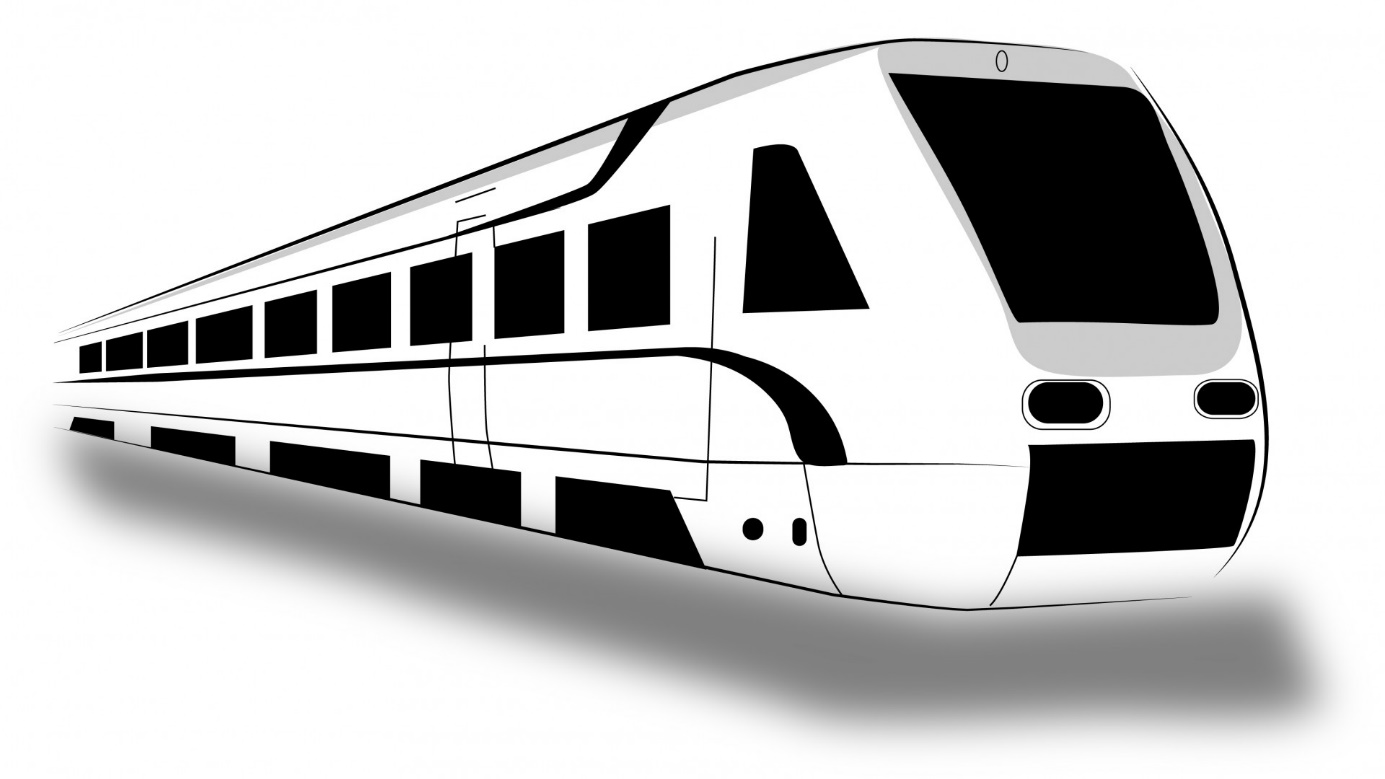
**M318 Projektarbeit**

**ÖV-Suche**







**Lea Schöpfer**

KR5, ICT. Berufsbildung Zentralschweiz

17.12.2020 Adligenswil

# Inhalt

[1 Inhalt 1](#_Toc59060567)

[2 Einleitung 2](#_Toc59060568)

[2.1 Aufgabenstellung 2](#_Toc59060569)

[2.2 Zweck der Dokumentation 2](#_Toc59060570)

[3 Mockup 2](#_Toc59060571)

[4 Use Case 4](#_Toc59060572)

[5 User Stories 4](#_Toc59060573)

[5.1 Überblick 4](#_Toc59060574)

[5.2 Ausführung User-Stories mit Akzeptanzkriterien 4](#_Toc59060575)

[5.2.1 Priorität 1 4](#_Toc59060576)

[5.2.2 Priorität 2 5](#_Toc59060577)

[5.2.3 Priorität 3 6](#_Toc59060578)

[6 Aktivitätsdiagramm 7](#_Toc59060579)

[7 Funktionen 8](#_Toc59060580)

[7.1 Umgesetzte Funktionen 8](#_Toc59060581)

[7.2 Fehlende Funktionen 9](#_Toc59060582)

[8 Test 10](#_Toc59060583)

[8.1 Testplan 10](#_Toc59060584)

[8.2 Testprotokoll 10](#_Toc59060585)

[9 Installation und Deinstallation 11](#_Toc59060586)

[9.1 Installationsanleitung 11](#_Toc59060587)

[9.1.1 Voraussetzung 11](#_Toc59060588)

[9.1.2 Vorgehen 11](#_Toc59060589)

[9.2 Deinstallationsanleitung 13](#_Toc59060590)

[10 Fazit 13](#_Toc59060591)

# Einleitung

Im ÜK M318 "Analysieren und objektbasiert programmieren" war es unsere Aufgabe eine Applikation zu planen, entwickeln, testen und dokumentieren. Wir führten ein ganzes Projekt angefangen bei der Kundenwunschinterpretation, der Ausarbeitung eines Mockups und User Stories bis zur eigentlichen Umsetzung und Testung der Applikation durch.

## Aufgabenstellung

Die Aufgabe war es, die Entwicklung einer Applikation, die öffentliche Verkehrsverbindungen von Punkt A nach B anzeigt. Dazu gehören Funktionen wie die Eingabe des Datums und der Uhrzeit sowie eine eigenständige Abfahrtsanzeige einzelner Stationen und einer Kartenansicht. Genauere Anforderungen des Kunden sind in den User Stories zu finden. Alle Daten werden mit dem frei zur Verfügung stehenden API eingebunden: <https://transport.opendata.ch/>

Das ganze Projekt ist im GIT-Repository unter folgender Adresse zu finden: <https://github.com/leasch36/modul-318-student>

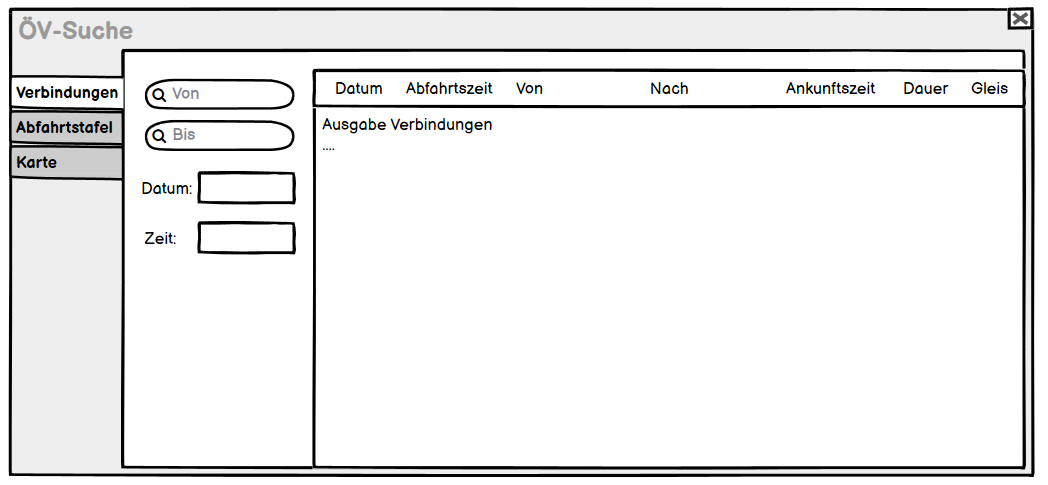
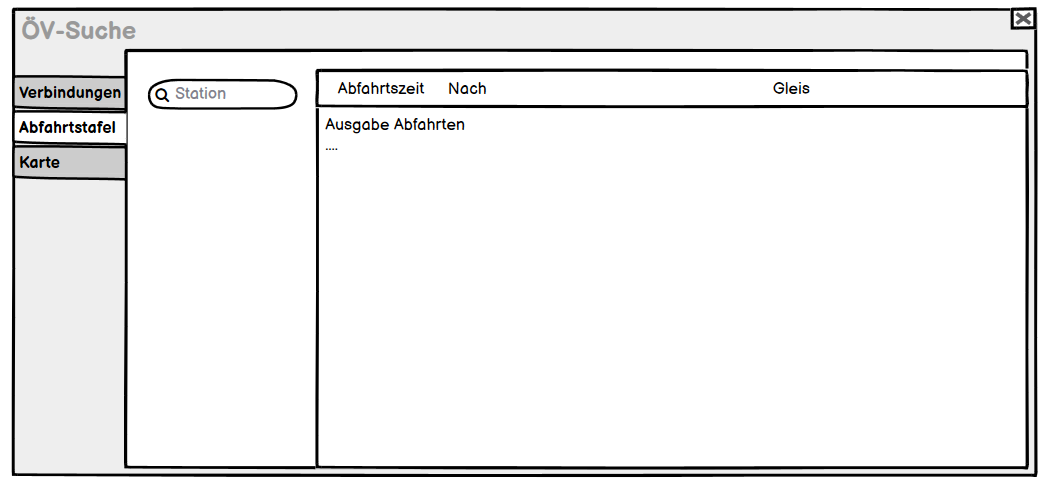
## Zweck der Dokumentation

Diese Dokumentation dient dazu die Entwicklung und Architektur meiner Applikation 'ÖV-Suche' zu dokumentieren sowie dessen Funktionen und Installation darzustellen.

# Mockup

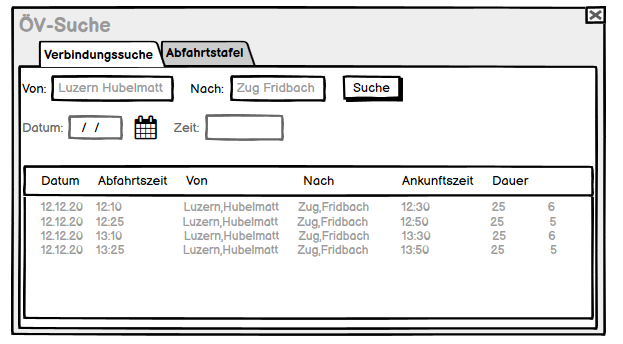
Ich habe mich bei der Erstellung des Mockups an die grundlegendsten Funktionalitäten gehalten. Die Mockups habe ich mithilfe von Balsamiq Wireframes erstellt.

Beim Erstellen der Benutzeroberfläche habe ich schnell bemerkt, dass ich für die Umsetzung des ersten Mockups lange suchen müsste. Beispielsweise wusste ich nicht, wie ich eine vertikale Tabansicht erzeuge.

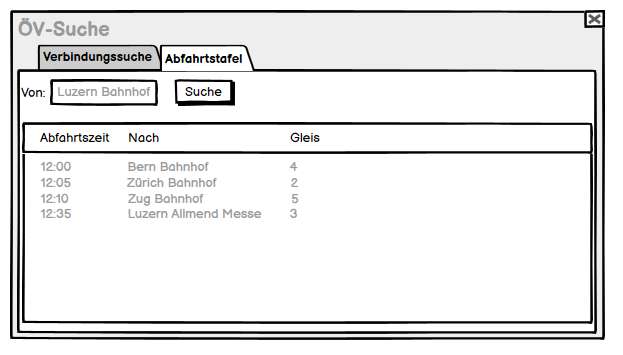
 

Stattdessen habe ich nach dem Durchsehen der Tools ein neues Mockup erstellt. Beispielsweise mit einer horizontalen Tabansicht.

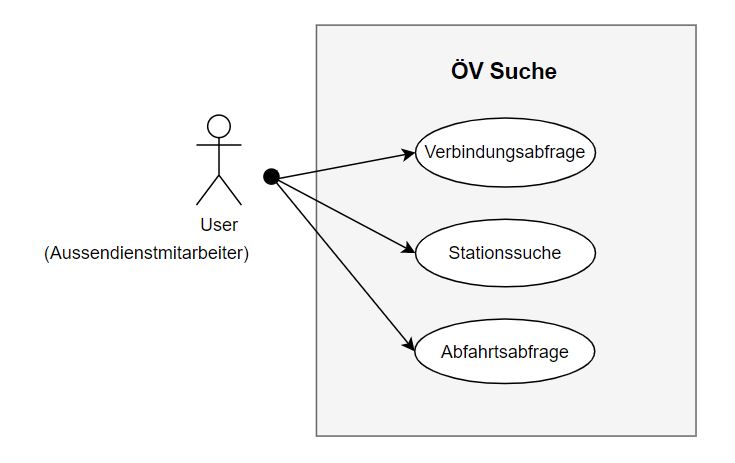
**Verbindungssuche**



**Abfahrtstafel**



# Use Case



# User Stories

## Überblick

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Was** | **Priorität\*** | **Status** |
| US01 | Eingabe Start- und Endstation | 1 | umgesetzt |
| US02 | Nächsten 4 Verbindungen zwischen Start- und Endstation ausgegeben | 1 | umgesetzt |
| US03 | Abfahrtstafel einer bestimmten Station | 1 | umgesetzt |
| US04 | Verbindungssuche nach Datum und Uhrzeit filtern | 2 | umgesetzt |
| US05 | Suchvorschläge während der Eingabe einer Station | 2 | umgesetzt |
| US06 | Verbindung per E-Mail verschicken | 3 | umgesetzt |
| US07 | Bestimmte Station auf einer Karte anzeigen | 3 | umgesetzt |
| US08 | Stationen im Umkreis von 10km auf einer Karte anzeigen | 3 | offen |
| US09 | Start- und Endstation für Rückfahrt tauschen | 3 | umgesetzt |

\*Priorität: hoch[1], mittel[2], tief[3]

## Ausführung User-Stories mit Akzeptanzkriterien

### Priorität 1

**US01- Eingabe Start- und Endstation:**

**Als** Aussendienstmitarbeiter

**möchte ich** Start- und Endstation eingeben können,

**damit** ich Verbindungen heraussuchen kann.

*Akzeptanzkriterien:*

**-** Start- und Endstation kann eingeben werden

**-** Start- und Endstation sind ein Pflichtfeld

**-** Verbindungen werden ausgegeben

*Priorität: Status:*

1 umgesetzt

**US02- Verbindungsvorschläge**  
**Als** Aussendienstmitarbeiter

**möchte ich** die nächsten 4 Verbindungen zwischen der eingegebenen Start- und Endstation angezeigt bekommen,

**damit** ich weiss welche Verbindung ich nehmen muss.

*Akzeptanzkriterien:*

- mind. nächsten 4 Verbindungen werden in einer Liste ausgegeben.

- Die Verbindungen werden aufsteigend nach Abfahrtszeig sortiert

- Es wird die Abfahrts- und Ankunftszeit angezeigt

*Priorität: Status:*

1 umgesetzt

**US03- Abfahrtstafel**

**Als** Aussendienstmitarbeiter

**möchte ich** sehen, welche Verbindungen ab einer bestimmten Station vorhanden sind (Abfahrtstafel),

**damit** ich die nächsten Verbindungen von dieser Station finde.

*Akzeptanzkriterien:*  
- Es werden alle nächsten Fahrten ab dieser Station ausgegeben

**-** Die Fahrten werden zeitlich aufsteigend sortiert

**-** Es wird angezeigt wohin diese Verbindung fährt

*Priorität: Status:*

1 umgesetzt

### Priorität 2

**US04- Datum- und Uhrzeiteingabe**  
**Als** Aussendienstmitarbeiter

**möchte ich** Verbindungssuchen nach Datum und Uhrzeit filtern können,

**damit** ich zukünftige Fahrten im Voraus planen kann.

*Akzeptanzkriterien:*  
- Datum und Uhrzeit kann man eingeben

**-** Entsprechende Verbindungen werden gesucht und ausgegeben

*Priorität: Status:*

2 umgesetzt

**US05- Suchvorschläge**  
**Als** Aussendienstmitarbeiter

**möchte ich** bereits während der Eingabe einer Station, Suchvorschläge erscheinen,

**damit** ich weniger tippen muss.

*Akzeptanzkriterien:*

- Während dem Tippen werden in einer Liste unter dem Eingabefeld mögliche Vorschläge angezeigt.

- Durch klicken auf den Vorschlag wird dieser in das Eingabefeld eingefügt.

*Priorität: Status:*

2 umgesetzt

### Priorität 3

**US06- Mailversand**  
**Als** Aussendienstmitarbeiter

**möchte ich** eine Verbindung meiner Wahl per E-Mail verschicken können,

**damit** andere Mitarbeiter von meinen Recherchen profitieren können.

*Akzeptanzkriterien:*

- Ich kann jede E-Mail-Adresse eingeben

- Ich kann eine beliebige Verbindung senden

- Mail wird verschickt

*Priorität: Status:*

3 umgesetzt

**US07- Karte**  
**Als** Aussendienstmitarbeiter

**möchte ich** eine bestimmte Station auf einer Karte anzeigen lassen,

**damit** ich den Weg zur Station finde.

*Akzeptanzkriterien:*

- Karte wird angezeigt

- Station wird angezeigt

*Priorität: Status:*

3 umgesetzt

**US08- Nahegelegene Stationen**  
**Als** Aussendienstmitarbeiter

**möchte ich** alle Stationen im Umkreis von 10km auf einer Karte angezeigt haben,

**damit** ich keine Umwege machen muss.

*Akzeptanzkriterien:*

- Stationen werden in einer Liste ausgegeben

- Es wird der Name und die Distanz des aktuellen Standorts angezeigt

*Priorität: Status:*

3 offen

**US09- Rückfahrt** (eigene User-Story)  
**Als** Aussendienstmitarbeiter

**möchte ich** die Feldinhalte von Start- und Endstation mit einem Klick tauschen können,

**damit** ich bei der Abfrage einer Verbindung für die Rückfahrt die Stationsnamen nicht erneut eingeben muss.

*Akzeptanzkriterien:*

- Button kann gedrückt werden.

- Inhalt der Suchfelder Start- und Endstation werden getauscht.

*Priorität: Status:*

3 umgesetzt

Hinweis: Meine eigene User-Story habe ich im Verlauf des Projekts angepasst.

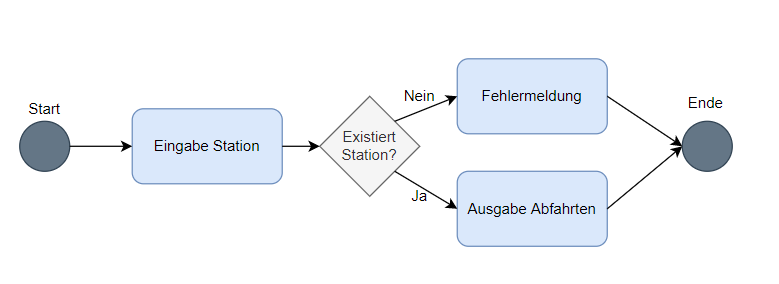
(Alte User-Story: Als Aussendienstmitarbeiter möchte ich den momentanen Stand vom

Verkehr sehen, damit ich nicht im Stau stecken bleibe.)

# Aktivitätsdiagramm

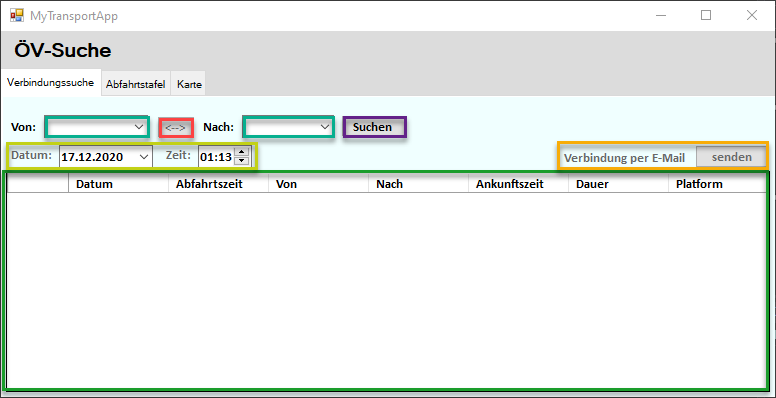
Aktivitätsdiagramm für die User-Story US03:

**Als** Aussendienstmitarbeiter **möchte ich** sehen, welche Verbindungen ab einer bestimmten Station vorhanden sind (Abfahrtstafel), **damit** ich die nächsten Verbindungen von dieser Station finde.



# Funktionen

## Umgesetzte Funktionen

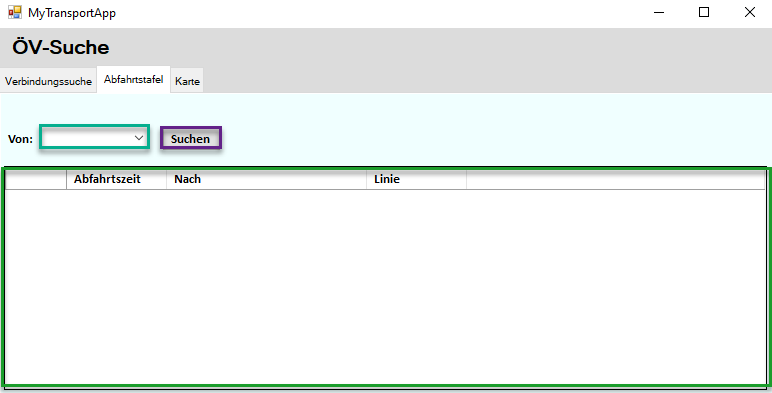


Im ersten Tab des Programmes befindet sich die Verbindungssuche. Es kann eine Abfahrts- und Ankunftsstation eingegeben werden. Während der Eingabe werden Stationsvorschläge angezeigt. Die beiden Felder Abfahrts- und Ankunftsstation sind Pflichtfelder. Mit dem Button Suchen werden Verbindungen gesucht und in der Tabelle ausgegeben. Sofern kein Datum oder eine Uhrzeit eingegeben wurde, so wird das aktuelle Datum und Uhrzeit genommen. Es werden die zeitlich nächsten 7 Verbindungen zwischen den eingegebenen Abfahrts- und Ankunftsstationen ausgegeben.

Die Suche kann zusätzlich mit dem Datum und Zeit Feld gefiltert werden. Dies ist allerdings optional. Mit dem Drücken auf den Knopf Suchen, werden Verbindungen für die gewünschte Strecke zum gewünschten Datum und Zeitpunkt ausgegeben. Die Verbindungen sind nach Datum und Uhrzeit aufsteigend sortiert.

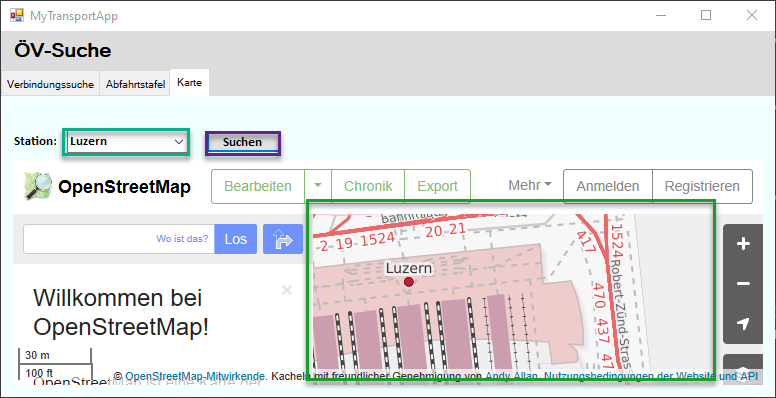
Mit dem Button <-> können die eingegebene Abfahrts- und Ankunftsstation getauscht werden.

In der Tabelle kann eine Verbindung ausgewählt werden und mit dem Klick auf den Knopf senden, kann die Verbindung per E-Mail verschickt werden.



Im zweiten Tab des Programmes befindet sich die Abfahrtstafel. Es kann eine Abfahrtsstation eingegeben werden. Während der Eingabe werden Stationsvorschläge angezeigt. Mit dem Button Suchen werden Verbindungen gesucht und in der Tabelle ausgegeben. Es werden die zeitlich nächsten 7 Verbindungen ausgegeben. Die Abfahrtstafel ist ebenfalls nach Datum und Uhrzeit aufsteigend sortiert.

**Karte**



Im dritten Tab des Programms befindet sich eine Karte. Es kann eine Station eingegeben werden. Während der Eingabe werden Stationsvorschläge angezeigt. Mit dem Button Suchen wird die Station auf der Karte angezeigt.

## Fehlende Funktionen

Die Funktionalität der User-Story US08 fehlt noch, da ich zu wenig Zeit für diese hatte:

**"Als** Aussendienstmitarbeiter **möchte ich** alle Stationen im Umkreis von 10km auf einer Karte angezeigt haben, **damit** ich keine Umwege machen muss."

# Test

## Testplan

Dies ist ein Testplan für die User-Stories mit Priorität 1.

Voraussetzung: Programm ist gestartet. Der Tester befindet sich im Tab <<Verbindungssuche>>.

**Stationssuche**

**Verbindungssuche**

|  |  |  |
| --- | --- | --- |
| **Schritt** | **Aktivität** | **Erwartetes Resultat** |
| 1 | In das Eingabefeld **Von** 'Luzern' eintippen. | Luzern kann eingegeben werden. |
| 2 | In das Eingabefeld **Nach** 'Zug' eintippen. | Zug kann eingegeben werden. |
| 3 | Den Knopf **Suchen** drücken. | Es werden Verbindungen mit Abfahrtsstation Luzern und Ankunftsstation Zug ausgegeben.  Es werden die (zeitlich) nächsten 7 Verbindungen ausgegeben.  Die Verbindungen sind aufsteigend nach Datum und Zeit der Abfahrt sortiert. |

**Abfahrtstafel**

|  |  |  |
| --- | --- | --- |
| **Schritt** | **Aktivität** | **Erwartetes Resultat** |
| 1 | In das Feld **Von** 'Luzern' eintippen. | Luzern kann eingegeben werden. |
| 2 | Den Knopf **Suchen** drücken. | Es werden alle Verbindungen mit der Abfahrtsstation Luzern ausgegeben.  Es werden die (zeitlich) nächsten 7 Verbindungen ausgegeben.  Die Verbindungen sind aufsteigend nach Datum und Zeit der Abfahrt sortiert.  Es wird angezeigt wohin die Verbindungen fahren. |

## Testprotokoll

Voraussetzung: Programm ist gestartet.

Durchführungsdatum: 17.12.2020

Tester/in: Lea Schöpfer

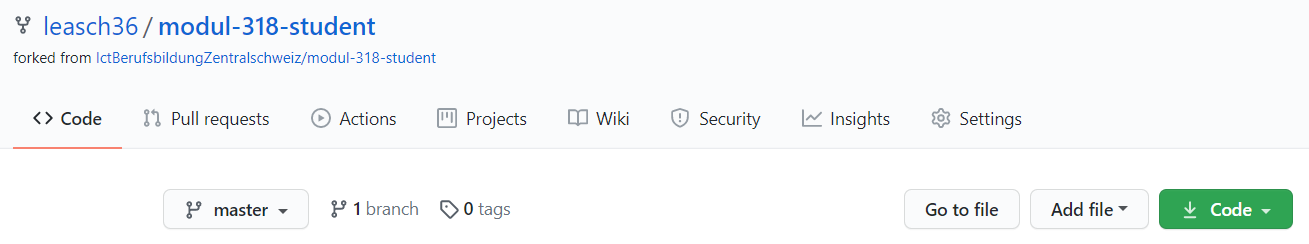
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Schritt** | **Aktivität** | **Erwartetes Resultat** | **Abw. Resultat** | **Erfüllt** |
|  |  |  | keine | Ja |
|  |  |  |  | Nein |
|  |  |  |  |  |

# Installation und Deinstallation

## Installationsanleitung

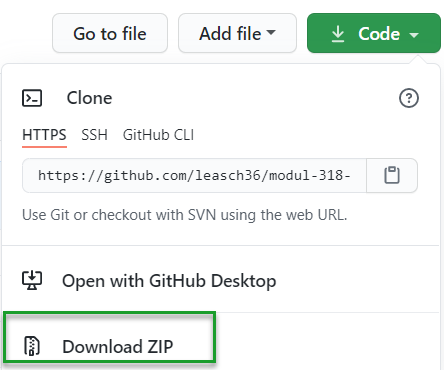
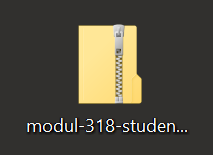
### Voraussetzung

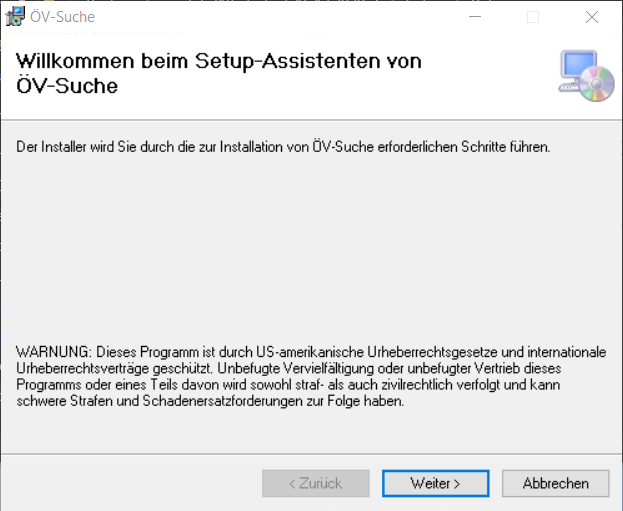
* Internet Verbindung, um Projekt von GitHub herunterzuladen.
* GitHub Name: **leasch36**
* Projektname: [**modul-318-student**](https://github.com/edp-justinek/modul-318-student)



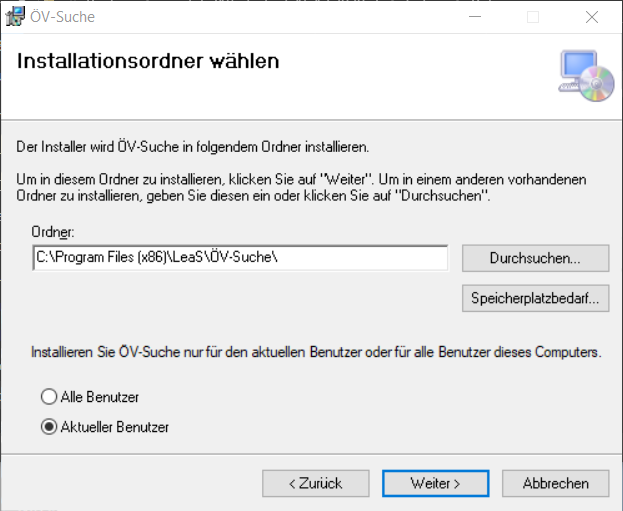
### Vorgehen

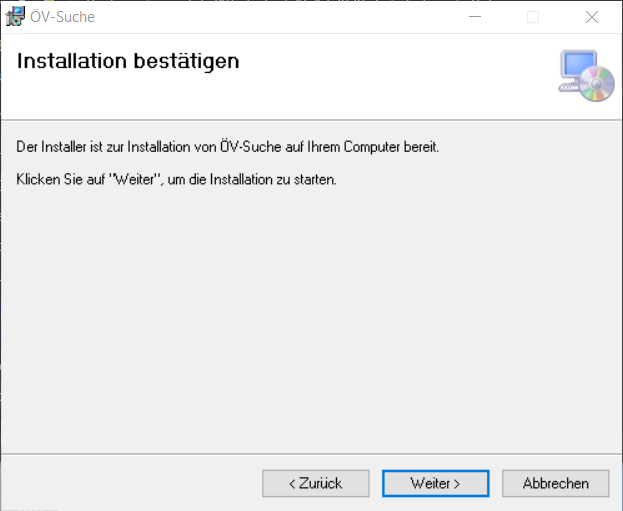
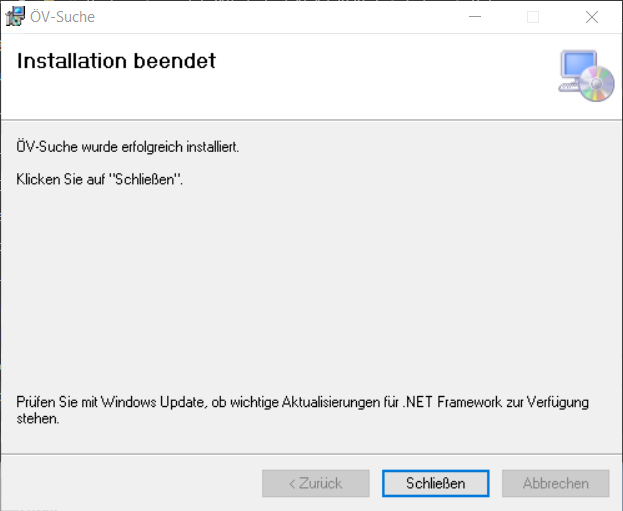
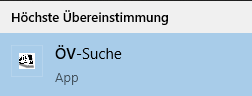
* Download von Github als Zip

* Zip Datei entpacken
* Folgenden Pfad im entpackten Ordner öffnen: **Setup\Release**
* Anschliessend die Datei **setup.exe** mit einem Doppelklick öffnen und die Installation durchführen  
  
* Als erstes wird darauf hingewiesen, das unbefugte Vervielfältigung und unbefugter Vertrieb des Programms strafbar ist. Mit «Weiter» kommt man zum nächsten Schritt.  
  
* Als nächstes kann bestimmt werden in welchem Pfad das Programm gespeichert werden soll.

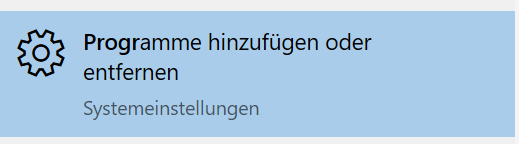
Es wird gefragt, ob es für alle Benutzer oder nur für den aktuellen Installiert werden soll.

Mit «Weiter» bestätigt man die ausgewählten Optionen.  


* Zuletzt muss man die Installation mit dem Klick auf «Weiter» bestätigt werden, dass die Installation starten kann.  
  
* Nach der Installation kann das Setup mit dem Button «Schliessen» beendet werden.  
  
* Nach Installationsabschluss kann das Programm über das Desktop Symbol oder das Windows-Menü geöffnet werden.  
   

## Deinstallationsanleitung

* Um das Programm zu deinstallieren, muss zuerst über die Windows – Taste «Programme hinzufügen oder entfernen» geöffnet werden.



* Danach wird das Programm in der Liste gesucht und mit dem Button «Deinstallieren» deinstalliert.  
  

# Fazit

Es war eine sehr spannende Projektarbeit. Ich habe gelernt Kundenvorgaben zu interpretieren und anhand dessen eine Applikation zu planen und umzusetzen. Der ganze Testablauf, die User-Stories sowie einzelne Programmfunktionen waren für mich komplett neu. Das Projekt war sehr zeitintensiv und die Zeit im ÜK allein reichte nicht aus. Durch das ergänzende Arbeiten am Projekt nach der Schule konnte ich ein für mich zufriedenstellendes Resultat liefern.