



Prüfungsleistung: Portfolio

Projektdokumentation

von Björn Winterleitner

Matrikelnummer: 321147129

Studiengang: Master Informatik

Prüfungsleistung im Modul:

DLMCSPSE01_D – Projekt: Software Engineering

Sommersemester 2022/2023

Abgabedatum: München, den 25.11.2023

Auswahl des Softwareprozesses / Vorgehensmodell:

Es wird nach dem Wasserfall-Modell vorgegangen. Aufgrund folgender Punkte sehe ich keine Notwendigkeit nach anderen Vorgehensweisen wie zum Beispiel dem V-Modell oder agilen Modellen vorzugehen:

- Durch eine eher geringe Entwicklungsdauer und Komplexität ist keine sinnvolle Aufteilung in mehrere Zyklen, wie im agilen Modell, möglich.
- Aufgrund keiner weiteren Entwickler entfällt die Aufteilung in verschiedene Arbeitspakete und die dazugehörigen Abstimmungen und Tests beim Integrieren neuer Funktionen.

Verwendete Technologien und Tools:

Die Software wird aufgrund von bestehenden und tiefergehenden Kenntnissen in der Programmiersprache „Python“ entwickelt.

Zur Entwicklung der grafischen Oberfläche (GUI) wird die Python-Bibliothek PyQt in der Version 5 verwendet und die benötigten GUI-Elemente in die Software eingebaut.

Um verschiedene Python-Umgebungen („Environments“) und deren installierte Bibliotheken zu verwalten, wird das Programm „Anaconda“ verwendet.

Die Software wird mithilfe der Entwicklungsumgebung „Visual Studio Code“ entwickelt.

Damit die Daten auch nach dem Beenden der Applikation dauerhaft erhalten bleiben, werden die Datensätze in einer CSV-Datei gespeichert und bei jeder Aktion des Benutzers dementsprechend aktualisiert.

Architektur

Die vorliegende Applikation wurde nach der Model-View-Controller-Architektur (MVC) implementiert. Dabei ist die Klasse „CsvFileAccess“ Teil der Model-Komponenten. Die Klassen der View-Komponenten können nur lesend auf die Daten aus dem Modell zugreifen, zum Beispiel um diese anzuzeigen. Werden Daten durch die View-Klassen manipuliert, müssen diese über die Controller-Komponente geführt werden, welche durch die Klasse „Controller“ repräsentiert wird. Als Teil der Controller-Komponente übernimmt sie die Steuerung der View-Komponenten und die Validierung und Speicherung der Benutzereingaben.

Design-Pattern

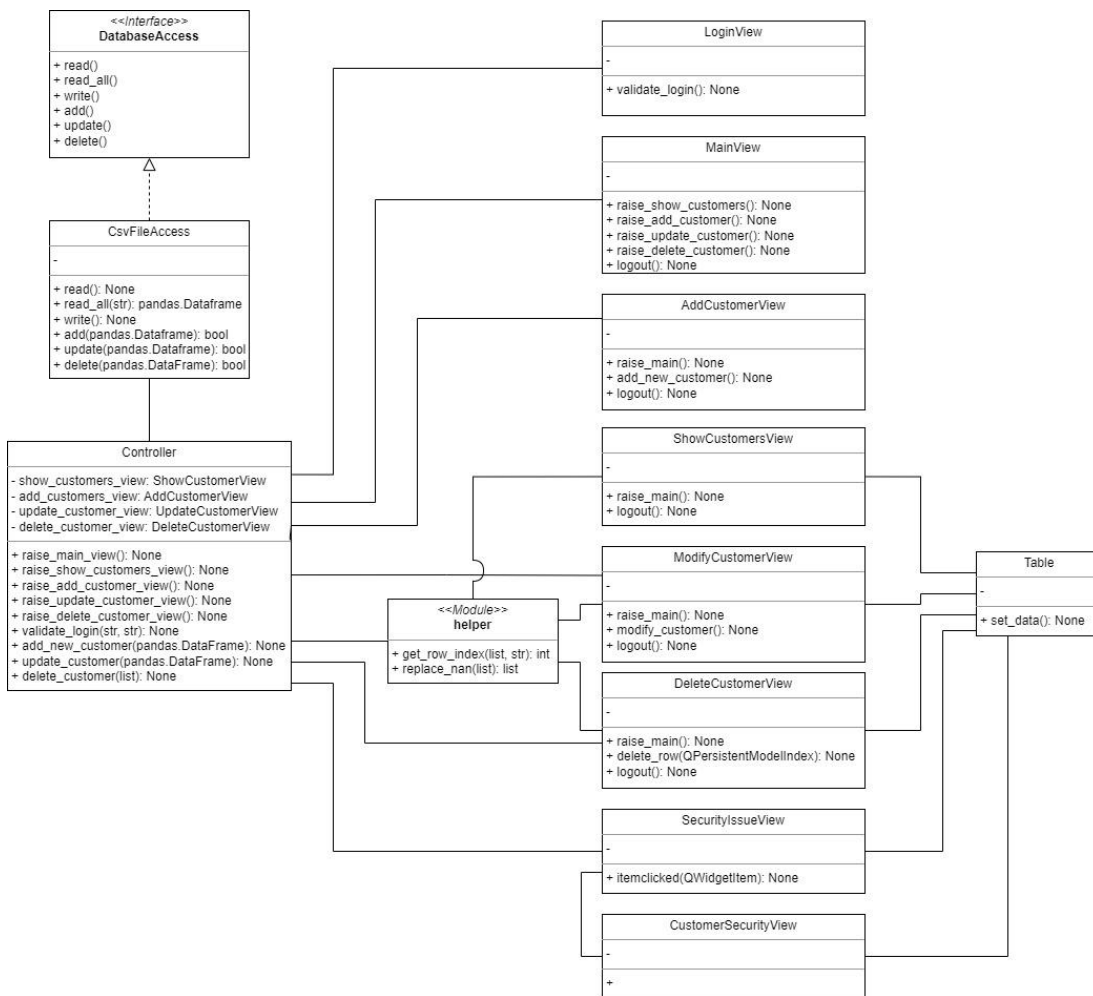
Als Design-Pattern wurde das „Singleton“-Pattern für die Klasse bzw. die Instanz der Klasse „CsvFileAccess“ implementiert.

Es wurde deshalb gewählt, da es nur genau eine Instanz dieser Klasse geben soll und somit sichergestellt wird, dass immer nur genau eine Instanz Zugriffe auf die Datenbank ausführt.

Umgesetzt wurde das Design-Pattern im Programm-Code wie folgt:

```
def __new__(cls):
    if not hasattr(cls, 'instance'):
        cls.instance = super(CsvFileAccess, cls).__new__(cls)
    return cls.instance
```

UML-Klassendiagramm:



Interface „DatabaseAccess“:

Das Interface „DatabaseAccess“ ist eine abstrakte Klasse und definiert die zu implementierenden Methoden zur Verwaltung von Daten in einem Datenspeicher z.B. Datenbank.

Klasse „CsvFileAccess“:

Die Klasse „CsvFileAccess“ implementiert das Interface „DatabaseAccess“. Objekte dieser Klasse sind für die Verwaltung der Daten in einer CSV-Datei zuständig. Es wird konkret festgelegt, wie die Daten aus einer CSV-Datei gelesen werden, Daten hinzugefügt und überschrieben werden.

Klasse „Controller“:

Die Klasse „Controller“ repräsentiert die Controller-Komponenten einer MVC-Architektur. Der Controller nimmt Daten und Befehle der Views entgegen und ist für die Speicherung, Löschung und Aktualisierung der Daten in der Datenbank zuständig.

Klasse „LoginView“:

Die Klasse „LoginView“ repräsentiert die Login-Oberfläche.

Klasse „MainView“:

Die Klasse „MainView“ repräsentiert die Hauptmenü-Oberfläche.

Klasse „AddCustomerView“:

Die Klasse „AddCustomerView“ repräsentiert die Oberfläche, in der ein neuer Kunde angelegt werden kann.

Klasse „ShowCustomersView“:

Die Klasse „ShowCustomersView“ repräsentiert die Oberfläche, welche alle bestehenden Kunden in einer Tabelle anzeigt.

Klasse „ModifyCustomerView“:

Die Klasse „ModifyCustomerView“ repräsentiert die Oberfläche, welche alle bestehenden Kunden in einer Tabelle anzeigt. Die einzelnen Kundendaten können bearbeitet und in der Datenbank gespeichert werden.

Klasse „DeleteCustomerView“:

Die Klasse „DeleteCustomerView“ repräsentiert die Oberfläche, welche alle bestehenden Kunden in einer Tabelle anzeigt. In dieser Ansicht können einzelne Kunden gelöscht werden.

Klasse „SecurityIssuesView“:

Die Klasse „SecurityIssuesView“ repräsentiert die Oberfläche, in der die bekannte Sicherheitslücken zu den vom Kunden eingesetzten Produkten und Versionen angezeigt werden.

Klasse „CustomerSecurityView“:

Die Klasse „CustomerSecurityView“ funktioniert gleich wie die Klasse „SecurityIssuesView“, mit dem Unterschied, dass hier nur die auf dem Kundennamen gefilterten Sicherheitslücken angezeigt werden.

Klasse „Table“:

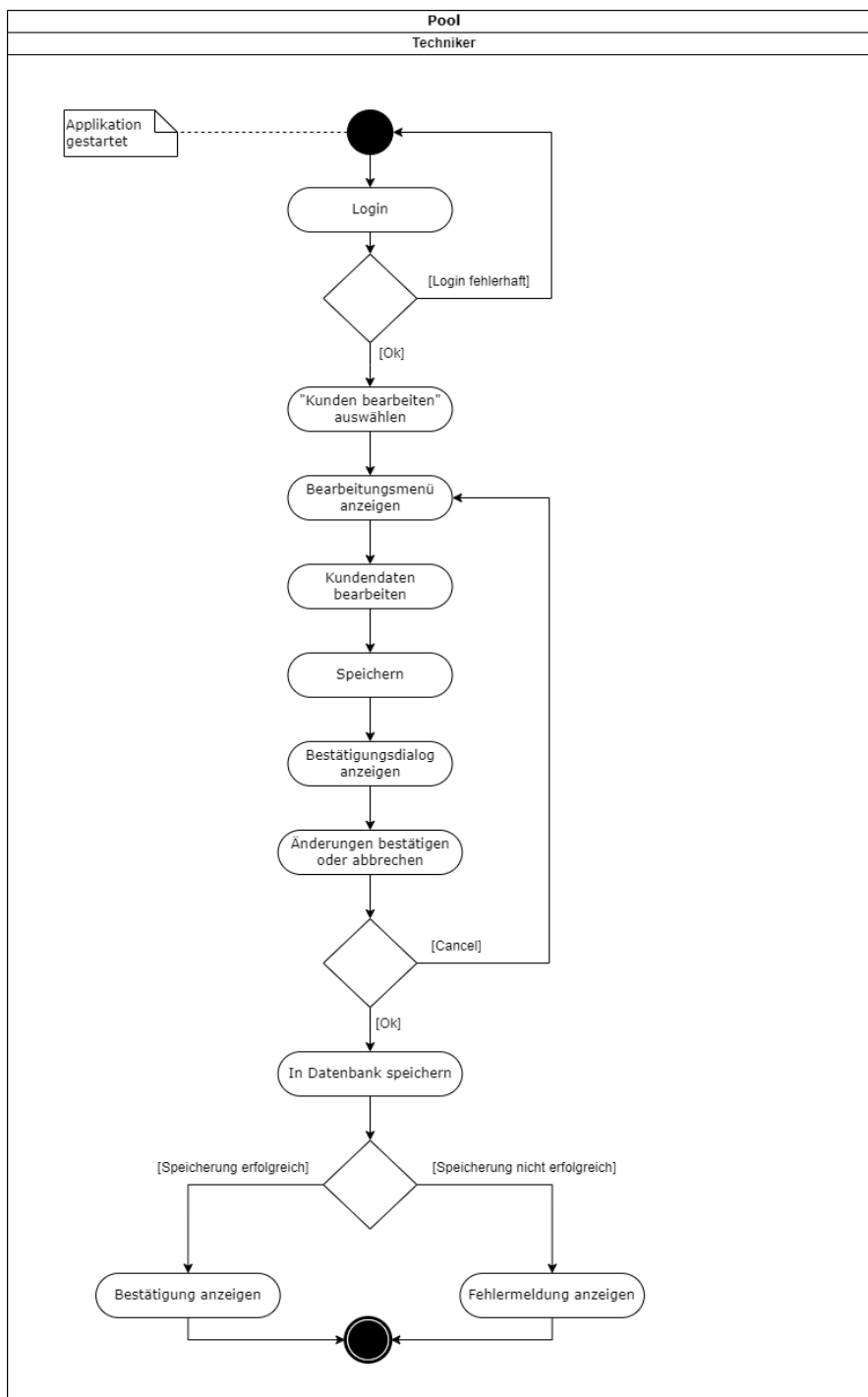
Die Klasse „Table“ erzeugt Objekte der Oberklasse QTableWidgetItem aus der PyQt-Bibliothek. Diese Objekte werden in den einzelnen Views bereits mit den Tabellen-Daten initialisiert und in der Oberfläche angezeigt.

Modul „helper“:

Das Modul „helper“ stellt verschiedene Funktionen bereit, die von den Objekten verschiedener Klassen verwendet werden.

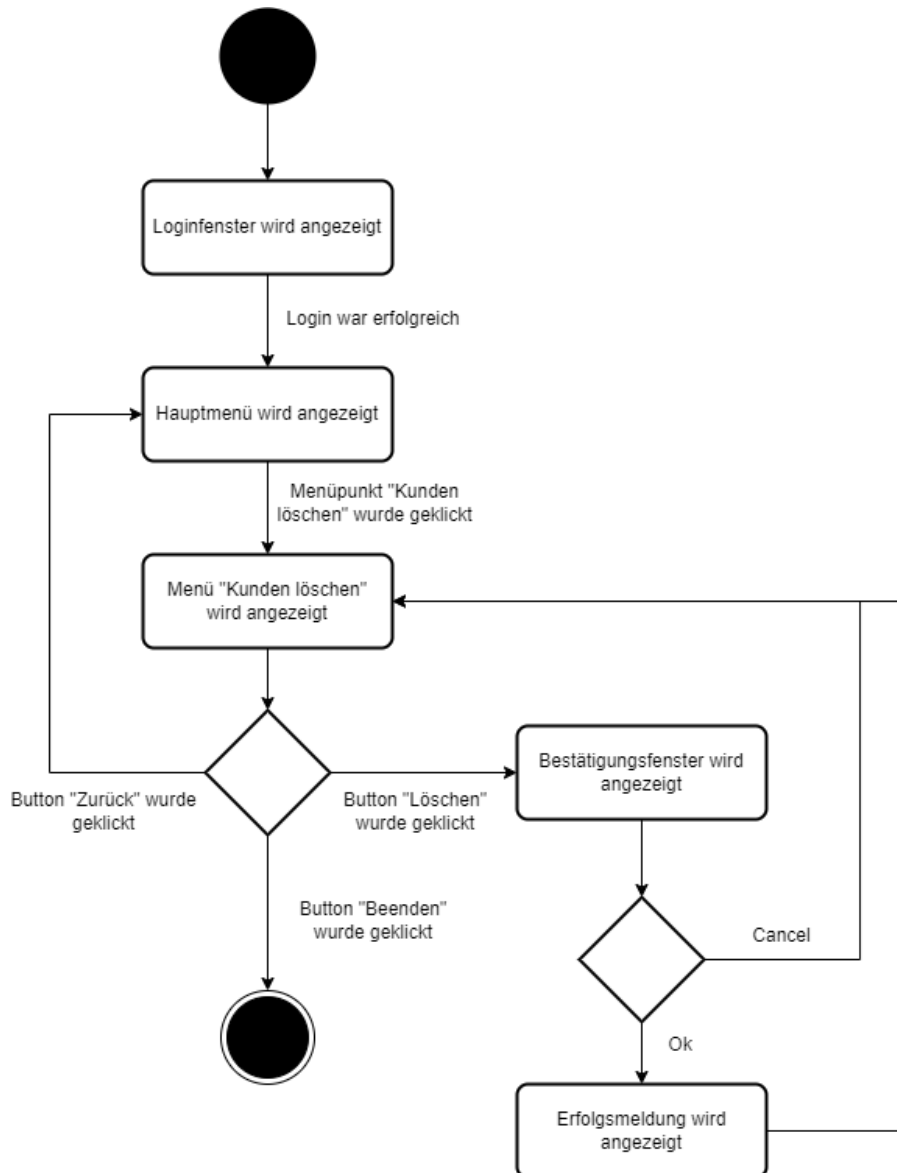
UML-Aktivitätsdiagramm

Folgendes Aktivitätsdiagramm zeigt den Ablauf des Kernprozesses „Kunden bearbeiten“. Hierbei können die Daten eines oder mehrere Kunden bearbeitet werden und anschließend in der Datenbank abgespeichert werden.



UML-Zustandsdiagramm

Folgendes Zustandsdiagramm zeigt den Ablauf des Kernprozesses „Kunden löschen“. Hierbei kann ein Kunde aus der Datenbank gelöscht werden, wenn der dazugehörige „Löschen“-Button in der Tabelle geklickt und die Löschung durch eine Bestätigungsabfrage bestätigt wird.



Link zum Git-Repository

<https://github.com/wwparzival/productmgttool>

Installationsvoraussetzungen:

Python-Version: 3.11.4

PyQt5: 5.15.7

Pandas: 1.5.3

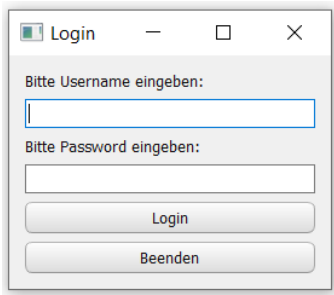
Test-User

User mit Read-Only-Rechten: Username „test1“ / Password „test1“

User mit vollen Rechten: Username „test“ / Password „test“

Benutzeranleitung

Die Anwendung startet mit einem Login-Fenster, in dem die persönlichen Anmeldedaten eingegeben werden müssen.

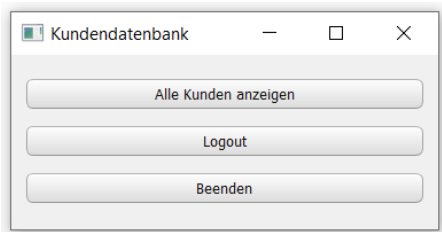


A screenshot of a Windows-style login window titled "Login". It contains two input fields: "Bitte Username eingeben:" and "Bitte Password eingeben:". Below the fields are two buttons: "Login" and "Beenden".

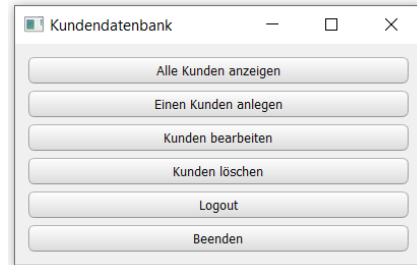
Nach dem Login wird das Hauptmenü angezeigt, welches sich je nach Berechtigungsstufe des eingeloggteten Benutzers unterscheidet.

Die linke Darstellung zeigt das Hauptmenü für Benutzer, die nur die Berechtigung zum Lesen der Kundendaten haben.

Die rechte Darstellung zeigt das Hauptmenü für Benutzer mit voller Berechtigung.



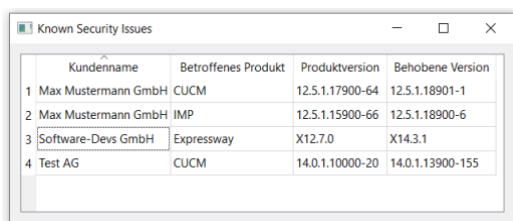
A screenshot of the "Kundendatenbank" main menu for a user with Read-Only permissions. It features three buttons: "Alle Kunden anzeigen", "Logout", and "Beenden".



A screenshot of the "Kundendatenbank" main menu for a user with full permissions. It features six buttons: "Alle Kunden anzeigen", "Einen Kunden anlegen", "Kunden bearbeiten", "Kunden löschen", "Logout", and "Beenden".

Das Hauptmenü bietet neben dem Beenden und dem Logout die verschiedenen Möglichkeiten zum Anlegen, Bearbeiten, Löschen und Anzeigen der Kundendatensätze an.

Neben dem Hauptmenü öffnet sich auch noch ein zusätzliches Fenster, welches einen Überblick der Kunden gibt, welche Produkte im Einsatz haben, die von bekannten Sicherheitslücken betroffen sind.



	Kundenname	Betroffenes Produkt	Produktversion	Behobene Version
1	Max Mustermann GmbH	CUCM	12.5.1.17900-64	12.5.1.18901-1
2	Max Mustermann GmbH	IMP	12.5.1.15900-66	12.5.1.18900-6
3	Software-Devs GmbH	Expressway	X12.7.0	X14.3.1
4	Test AG	CUCM	14.0.1.10000-20	14.0.1.13900-155

Bei einem Klick auf dem Kundennamen öffnet sich ein weiteres Fenster, welches nur die jeweiligen Sicherheitslücken für den angeklickten Kunden anzeigt.

	Kundenname	Betroffenes Produkt	Produktversion	Behobene Version
1	Max Mustermann GmbH	CUCM	12.5.1.17900-64	12.5.1.18901-1
2	Max Mustermann GmbH	IMP	12.5.1.15900-66	12.5.1.18900-6

Im Untermenü „Alle Kunden anzeigen“ kann der vollständige Kundenstamm angezeigt werden.

	Kundenname	Kundennummer	CUCM - Version	IMP - Version	CUC - Version	EXP - Version	Vertragsende
1	1111	12313	11.5.1.18000-21	11.5.1.18000-21			11/11/2023
2	6666	6666	12.1.2.12312-12	21.1.3.32131-31	21.1.2.12312-12	X12.1.12	01/01/2025
3	m21313annschroder1	21222	11.5.1.18000-21	11.5.1.19000-55	11.5.1.20000-21	X14.0.9	31/1/2023
4	m21313annschroder2	21223	11.5.1.18000-21	11.5.1.19000-55	11.5.1.20000-21	X14.0.9	31/1/2024
5	m21313annschroder3	21224	11.5.1.18000-21	11.5.1.19000-55	11.5.1.20000-21	X14.0.9	31/1/2025
6	test124	12312	11.5.1.18000-21	11.5.1.18000-21		X12.2.23	1/1/2024

Zurück
Logout
Beenden

Generell können alle angezeigten Tabellenspalten auch sortiert werden.

Die gefärbten Kundennamen geben Aufschluss über die Restlaufzeit des Vertrages wieder:

- Grün: Die Vertragslaufzeit ist noch länger als ein Jahr.
- Gelb: Die Vertragslaufzeit ist kürzer als ein Jahr.
- Rot: Der Vertrag ist bereits abgelaufen.

Im Untermenü „Einen Kunden anlegen“ kann ein neuer Kunde angelegt werden.

Kundename
Maximal 30 Zeichen

Kundennummer
1 - 99999

CUCM - Version
Bsp.: 12.5.1.17900-22

IMP - Version
Bsp.: 12.5.1.17900-22

CUC - Version
Bsp.: 12.5.1.17900-22

EXP - Version
Bsp.: X14.0.2

Vertragsende
Bsp.: 1/10/2025

Kunden anlegen
Zurück
Logout
Beenden

Die Felder „Kundename“, „Kundennummer“ und „Vertragsende“ sind Pflichtfelder und müssen ausgefüllt werden. Zusätzlich müssen die beiden Felder „Kundename“ und „Kundennummer“ eindeutig sein. Es kann somit kein neuer Kunde angelegt, werden wenn der Kundename und / oder die Kundennummer bereits existiert.

Bestehende Kunden können im Untermenü „Kunden bearbeiten“ geändert werden. Dazu werden die Daten in den entsprechenden Tabellenzellen bearbeitet und anschließend mit dem Button „Änderungen speichern“ in die Datenbank abgespeichert.

Ein bestehender Kunde kann im Untermenü „Kunden löschen“ gelöscht werden, in dem der entsprechende Button in der letzten Spalte des entsprechenden Kunden gedrückt wird.

Einen Kunden löschen							
	Kundenname	Kundennummer	CUCM - Version	IMP - Version	CUC - Version	EXP - Version	Vertragsende Button
1	1111	12313	11.5.1.18000-21	11.5.1.18000-21			11/11/2023 Löschen
2	6666	6666	12.1.2.12312-12	21.1.3.32131-31	21.1.2.12312-12	X12.1.12	01/01/2025 Löschen
3	m21313annschroder1	21222	11.5.1.18000-21	11.5.1.19000-55	11.5.1.20000-21	X14.0.9	31/1/2023 Löschen
4	m21313annschroder2	21223	11.5.1.18000-21	11.5.1.19000-55	11.5.1.20000-21	X14.0.9	31/1/2024 Löschen
5	m21313annschroder3	21224	11.5.1.18000-21	11.5.1.19000-55	11.5.1.20000-21	X14.0.9	31/1/2025 Löschen
6	test124	12312	11.5.1.18000-21	11.5.1.18000-21		X12.2.23	1/1/2024 Löschen
Zurück							
Logout							
Beenden							

Erst nach der Bestätigung des Vorgangs wird der Kunde endgültig aus der Datenbank gelöscht.