	Bezeichnung (optional)	Anzahl (optional)	Einzelpreis (optional)

Leistungszeitraum  
01.05.2020 bis 31.12.2020

Rechnung erstellen

Die Rechnung wurde erfolgreich erstellt.

Details  
als PDF öffnen



Name und Anschrift Rechnungsempfänger

Kunde: Daten manuell eingeben

Anschrift: Richard Reed  
Hauptstraße 1  
1220 Wien

Weiter

Kundendaten  
Richard Reed

Rechnungsadresse  
Hauptstraße 1  
1220 Wien

Positionen

1	Außerordentliche Leistung 4	1	653,00
2	Bezeichnung (optional)	Anzahl (optional)	Einzelpreis (optional)

Leistungszeitraum  
01 . 05 . 2020 bis 31 . 08 . 2020

Rechnung erstellen

Die Rechnung wurde erfolgreich erstellt.

Details  
als PDF öffnen



Name und Anschrift Rechnungsempfänger

Kunde:

Anschrift:

Kundendaten  
Anna Markel

Rechnungsadresse  
Flach-Erdenstraße 3  
1140 Wien

Positionen

1	Außerordentliche Leistung 5	2	12,00
2	Bezeichnung (optional)	Anzahl (optional)	Einzelpreis (optional)

Leistungszeitraum  
15.06.2020 bis 31.08.2020

Die Rechnung wurde erfolgreich erstellt.

[Details](#)  
[als PDF öffnen](#)



### 10.3.1.6. Leistungen

Es sollen 5 Leistungen für das Jahr 2019 angelegt werden.

Ergebnis: erfolgreich

Details:

Leistung			
Bezeichnung:	Leistung1	Einzeltarif:	40,00
Jahr (Preisbasis):	2019	Mengentarif:	30,00
Einheit:	Stunde	Ab (Anzahl):	2

**Leistung anlegen**

Die neue Leistung wurde erfolgreich angelegt

[Neue Leistung anlegen](#)

[Leistungen für 2019 anzeigen](#)

Leistung			
Bezeichnung:	Leistung2	Einzeltarif:	99,00
Jahr (Preisbasis):	2019	Mengentarif:	75,00
Einheit:	Tag	Ab (Anzahl):	2

**Leistung anlegen**

Die neue Leistung wurde erfolgreich angelegt

[Neue Leistung anlegen](#)

[Leistungen für 2019 anzeigen](#)

Leistung			
Bezeichnung:	Leistung3	Einzeltarif:	150,00
Jahr (Preisbasis):	2019	Mengentarif:	100,00
Einheit:	Woche	Ab (Anzahl):	2

**Leistung anlegen**



Die neue Leistung wurde erfolgreich angelegt

[Neue Leistung anlegen](#)

[Leistungen für 2019 anzeigen](#)

#### Leistung

Bezeichnung:	Leistung4	Einzeltarif:	250,00
Jahr (Preisbasis):	2019	Mengentarif:	200,00
Einheit:	Monat	Ab (Anzahl):	2

[Leistung anlegen](#)

Die neue Leistung wurde erfolgreich angelegt

[Neue Leistung anlegen](#)

[Leistungen für 2019 anzeigen](#)

Bitte alle Pflichtfelder ausfüllen!

#### Leistung

Bezeichnung:	Leistung5	Einzeltarif:	30,00
Jahr (Preisbasis):	2019	Mengentarif:	25,00
Einheit:	Stück	Ab (Anzahl):	3

[Leistung anlegen](#)

Die neue Leistung wurde erfolgreich angelegt

[Neue Leistung anlegen](#)

[Leistungen für 2019 anzeigen](#)

#### Leistung

Bezeichnung:	Leistung6	Einzeltarif:	50,00
Jahr (Preisbasis):	2019	Mengentarif:	45,00
Einheit:	Besuch	Ab (Anzahl):	7

[Leistung anlegen](#)



Die neue Leistung wurde erfolgreich angelegt

[Neue Leistung anlegen](#)

[Leistungen für 2019 anzeigen](#)

**Leistung**

Bezeichnung:	Leistung7	Einzeltarif:	45,00
Jahr (Preisbasis):	2019	Mengentarif:	40,00
Einheit:	Einheit	Ab (Anzahl):	15

**Leistung anlegen**

Die neue Leistung wurde erfolgreich angelegt

[Neue Leistung anlegen](#)

[Leistungen für 2019 anzeigen](#)

**Leistung**

Bezeichnung:	Leistung8	Einzeltarif:	25,00
Jahr (Preisbasis):	2019	Mengentarif:	20,00
Einheit:	Stunde	Ab (Anzahl):	14

**Leistung anlegen**

Die neue Leistung wurde erfolgreich angelegt

[Neue Leistung anlegen](#)

[Leistungen für 2019 anzeigen](#)

**Leistung**

Bezeichnung:	Leistung9	Einzeltarif:	80,00
Jahr (Preisbasis):	2019	Mengentarif:	70,00
Einheit:	Tag	Ab (Anzahl):	2

**Leistung anlegen**



Die neue Leistung wurde erfolgreich angelegt

[Neue Leistung anlegen](#)

[Leistungen für 2019 anzeigen](#)

---

Leistung			
Bezeichnung:	Leistung10	Einzeltarif:	15,00
Jahr (Preisbasis):	2019	Mengentarif:	10
Einheit:	Stunde	Ab (Anzahl):	30

[Leistung anlegen](#)

---

Die neue Leistung wurde erfolgreich angelegt

[Neue Leistung anlegen](#)

[Leistungen für 2019 anzeigen](#)

---

Leistung			
Bezeichnung:	Leistung11	Einzeltarif:	150,00
Jahr (Preisbasis):	2019	Mengentarif:	100,00
Einheit:	Woche	Ab (Anzahl):	10

[Leistung anlegen](#)

---

Die neue Leistung wurde erfolgreich angelegt

[Neue Leistung anlegen](#)

[Leistungen für 2019 anzeigen](#)

---

Leistung			
Bezeichnung:	Leistung12	Einzeltarif:	300,00
Jahr (Preisbasis):	2019	Mengentarif:	250,00
Einheit:	Monat	Ab (Anzahl):	10

[Leistung anlegen](#)



Die neue Leistung wurde erfolgreich angelegt

[Neue Leistung anlegen](#)

[Leistungen für 2019 anzeigen](#)

#### Leistung

Bezeichnung:

Jahr (Preisbasis):

Einheit:

Einzeltarif:



Mengentarif:



Ab (Anzahl):



[Leistung anlegen](#)

Die neue Leistung wurde erfolgreich angelegt

[Neue Leistung anlegen](#)

[Leistungen für 2019 anzeigen](#)

#### Leistung

Bezeichnung:

Jahr (Preisbasis):

Einheit:

Einzeltarif:



Mengentarif:



Ab (Anzahl):



[Leistung anlegen](#)

Die neue Leistung wurde erfolgreich angelegt

[Neue Leistung anlegen](#)

[Leistungen für 2019 anzeigen](#)

#### Leistung

Bezeichnung:

Jahr (Preisbasis):

Einheit:

Einzeltarif:



Mengentarif:



Ab (Anzahl):



[Leistung anlegen](#)



Die neue Leistung wurde erfolgreich angelegt

[Neue Leistung anlegen](#)

[Leistungen für 2019 anzeigen](#)

Leistung

Bezeichnung:

Jahr (Preisbasis):

Einheit:

Einzeltarif:

Mengentarif:

Ab (Anzahl):

Die neue Leistung wurde erfolgreich angelegt

[Neue Leistung anlegen](#)

[Leistungen für 2019 anzeigen](#)

Leistung

Bezeichnung:

Jahr (Preisbasis):

Einheit:

Einzeltarif:

Mengentarif:

Ab (Anzahl):

Die neue Leistung wurde erfolgreich angelegt

[Neue Leistung anlegen](#)

[Leistungen für 2019 anzeigen](#)

Leistung

Bezeichnung:

Jahr (Preisbasis):

Einheit:

Einzeltarif:

Mengentarif:

Ab (Anzahl):



Die neue Leistung wurde erfolgreich angelegt

[Neue Leistung anlegen](#)

[Leistungen für 2019 anzeigen](#)

Leistung

Bezeichnung:	Leistung19	Einzeltarif:	150,00
Jahr (Preisbasis):	2019	Mengentarif:	100,00
Einheit:	Besuch	Ab (Anzahl):	15

[Leistung anlegen](#)

Die neue Leistung wurde erfolgreich angelegt

[Neue Leistung anlegen](#)

[Leistungen für 2019 anzeigen](#)

Leistung

Bezeichnung:	Leistung20	Einzeltarif:	1900,00
Jahr (Preisbasis):	2019	Mengentarif:	1500,00
Einheit:	Monat	Ab (Anzahl):	30

[Leistung anlegen](#)

Die neue Leistung wurde erfolgreich angelegt

[Neue Leistung anlegen](#)

[Leistungen für 2019 anzeigen](#)

Leistung

Bezeichnung:	Leistung21	Einzeltarif:	150,00
Jahr (Preisbasis):	2019	Mengentarif:	145,00
Einheit:	Stück	Ab (Anzahl):	5

[Leistung anlegen](#)



Die neue Leistung wurde erfolgreich angelegt

[Neue Leistung anlegen](#)

[Leistungen für 2019 anzeigen](#)

#### Leistung

Bezeichnung:	Leistung22	Einzeltarif:	150,00
Jahr (Preisbasis):	2019	Mengentarif:	145
Einheit:	Einheit	Ab (Anzahl):	7

[Leistung anlegen](#)

Die neue Leistung wurde erfolgreich angelegt

[Neue Leistung anlegen](#)

[Leistungen für 2019 anzeigen](#)

#### Leistung

Bezeichnung:	Leistung23	Einzeltarif:	78,00
Jahr (Preisbasis):	2019	Mengentarif:	70,00
Einheit:	Stunde	Ab (Anzahl):	5

[Leistung anlegen](#)

Die neue Leistung wurde erfolgreich angelegt

[Neue Leistung anlegen](#)

[Leistungen für 2019 anzeigen](#)

#### Leistung

Bezeichnung:	Leistung24	Einzeltarif:	1450,00
Jahr (Preisbasis):	2019	Mengentarif:	99,99
Einheit:	Monat	Ab (Anzahl):	0

[Leistung anlegen](#)



Die neue Leistung wurde erfolgreich angelegt

[Neue Leistung anlegen](#)

[Leistungen für 2019 anzeigen](#)

Leistung

Bezeichnung:

Leistung25

Einzeltarif:

25,00

Jahr (Preisbasis):

2019

Mengentarif:

20

Einheit:

Stunde

Ab (Anzahl):

16

[Leistung anlegen](#)

Die neue Leistung wurde erfolgreich angelegt

[Neue Leistung anlegen](#)

[Leistungen für 2019 anzeigen](#)



**Die Leistungen sollen einem Kunden zugeordnet werden.**

Ergebnis: erfolgreich

Details:

Allgemein Vermerke Pflegeplan Tarife Rechnungen

Leistungen von Kunden:

Bezeichnung	Einheit	Preis/Einheit	Mengenpreis	Mengenpreis ab	Entfernen
Leistung1	Stunde	40.00 €	30.00 €	2	<a href="#">löschen</a>
Leistung3	Woche	150.00 €	100.00 €	2	<a href="#">löschen</a>
Leistung4	Monat	250.00 €	200.00 €	2	<a href="#">löschen</a>
Leistung5	Stück	30.00 €	25.00 €	3	<a href="#">löschen</a>
Leistung2	Tag	99.00 €	75.00 €	2	<a href="#">löschen</a>

Verfügbare Leistungen:

Jahr: 2019

Allgemein Vermerke Pflegeplan Tarife Rechnungen

Leistungen von Kunden:

Bezeichnung	Einheit	Preis/Einheit	Mengenpreis	Mengenpreis ab	Entfernen
Leistung7	Einheit	45.00 €	40.00 €	15	<a href="#">löschen</a>
Leistung8	Stunde	25.00 €	20.00 €	14	<a href="#">löschen</a>
Leistung6	Besuch	50.00 €	45.00 €	7	<a href="#">löschen</a>
Leistung9	Tag	80.00 €	70.00 €	2	<a href="#">löschen</a>
Leistung10	Stunde	15.00 €	10.00 €	30	<a href="#">löschen</a>

Verfügbare Leistungen:

Jahr: 2019

Allgemein Vermerke Pflegeplan Tarife Rechnungen

Leistungen von Kunden:

Bezeichnung	Einheit	Preis/Einheit	Mengenpreis	Mengenpreis ab	Entfernen
Leistung12	Monat	300.00 €	250.00 €	10	<a href="#">löschen</a>
Leistung14	Stunde	78.00 €	75.00 €	16	<a href="#">löschen</a>
Leistung11	Woche	150.00 €	100.00 €	10	<a href="#">löschen</a>
Leistung13	Stück	80.00 €	75.00 €	10	<a href="#">löschen</a>
Leistung15	Monat	1400.00 €	1250.00 €	30	<a href="#">löschen</a>

Verfügbare Leistungen:

Jahr: 2019



Allgemein Vermerke Pflegeplan Tarife Rechnungen

Leistungen von Kunden:

Bezeichnung	Einheit	Preis/Einheit	Mengenpreis	Mengenpreis ab	Entfernen
Leistung17	Tag	150.00 €	100.00 €	20	<a href="#">löschen</a>
Leistung18	Woche	750.00 €	650.00 €	20	<a href="#">löschen</a>
Leistung19	Besuch	150.00 €	100.00 €	15	<a href="#">löschen</a>
Leistung20	Monat	1900.00 €	1500.00 €	30	<a href="#">löschen</a>
Leistung16	Stunde	60.00 €	45.00 €	15	<a href="#">löschen</a>

Verfügbare Leistungen:

Jahr: 2019 ▾

Allgemein Vermerke Pflegeplan Tarife Rechnungen

Leistungen von Kunden:

Bezeichnung	Einheit	Preis/Einheit	Mengenpreis	Mengenpreis ab	Entfernen
Leistung21	Stück	150.00 €	145.00 €	5	<a href="#">löschen</a>
Leistung22	Einheit	150.00 €	145.00 €	7	<a href="#">löschen</a>
Leistung24	Monat	1450.00 €	-	-	<a href="#">löschen</a>
Leistung25	Stunde	25.00 €	20.00 €	16	<a href="#">löschen</a>
Leistung23	Stunde	78.00 €	70.00 €	5	<a href="#">löschen</a>

Verfügbare Leistungen:

Jahr: 2019 ▾



**Diese Leistungen (nur die neu angelegten) sollen indexiert werden mit einem Index von 1,62% (= Faktor 1,0162) für 2020.**

Ergebnis: erfolgreich

Details:

Screenshot of the Care CRM software interface showing the indexing configuration for services.

The sidebar menu includes: Kunden-, Pfleger-, Sachwälter-, Angehörige-, Pflegeplan-, Rechnungen-, Neue Rechnung-, Administration-, Leistungen-, Alle, Neu, Index (which is selected), Alle User, and User hinzufügen.

The main configuration area shows the following settings:

Jahr anzeigen:	2019
Faktor für Indexierung:	1,0162
Neues Jahr:	2020

A table lists the services and their indexing status:

Bezeichnung	Einheit	Preis/Einheit	Mengenpreis	Mengenpreis ab	Indexieren
Leistung1	Stunde	40.00 €	30.00 €	2	<input checked="" type="checkbox"/>
Leistung16	Stunde	60.00 €	45.00 €	15	<input checked="" type="checkbox"/>
Leistung14	Stunde	78.00 €	75.00 €	16	<input checked="" type="checkbox"/>
Leistung25	Stunde	25.00 €	20.00 €	16	<input checked="" type="checkbox"/>
Leistung4	Monat	250.00 €	200.00 €	2	<input checked="" type="checkbox"/>
Leistung7	Einheit	45.00 €	40.00 €	15	<input checked="" type="checkbox"/>
Leistung9	Tag	80.00 €	70.00 €	2	<input checked="" type="checkbox"/>
Leistung23	Stunde	78.00 €	70.00 €	5	<input checked="" type="checkbox"/>
Test1	Stunde	99.99 €	59.99 €	1	<input type="checkbox"/>
Leistung20	Monat	1900.00 €	1500.00 €	30	<input checked="" type="checkbox"/>
Leistung2	Tag	99.00 €	75.00 €	2	<input checked="" type="checkbox"/>
Leistung10	Stunde	15.00 €	10.00 €	30	<input checked="" type="checkbox"/>
Leistung8	Stunde	25.00 €	20.00 €	14	<input checked="" type="checkbox"/>
Leistung24	Monat	1450.00 €	-	-	<input checked="" type="checkbox"/>
Test2	Tag	120.00 €	60.00 €	2	<input type="checkbox"/>
Leistung5	Stück	30.00 €	25.00 €	3	<input checked="" type="checkbox"/>
Leistung12	Monat	300.00 €	250.00 €	10	<input checked="" type="checkbox"/>
Leistung17	Tag	150.00 €	100.00 €	20	<input checked="" type="checkbox"/>
Leistung21	Stück	150.00 €	145.00 €	5	<input checked="" type="checkbox"/>
Leistung15	Monat	1400.00 €	1250.00 €	30	<input checked="" type="checkbox"/>
Leistung13	Stück	80.00 €	75.00 €	10	<input checked="" type="checkbox"/>
Leistung19	Besuch	150.00 €	100.00 €	15	<input checked="" type="checkbox"/>
Leistung18	Woche	750.00 €	650.00 €	20	<input checked="" type="checkbox"/>
Leistung11	Woche	150.00 €	100.00 €	10	<input checked="" type="checkbox"/>
Leistung3	Woche	150.00 €	100.00 €	2	<input checked="" type="checkbox"/>
Leistung6	Besuch	50.00 €	45.00 €	7	<input checked="" type="checkbox"/>
Leistung22	Einheit	150.00 €	145.00 €	7	<input checked="" type="checkbox"/>

**Indexieren** button at the bottom left of the table.

The message "Die Leistungen wurden erfolgreich indexiert" (The services were successfully indexed) is displayed in green at the top of the result screen.

[Indexierte Leistungen anzeigen](#) (View indexed services) link at the bottom of the result screen.



**Die neuen Leistungen sollen nun in 2020 aufscheinen mit den um 1,62% erhöhten Preisen**

Ergebnis: erfolgreich

Details:

Bezeichnung	Einheit	Preis/Einheit	Mengenpreis	Mengenpreis ab
test 2	Woche	99.90 €	89.00 €	5
Leistung9	Tag	81.30 €	71.13 €	2
Leistung19	Besuch	152.43 €	101.62 €	15
Leistung6	Besuch	50.81 €	45.73 €	7
Leistung3	Woche	152.43 €	101.62 €	2
Leistung21	Stück	152.43 €	147.35 €	5
Leistung25	Stunde	25.41 €	20.32 €	16
Leistung13	Stück	81.30 €	76.22 €	10
Leistung20	Monat	1930.78 €	1524.30 €	30
Sonder Hausbesuch	Monat	1500.00 €	1500.00 €	2
Leistung2	Tag	100.60 €	76.22 €	2
Leistung16	Stunde	60.97 €	45.73 €	15
Leistung17	Tag	152.43 €	101.62 €	20
Leistung5	Stück	30.49 €	25.41 €	3
Leistung8	Stunde	25.41 €	20.32 €	14
Leistung1	Stunde	40.65 €	30.49 €	2
Leistung7	Einheit	45.73 €	40.65 €	15
Leistung23	Stunde	79.26 €	71.13 €	5
Leistung24	Monat	1473.49 €	-	-
Leistung18	Woche	762.15 €	660.53 €	20
Leistung11	Woche	152.43 €	101.62 €	10
Qualitätsicherung der 24h Betreuung	Besuch	150.00 €	-	-
Leistung10	Stunde	15.24 €	10.16 €	30
Leistung12	Monat	304.86 €	254.05 €	10
Leistung14	Stunde	79.26 €	76.22 €	16
Leistung22	Einheit	152.43 €	147.35 €	7
Leistung4	Monat	254.05 €	203.24 €	2
Leistung15	Monat	1422.68 €	1270.25 €	30
Stundenweise Betreuung	Stunde	56.80 €	49.00 €	5



**Beim Kunden sollen automatisch die neu indexierten Leistungen erscheinen**

Ergebnis: erfolgreich

Details:

Allgemein	Vermerke	Pflegeplan	Tarife	Rechnungen	
Leistungen von Kunden:					
Bezeichnung	Einheit	Preis/Einheit	Mengenpreis	Mengenpreis ab	Entfernen
Leistung1	Stunde	40.65 €	30.49 €	2	<a href="#">löschen</a>
Leistung4	Monat	254.05 €	203.24 €	2	<a href="#">löschen</a>
Leistung2	Tag	100.60 €	76.22 €	2	<a href="#">löschen</a>
Leistung5	Stück	30.49 €	25.41 €	3	<a href="#">löschen</a>
Leistung3	Woche	152.43 €	101.62 €	2	<a href="#">löschen</a>
Verfügbare Leistungen:					
Jahr:	2020	<input type="button" value="▼"/>			

Allgemein	Vermerke	Pflegeplan	Tarife	Rechnungen	
Leistungen von Kunden:					
Bezeichnung	Einheit	Preis/Einheit	Mengenpreis	Mengenpreis ab	Entfernen
Leistung7	Einheit	45.73 €	40.65 €	15	<a href="#">löschen</a>
Leistung9	Tag	81.30 €	71.13 €	2	<a href="#">löschen</a>
Leistung10	Stunde	15.24 €	10.16 €	30	<a href="#">löschen</a>
Leistung6	Besuch	50.81 €	45.73 €	7	<a href="#">löschen</a>
Leistung8	Stunde	25.41 €	20.32 €	14	<a href="#">löschen</a>
Verfügbare Leistungen:					
Jahr:	2020	<input type="button" value="▼"/>			

Allgemein	Vermerke	Pflegeplan	Tarife	Rechnungen	
Leistungen von Kunden:					
Bezeichnung	Einheit	Preis/Einheit	Mengenpreis	Mengenpreis ab	Entfernen
Leistung12	Monat	304.86 €	254.05 €	10	<a href="#">löschen</a>
Leistung13	Stück	81.30 €	76.22 €	10	<a href="#">löschen</a>
Leistung11	Woche	152.43 €	101.62 €	10	<a href="#">löschen</a>
Leistung14	Stunde	79.26 €	76.22 €	16	<a href="#">löschen</a>
Leistung15	Monat	1422.68 €	1270.25 €	30	<a href="#">löschen</a>
Verfügbare Leistungen:					
Jahr:	2020	<input type="button" value="▼"/>			

Allgemein	Vermerke	Pflegeplan	Tarife	Rechnungen	
Leistungen von Kunden:					
Bezeichnung	Einheit	Preis/Einheit	Mengenpreis	Mengenpreis ab	Entfernen
Leistung16	Stunde	60.97 €	45.73 €	15	<a href="#">löschen</a>
Leistung20	Monat	1930.78 €	1524.30 €	30	<a href="#">löschen</a>
Leistung17	Tag	152.43 €	101.62 €	20	<a href="#">löschen</a>
Leistung18	Woche	762.15 €	660.53 €	20	<a href="#">löschen</a>
Leistung19	Besuch	152.43 €	101.62 €	15	<a href="#">löschen</a>
Verfügbare Leistungen:					
Jahr:	2020	<input type="button" value="▼"/>			



Allgemein Vermerke Pflegeplan Tarife Rechnungen

Leistungen von Kunden:

Bezeichnung	Einheit	Preis/Einheit	Mengenpreis	Mengenpreis ab	Entfernen
Leistung23	Stunde	79.26 €	71.13 €	5	<a href="#">löschen</a>
Leistung24	Monat	1473.49 €	-	-	<a href="#">löschen</a>
Leistung21	Stück	152.43 €	147.35 €	5	<a href="#">löschen</a>
Leistung22	Einheit	152.43 €	147.35 €	7	<a href="#">löschen</a>
Leistung25	Stunde	25.41 €	20.32 €	16	<a href="#">löschen</a>

Verfügbare Leistungen:

Jahr:



### 10.3.1.7. User und Sicherheit

**Es soll eine neuer User angelegt werden (ohne Adminrechte)**

Ergebnis: erfolgreich

Details:

Userdaten:	
Vorname	Hans
Nachname	Jones
E-Mail	hans@jones.at
Admin:	<input type="checkbox"/>

**Datensatz anlegen**

Der neue User wurde erfolgreich angelegt

Userdaten:  
Username: hjones  
Passwort: asao21mG7\_2pEfAq

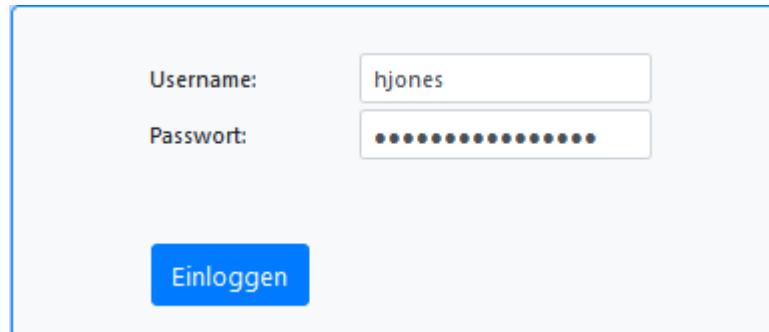
[Userdetails](#)  
[Neuen User anlegen](#)



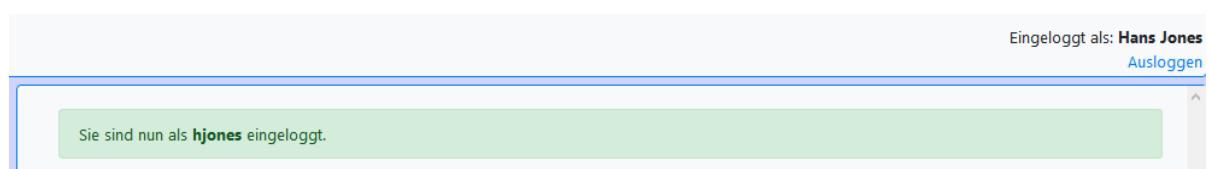
**Der User soll sich mit dem generierten Passwort erfolgreich anmelden können**

Ergebnis: erfolgreich

Details:



A screenshot of a login form. It has two input fields: "Username:" with the value "hjones" and "Passwort:" with a series of black dots representing the password. Below the fields is a blue button labeled "Einloggen".



A screenshot showing the result of a successful login. At the top right, it says "Eingeloggt als: **Hans Jones**" and "Ausloggen". A green banner at the bottom displays the message "Sie sind nun als **hjones** eingeloggt.".



**Der User soll sich 3 mal mit falschem Passwort anmelden und automatisch gesperrt werden**

Ergebnis: erfolgreich

Details:

Ihr Benutzerkonto wurde wegen zu vielen fehlgeschlagenen Login-Versuchen gesperrt.  
Bitte wenden Sie sich an einen Administrator.

Username:

Passwort:

**Einloggen**



**Nachdem ein Administrator ihn wieder freigeschaltet hat, soll der User sich wieder anmelden können**

Ergebnis: erfolgreich

Details:

Userdaten:

Vorname	Hans	Username:	<b>hjones</b>
Nachname	Jones	Aktiv:	<input checked="" type="checkbox"/>
E-Mail	hans@jones.at		
Admin:	<input type="checkbox"/>		

**Übernehmen**   **Verwerfen**

Userdaten:

Vorname	<b>Hans</b>	Username:	<b>hjones</b>
Nachname	<b>Jones</b>	Status:	<b>aktiv</b>
E-Mail	<b>hans@jones.at</b>	Neues Passwort:	<b>cjEvXg.y_C/\$mD/Z</b>
Art:	<b>User</b>		

**Datensatz ändern**

Sie sind nun als **hjones** eingeloggt.



**Der User soll versuchen, auf die Adminfunktionen (`Igneu.php`, `Igalle.php`, `Igindex.php`, `usalle.php`, `usneu.php`, `usaendern.php`) zuzugreifen versuchen und einen Fehler erhalten.**

Ergebnis: erfolgreich

Details:

The image contains three screenshots of a web browser displaying error messages. Each screenshot has a header bar with various links and a status bar indicating 80% zoom.

- Screenshot 1: 192.168.0.167/carecrm/usdetails.php?us\_id=7**  
The page title is "Administration → Userdetails". On the left, a sidebar menu includes "Kunden", "Pfleger", "Sachwalter", "Angehörige", "Pflegeplan", "Rechnungen", and "Neue Rechnung". A message box in the center states: "Sie besitzen nicht die notwendigen Berechtigungen, um diese Funktion zu nutzen. Bitte wenden Sie sich an einen Administrator." (You do not have the necessary permissions to use this function. Please contact an administrator.)
- Screenshot 2: 192.168.0.167/carecrm/galle.php**  
The page title is "Leistungen → Alle Leistungen". The sidebar menu is identical to the first screenshot. A message box states: "Sie besitzen nicht die notwendigen Berechtigungen, um diese Funktion zu nutzen. Bitte wenden Sie sich an einen Administrator." (You do not have the necessary permissions to use this function. Please contact an administrator.)
- Screenshot 3: 192.168.0.167/carecrm/igindex.php**  
The page title is "Leistungen → Indexieren". The sidebar menu is identical to the first screenshot. A message box states: "Sie besitzen nicht die notwendigen Berechtigungen, um diese Funktion zu nutzen. Bitte wenden Sie sich an einen Administrator." (You do not have the necessary permissions to use this function. Please contact an administrator.)



192.168.0.167/carecrm/usalle.php

Welcome to phpMyA... M Posteingang (534) - b... ⓘ Startseite - Proxer.Me 🌐 Neuer Tab ⚙ Setup UFW on CentOS... ⓘ A Simple Bash Script f... ⓘ How to Install and Co... MySQL :: MySQL NDB ...

Eingeloggt als: **Hans Jones** Ausloggen

Administration → Alle User

Kunden ▾ Pfleger ▾ Sachwälter ▾ Angehörige ▾ Pflegeplan ▾ Rechnungen ▾ Neue Rechnung ▾

Sie besitzen nicht die notwendigen Berechtigungen, um diese Funktion zu nutzen.  
Bitte wenden Sie sich an einen Administrator.

192.168.0.167/carecrm/usneu.php

Welcome to phpMyA... M Posteingang (534) - b... ⓘ Startseite - Proxer.Me 🌐 Neuer Tab ⚙ Setup UFW on CentOS... ⓘ A Simple Bash Script f... ⓘ How to Install and Co... MySQL :: MySQL NDB ...

Eingeloggt als: **Hans Jones** Ausloggen

Administration → Neuer User

Kunden ▾ Pfleger ▾ Sachwälter ▾ Angehörige ▾ Pflegeplan ▾ Rechnungen ▾ Neue Rechnung ▾

Sie besitzen nicht die notwendigen Berechtigungen, um diese Funktion zu nutzen.  
Bitte wenden Sie sich an einen Administrator.

192.168.0.167/carecrm/usaendern.php

Welcome to phpMyA... M Posteingang (534) - b... ⓘ Startseite - Proxer.Me 🌐 Neuer Tab ⚙ Setup UFW on CentOS... ⓘ A Simple Bash Script f... ⓘ How to Install and Co... MySQL :: MySQL NDB ...

Eingeloggt als: **Hans Jones** Ausloggen

Administration → User Ändern

Kunden ▾ Pfleger ▾ Sachwälter ▾ Angehörige ▾ Pflegeplan ▾ Rechnungen ▾ Neue Rechnung ▾

Sie besitzen nicht die notwendigen Berechtigungen, um diese Funktion zu nutzen.  
Bitte wenden Sie sich an einen Administrator.



**Nachdem ein Admin den User zum Admin gemacht hat, soll der User versuchen, auf die Seiten zuzugreifen**

Ergebnis: erfolgreich

Details:

Userdaten:

Vorname	Hans	Username:	<b>hjones</b>
Nachname	Jones	Aktiv:	<input checked="" type="checkbox"/>
E-Mail	hans@jones.at		
Admin:	<input checked="" type="checkbox"/>		

**Übernehmen**    **Verwerfen**

Jones	Hans	hjones	User	aktiv	<a href="#">Details</a>
Jones	Hans	hjones	Admin	aktiv	<a href="#">Details</a>



**User soll dann von einem anderen Admin manuell gesperrt werden und keinen Zugang mehr zum System haben.**

Ergebnis: erfolgreich

Details:

The screenshot shows the care CRM administration interface for managing users. In the top right corner, it says "Eingeloggt als: Bernhard Bonengl" and "Ausloggen". On the left, there's a sidebar with navigation links like Kunden, Pfleger, Sachwalter, Angehörige, Pflegeplan, Rechnungen, Neue Rechnung, Administration, Leistungen, Alle User, and User hinzufügen. The main area is titled "Administration → User Ändern". It shows a form for "Userdaten" with fields for Vorname (Hans), Nachname (Jones), E-Mail (hans@jones.at), and Username (hjones). There's also a checked checkbox for "Admin:". Below the form are two buttons: "Übernehmen" and "Verwerfen". A message box at the bottom says "Ihr Benutzerkonto ist inaktiv. Bitte wenden Sie sich an einen Administrator." (Your user account is inactive. Please contact an administrator.) At the bottom, there's a login form with "Username: hjones" and a password field, followed by a large blue "Einloggen" button.



**Nachdem der User wieder aktiviert wurde, soll für ihn ein neues Passwort generiert werden, mit dem er sich einloggen kann**

Ergebnis: erfolgreich

Details:

Userdaten:

Vorname	Hans	Username:	<b>hjones</b>
Nachname	Jones	Aktiv:	<input checked="" type="checkbox"/>
E-Mail	hans@jones.at		
Admin:	<input type="checkbox"/>		

**Übernehmen**   **Verwerfen**

Userdaten:

Vorname	<b>Hans</b>	Username:	<b>hjones</b>
Nachname	<b>Jones</b>	Status:	<b>aktiv</b>
E-Mail	<b>hans@jones.at</b>		
Art:	<b>User</b>		

Neues Passwort: `=aeO6iOnwRoVy7b?`

**Datensatz ändern**

Eingeloggt als: **Hans Jones**  
[Ausloggen](#)

Sie sind nun als **hjones** eingeloggt.





### 10.3.2. Umgehen Userlogins

In diesen Tests wurde überprüft, ob man ohne angemeldet zu sein auf die Seiten, Daten oder Funktionen des CareCRM zugreifen kann (siehe 7.3)

Details:

The image contains four screenshots of a web browser window, each showing a different page from the CareCRM system. All four screenshots display a 'Sie sind nicht eingeloggt.' (You are not logged in.) message and a 'Bitte loggen Sie sich ein, um auf diese Seite zugreifen zu können.' (Please log in to access this page.) button.

- Screenshot 1:** URL: 192.168.0.167/carecrm/agalle.php  
Title: Angehörige → Alle Angehörigen  
Message: Sie sind nicht eingeloggt. Einloggen
- Screenshot 2:** URL: 192.168.0.167/carecrm/agaeandern.php  
Title: Angehörige → Angehörigen ändern  
Message: Sie sind nicht eingeloggt. Einloggen
- Screenshot 3:** URL: 192.168.0.167/carecrm/agdetails.php  
Title: Angehörige → Angehörigendetails  
Message: Sie sind nicht eingeloggt. Einloggen
- Screenshot 4:** URL: 192.168.0.167/carecrm/agdetails2.php  
Title: Angehörige → Angehörigendetails  
Message: Sie sind nicht eingeloggt. Einloggen



Sie sind nicht eingeloggt.  
Einloggen

Angehörige → Neuer Angehöriger

Kunden ▾  
Pfleger ▾

Bitte loggen Sie sich ein, um auf diese Seite zugreifen zu können.

Sie sind nicht eingeloggt.  
Einloggen

Angehörige → Angehörige suchen

Kunden ▾  
Pfleger ▾

Bitte loggen Sie sich ein, um auf diese Seite zugreifen zu können.

Sie sind nicht eingeloggt.  
Einloggen

Kunden → Kunde ändern

Kunden ▾  
Suchen

Bitte loggen Sie sich ein, um auf diese Seite zugreifen zu können.

Sie sind nicht eingeloggt.  
Einloggen

Kunden → Alle Kunden

Kunden ▾  
Suchen

Bitte loggen Sie sich ein, um auf diese Seite zugreifen zu können.

Sie sind nicht eingeloggt.  
Einloggen

Kunden → Kundendetails

Kunden ▾  
Suchen

Bitte loggen Sie sich ein, um auf diese Seite zugreifen zu können.



Sie sind nicht eingeloggt.  
Einloggen

Kunden → Kundendetails

Kunden ▾  
Suchen

Bitte loggen Sie sich ein, um auf diese Seite zugreifen zu können.

Sie sind nicht eingeloggt.  
Einloggen

Kunden → Kundendetails

Kunden ▾  
Suchen

Bitte loggen Sie sich ein, um auf diese Seite zugreifen zu können.

Sie sind nicht eingeloggt.  
Einloggen

Kunden → Kundendetails

Kunden ▾  
Suchen

Bitte loggen Sie sich ein, um auf diese Seite zugreifen zu können.

Sie sind nicht eingeloggt.  
Einloggen

Kunden → Neuer Kunde

Kunden ▾  
Suchen

Bitte loggen Sie sich ein, um auf diese Seite zugreifen zu können.



Sie sind nicht eingeloggt.  
Einloggen

Kunden → Kunden suchen

Kunden ▾  
Suchen

Bitte loggen Sie sich ein, um auf diese Seite zugreifen zu können.

Sie sind nicht eingeloggt.  
Einloggen

Kunden ▾  
Pfleger ▾

Bitte loggen Sie sich ein, um auf diese Seite zugreifen zu können.

Sie sind nicht eingeloggt.  
Einloggen

Kunden ▾  
Pfleger ▾

Bitte loggen Sie sich ein, um auf diese Seite zugreifen zu können.

Sie sind nicht eingeloggt.  
Einloggen

Kunden ▾  
Pfleger ▾

Bitte loggen Sie sich ein, um auf diese Seite zugreifen zu können.

Sie sind nicht eingeloggt.  
Einloggen

Kunden ▾  
Pfleger ▾

Bitte loggen Sie sich ein, um auf diese Seite zugreifen zu können.



192.168.0.167/carecrm/pgaendern.php

Pfleger → Pfleger ändern

Sie sind nicht eingeloggt.  
Einloggen

Kunden ▾  
Pfleger ▾

Bitte loggen Sie sich ein, um auf diese Seite zugreifen zu können.

192.168.0.167/carecrm/pgalle.php

Pfleger → Alle Pfleger

Sie sind nicht eingeloggt.  
Einloggen

Kunden ▾  
Pfleger ▾

Bitte loggen Sie sich ein, um auf diese Seite zugreifen zu können.

192.168.0.167/carecrm/pgdetails.php

Pfleger → Pflegerdetails

Sie sind nicht eingeloggt.  
Einloggen

Kunden ▾  
Pfleger ▾

Bitte loggen Sie sich ein, um auf diese Seite zugreifen zu können.

192.168.0.167/carecrm/pgdetails2.php

Pfleger → Pflegerdetails

Sie sind nicht eingeloggt.  
Einloggen

Kunden ▾  
Pfleger ▾

Bitte loggen Sie sich ein, um auf diese Seite zugreifen zu können.

192.168.0.167/carecrm/pgdetails3.php

Pfleger → Pflegerdetails

Sie sind nicht eingeloggt.  
Einloggen

Kunden ▾  
Pfleger ▾

Bitte loggen Sie sich ein, um auf diese Seite zugreifen zu können.



Four screenshots of a web application interface for "care CRM". Each screenshot shows a top navigation bar with a search field, a user dropdown menu, and a message box. The main content area displays a title, a sidebar with "Kunden" and "Pfleger" dropdowns, and a large message box.

- Screenshot 1:** Title: Pfleger → Neuer Pfleger. Message: Sie sind nicht eingeloggt. Einloggen. Content: Bitte loggen Sie sich ein, um auf diese Seite zugreifen zu können.
- Screenshot 2:** Title: Pfleger → Pfleger suchen. Message: Sie sind nicht eingeloggt. Einloggen. Content: Bitte loggen Sie sich ein, um auf diese Seite zugreifen zu können.
- Screenshot 3:** Title: Pflegeplan → Details. Message: Sie sind nicht eingeloggt. Einloggen. Content: Bitte loggen Sie sich ein, um auf diese Seite zugreifen zu können.
- Screenshot 4:** Title: Pflegeplan → Export - Kundenplan. Message: Sie sind nicht eingeloggt. Einloggen. Content: Bitte loggen Sie sich ein, um auf diese Seite zugreifen zu können.
- Screenshot 5:** Title: Pflegeplan → Export - Pflegerplan. Message: Sie sind nicht eingeloggt. Einloggen. Content: Bitte loggen Sie sich ein, um auf diese Seite zugreifen zu können.



Sie sind nicht eingeloggt.  
Einloggen

Pflegeplan → Export - Hausbesuche

Kunden ▾  
Pfleger ▾

Bitte loggen Sie sich ein, um auf diese Seite zugreifen zu können.

Sie sind nicht eingeloggt.  
Einloggen

Pflegeplan → Hausbesuche

Kunden ▾  
Pfleger ▾

Bitte loggen Sie sich ein, um auf diese Seite zugreifen zu können.

Sie sind nicht eingeloggt.  
Einloggen

Pflegeplan → Neue Tätigkeit

Kunden ▾  
Pfleger ▾

Bitte loggen Sie sich ein, um auf diese Seite zugreifen zu können.

Sie sind nicht eingeloggt.  
Einloggen

Rechnungen → Alle

Kunden ▾  
Pfleger ▾

Bitte loggen Sie sich ein, um auf diese Seite zugreifen zu können.

Sie sind nicht eingeloggt.  
Einloggen

Rechnungen → Export

Kunden ▾  
Pfleger ▾

Bitte loggen Sie sich ein, um auf diese Seite zugreifen zu können.



Sie sind nicht eingeloggt.  
[Einloggen](#)

Neue Rechnung → **Rechnung für  
Kunde**

Kunden ▾  
Pfleger ▾

Bitte loggen Sie sich ein, um auf diese Seite zugreifen zu können.

Sie sind nicht eingeloggt.  
[Einloggen](#)

Neue Rechnung → **Rechnung  
Manuell**

Kunden ▾  
Pfleger ▾

Bitte loggen Sie sich ein, um auf diese Seite zugreifen zu können.

Sie sind nicht eingeloggt.  
[Einloggen](#)

Rechnungen → **Offene**

Kunden ▾  
Pfleger ▾

Bitte loggen Sie sich ein, um auf diese Seite zugreifen zu können.

Sie sind nicht eingeloggt.  
[Einloggen](#)

Rechnungen → **Details**

Kunden ▾  
Pfleger ▾

Bitte loggen Sie sich ein, um auf diese Seite zugreifen zu können.

Sie sind nicht eingeloggt.  
[Einloggen](#)

Rechnungen → **ToDo**

Kunden ▾  
Pfleger ▾

Bitte loggen Sie sich ein, um auf diese Seite zugreifen zu können.



A screenshot of a web browser window. The address bar shows the URL "192.168.0.167/carecrm/rechnung.php". The page content displays a "Sie sind nicht eingeloggt." message and a "Bitte loggen Sie sich ein, um auf diese Seite zugreifen zu können." button.

Bitte [loggen Sie sich ein](#), um auf diese Seite zugreifen zu können.

A screenshot of a web browser window. The address bar shows the URL "192.168.0.167/carecrm/start.php". The page content displays a "Sie sind nicht eingeloggt." message and a "Bitte loggen Sie sich ein, um auf diese Seite zugreifen zu können." button.

A screenshot of a web browser window. The address bar shows the URL "192.168.0.167/carecrm/swaendern.php". The page content displays a "Sie sind nicht eingeloggt." message and a "Bitte loggen Sie sich ein, um auf diese Seite zugreifen zu können." button.

A screenshot of a web browser window. The address bar shows the URL "192.168.0.167/carecrm/swalle.php". The page content displays a "Sie sind nicht eingeloggt." message and a "Bitte loggen Sie sich ein, um auf diese Seite zugreifen zu können." button.

A screenshot of a web browser window. The address bar shows the URL "192.168.0.167/carecrm/swdetails.php". The page content displays a "Sie sind nicht eingeloggt." message and a "Bitte loggen Sie sich ein, um auf diese Seite zugreifen zu können." button.



Sie sind nicht eingeloggt.  
Einloggen

Sachwalter → **Sachwalterdetails**

Kunden ▾ Pfleger ▾

Bitte loggen Sie sich ein, um auf diese Seite zugreifen zu können.

192.168.0.167/carecrm/swdetails2.php

Sie sind nicht eingeloggt.  
Einloggen

Sachwalter → **Neuer Sachwalter**

Kunden ▾ Pfleger ▾

Bitte loggen Sie sich ein, um auf diese Seite zugreifen zu können.

192.168.0.167/carecrm/swneu.php

Sie sind nicht eingeloggt.  
Einloggen

Sachwalter → **Sachwalter suchen**

Kunden ▾ Pfleger ▾

Bitte loggen Sie sich ein, um auf diese Seite zugreifen zu können.

192.168.0.167/carecrm/swsuche.php

Sie sind nicht eingeloggt.  
Einloggen

Administration → **User Ändern**

Kunden ▾ Pfleger ▾

Bitte loggen Sie sich ein, um auf diese Seite zugreifen zu können.

192.168.0.167/carecrm/usaendern.php

Sie sind nicht eingeloggt.  
Einloggen

Administration → **Alle User**

Kunden ▾ Pfleger ▾

Bitte loggen Sie sich ein, um auf diese Seite zugreifen zu können.

192.168.0.167/carecrm/usalle.php



Three screenshots of a web browser showing the care CRM application interface:

- Userdetails Page:** The URL is 192.168.0.167/carecrm/usdetails.php. The page title is "Administration → Userdetails". It features a navigation menu with "Kunden" and "Pfleger" dropdowns. A message box says "Bitte loggen Sie sich ein, um auf diese Seite zugreifen zu können." (Please log in to access this page). On the right, it says "Sie sind nicht eingeloggt." (You are not logged in.) and has a "Einloggen" link.
- Neuer User Page:** The URL is 192.168.0.167/carecrm/usneu.php. The page title is "Administration → Neuer User". It has the same navigation and message box as the previous page, along with the "Sie sind nicht eingeloggt." and "Einloggen" message.
- Kunden → Neuer Vermerk Page:** The URL is 192.168.0.167/carecrm/veneu.php. The page title is "Kunden → Neuer Vermerk". It has the same navigation and message box as the previous pages, along with the "Sie sind nicht eingeloggt." and "Einloggen" message.



### 10.3.3. Injection-Tests

Die Injection-Tests wurden auf jeweils einer Seite jedes Moduls durchgeführt, um zu testen, ob ungewünschte Eingriffe in die Datenbank durchführbar sind (siehe 7.2)

#### Seite: *kdalle.php*

Ergebnis: Die Anzeige hat sich nicht verändert. Es werden nach wie vor die Kunden angezeigt, welche der Auswahl entsprechen.

end/kdalle.php?kd\_tot=1%20OR%201=1

Anrede	Nachname	Vorname	PLZ	Adresse
Herr	Musterman	Mario	1130	Anton-Langer
Herr	Kleinmann	Peter	1110	Luise Montag
Herr	Rüdiger	Richard	1110	Teststraße 69/
Herr	Sedlacek	Markus	1110	Grillgasse 66/
Herr	Österreicher	Markus	1010	Zweigeltstraße
Frau	Kraft	Elena	1010	Markusplatz 1



### Seite: *kdneu.php*

Ergebnis: Weder mit ‘;DROP TABLE test, ’);DROP TABLE test, noch mit *DROP TABLE test* war eine Injection möglich. Die Ausdrücke wurden als Teil des Akts in die Datenbank gespeichert und nicht ausgeführt.

Allgemeine Daten

Anrede:	Herr	Sachwalte
Titel:	Titel	Festnetz:
Vorname	Bernhard	Mobil:
Nachname	:DROP TABLE test	E-Mail:
Geburtsdatum	23.02.1988	Rechnung

Angehörige

Angehöriger 1:

Rechnungsadresse

Straße:	Nr.:	Stg:	S
---------	------	------	---

Allgemeine Daten

Anrede:	<b>Herr</b>
Titel:	
Vorname:	<b>Bernhard</b>
Nachname:	<b>:DROP TABLE test</b>
Geburtsdatum:	<b>23.02.1988</b>
Verstorben:	<b>nein</b>

Vertragsdaten:



### Seite: pgaendern.php

Ergebnis: analog zum vorhergehenden Test wurden die Statements auch hier ohne Ausführung als Text in die Datenbank eingetragen.

Allgemeine Daten

Anrede:	Herr
Vorname	Abraham
Nachname	;DROP TABLE test
Geburtsdatum	03.04.1977
Beginn	tt.mm.jjjj
Ende	tt.mm.jjjj
aktiv	<input checked="" type="checkbox"/>

Allgemeine Daten

Anrede:	<b>Herr</b>
Vorname	<b>Abraham</b>
Nachname	<b>;DROP TABLE test</b>
Geburtsdatum	<b>03.04.1977</b>

Adresse



### Seite: swneu.php

Ergebnis: analog zum vorhergehenden Test wurden die Statements auch hier ohne Ausführung als Text in die Datenbank eingetragen. Der Test wurde auf agneu.php nicht durchgeführt, da diese zwei Module identisch sind.

Allgemeine Daten	
Anrede:	Anrede
Titel:	Titel
Vorname	Mario
Nachname	'; DROP TABLE test
Geburtsdatum	30.06.2020

Anrede:	<b>Herr</b>
Titel:	
Vorname	<b>Mario</b>
Nachname	<b>'; DROP TABLE test</b>
Geburtsdatum	<b>30.06.2020</b>



## Seite: ppneu.php

Auf dieser Seite gibt es kein Textfeld. Durch die Funktionen des Webbrowsers konnte jedoch das HTML verändert und das Feld in ein Textfeld geändert werden, um den Test durchzuführen.

Ergebnis: Ein MySQL Fehler wurde verursacht und dem Benutzer angezeigt. Es hat keine Änderung in der Datenbank stattgefunden.

A screenshot of a web browser window showing a form for creating an activity. The form has fields for 'Kunde' (Customer), 'Pfleger' (Nurse), 'Beginn' (Start), and 'Ende' (End). The 'Beginn' field contains the value '24.06.jjjj'. To the right of the browser window, a developer tool's element inspector is open, specifically the 'Elements' tab. It shows the HTML structure of the 'Beginn' input field. The original HTML is: <input type="text" class="form-control form-control-sm" name="pp\_beginn" id="pp\_beginn" value=";DROP TABLE test">. A mouse cursor is hovering over the 'value' attribute, which is highlighted in blue. A tooltip or status bar at the bottom of the browser window displays the date and time as 'Datum: 24.06.2020' and 'Uhrzeit: 14:06:10'.

Datum: 24.06.2020  
Uhrzeit: 14:06:10  
Seite: /CareCRM/frontend/ppneu.php  
User: bbonen

Fehler:  
mysqli\_error: Incorrect date value: ';DROP TABLE test' for column 'pp\_beginn' at row 1



### Seite: *rgtodo.php*

Ergebnis: Keine Änderung der angezeigten Daten. Das Ausgewählte Monat hat sich nicht verändert.

d/rgtodo.php?rg\_jahr=2020&rg\_monat=03' OR 1=1

---

---

---

Zu erledigende Verrechnungen anzeigen

Jahr/Monat:

Kunde	Adresse
Herr Kleinmann Peter	1110, Luise Montag Gasse 1
Herr Zimmer Peter	1200, Mariusplatz 12/1/2/4



rgtodo.php?rg\_jahr=2020&rg\_monat=03%20OR%201=1

Zu erledigende Verrechnungen anzeigen

Jahr/Monat:

**Kunde**

Herr Kleinmann Peter

**Adresse**

1110, Luise Montag Gasse 1

Herr Zimmer Peter

1200, Mariusplatz 12/1/2/4



### Seite: rgneuman.php

Es wurde sowohl im ersten, als auch im zweiten Schritt versucht, die Injection durchzuführen.

Ergebnis: im zweiten Schritt ist ersichtlich, dass das Anführungszeichen escaped wurde. Trotz mehreren Versuchen, bei welchen auch das Wissen der Datenbankstruktur genutzt wurde, wurde jeweils ein Datenbankfehler ausgegeben und keine Änderung durchgeführt.

Name und Anschrift Rechnungsempfänger

Kunde:	Daten manuell eingeben
Name:	Bernhard Testmann
Straße + Nr:	Mustergasse 1
PLZ Ort:	'; DROP TABLE test

Weiter

Kundendaten  
Bernhard Testmann

Rechnungsadresse  
Mustergasse 1  
\'; DROP TABLE test

Positionen

1	Bezeichnung	Anzahl	Einzelpreis
---	-------------	--------	-------------

Leistungszeitraum

01.05.2020

bis

31.05.2020

Rechnung erstellen



Kundendaten  
Bernhard Testmann

Rechnungsadresse  
Mustergasse 1  
\'; DROP TABLE test

Positionen

1	:);DROP TABLE test	1	1
2	Bezeichnung (optional)	Anzahl (optional)	Einze

Leistungszeitraum  
01.05.2020  bis 31.05.2020

**Rechnung erstellen**

Datum: 24.06.2020  
Uhrzeit: 14:21:11  
Seite: /CareCRM/frontend/rgneuman.php  
User: bbonen

Fehler:  
mysqli\_error: You have an error in your SQL syntax; check the manual that corresponds to your MariaDB server version for the right syntax to use near 'DROP TABLE test;', 'e', '1', '0000' at line 1



### Seite: usneu.php

Die Injection wurde in das Mail-Feld eingetragen und an eine gültige E-Mail Adresse angefügt, um diese Prüfung zu übergehen.

Ergebnis: Das System erkennt, dass es sich um ungültige Zeichen handelt und wirft eine Warnung aus.

Userdaten:

Vorname	Bernhard
Nachname	Testmann
E-Mail	b.t@mail.com');DRC
Admin:	<input checked="" type="checkbox"/>

**Datensatz anlegen**

Bitte die E-Mail Adresse im Format "name@beispiel.at" angeben!

Userdaten:

Vorname	Bernhard	Das Passwort fü
Nachname	Testmann	
E-Mail	b.t@mail.com');DRC	
Admin:	<input checked="" type="checkbox"/>	

**Datensatz anlegen**