

Fahrplan

Dominik Bielser

ÜK M318

21.12.2018



Inhaltsverzeichnis

Einleitung.....	3
Zweck dieser Dokumentation.....	3
Planung.....	3-4
Richtlinien vom Programmieren.....	5
Umsetzung.....	5-6
Anwendung im Betrieb.....	7-9
Bekannte Bugs.....	10
Testfälle.....	10-12
Installationsanleitung.....	12-14

1. Einleitung

Wir haben den Auftrag bekommen, einen Fahrplan zu entwickeln und eine Dokumentation dazuzuschreiben.

Es wurden uns viele verschiedene Präsentationen gezeigt, die uns zeigten wie wir an einem Projekt korrekt vorgehen sollten. Wir mussten auch am Anfang viele verschiedene Übungsaufgaben lösen, die uns unseren Start am Projekt vereinfachen sollten.

2. Zweck dieser Dokumentation

Dieses Dokument dient dazu mein Vorgehen und einzelnen Arbeitsschritte aufzuzeigen. Es soll auch zeigen wie meine Applikation aufgebaut ist.

3. Planung

Bevor wir mit dem Projekt beginnen konnten, mussten wir zuerst ein GUI-Design auf balsamiq.cloud erstellen. Das sich an unseren finalen Version der Fahrplan Anwendung orientieren soll.

Damit wir Wissen wie unser GUI vom Programm dann aussehen soll.

Erstelltes GUI auf balsamiq.cloud

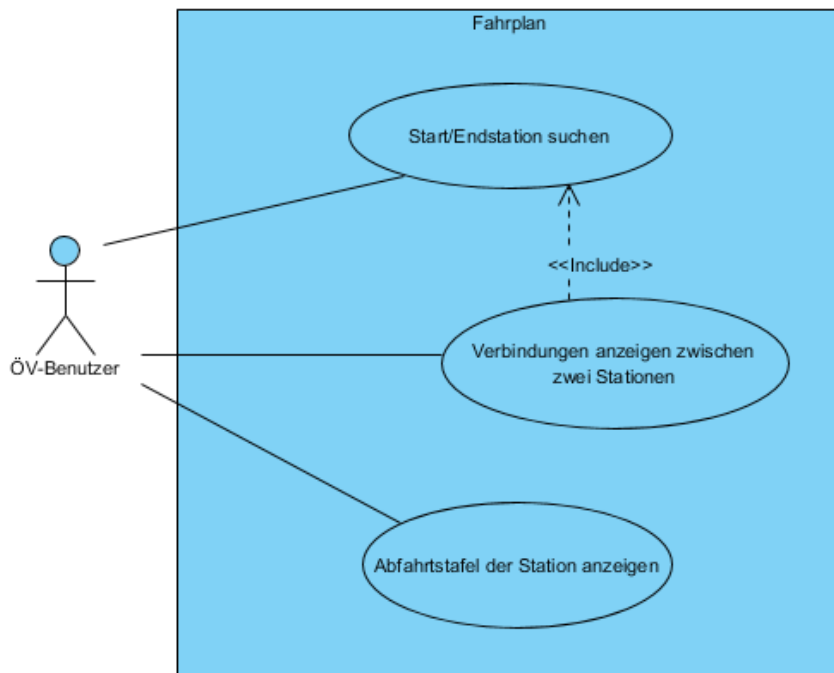
The wireframe shows a user interface for a train schedule application. It includes input fields for departure location, destination, time, and date. There are buttons for 'Verbindung Suchen' (Search Connection), 'Abfahrtsstafel' (Departure Board), 'E-Mail versenden' (Send Email), and 'Google Maps'. A table at the bottom displays the schedule with columns for Date, Departure Location, Departure, Destination, Arrival, and Platform.

Datum	Abfahrtsort	Abfahrt	Zielort	Ankunft	Gleis

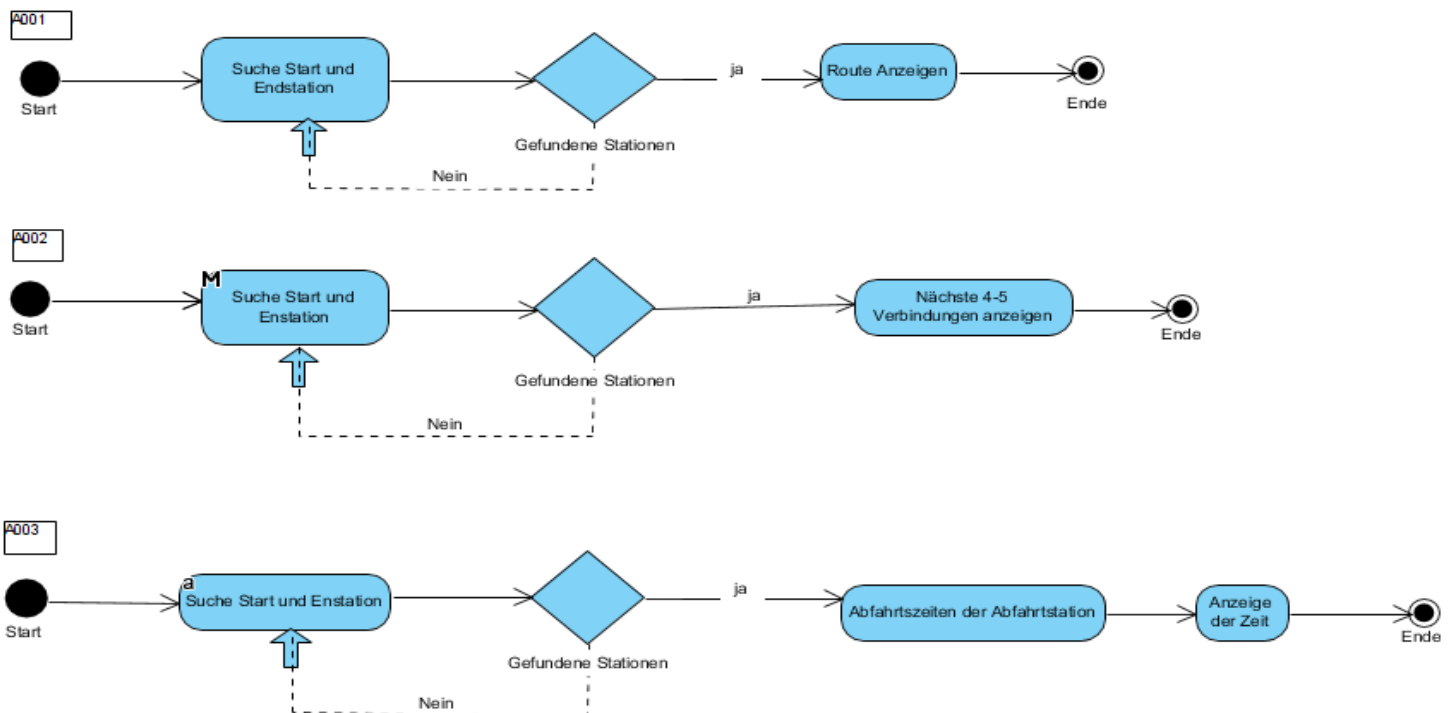
Zu unserem Programm haben wir noch ein Use-Case-Diagramm und ein Aktivitätsdiagramm erstellt.

Folgende sieht man in den Bilder unten.

USE-CASE-DIAGRAMM



AKTIVITÄTSDIAGRAMM



Richtlinien vom Programmieren

Die Richtlinien zeigen wie ich meinen Code programmiert habe zur Anwendung.

Bei den Buttons, Label, und Textboxen benannte ich sie immer so. Kürzel am Anfang mit 3. Buchstaben (kleinbuchstaben) danach passender Name.

Beispiele:

Buttons: btnSuchen, btnMaps, btnTafel ...

Textboxen: txtAbfahrtort, txt Zielort, txtMail

Label: lbsAbfahrt, lbsZielort ...

Beim GUI wollte ich auch noch das alles auf einem Panel platz hat, damit alles miteinander in der Nähe ist.

Die Methoden sehen immer so aus Get_Grid oder Get_Stations

Umsetzung

Was wurde in meinem Projekt erfolgreich umgesetzt? Die Anforderungen A001, A002, A003, A004, A005, A006 und A008 habe ich erfolgreich umgesetzt. Die Anforderung A007 wusste ich nicht genau wie ich sie umsetzen soll, daher habe ich diese nicht erfüllen können.

Beim Suchen nach einer Verbindung vom (Abfahrtsort/Zielort) wird ein Automatischer Vorschlag angezeigt in der Listbox, dann kann man die gewählte Station aus der Liste aussuchen.

Das ändern des Datum und der Uhrzeit klappt auch, wenn man möchte kann man sich noch eine Abfahrtsort anzeigen lassen für den Abfahrtsort der gewählt wurde.

Das Anzeigen des gewählten Abfahrtsort auf GoogleMaps funktioniert auch, so kann man genaueres erfahren über den Abfahrtsort.

Wenn man eine E-Mail des gewählten Abfahrtsplan möchte, kann man sich eine E-Mail zusenden lassen.

ID	Beschreibung	Priorität
A001	Als ÖV-Benutzer möchte ich Start- und Endstation mittels Textsuche suchen können, damit ich nicht alle Stationsnamen auswendig lernen muss.	Ja
A002	Als ÖV-Benutzer möchte ich die aktuellen, d.h. mindestens die nächsten vier bis fünf Verbindungen zwischen den beiden gefundenen und ausgewählten Stationen sehen, damit ich weiss wann ich zur Station muss, um den für mich idealen Anschluss zu erwischen.	Ja
A003	Als ÖV-Benutzer möchte ich sehen, welche Verbindungen ab einer bestimmten Station vorhanden sind, damit ich bei mir zuhause eine Art Abfahrtstafel haben kann.	Ja
A004	Als ÖV-Benutzer möchte ich, dass schon während meiner Eingabe erste Such-Resultate erscheinen, damit ich effizienter nach Stationen suchen kann.	Ja
A005	Als ÖV-Benutzer möchte ich nicht nur aktuelle Verbindungen suchen können, sondern auch solche zu einem beliebigen anderen Zeitpunkt, damit ich zukünftige Reisen planen kann.	Ja
A006	Als ÖV-Benutzer möchte ich sehen, wo sich eine Station befindet, damit ich mir besser vorstellen kann, wie die Situation vor Ort aussieht.	Ja
A007	Als ÖV-Benutzer möchte Stationen finden, die sich ganz in der Nähe meiner aktuellen Position befinden, damit ich schnell einen Anschluss erreichen kann.	Nein
A008	Ich möchte meine gefundenen Resultate via Mail weiterleiten können, damit auch andere von meinen Recherchen profitieren können.	Ja

Anwendung im Betrieb

Hier seht ihr ein paar Bilder meiner Anwendung wie Sie in Betrieb aussieht.

This screenshot shows the application interface with all input fields empty. The layout includes:

- Abfahrtsort:** A text input field.
- Zielort:** A text input field.
- Uhrzeit:** A time selection dropdown menu showing 08:37:39.
- Datum:** A date selection dropdown menu showing Freitag, 21. Dezember 2018.
- E-Mail Adresse:** A text input field.
- Buttons:** "Verbindung Suchen", "Abfahrtstafel", "E-Mail versenden", and "Google Maps".
- Map Area:** A large gray rectangular placeholder for a map.

This screenshot shows the application interface with pre-filled data. The layout includes:

- Abfahrtsort:** A text input field containing "Basel" with a dropdown menu showing "Basel SBB", "Basel, Schifflande", and "Basel Bad Bf".
- Zielort:** A text input field containing "Genf" with a dropdown menu showing "Genève", "Genève, Cornavin", and "Genève, Bel-Air".
- Uhrzeit:** A time selection dropdown menu showing 08:37:39.
- Datum:** A date selection dropdown menu showing Freitag, 21. Dezember 2018.
- E-Mail Adresse:** A text input field.
- Buttons:** "Verbindung Suchen", "Abfahrtstafel", "E-Mail versenden", and "Google Maps".
- Map Area:** A large gray rectangular placeholder for a map.

Abfahrtsort: Zielort:

Uhrzeit:

Datum: E-Mail Adresse:

	Datum	Abfahrtsort	Abfahrt	Zielort	Ankunft	Gleis
►	21.12.2018	Basel SBB	08:31	Genève-Aér...	11:27	5
	21.12.2018	Basel SBB	09:04	Genève-Aér...	11:54	8
	21.12.2018	Basel SBB	09:31	Genève-Aér...	12:27	5
	21.12.2018	Basel SBB	10:04	Genève-Aér...	12:54	8
	21.12.2018	Basel SBB	10:31	Genève-Aér...	13:27	5
	21.12.2018	Basel SBB	11:04	Genève-Aér...	13:56	8
*						

Abfahrtsort: Zielort:

Uhrzeit:

Datum: E-Mail Adresse:

	Zeit	Nach	Linie
►	08:50	Laufenburg	S 1
	08:51	Saint-Louis (Haut-Rhin)	TE2 TE2 96216
	08:59	Interlaken Ost	IC 61
	09:01	Olten	S 3
	09:03	Biel/Bienne	IC 51
	09:04	Lugano	IC 21
	09:04	Basel Bad Bf	S 6
	09:06	Basel Bad Bf	ICE ICE 76
	09:06	Laufen	S 3
	09:07	Chur	ICE ICE 271
	09:09	Saint-Louis (Haut-Rhin)	TER TER 96198

Abfahrtsort: Zielort:

Uhrzeit:

Datum: E-Mail Adresse:

Abfahrtsort: Zielort:

Uhrzeit:

Datum: E-Mail Adresse:

Zeit	Linie
08:50	S 1
08:51	Rhin) TE2 TE2 96216
08:59	IC 61
09:01	S 3
09:03	Biel/Bienne IC 51
09:04	Lugano IC 21
09:04	Basel Bad Bf S 6
09:06	Basel Bad Bf ICE ICE 76
09:06	Laufen S 3
09:07	Chur ICE ICE 271
09:09	Saint-Louis (Haut-Rhin) TER TER 96198

Senden der E-Mail war erfolgreich!

Bekannte Bugs

Bei Google-Maps kommt ein Problem mit der Kompatibilitätsansicht im Internet-Explorer.

Ein möglicher Lösungsvorschlag wäre, die Kompatibilitätsansicht deaktivieren, im Internet Explorer, was ich auch versucht habe.

Dies hat Jedoch den Fehler nicht behoben.

Testfall «E-Mail versenden der Abfahrtstafel inkl. Google Maps anzeigt»

Schritt	Aktivität	Erwartetes Resultat
1.	Anwender will eine Zugverbindung suchen	Panel der Anwendung startet
2.	Gibt den Abfahrtsort Basel ein, und wählt aus der Liste Basel SBB, das gleiche macht er beim Zielort für Genf Aéroport. Anschliessend klickt er auf den Button Verbindung suchen.	Es werden die nächsten 6. Verbindungen von Basel SBB – Genève Aéroport angezeigt, im Anzeige-Fenster.
3.	Anwender möchte genaueres über den Standort Basel SBB erfahren, klickt auf Google Maps.	Der Standort Basel SBB, wird im Anzeige-Fenster auf Google Maps geöffnet.
4.	Anwender möchte die Abfahrtstafel von Basel SBB wissen, klickt auf Abfahrtstafel.	Die Abfahrtstafel für Basel SBB wird im Anzeige-Fenster angezeigt
5.	Er möchte die Abfahrtstafel als E-Mail erhalten, füllt seine E-Mail-Adresse in der Textbox E-Mail Adresse ein. Und klickt auf E-Mail versenden	Die Messagebox, mit der Nachricht. «Senden der E-Mail war erfolgreich» Erscheint.

Testfall «Suchvorschlag automatisch Anzeigen, inkl Verbindung suchen»

Schritt	Aktivität	Erwartetes Resultat
1.	Anwender will eine Zugverbindung suchen	Panel der Anwendung erscheint
2.	Anwender gibt folgende Daten ein: Abfahrtsort: Bas	Vorschläge zum Aussuchen des Abfahrtort erscheinen.
3.	Anwender klickt auf Basel SBB	Textbox bekommt den Inhalt Basel SBB
4.	Anwender gibt folgende Daten ein. Zielort: St.Gallen Datum: 21. Dezember 2018 Zeit: 9:26:00 Anwender klickt auf Verbidnug Suchen	Beim Anzeige-Fenster werden die nächsten 6. Verbindungen angezeigt

Testfall «Anzeigen der Abfahrtstafel»

Schritt	Aktivität	Erwartetes Resultat
1	Anwender gibt foglende Daten ein: Abfahrtsort: Zürich HB Anwender klickt auf Abfahrtstafel	Im Anzeige-Fenster wird die aktuelle Abfahrtstafel für die Station Zürich HB angezeigt.

Unit-Test

Beim Unit-Test musste ich noch im TransportTest.cs noch das korrekte Datum und die Uhrzeit hinzufügen im richtigen Format.

Der Test wurde bei mir erfolgreich durchgeführt.

Dies sieht so aus.

```
[TestMethod]
    public void Connections()
    {
        testee = new Transport();
        var connections = testee.GetConnections("Sursee", "Luzern", "2018-02-23",
"13:25");

        Assert.IsNotNull(connections);
    }
}
```

Installationsanleitung

1.

Downloaden Sie sich den Ordner <https://github.com/DominikBielser/modul-318-student> und öffnen Sie ihn.

2.

Machen sie einen Doppelklick auf den Ordner Applikation, dann sollte er sich öffnen

3.

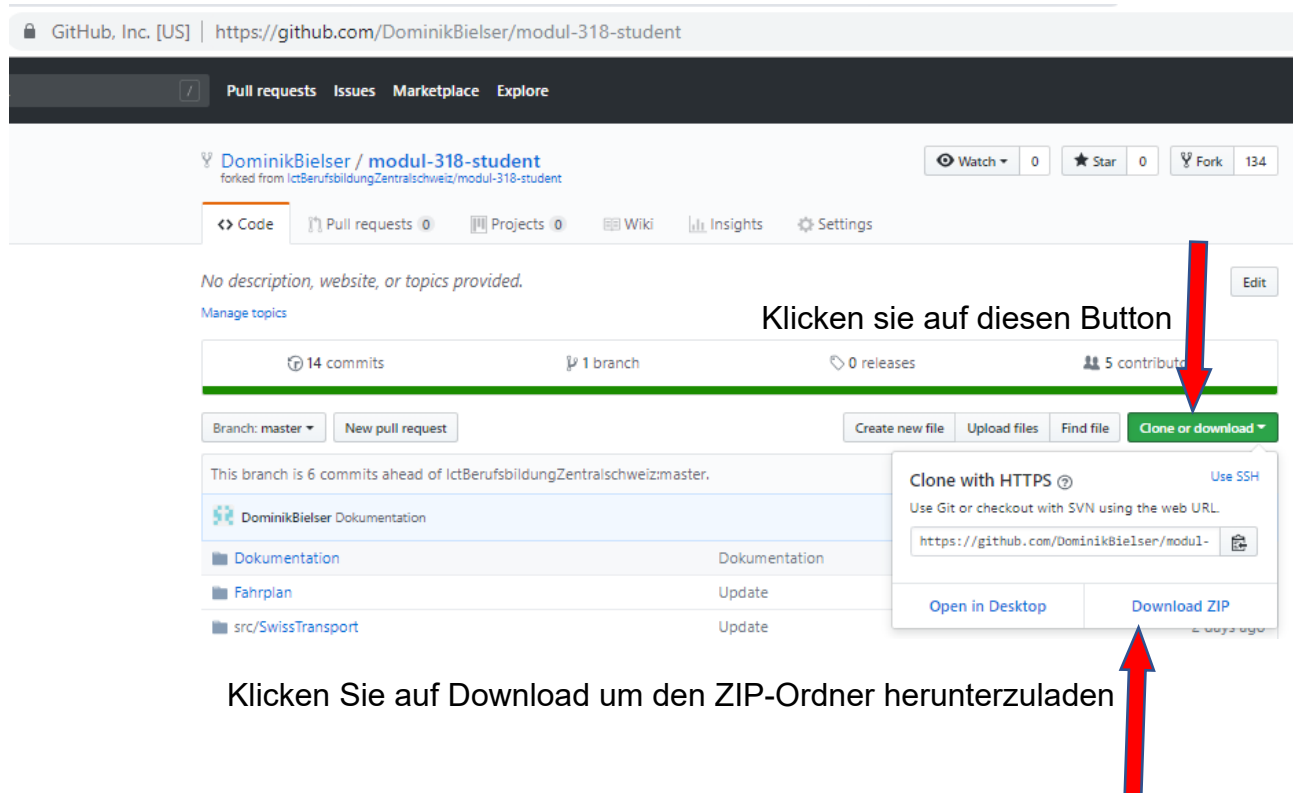
Wenn Sie ihm Ordner Applikation sind, machen sie einen Doppelklick auf die Anwendung Fahrplan, dann sollte sich die Applikation öffnen.

4.

Sie können Ihre Anwendung deinstallieren in dem Sie den Ordner Applikation deinstallieren.

Installationsanleitung in Bilder

1.



Klicken sie auf diesen Button

Klicken Sie auf Download um den ZIP-Ordner herunterzuladen

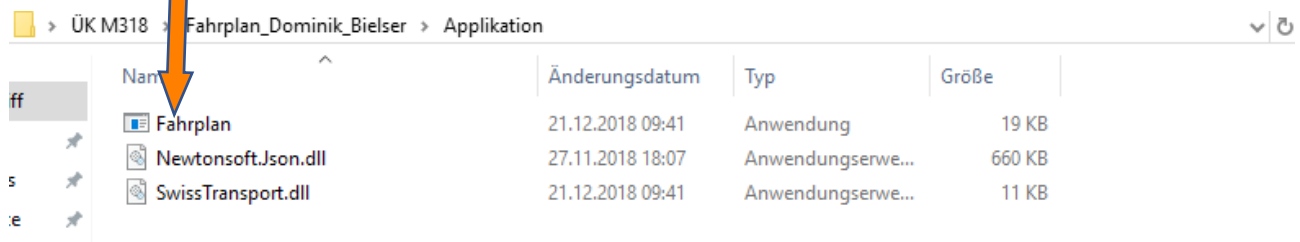
2.



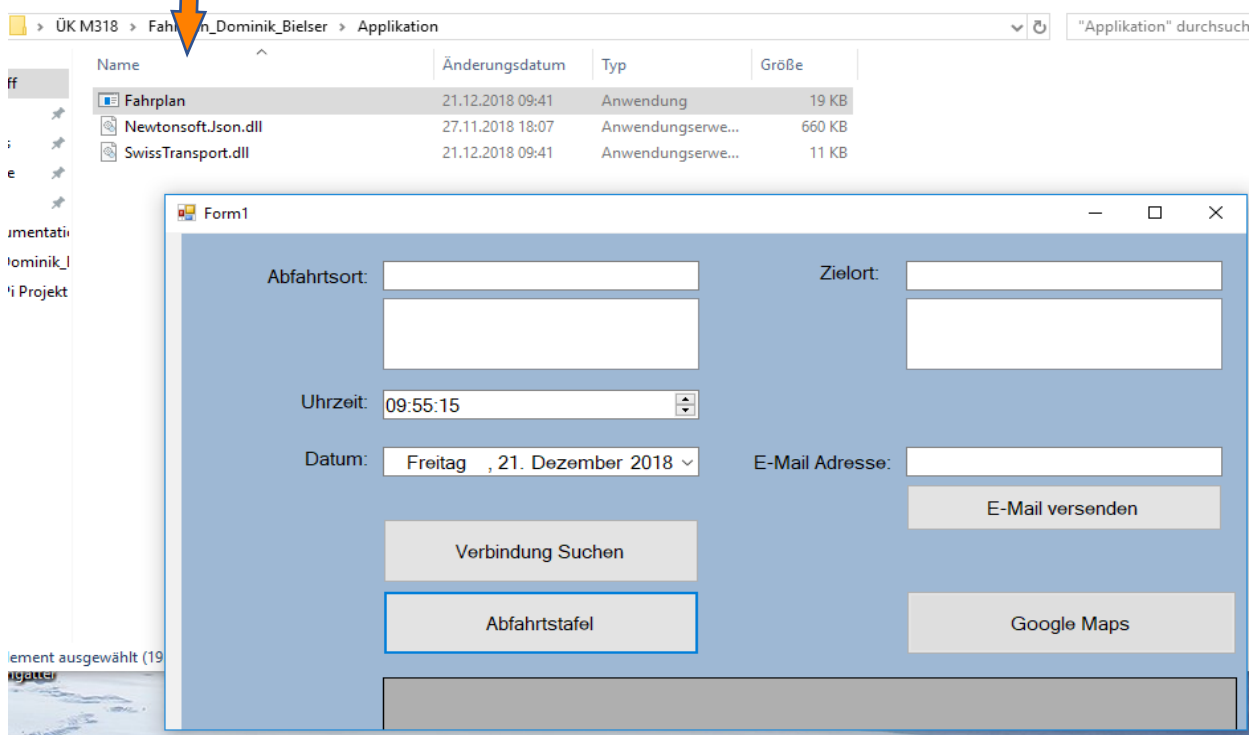
Doppelklick (linke Maustaste zweimal drücken) auf den Ordner Applikation

Name	Änderungsdatum	Typ	Größe
Applikation	21.12.2018 09:47	Dateiordner	
Dokumentation	20.12.2018 16:19	Dateiordner	
Fahrplan	21.12.2018 09:41	Dateiordner	
packages	19.12.2018 10:48	Dateiordner	
src	19.12.2018 10:39	Dateiordner	
tests	19.12.2018 10:39	Dateiordner	
modul-318-student.sln	19.12.2018 10:58	Visual Studio Solu...	3 KB
README.md	19.12.2018 10:39	MD-Datei	1 KB

3. Doppelklick (linke Maustaste zweimal drücken) auf Fahrplan



4. Das Programm öffnet sich nach dem Doppelklick auf den Fahrplan, so sieht es dann aus.



5. So sieht der Papierkorb in Windows aus

