**Projekt Dokumentation**



Inhaltsverzeichnis

[Einleitung 3](#_Toc57811333)

[Zweck des Dokuments 3](#_Toc57811334)

[Bekannte Mängel im Projekt 3](#_Toc57811335)

[Verschwindende Pins 3](#_Toc57811336)

[Drucken 3](#_Toc57811337)

[Mockups 3](#_Toc57811338)

[Verbindungen suchen 4](#_Toc57811339)

[Abfahrtstafel 4](#_Toc57811340)

[User Stories 5](#_Toc57811341)

[Aktivitätsprogramm 7](#_Toc57811342)

[Testfälle 8](#_Toc57811343)

[Verbindungen 8](#_Toc57811344)

[Abfahrtstafel 8](#_Toc57811345)

[Karte 9](#_Toc57811346)

[Nächste Station 9](#_Toc57811347)

[Testprotokoll 10](#_Toc57811348)

[Verbindungen 10](#_Toc57811349)

[Abfahrtstafel 11](#_Toc57811350)

[Karte 11](#_Toc57811351)

[Nächste Station 11](#_Toc57811352)

[Installationsanleitung 12](#_Toc57811353)

# Einleitung

Der ÜK 318 behandelt das Objektorientierte Programmieren, welchen wir von Ende November bis Anfang Dezember besuchten. In diesem ÜK haben wir kennengelernt, wie ein GUI-Designed sein soll, dass es einen User anspricht und einfach verständlich ist. Wichtig ist auch der Clean Code, dass der Code leicht verständlich für andere Programmierer erweiterbar ist. Zum Schluss lernten wir noch die wichtigste Arbeit eines Informatikers kennen, das Testing. All unsere Neuerlernten Fähigkeiten durften wir zum Schluss in einem Projekt unter Beweis stellen. Das Thema war eine TravelingApp zu Programmieren mit genau vorgegebenen wüschen. Zur Starthilfe bekamen wir eine API, welche alle Daten enthielt. Der ÜK-Leiter hat uns durch das Projekt begleitet und unterstützt.

# Zweck des Dokuments

In diesem Dokument soll das Programm erläutert werden. Es werden alle Bekannten Mängel genannt, das Mockup des Programms abgebildet und beschrieben, die User Stories aufgelistet, ein Aktivitätsprogramm einer User Story abgebildet, das Testing behandelt und eine Installationsanleitung angegeben. Unsere schlussendliche Note des Projekts besteht aus 60% des Programms und 40% aus der Dokumentation.

# Bekannte Mängel im Projekt

## Verschwindende Pins

Der Eingefügte Pin auf der GMap verschindet, sobald man scrollt. Der bleit vorhanden, wenn man nur die Kamera nur hin und her schiebt mit der Maus.

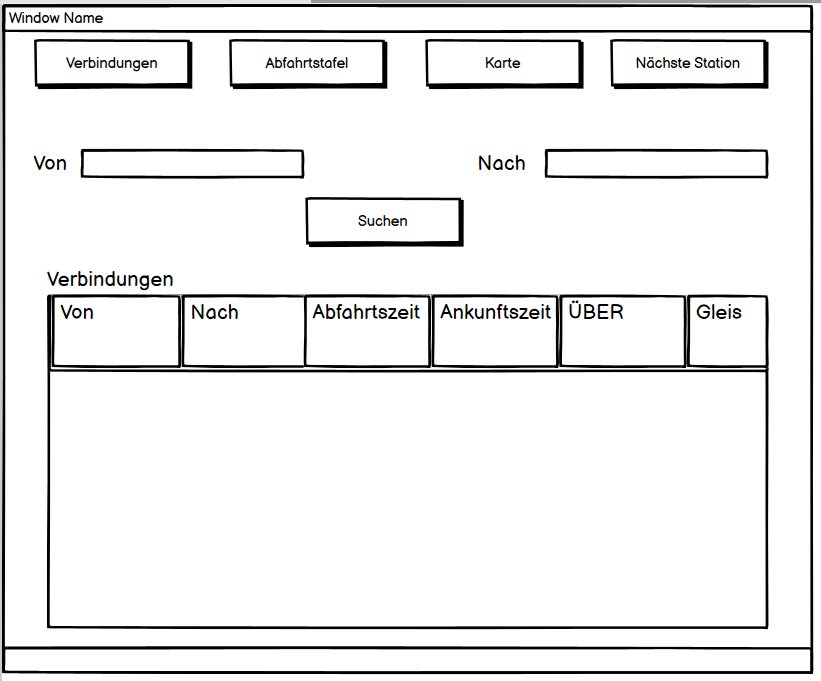
## Drucken

Die Druckfunktion erstellt eine OneNote Seite mit dem String, anstatt ein Blatt auszudrucken.

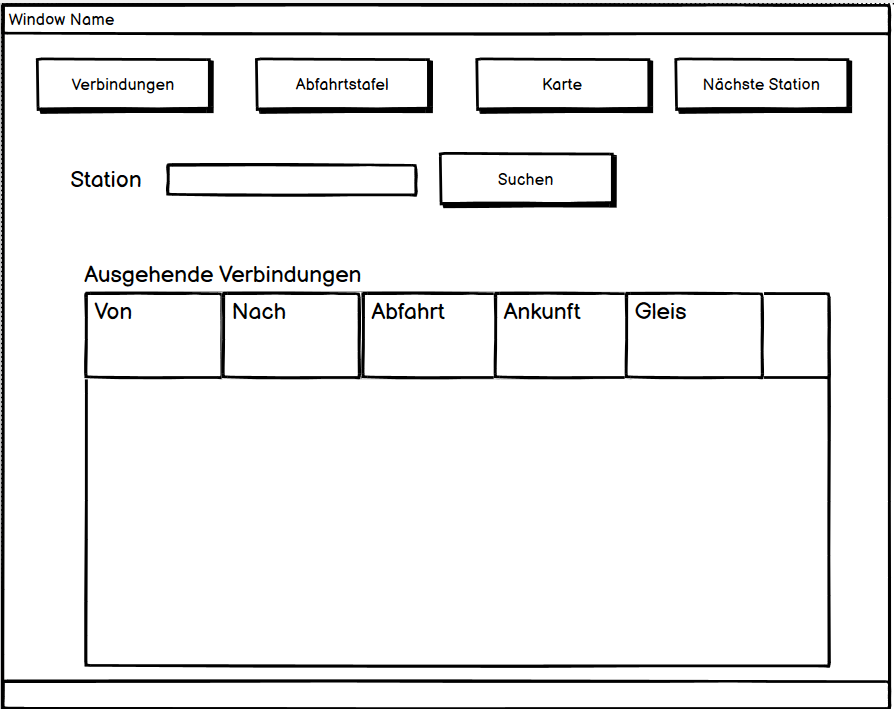
# Mockups

Mein Motto beim Programm Mockup war, weniger ist mehr. Ich wollte das Programm übersichtlich und einfach halten. Dies hat mehrere Vorteile, es ist einfacher für den User kennenzulernen und zu verstehen, aber es ist auch weniger Arbeit für den Entwickler. Unten sind die Mockups der Prio 1 User Stories gezeigt, sie haben sie während des Projekt noch verändert, da weitere Funktionen von Prio 2 und 3 Stories hinzugekommen sind. Das Mockup hat 2 grundsätzliche Dinge festgelegt. Ich wollte oben eine Menübar mit den Tabs und unten immer ein Fenster mit der Anwendung. Das Grundgerüst konnte ich beibehalten.

## Verbindungen suchen



## Abfahrtstafel

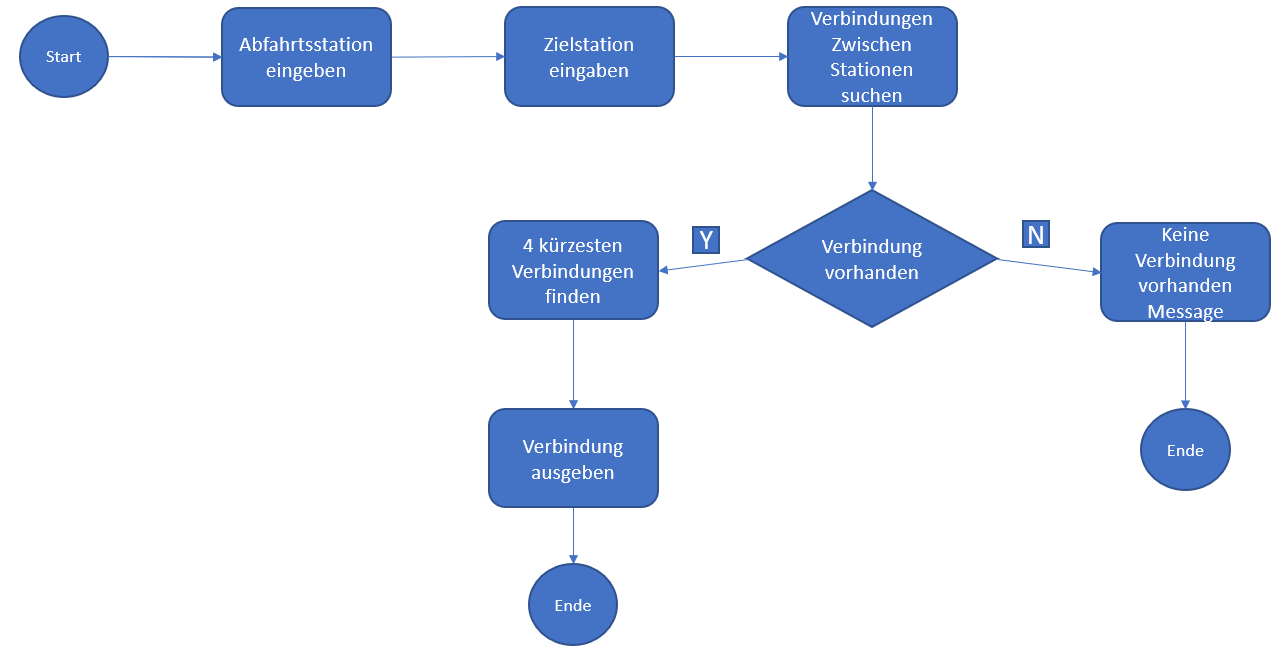


# User Stories

|  |  |
| --- | --- |
| 1. Verbindungen suchen, Priorität 1 | Als User möchte ich die Zugverbindungen zwischen zwei Stationen suchen. Somit weiss der User, welchen Zug er nehmen muss.  -2 Stationen können angegeben werden  -Zeigt die 4 nächsten Verbindungen an.  -Die Verbindungen und ihre Daten stimmen.  -Die API gibt die Verbindungen für die Stationen aus, welche am nächsten an der Usereingabe liegen. |
| 1. Abfahrtstafel, Priorität 1 | Als User möchte ich die ausgehenden Verbindungen einer Station anschauen können.  Als User gelten alle Benutzer des Programms.  -Station auswählbar.  -Es werden die ausgehenden Verbindungen angezeigt  -Falls keine vorhanden ist, wird dies erklärt. |
| 1. Suchhilfe, Priorität 2 | Als User kenne ich nur den Anfang der Stationsnahmen, das Programm soll mir Vorschläge daraus geben. Es soll helfen, wenn man einen Stationsnamen vergisst.  -Es werden passende Stationsvorschläge für die Eingabe gegeben.  -Die Vorschläge werden bei der Eingabe aktualisiert. |
| 1. Zukünftige Verbindungen, Priorität 2 | Als User möchte ich die Verbindungen nach Ankunftszeit suchen können. Somit kann ich mich im Vornherein für eine Fahrt informieren.  Als User gelten alle Benutzer des Programms.  -Zeit bei Verbindungssuche auswählbar  -Wenn es keine Fahrt gibt, wird dies angegeben.  -Die Auswählfunktion der Station gehört nicht zu dieser Story.  -Es werden die nächsten Resultate nach der angegebenen Abfahrtszeit ausgegeben. |
| 1. Stationen auf Karte, Priorität 3 | Als User möchte eine Station auf einer Karte suchen können. Somit kann ich auf einer Karte meine Route einfach planen.  Als User gelten alle Benutzer des Programms.  -Es wird eine Karte angezeigt  -Auf der Karte ist die Station markiert  -Die Karte öffnet sich auf der Station und muss nicht gesucht werden. |
| 1. Nächste Station, Priorität 3 | Als User sollen mir die nächsten Stationen von meiner momentanen Lage angezeigt werden. Es dient dazu, schnell eine nahe Station von überall in der Schweiz zu finden.  Als User gelten alle Benutzer des Programms.  -Mir werden die Nächsten 5 Stationen in meinem Umfeld ausgegeben.  -Mir wird die Distanz zu jeder Station genannt  -Das Programm erkennt automatisch meinen momentanen Standpunkt. |
| 1. Verbindung per senden, Priorität 3 | Als User möchte ich meine Arbeitskollegen/ Mitarbeitern eine Verbindung per E-Mail senden können.  Als User gelten alle Benutzer des Programms.  -Das Auswählen dieser Verbindung gehört zu einer anderen Story.  -Eine Ausgewählte Verbindung ist per E-Mail versendbar.  -Man kann die Empfänger-E-Mail selbst angeben.  -Die Daten der Verbindung werden automatisch in die E-Mail übertragen. |
| 1. Verbindungen drucken, Vorschlag | Als User möchte ich gesuchte Verbindungen einfach Ausdrucken können, somit habe ich eine Archivierungsmöglichkeit der Daten.  -Als User gelten alle Benutzer des Programms.  -Mit Doppelklick auf eine Verbindung wird diese an den Drucker gesendet.  -Es werden alle Informationen der Verbindung ausgedruckt. |

# Aktivitätsprogramm

Das untenstehende Aktivitätsprogamm beschreibt den Ablauf der User Story Verbindungen suchen. Vom User werden die Abfahrt- und Zielstation eingegeben und er erwarten danach die kürzesten Verbindungen zwischen diesen Orten zu erhalten.



# Testfälle

## Verbindungen

|  |  |  |  |
| --- | --- | --- | --- |
| **Schritt** | **Aktivität** | **Erw. Resultat** | **Erfüllt** |
| 1 | Doppelklick auf MyTransportApp.exe | Das Programm MyTransportApp öffnets sich und zeigt die Verbindungen-Seite. |  |
| 2 | Im Von-Textfeld «Lu» eingeben und Combobox öffnen. | In der Combobox werden Vorschläge von Stationen für «Lu» geben. |  |
| 3 | Im Von-Textfeld «Luzern» eingeben. | Im Von-Textfeld wird flüssig «Luzern» eingeben. |  |
| 4 | Im Nach-Textfeld «Ber» eingeben und Combobox öffnen. | In der Combobox werden Stationsvorschläge für «Ber» geben. |  |
| 5 | Im Nach-Textfeld «Bern» eingeben. | Es wird flüssig im Nach-Textfeld «Bern» eingeben. |  |
| 6 | Im Abfahrt-Datum-Feld 02.12.2020 eingeben. | Es öffnet sich eine Kalendersuche, bei welcher das gewünschte Datum eingeben werden kann. |  |
| 7 | Im Abfahrt-Zeit-Feld 12.00 eingeben. | Das Abfahrt-Zeit-Feld gibt nur die angegebene Zeit an. |  |
| 8 | Button Suchen betätigen. | Es werden 4 Resultate in der DataGridView von Verbindungen von Luzern nach Bern ausgegeben. Jedes Resultat hat eine Abfahrts- und Ankunftszeit, wie eine Gleisangabe. |  |
| 9 | Die oberste Datagridreihe auswählen unter dem Head. | Die ganze Datagridreihe wird blau markiert. |  |
| 10 | Den Button senden betätigen. | Es öffnet sich ein E-Mail-Tool und schreibt in eine E-Mail die Angaben der ausgewählten DatagridReihe. |  |
| 11 | Doppelklick auf die oberste DatagridReihe. | Es öffnet sich eine Printvorlage mit den Angaben der doppelgeklickten Reihe. |  |

## Abfahrtstafel

|  |  |  |  |
| --- | --- | --- | --- |
| **Schritt** | **Aktivität** | **Erw. Resultat** | **Erfüllt** |
| 1 | Doppelklick auf MyTransportApp.exe. | Das Programm MyTransportApp öffnets sich und zeigt die Verbindungen-Seite. |  |
| 2 | Den Button Abfahrtstafel betätigen. | Das Programm öffnet die Abfahrtstafel form. |  |
| 3 | Ins Textfeld «Luzern» eingeben. | Es wird flüssig in das Textfeld «Luzern» eingegeben. |  |
| 4 | Den Button Suchen betätigen. | Es erscheinen im Datagridview alle ausgehenden Verbindungen mit Nach, Nr und Name. |  |

## Karte

|  |  |  |  |
| --- | --- | --- | --- |
| **Schritt** | **Aktivität** | **Erw. Resultat** | **Erfüllt** |
| 1 | Doppelklick auf MyTransportApp.exe. | Das Programm MyTransportApp öffnets sich und zeigt die Verbindungen-Seite. |  |
| 2 | Den Button Karte betätigen. | Das Programm öffnet das Karten-Form. |  |
| 3 | «Luzern» in die Textbox eingeben. | Es wird flüssig «Luzern» in die Textbox eingegeben. |  |
| 4 | Den Button «Suche Station» betätigen | Im GMap-Fenster erscheint eine Karte, welche Luzern zeigt und dies mit einem grünen Pin markiert. |  |

## Nächste Station

|  |  |  |  |
| --- | --- | --- | --- |
| **Schritt** | **Aktivität** | **Erw. Resultat** | **Erfüllt** |
| 1 | Doppelklick auf MyTransportApp.exe. | Das Programm MyTransportApp öffnets sich und zeigt die Verbindungen-Seite. |  |
| 2 | Den Button Nächstestation betätigen. | Das Programm öffnet das Karten-Form. Ev. Kommt eine Fehlermeldung, welche gewollt ist. |  |
| 3 | Fehlermeldung bestätigen und bis 15 warten. | Im Datagridview erscheinen 5 Stationen mit Namen und der Distanz. |  |

# Testprotokoll

## Verbindungen

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Schritt** | **Aktivität** | **Erw. Resultat** | **Abw. Result** | **Erfüllt** |
| 1 | Doppelklick auf MyTransportApp.exe | Das Programm MyTransportApp öffnets sich und zeigt die Verbindungen-Seite. |  | ✅ |
| 2 | Im Von-Textfeld «Lu» eingeben und Combobox öffnen. | In der Combobox werden Vorschläge von Stationen für «Lu» geben. |  | ✅ |
| 3 | Im Von-Textfeld «Luzern» eingeben. | Im Von-Textfeld wird flüssig «Luzern» eingeben. |  | ✅ |
| 4 | Im Nach-Textfeld «Ber» eingeben und Combobox öffnen. | In der Combobox werden Stationsvorschläge für «Ber» geben. |  | ✅ |
| 5 | Im Nach-Textfeld «Bern» eingeben. | Es wird flüssig im Nach-Textfeld «Bern» eingeben. |  | ✅ |
| 6 | Im Abfahrt-Datum-Feld 02.12.2020 eingeben. | Es öffnet sich eine Kalendersuche, bei welcher das gewünschte Datum eingeben werden kann. |  | ✅ |
| 7 | Im Abfahrt-Zeit-Feld 12.00 eingeben. | Das Abfahrt-Zeit-Feld gibt nur die angegebene Zeit an. |  | ✅ |
| 8 | Button Suchen betätigen. | Es werden 4 Resultate in der DataGridView von Verbindungen von Luzern nach Bern ausgegeben. Jedes Resultat hat eine Abfahrts- und Ankunftszeit, wie eine Gleisangabe. |  | ✅ |
| 9 | Die oberste Datagridreihe unter dem Head auswählen. | Die ganze Datagridreihe wird blau markiert. |  | ✅ |
| 10 | Den Button senden betätigen. | Es öffnet sich ein E-Mail-Tool und schreibt in eine E-Mail die Angaben der ausgewählten DatagridReihe. |  | ✅ |
| 11 | Doppelklick auf die oberste DatagridReihe. | Es öffnet sich eine Printvorlage mit den Angaben der doppelgeklickten Reihe. | Beim Computer kommt die Meldung, 1 Seite wird gedruckt. Es wird jedoch nichts gedruckt. Jedoch wird eine neue OneNoteSeite mit dem String generiert. | 🚫 |

## Abfahrtstafel

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Schritt** | **Aktivität** | **Erw. Resultat** | **Abw. Resultat** | **Erfüllt** |
| 1 | Doppelklick auf MyTransportApp.exe. | Das Programm MyTransportApp öffnets sich und zeigt die Verbindungen-Seite. |  | ✅ |
| 2 | Den Button Nächste Station betätigen. | Das Programm öffnet das Karten-Form. Ev. Kommt eine Fehlermeldung, welche gewollt ist. |  | ✅ |
| 3 | Fehlermeldung bestätigen und bis 15 warten. | Im Datagridview erscheinen 5 Stationen mit Namen und der Distanz. |  | ✅ |

## Karte

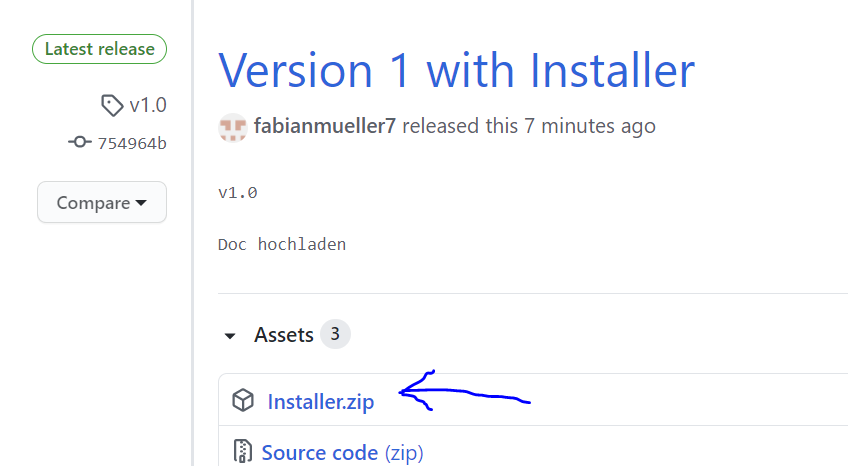
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Schritt** | **Aktivität** | **Erw. Resultat** | **Abw. Resultat** | **Erfüllt** |
| 1 | Doppelklick auf MyTransportApp.exe. | Das Programm MyTransportApp öffnets sich und zeigt die Verbindungen-Seite. |  | ✅ |
| 2 | Den Button Karte betätigen. | Das Programm öffnet das Karten-Form. |  | ✅ |
| 3 | «Luzern» in die Textbox eingeben. | Es wird flüssig «Luzern» in die Textbox eingegeben. |  | ✅ |
| 4 | Den Button «Suche Station» betätigen | Im GMap-Fenster erscheint eine Karte, welche Luzern zeigt und dies mit einem grünen Pin markiert. |  | ✅ |

## Nächste Station

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Schritt** | **Aktivität** | **Erw. Resultat** | **Abw. Resultat** | **Erfüllt** |
| 1 | Doppelklick auf MyTransportApp.exe. | Das Programm MyTransportApp öffnets sich und zeigt die Verbindungen-Seite. |  | ✅ |
| 2 | Den Button Nächste Station betätigen. | Das Programm öffnet das Karten-Form. Ev. Kommt eine Fehlermeldung, welche gewollt ist. |  | ✅ |
| 3 | Fehlermeldung bestätigen und bis 15 warten. | Im Datagridview erscheinen 5 Stationen mit Namen und der Distanz. |  | ✅ |

# Installationsanleitung

-Downloaden Sie das Installer.zip von der [Github-Seite](https://github.com/fabianmueller7/modul-318-student/releases/tag/v1.0).



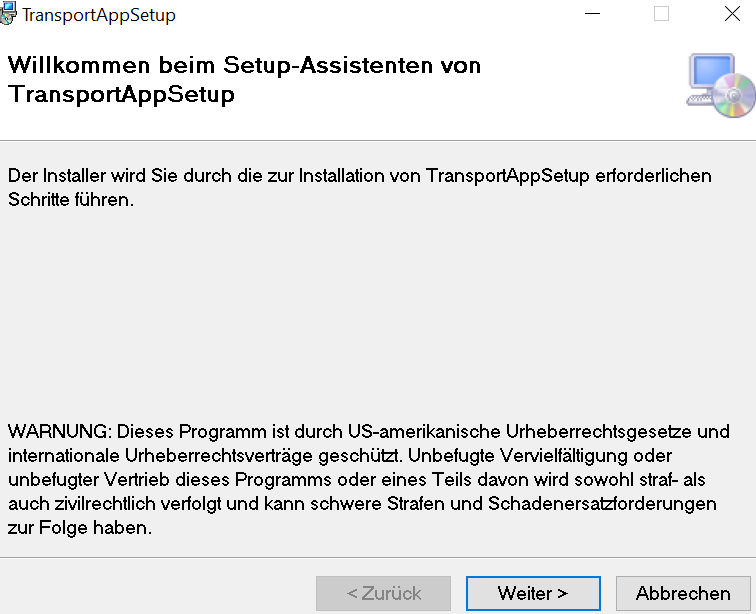
-Entpacken Sie das Zipfile.

-Führen Sie das TransportAppSetup.msi aus.

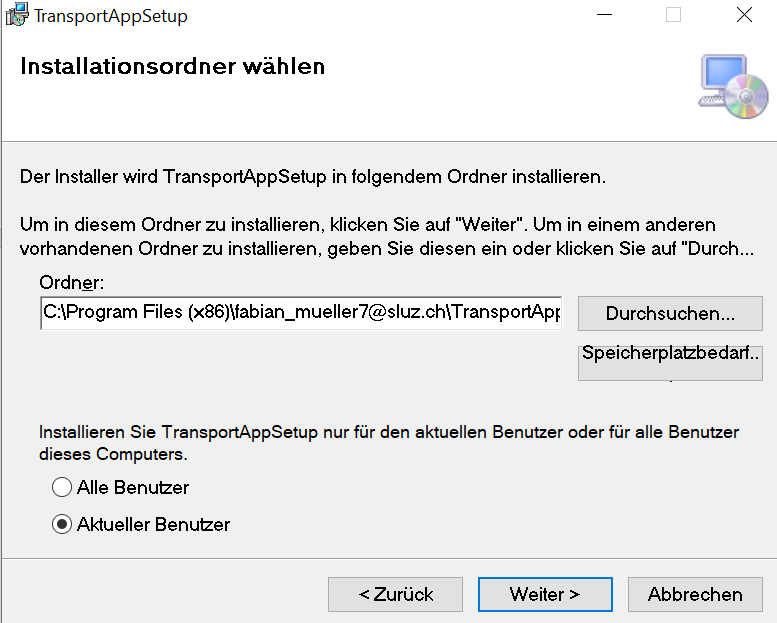
-Erlauben Sie der Exe die Berechtigung mit Weiter Informationen.



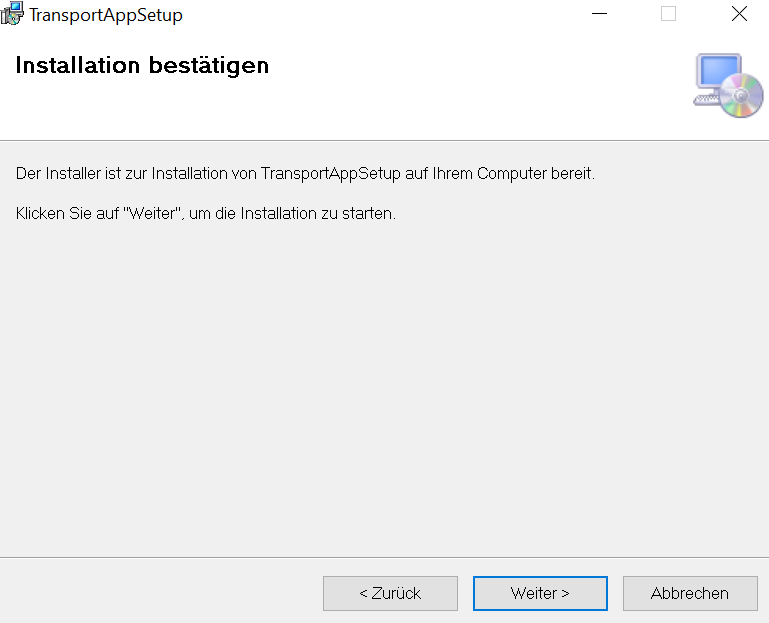
-Weiter



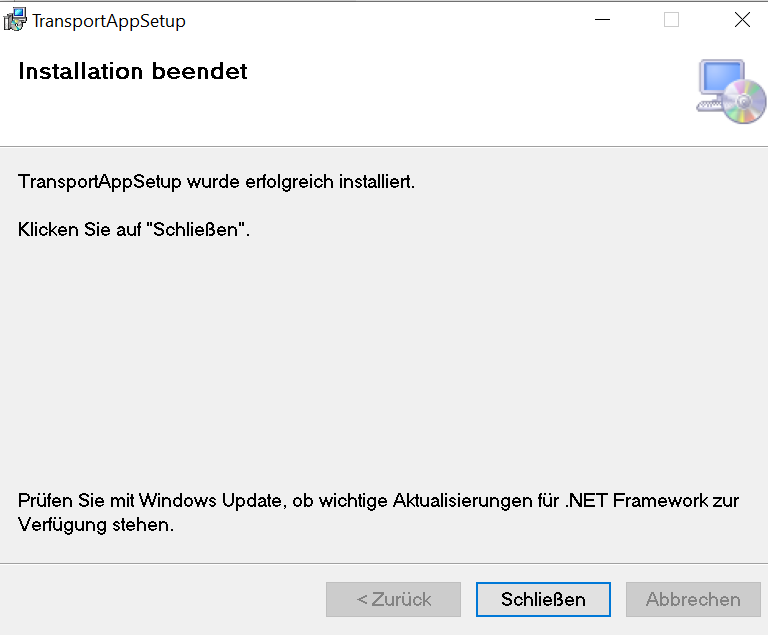
-Wählen Sie den Speicherort des Programms mit durchsuchen und für welche Benutzer es installiert werden soll.



-Bestätigen Sie die Installation mit Weiter.



-Installation abgeschlossen.



-Schalten sie die Positionserkennung für Apps ein in den Einstellungen unter Position.



-Führen Sie die Verknüpfung auf dem Desktop aus.

