SWO5 - UE03 Dokumentation

Aufwand: 12h

Inhaltsverzeichnis

[1. Architekturbeschreibung: 3](#_Toc95075077)

[1.1 Erweiterung UE03 4](#_Toc95075078)

[1.2 Domäne 5](#_Toc95075079)

[1.2.1 Ergänzungen Übung 3 5](#_Toc95075080)

[1.3. Logic 6](#_Toc95075081)

[1.4 Persönliche Änderungen 6](#_Toc95075082)

[2. Quellcode 6](#_Toc95075083)

[3. Unit-Tests 7](#_Toc95075084)

[3. Inbetriebnahme 8](#_Toc95075085)

[4. Benutzerdokumentation - WPF 9](#_Toc95075086)

[4.1. Übersicht 9](#_Toc95075087)

[4.2. Filter 10](#_Toc95075088)

[4.3. Diagramm-Ansicht 12](#_Toc95075089)

[4.4. Tabellen-Ansicht 13](#_Toc95075090)

[4.5. Export-Funktion 14](#_Toc95075091)

[5. Benutzerdokumentation – Web / ASP.NET 16](#_Toc95075092)

[5.2 Bundesländer-Verwaltung 16](#_Toc95075093)

[5.1.1 Erstellen eines neuen Bundeslandes 17](#_Toc95075094)

[5.1.2 Editieren eines bestehenden Bundeslandes 18](#_Toc95075095)

[5.1.3 Entfernen eines bestehenden Bundeslandes 19](#_Toc95075096)

[5.2 Bezirke 20](#_Toc95075097)

[5.2.1 Erstellen eines neuen Bezirks 21](#_Toc95075098)

[5.2.2 Editieren eines bestehenden Bezirks 22](#_Toc95075099)

[5.2.3 Entfernen eines bestehenden Bezirks 23](#_Toc95075100)

[5.3 Benutzer 24](#_Toc95075101)

[5.3.1 Erstellen von neuen Benutzern 25](#_Toc95075102)

[5.3.2 Editieren eines bestehenden Benutzers 27](#_Toc95075103)

[5.3.3 Entfernen eines bestehenden Benutzers 29](#_Toc95075104)

[6. REST API 30](#_Toc95075105)

[6.1 Test der Methoden ohne Authorisierung 30](#_Toc95075106)

[6.2 Test der Methoden mit Authorisierung 30](#_Toc95075107)

[Abbildung 1: Klassendiagramm 3](#_Toc95074868)

[Abbildung 2: UML Übung 03 4](#_Toc95074869)

[Abbildung 3: ER-Diagramm 5](#_Toc95074870)

[Abbildung 4: Unit Tests 7](#_Toc95074871)

[Abbildung 5: How-To Datenbank hinzufügen 8](#_Toc95074872)

[Abbildung 6: How-to: Daten zur Datenbank hinzufügen 8](#_Toc95074873)

[Abbildung 7: UI - Übersichts-Tab 9](#_Toc95074874)

[Abbildung 8: UI - Bundesland Dropdown 10](#_Toc95074875)

[Abbildung 9: UI - Bezirk Dropdown 10](#_Toc95074876)

[Abbildung 10: UI - Zeitraum Dropdown 11](#_Toc95074877)

[Abbildung 11: UI Datumsauswahl 11](#_Toc95074878)

[Abbildung 12: UI - Diagrammansicht - Gesamt 12](#_Toc95074879)

[Abbildung 13: UI - Diagrammansicht - Infiziert 13](#_Toc95074880)

[Abbildung 14: UI – Tabellenansicht 13](#_Toc95074881)

[Abbildung 15: Export Dialog 14](#_Toc95074882)

[Abbildung 16: Export File 14](#_Toc95074883)

[Abbildung 17: Export Ergebnis 15](#_Toc95074884)

[Abbildung 18: Start – Bundesland – Anzeige 16](#_Toc95074885)

[Abbildung 19: Menüauswahl im Header 16](#_Toc95074886)

[Abbildung 20: Bundesland Erstellung - Required 17](#_Toc95074887)

[Abbildung 21: Bundesland Erstellung – Mindestlänge 17](#_Toc95074888)

[Abbildung 22: Bundesland Erstellung – Erfolgreich 17](#_Toc95074889)

[Abbildung 23: Bundesland Editierung - Required 18](#_Toc95074890)

[Abbildung 24: Bundesland Editierung - Mindestlänge 18](#_Toc95074891)

[Abbildung 25: Bundesland Editierung – Erfolgreich 18](#_Toc95074892)

[Abbildung 26: Bundesland Entfernung - Bezirk-Referenzen 19](#_Toc95074893)

[Abbildung 27: Bundesland Entfernung - Keine Referenz 19](#_Toc95074894)

[Abbildung 28: Bundesland Entfernung – Erfolgreich 19](#_Toc95074895)

[Abbildung 29: Bezirke – Anzeige 20](#_Toc95074896)

[Abbildung 30: Bezirk Erstellung - Required 21](#_Toc95074897)

[Abbildung 31: Bezirk Erstellung – Mindestlänge 21](#_Toc95074898)

[Abbildung 32: Bezirk Erstellung – Erfolgreich 22](#_Toc95074899)

[Abbildung 33: Bundesland Entfernung - Referenzierte Eilmeldungen 23](#_Toc95074900)

[Abbildung 34: Bundesland Entfernung - Keine Referenzen 23](#_Toc95074901)

[Abbildung 35: Bundesland Entfernung – Erfolgreich 23](#_Toc95074902)

[Abbildung 36: Benutzer - Anzeige 24](#_Toc95074903)

[Abbildung 37: Benutzer Erstellung – Required 25](#_Toc95074904)

[Abbildung 38: Benutzer Erstellung - Mindestlänge 25](#_Toc95074905)

[Abbildung 39: Benutzer Erstellung - Passwort bestätigen 26](#_Toc95074906)

[Abbildung 40: Benutzer Erstellung - Erfolgreich 26](#_Toc95074907)

[Abbildung 41: Benutzer Editierung - Altes Passwort 27](#_Toc95074908)

[Abbildung 42: Benutzer Editierung - Nur Name 27](#_Toc95074909)

[Abbildung 43: Benutzer Editierung - Neues Passwort 28](#_Toc95074910)

[Abbildung 44: Benutzer Entfernung – Referenzierte Eilmeldungen 29](#_Toc95074911)

[Abbildung 45: Benutzer Entfernung - Keine Referenzen 29](#_Toc95074912)

[Abbildung 46: Benutzer Entfernung – Erfolgreich 29](#_Toc95074913)

[Abbildung 47: API Abfrage - Keine Authorisierung 30](#_Toc95074914)

[Abbildung 48: API Login – Token 30](#_Toc95074915)

[Abbildung 49: API Abfrage - Alle Reports 31](#_Toc95074916)

[Abbildung 50: API Abfrage - Ein Report 32](#_Toc95074917)

[Abbildung 51: API POST - Erstellung 33](#_Toc95074918)

# 1. Architekturbeschreibung:

Das Projekt orientiert sich stark an demjenigen Projekt, welches im Laufe der Übung durchbesprochen wurde, weshalb der Großteil der Struktur gleich ist. Das entworfene Klassendiagramm sieht wie folgt aus:

Graphical user interface

Description automatically generated

Abbildung 1: Klassendiagramm

Die zwei größten Änderungen stellen die Komponenten „Domain“ und „Logic“ dar, was auch Sinn macht, da das grundlegende Gerüst gleich ist, während sich aber die Daten und die Anwendungsfälle ändern.

Zudem habe ich mich dazu entschieden die Verbindung des DAL-Package mit der Implementierung wie in der Übung zu gestalten, weil ich das Konzept von Attributes und Assemblies verinnerlichen wollte. Ich kann mir vorstellen kann, dass derartige Flexibilität, auch wenn in diesem Fall nicht gebraucht, sehr hilfreich sein kann.

Dadurch, dass sich Domäne und Logik am meisten verändert haben, möchte ich in den nächsten Kapiteln noch näher auf meine Entscheidungen eingehen.

## 1.1 Erweiterung UE03

Als Erstes möchte ich kurz darauf hinweisen, dass ich leider auf die Web-Version von Visual-Paradigm ausweichen musste, und deswegen zum einen leichte Inkonsistenzen in der Datendarstellung habe, sowie nur den neuen Teil des UMLs samt deren Abhängigkeiten darstelle.

Für die Erstellung der Webanwendung wurden Controller für jeden Datentyp, sowie die zugehörigen Razor-Views und Viewmodels für das Model-Binding im Projekt „SWO5.Dashboard.Web“ implementiert.

Zusätzlich dazu, wurde das Logic Package überarbeitet, sodass nun IManager existieren, die die bestehenden Daos verwenden, um einen einfachen Datenzugriff zu ermöglichen, ohne die gesamte Businesslogik (bzw. das vorherige CovidDashboard) einzubinden.

Für die Erstellung der API wurden entsprechende DTOs erstellt, welche den Klassen aus der Domain gleichkommen. Zusätzlich dazu wurden die notwendigen Controller für die Verwendung der Report-Requests sowie für den Login erstellt.

Überarbeitet wurde also das Logic-Projekt, während das Web, das DTO und das API Projekt hinzugekommen sind.

Diagram

Description automatically generated

Abbildung : UML Übung 03

## 1.2 Domäne

Um die Entscheidungen besser erklären zu können, möchte ich kurz das Datenbankschema illustrieren, da die Klassen in C# ähnlich aufgebaut sind:

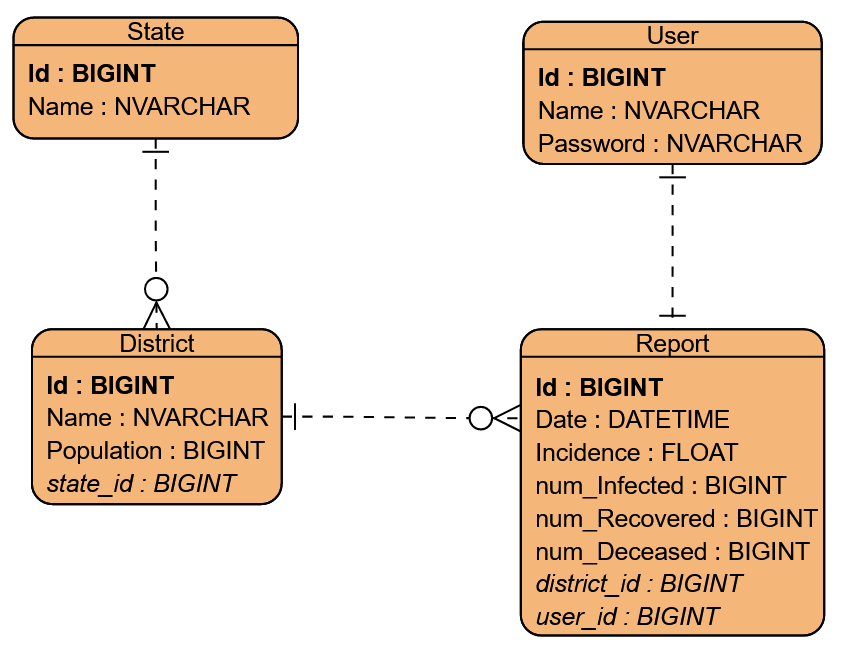


Abbildung 3: ER-Diagramm

Auf den ersten Blick wurden alle relevanten Daten in den Tabellen/Klassen gekapselt.

Anfangs dachte ich daran für District einen Composite-Key zu verwenden, was sich aber mit der Lösung aus dem Unterricht nicht gut vereinbaren lasst, da es wünschenswert ist, dass beim Einfügen bzw. Updaten die ID des Elements retourniert wird um dieses lokal zu vervollständigen bzw. eine Rückmeldung zu erhalten, ob die Datenbankbefehle erfolgreich waren oder nicht. Deswegen habe ich schlussendlich die gesamte GKZ (die indirekt das Bundesland enthält) als Key definiert.

Im Zuge meines Designs habe ich mich außerdem dazu entschieden nicht nur die IDs der Fremd-Entitäten in der C# Implementierung zu speichern, sondern vollständige Referenzen auf die entsprechenden Instanzen, um zu jeder Zeit vollständige Informationen zu haben.

### 1.2.1 Ergänzungen Übung 3

Im Laufe der Übung 3 mussten auch neue Bezirke/Bundesländer erstellt werden, was sich mit meiner bestehenden Datenbank schwierig gestaltete, da meine Primary Keys für diese Tabellen keine IDENTITY Spalten waren und somit die Keys nicht auto-inkrementierten. Das, sowie einige kleine damit verbundene Änderungen im DAL, wurden im Laufe dieser Übung eingearbeitet.

## 1.3. Logic

Wie im Unterricht wurde eine entsprechende Factory und ein entsprechendes Interface für alle gekapselten Funktionalitäten erstellt.

Für die User-Verwaltung wurden die Methoden UserExists und HashPassword angefügt, da ich die Passwörter sicher speichern will und ich mir vorstellen kann, dass es relevant ist, ob ein User (mit bestimmtem Namen) bereits existiert.

Außerdem wurde eine Funktion vorgesehen, welche ein gegebenes CSV der AGES ausliest und dessen Daten in die Datenbank schreibt.

Im Vergleich zur letzten Iteration (Übung 1) wurden die Funktionen, welche ein gewisses Zeitfenster betreffen, entfernt, da diese Funktionalität auch einfach in den entsprechenden Funktionen im GUI verwirklicht werden kann, wo ohnehin LINQ benötigt wird.

Hinzugekommen sind jedoch die Unit-Tests für alle Datenbank-Operationen, die Bundesländer und Bezirke betreffen. Bei der letzten Iteration wurden diese bewusst weggelassen, da diese Datensatze normalerweise konstant bleiben. Diese wurden vollständigkeitshalber ergänzt, da es bei anderen Personen deswegen zu einem Punkteabzug gekommen ist.

Wie bereits beim UML erwähnt sind bei der Logik im Rahmen der Übung 3 nur die Manager-Klassen hinzugekommen.

## 1.4 Persönliche Änderungen

Ich habe mich, konträr zur Übung, dazu entschieden, dass ich die JOINS wirklich in den Commands/Selects selbst verwirklichen will und nicht nachträglich im Code die Einträge zusammenstöpseln muss. Daraus ergibt sich, dass sämtliche Read-Methoden für District und Report auch in der spezialisierten Klasse genauer ausgeführt werden müssen, da sich die select-statements nicht mehr in AbstractDao zusammenfassen lassen.

Als Lösung dafür würde ich eine dynamische Bindung implementieren, indem ich die generell formulierten Methoden in AbstractDao als virtual kennzeichne und dann in den relevanten Subklassen überschreibe.

# 2. Quellcode

Der Quellcode wurde aufgrund der schieren Menge an Dateien nicht direkt ins PDF kopiert, weshalb ich hier auf das Visual Studio Projekt verweise.

Um den Code in Betrieb zu nehmen, muss vorher noch die Datenbank mithilfe des beigelegten SQL-Scripts erstellt und befüllt werden. Mehr dazu auch in der Benutzerdokumentation.

# 3. Unit-Tests

Ähnlich zum Quellcode werden auch diese im Detail im Code durchgegangen.

Die Testfälle, welche alle Funktionen der Business-Logik abdecken, waren leider teils (nicht vermeidbar) abhängig voneinander, es wurde aber so gut wie möglich versucht, dies zu reduzieren.

Anbei noch ein Screenshot des erfolgreichen Durchlaufs:

A picture containing text

Description automatically generated

Abbildung 4: Unit Tests

Wie oben bereits erwähnt, wurden die Funktionen der Business-Logik um die CRUD-Methoden der Bundesländer und Bezirke erweitert bzw. die Funktionen mit Zeitfenstern entfernt.

# 3. Inbetriebnahme

Um das Programm in Betrieb nehmen zu können, muss zuerst die Datenbasis importiert werden. Die gesamte Datenstruktur, sowie alle Meldungen und zusätzlich notwendigen Daten können mithilfe der Datei „CREATE\_DATABASE.sql“ angelegt werden.

Hierfür wird in Visual Studio im SQL Server Object Explorer eine Query auf den zuständigen SQL-Express-Server angelegt und eine Datenbank namens „Covid“ erstellt (siehe erste Zeile im SQL-File).

Graphical user interface, text

Description automatically generated

Abbildung 5: How-To Datenbank hinzufügen

Danach kann wiederum eine Verbindung/Query zu der neu erstellten Datenbank aufgebaut werden, um mit den restlichen Zeilen alle Daten anzulegen.

Text

Description automatically generated

Abbildung 6: How-to: Daten zur Datenbank hinzufügen

# 4. Benutzerdokumentation - WPF

In diesem Bereich werden die wichtigsten Funktionen der Desktopanwendung erläutert.

## 4.1. Übersicht

Beim Programmstart wird der Benutzer vom UI im „Übersicht“-Tab begrüßt. In diesem sind die momentan aktiven Fälle, die neuen Fälle für den heutigen Tag, sowie eine Übersicht über die aktuellen 7-Tage-Inzidenzen aller Bundesländer zu sehen. Diese Felder sind von den Einstellungen oben unbeeinflusst und werden nur zum Start des Programms berechnet.

Graphical user interface

Description automatically generated with low confidence

Abbildung 7: UI - Übersichts-Tab

Anmerkung für Tutoren:

(Stand 10.12.2022, neuester Datensatz)

Die momentanen aktiven Fälle fallen bei meiner Berechnung deutlich höher aus als erwartet (ist: 452361, soll: 77165). Im Endeffekt sollten sich die aktiven Fälle aus der Differenz zwischen Summe der Infizierten Personen und der Summe der Genesenen/Verstorbenen Personen ergeben. Also praktisch Insgesamt\_Infiziert – (Insgesamt\_Genesen + Insgesamt\_Verstorben) – das scheint jedoch bei diesem Datenset nicht zu funktionieren.

Die neuen Fälle weichen leicht ab mit einem erwarteten Ergebnis von 10842 im Vergleich zu dem hier präsentierten Ergebnis von 11288.

## 4.2. Filter

Oben in der Ansicht zu sehen, sind die Filter welche festgelegt werden können. Über Dropdowns kann eine Kombination von Bundesland, Bezirk und einem Datumsbereich ausgewählt werden, anhand dessen die Diagramme und die Tabelle neu berechnet werden.

Chart

Description automatically generated

Abbildung 8: UI - Bundesland Dropdown

Graphical user interface

Description automatically generated with medium confidence

Abbildung 9: UI - Bezirk Dropdown

Graphical user interface, application

Description automatically generated

Abbildung 10: UI - Zeitraum Dropdown

Graphical user interface, chart

Description automatically generated

Abbildung 11: UI Datumsauswahl

## 4.3. Diagramm-Ansicht

Klickt man auf den Tab „Diagramme“, welcher schon während der Präsentation der Filter hergezeigt wurde, so erhält man einen Überblick über das Infektionsgeschehen in Österreich bzw. in den ausgewählten Regionen im gewählten Zeitraum.

Graphical user interface, chart

Description automatically generated

Abbildung 12: UI - Diagrammansicht - Gesamt

Standardmäßig werden alle Graphen gleichzeitig angezeigt, jedoch führt das zum Beispiel in diesem Szenario dazu, dass die Werte der Infektionen, Genesungen und Tode untergeht, da die Inzidenz zu hoch ist.

Aus diesem Grund, ist es möglich über die Combobox „Typ“ auszuwählen, welcher Graph dargestellt werden soll. Die Werte werden auch entsprechend für eine anständige Darstellung skaliert:

Chart

Description automatically generated

Abbildung 13: UI - Diagrammansicht - Infiziert

## 4.4. Tabellen-Ansicht

Alternativ zum Diagramm können die Daten auch als Tabelle visualisiert werden.

Graphical user interface, application, table

Description automatically generated

Abbildung 14: UI – Tabellenansicht

## 4.5. Export-Funktion

Über den Button „Export“ können Benutzer die dargestellte Tabelle als CSV exportieren. Aufgrund der Tatsache, dass der Decimal-Point im deutschen das Komma ist, wurde als Delimiter das Semicolon verwendet.

Für einen Export muss der Nutzer nur den Button „Exportieren“ anklicken und einen Speicherort festlegen.

A screenshot of a computer

Description automatically generated

Abbildung 15: Export Dialog

Graphical user interface, text

Description automatically generated

Abbildung 16: Export File

A screenshot of a computer

Description automatically generated with low confidence

Abbildung 17: Export Ergebnis

# 5. Benutzerdokumentation – Web / ASP.NET

In diesem Teil wird die Funktion der Web-Implementierung zur Verwaltung der Stammdaten erläutert.

## 5.2 Bundesländer-Verwaltung

Beim Aufrufen der Anwendung/Website wird der User durch die Anzeige der Bundesländer begrüßt.

Graphical user interface, table

Description automatically generated with medium confidence

Abbildung : Start – Bundesland – Anzeige

In diesem Menü werden nicht nur die Daten jedes Bundeslandes angezeigt – der User kann zudem neue Bundesländer erstellen oder bestehende Bundesländer editieren und löschen.

Um in ein anderes Menü zu wechseln, können die Links im Header-Bereich verwendet werden:

Graphical user interface, text, application, chat or text message

Description automatically generated

Abbildung : Menüauswahl im Header

### 5.1.1 Erstellen eines neuen Bundeslandes

Beim Erstellen des Bundeslandes muss der Benutzer lediglich den Namen des Bundeslandes eingeben. Dieser muss zwischen 2 und 50 Zeichen lang sein und ist Voraussetzung für die korrekte Erstellung. Werden diese Bedingungen nicht erfüllt, kommt es zur Fehlermeldung:

Graphical user interface, text, application

Description automatically generated with medium confidence

Abbildung : Bundesland Erstellung - Required

Graphical user interface, text, application, Word, email

Description automatically generated

Abbildung : Bundesland Erstellung – Mindestlänge

Werden die Bedingungen erfüllt, so findet man den Eintrag nach Umleitung auf die Hauptseite in der Liste:

Table

Description automatically generated

Abbildung : Bundesland Erstellung – Erfolgreich

### 5.1.2 Editieren eines bestehenden Bundeslandes

Beim Editieren wird das gleiche Formular wie beim Erstellen aufgerufen, jedoch vor ausgefüllt mit den bestehenden Daten. Bei der Editierung gelten die gleichen Regeln wie bei der Erstellung.

Graphical user interface, application, Word

Description automatically generated

Abbildung : Bundesland Editierung - Required

Graphical user interface, text, application

Description automatically generated

Abbildung : Bundesland Editierung - Mindestlänge

Werden die Daten widerum korrekt eingegeben, so findet man das veränderte Element auf:

Table

Description automatically generated

Abbildung : Bundesland Editierung – Erfolgreich

### 5.1.3 Entfernen eines bestehenden Bundeslandes

Beim Löschen von Bundesländern gibt es nicht viel zu beachten. Der User wird lediglich nochmals gefragt, ob er das Bundesland wirklich löschen will. Versucht der Benutzer jedoch ein Bundesland zu löschen, dass durch bestehende Bezirke referenziert wird, so erhält der Benutzer eine Fehlermeldung:

Graphical user interface, text, application

Description automatically generated

Abbildung : Bundesland Entfernung - Bezirk-Referenzen

Wird ein Bundesland gelöscht, zu dem es keine Bezirke gibt, so funktioniert der Vorgang jedoch und das Element verschwindet aus der Liste.

Graphical user interface, text, application

Description automatically generated

Abbildung : Bundesland Entfernung - Keine Referenz

Table

Description automatically generated

Abbildung : Bundesland Entfernung – Erfolgreich

## 5.2 Bezirke

Zusätzlich zu den Bundesländern können natürlich auch die Bezirke verwaltet werden. Ähnlich zu den Bundesländern können auch diese erstellt, editiert und entfernt werden.

Angezeigt wird anfangs allerdings wieder die Liste:

Table

Description automatically generated

Abbildung : Bezirke – Anzeige

### 5.2.1 Erstellen eines neuen Bezirks

Beim Erstellen eines neuen Bezirks muss der Benutzer verpflichtend dessen Name, die Population, sowie das darüberliegende Bundesland auswählen.

Alle Felder sind verpflichtend. Zusätzlich gilt wieder die Einschränkung, dass der Name zwischen 2 und 50 Zeichen lang sein muss, sowie, dass die Population nur positive Zahlen annehmen darf.

Graphical user interface, text, application, email

Description automatically generated

Abbildung : Bezirk Erstellung - Required

Graphical user interface, application, Word

Description automatically generated

Abbildung : Bezirk Erstellung – Mindestlänge

Werden die Daten korrekt ausgefüllt, so findet sich der Bezirk in der Liste wieder:

Table

Description automatically generated

Abbildung : Bezirk Erstellung – Erfolgreich

### 5.2.2 Editieren eines bestehenden Bezirks

Wie bei den Bundesländern sieht das Editier-Formular wieder analog zum Erstell-Formular aus. Beim Aufrufen werden wieder die bestehenden Daten vorausgefüllt:

Graphical user interface, application

Description automatically generated

Auf die restlichen Anwendungsfälle bezüglich der Validierung bei der Editierung wird aufgrund der Redundanz nicht mehr eingegangen und kann im Zweifel durch Ausführen des Programms kontrolliert werden.

### 5.2.3 Entfernen eines bestehenden Bezirks

Wie beim Entfernen der Bundesländer wird auch hier zuerst der Nutzer nochmals gefragt, sowie beim Löschen überprüft, ob der Bezirk in einer Eilmeldung referenziert wurde.

Graphical user interface, text, application

Description automatically generated

Abbildung : Bundesland Entfernung - Referenzierte Eilmeldungen

Graphical user interface, text, application, Word

Description automatically generated

Abbildung : Bundesland Entfernung - Keine Referenzen

Table

Description automatically generated

Abbildung : Bundesland Entfernung – Erfolgreich

## 5.3 Benutzer

Zu guter letzt können auch noch Benutzer in der Webanwendung verwaltet werden. Wiederum gestaltet sich dies analog zu Bezirken und Bundesländern.

Graphical user interface, text, application, email

Description automatically generated

Abbildung : Benutzer - Anzeige

### 5.3.1 Erstellen von neuen Benutzern

Benutzer bestehen aus einem Namen und einem Passwort.

Der Name muss zwischen 2 und 50 Zeichen lang sein, während das Passwort mindestens aus 6 Zeichen bestehen muss und keine Maximal-Länge hat, da es ohnehin mittels SHA256 in einen 64-Zeichen langen Hash transformiert wird.

Graphical user interface, application

Description automatically generated

Abbildung : Benutzer Erstellung – Required

Graphical user interface, application

Description automatically generated

Abbildung : Benutzer Erstellung - Mindestlänge

Hat der Benutzer valide Daten eingegeben, muss der Benutzer das Passwort zudem noch bestätigen, indem er es ein zweites Mal eintippt.

Graphical user interface, application

Description automatically generated

Abbildung : Benutzer Erstellung - Passwort bestätigen

Wurden die Daten richtig eingegeben und das Passwort bestätigt, so findet man den Benutzer mitsamt seinem Hash in der Liste wieder.

Application

Description automatically generated with low confidence

Abbildung : Benutzer Erstellung - Erfolgreich

### 5.3.2 Editieren eines bestehenden Benutzers

Beim Editieren eines Benutzers ist es wichtig, dass dieser sein bestehendes Passwort weiß, sodass niemand anders seine Daten editieren kann.

Es kann der Name sowie das Passwort geändert werden. Bei Leerlassen des „New Password“-Feldes, wird das Passwort beim Alten belassen.

Graphical user interface, text, application, email

Description automatically generated

Abbildung : Benutzer Editierung - Altes Passwort

Graphical user interface, application

Description automatically generated

Graphical user interface, application

Description automatically generated

Abbildung : Benutzer Editierung - Nur Name

Graphical user interface, application

Description automatically generated

A picture containing application

Description automatically generated

Abbildung : Benutzer Editierung - Neues Passwort

Es ist aufgrund der ähnlichen Hashes leider schwerer zu erkennen, jedoch wurde das Passwort geändert.

### 5.3.3 Entfernen eines bestehenden Benutzers

Für den Benutzer gilt beim Entfernen das gleiche wie beim Bezirk – er darf nicht in einem Report referenziert sein!

Graphical user interface, text, application

Description automatically generated

Abbildung : Benutzer Entfernung – Referenzierte Eilmeldungen

Graphical user interface, text, application

Description automatically generated

Abbildung : Benutzer Entfernung - Keine Referenzen

Graphical user interface, application

Description automatically generated

Abbildung : Benutzer Entfernung – Erfolgreich

# 6. REST API

Für den Zugriff auf Eilmeldungen ist eine REST Schnittstelle vorgesehen. Mithilfe von dieser Schnittstelle können sich User authentifizieren und danach Reports abfragen und erstellen.

Für die Tests auf diese Schnittstelle bzw. Anleitungen zur Erstellung von gültigen Datenstrukturen in JSON liegt ein File „API\_TEST.http“ vor, welches in VS Code geöffnet werden kann.

## 6.1 Test der Methoden ohne Authorisierung

Der Aufruf jeglicher Methoden ohne Authorisierung bzw. vorangehenden Login führt zu folgender Nachricht:

Text

Description automatically generated

Abbildung : API Abfrage - Keine Authorisierung

## 6.2 Test der Methoden mit Authorisierung

Als Erstes muss sich der Benutzer anmelden. Das bereitgestellte File enthält einen fertigen User zur Anmeldung.

Stimmen die Daten mit jenen in der Datenbank überein, so wird ein Token retourniert:

Text

Description automatically generated

Abbildung : API Login – Token

Der oben erhaltene Token muss nach dem Login bei den Requests mit übergeben werden, um sich zu authorisieren. Der gegebene Token muss im File nur ausgetauscht werden, damit die Requests funktionieren.

Beim Abfragen „aller“ Reports, musste die Größe eingeschränkt werden, da VS Code sowie der Browser sonst einfroren. Es werden daher nur 100 der ca. 65000 Datensätze retourniert.

Text

Description automatically generated

Abbildung : API Abfrage - Alle Reports

Alternativ dazu kann natürlich auch ein spezifischer Report (z.B. mit der ID 15000) abgefragt werden:

Text

Description automatically generated

Abbildung : API Abfrage - Ein Report

Zu guter letzt kann auch noch ein Report erstellt werden:

Text

Description automatically generated

Abbildung : API POST - Erstellung