

Projektarbeit TransportApp

Inhalt

Einleitung.....	2
Aufgabenstellung.....	2
Anforderungen	3
Erfüllt	3
Known-Issues.....	3
Mockup.....	4
Use Cases.....	8
Testing	10
Installation.....	12
Programmierrichtlinien	12
Naming Conventions	12
Declaration	12
Comments	12
Statements	12
Projektinfo	12

Einleitung

Der zweck dieses Dokument ist die Aufgabenstellung, gewisse Schritte und die Installation des Projektes also die TransportApp festzuhalten. Genauer noch, was musste ich machen, wie ich angefangen haben(nicht zu genau festgehalten), meine Planung, Probleme, die noch vorhanden sind und zu guter Letzt das Endprodukt nach einer Woche.

Aufgabenstellung

Die Aufgabe für dieses Projekt war eine Applikation zu machen, welche alle Anforderungen erfüllt die unter dem Punkt "Anforderungen" zu finden sind. Dafür haben wir eine Solution bekommen, welche schon alle nötigen Funktionen von der TransportApi implementiert hat.

Anforderungen

Priorities: 1 = must / 2 = should / 3 = nice to have

ID	Beschreibung	Priorität	Gemacht
A001	Als ÖV-Benutzer möchte ich Start- und Endstation mittels Textsuche suchen können, damit ich nicht alle Stationsnamen auswendig lernen muss.	1	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.	1	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.	1	Ja
A004	Als ÖV-Benutzer möchte ich, dass schon während meiner Eingabe erste Suchresultate erscheinen, damit ich effizienter nach Stationen suchen kann.	2	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	2	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.	3	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.	3	Nein
A008	Ich möchte meine gefundenen Resultate via Mail weiterleiten können, damit auch andere von meinen Recherchen profitieren können.	3	Nein

Erfüllt

Erfüllt wurden die Anforderungen A001-A006. A007 und A008 konnte ich leider nicht erfüllen, da ich keine Zeit mehr hatte. Die A007 musste ich leider im Code Behind umsetzen, da das Custom Control leider mit MVVM nicht funktioniert hat.

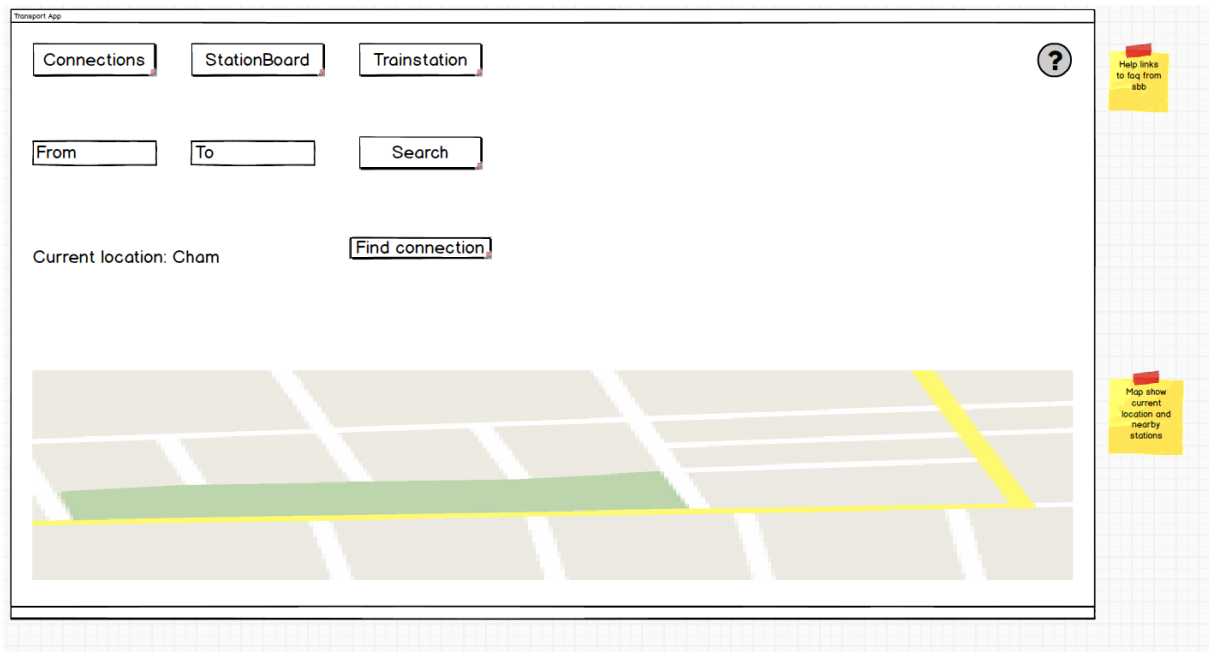
Known-Issues

Falls man viele Zeichen in ein Stationsfeld, welche nicht gefunden werden können, gibt es Performance Probleme.

Falls ein Station vorhanden ist, gibt die Suche immer noch etwas zurück obwohl eine Station nicht existiert. Die Station, die nicht existiert wird, einfach von der Api mit einem Standard-Ort ersetzt.

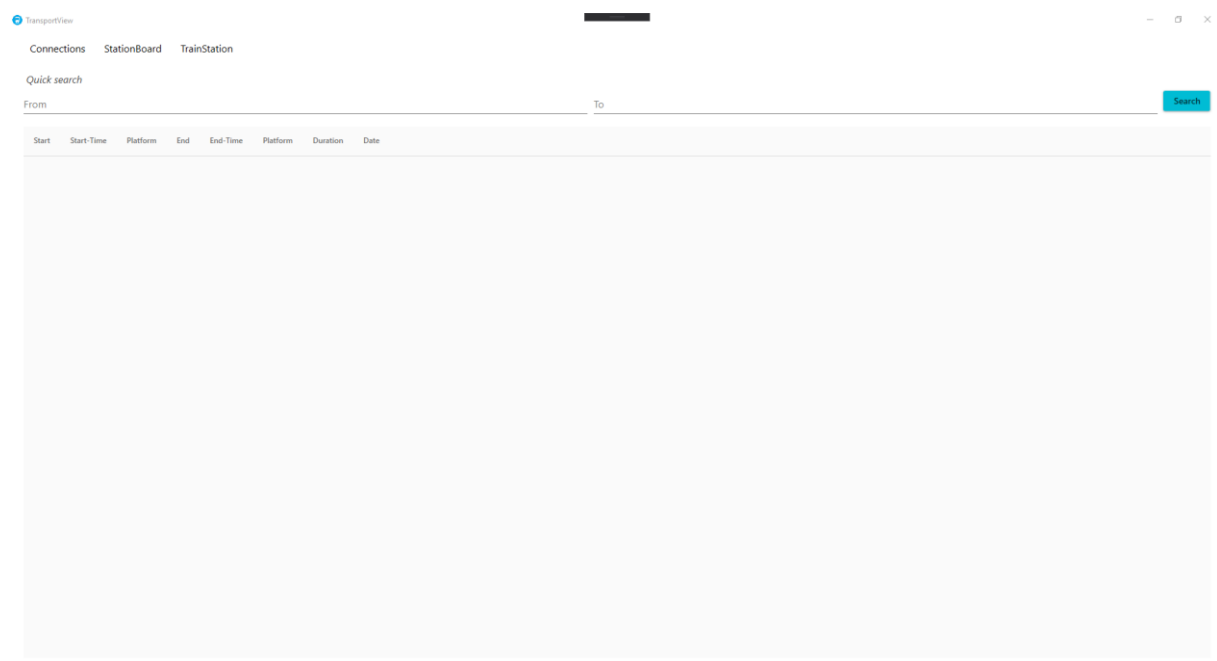
Mockup

Dies ist die Startseite der Applikation.



So sieht nun das Fenster wirklich aus. Die Map und die Current location habe ich nicht umgesetzt auf der Startseite, aber das Prinzip mit der schnell Suche habe ich gelassen. Ich habe einfach noch ein Grid hinzugefügt für die Verbindungen. Vorher wollte ich diese im "Connections" Fenster abfüllen.

Die "Current Location" habe ich weggelassen da ich keine einfache Möglichkeit gefunden habe, den aktuellen Ort herauszufinden. Die Map habe ich deswegen auch weggelassen.



[illegible]

Connections - TransportApp

Connections

<

From

To

Date

04.12.2019

Time

14:18

☐ Is time arrival-time

Search

Start

Start-Time

Platform

End

End-Time

Platform

Duration

Date

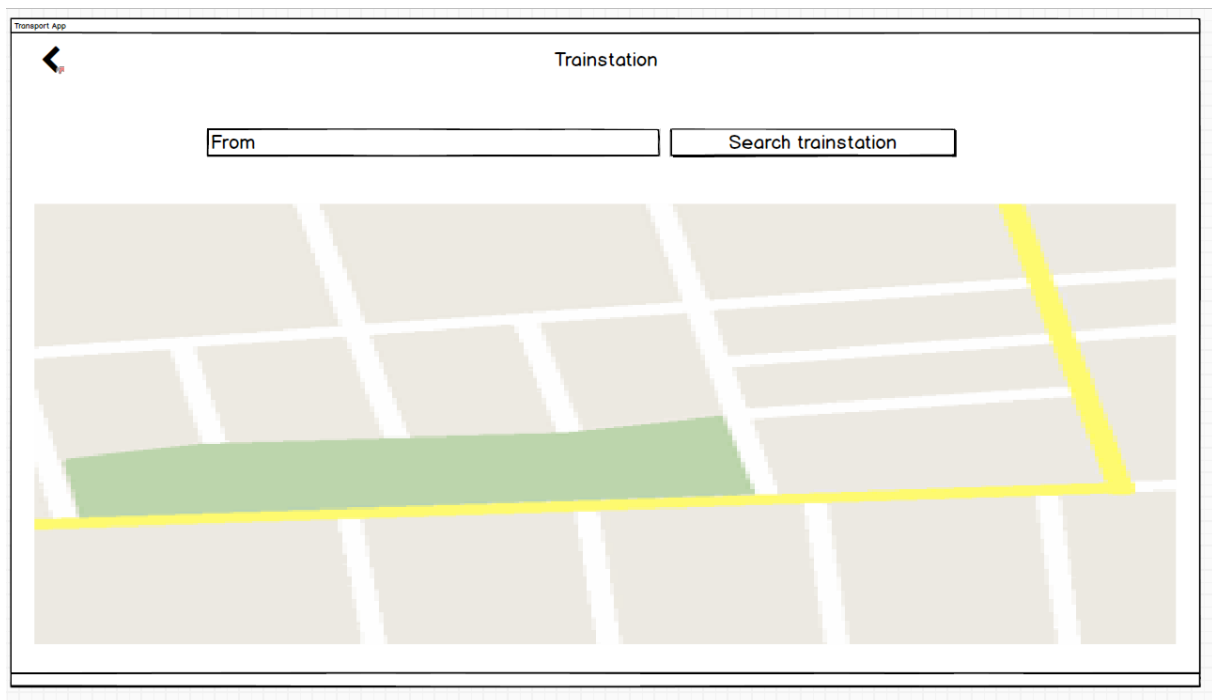
Dieses Fenster dient dazu eine Stationstabelle für eine bestimmte Station anzuzeigen.

[illegible]

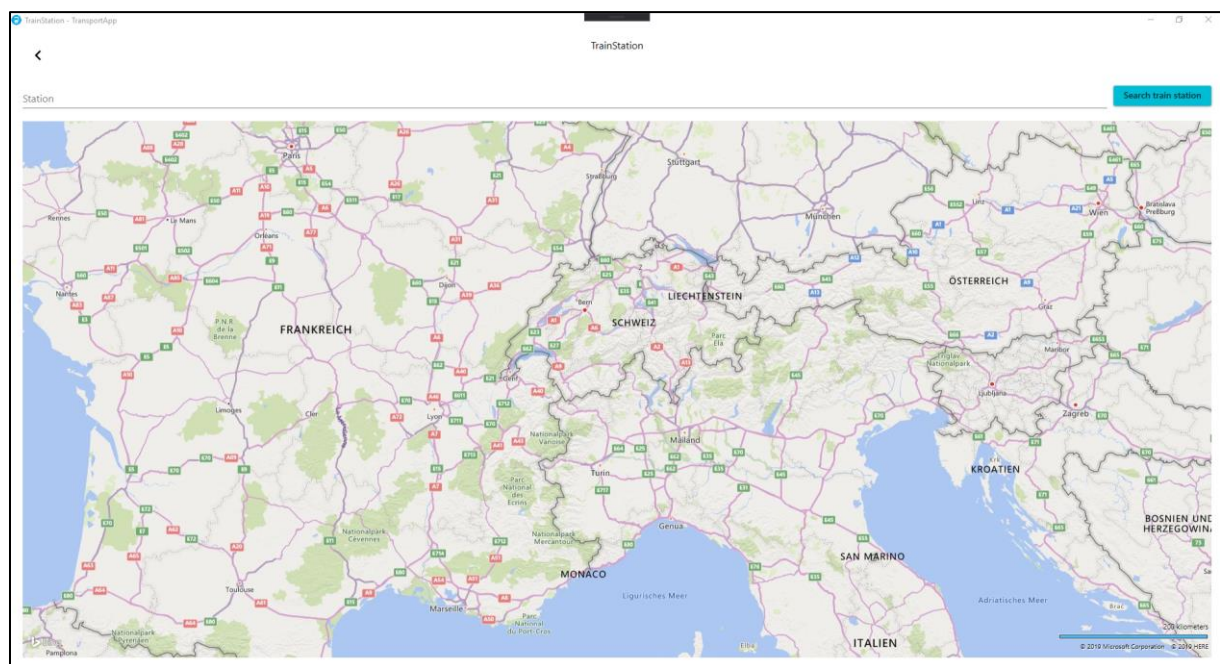
So sieht das Station-Board-Fenster aus, auch da nichts gross anders. Das einzige ist, es gibt kein Pdf, wie oben dargestellt, im Grid sondern es werden gleich Daten abgefüllt. Und die Station-TextBox und den Button habe ich einfach noch, schönheitshalber bis zum Rand gemacht.

The screenshot shows a web application titled "StationBoard - TransportApp" in the top-left corner. The main header area contains a back arrow icon on the left and the text "StationBoard" in the center. On the right side of the header, there is a search bar with the placeholder text "Search train station" and a blue button labeled "Search train station". Below the header, there is a table with a light gray background. The table has five columns with the following headers: "Category", "Name", "Number", "Operator", and "End-Location". The table body is currently empty, showing only the header row.

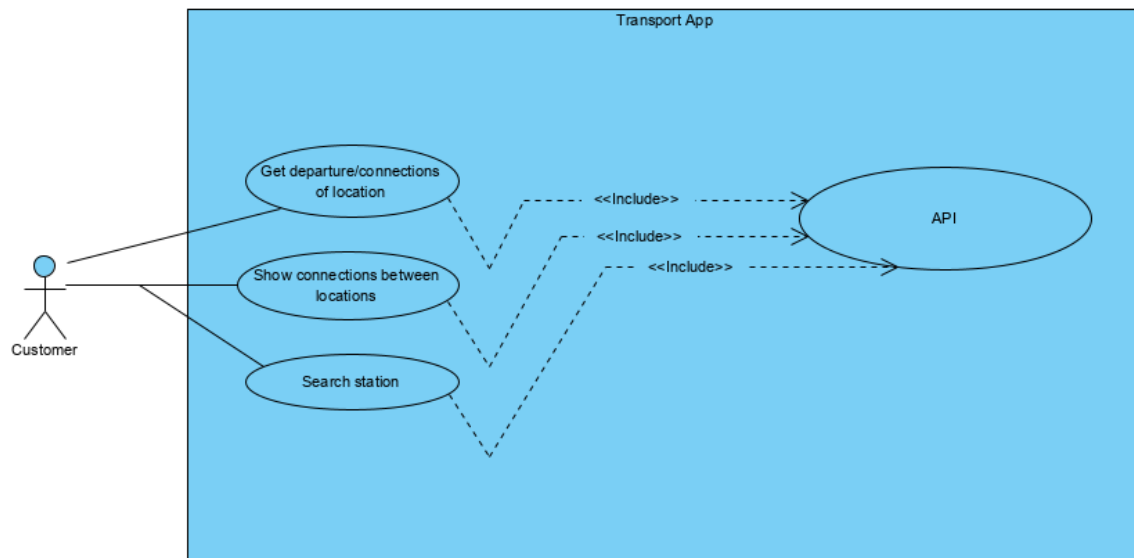
Mit diesem Fenster kann man eine Zugstation finden auf der Karte und die Umgebung anschauen.



So sieht die Umsetzung aus, auch hier habe ich einfach den Button und die TextBox bis zur Seite gemacht, sonst ist alles gleich.



Use Cases



USE CASE

A001, SEARCH STATION

BESCHREIBUNG

Kunde der Transport App, möchte nicht den vollständigen Name eingeben müssen. Sondern Vorschläge angezeigt bekommen.

AKTEUR

Eine zufällige Person, welche die Transport App heruntergeladen at.

VORBEDINGUNGEN

Der Kunde muss eine Internetverbindung haben.

ABLAUF

1. Auf Verbindung suchen klicken
2. Station Name anfangen einzugeben
3. Vorschläge werden angezeigt und sind auswählbar

ERGEBNIS

Ein oder mehrere Ergänzungsvorschläge für den bereits eingeben Text wird angezeigt.

USE CASE

A002, SHOW CONNECTIONS BETWEEN STATIONS

BESCHREIBUNG

Kunde der Transport App gibt zwei Stationen ein und möchte mindestens 5 Verbindungen zur gewünschten Zeit angezeigt bekommen.

AKTEUR

Eine zufällige Person, welche die Transport App heruntergeladen at.

VORBEDINGUNGEN

Der Kunde muss eine Internetverbindung haben.

ABLAUF

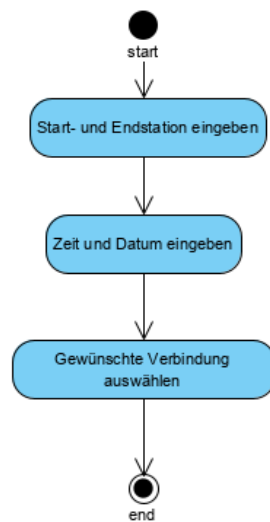
1. Auf "Verbindung suchen" klicken
2. Start- und Endstation eingeben
3. Zeit und Datum eingeben
4. Auf "suchen" klicken
5. Fünf Verbindungen werden angezeigt

ERGEBNIS

Fünf Verbindungen werden angezeigt von der Startstation zur Endstation.

USE CASE	A003, GET DEPARTURE TABLE OF LOCATION
BESCHREIBUNG	Kunde der Transport App kann eine Station eingeben und bekommt alle Verbindungen ab dieser Station.
AKTEUR	Eine zufällige Person, welche die Transport App heruntergeladen at.
VORBEDINGUNGEN	Der Kunde muss eine Internetverbindung haben.
ABLAUF	<ol style="list-style-type: none">1. Kunde klickt auf "Abfahrtsplan"2. Kunde gibt eine Station ein3. Es zeigt alle Abfahrtspläne an
ERGEBNIS	Eine Art Abfahrtstafel wird angezeigt für alle Linien, die an dieser Haltstelle halten.

Aktivität Diagramm zu A002

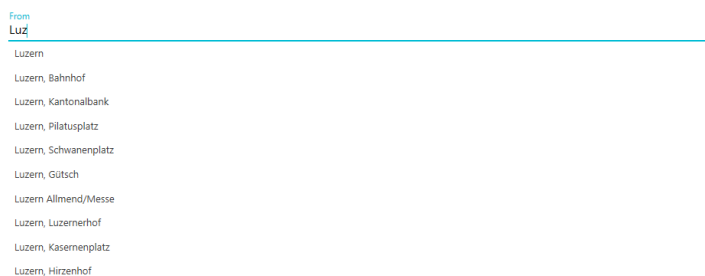


Testing

Search Station, A001

Schritt	Aktivität	Erwartetes Resultat
1	In einer TextBox, in welcher man eine Station suchen kann "Luz" eingeben.	In einer Dropdown-Liste wird "Luzern" angezeigt und noch einige andere Möglichkeiten mit "Luz".

Dieser Test wurde am 04.12.2019 von Raphael Härtel ausgeführt und gab folgendes Resultat zurück. Das Resultat stimmte mit dem Erwarteten Resultat überein.



Show connections between stations, A002

Schritt	Aktivität	Erwartetes Resultat
1	Auf "Connections" klicken.	Das Connection Fenster ist offen.
2	In der TextBox für die "Start-Location" "Luz" eingeben. Und "Luzern" aus der List auswählen.	Luzern wird zuoberst in der Liste angezeigt und wenn es ausgewählt wird, wird es in die TextBox abgefüllt und die Liste mit den Vorschlägen verschwindet.
3	Auf "Search train station" klicken.	Die Tabelle unten dran wird abgefüllt.

Dieser Test wurde am 04.12.2019 von Raphael Härtel ausgeführt und gab folgendes Resultat zurück. Das Resultat stimmte mit dem Erwarteten Resultat überein.

From

Luzern

To

Sursee

Date

06.12.2019

Time

16:00

Is time arrival-time

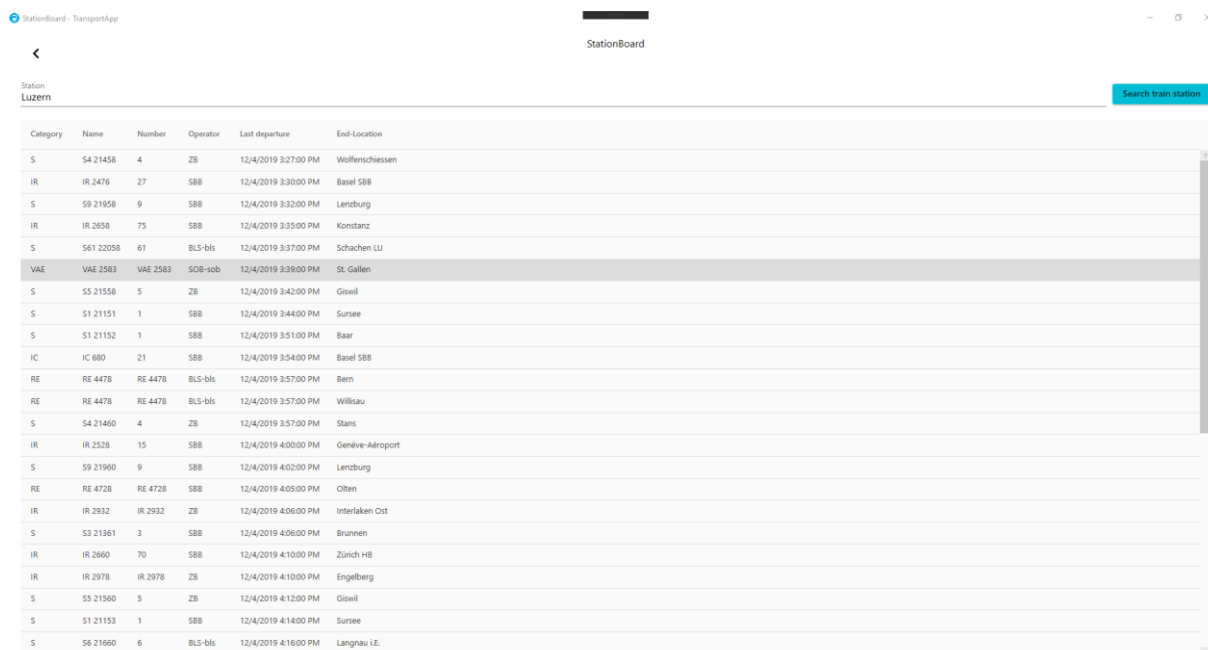
Search

Start	Start-Time	Platform	End	End-Time	Platform	Duration	Date
Luzern	16:00	8	Sursee	16:17	8	17 min	06.12.2019
Luzern	16:05	9	Sursee	16:24	9	19 min	06.12.2019
Luzern	16:14	4	Sursee	16:40	4	26 min	06.12.2019
Luzern	16:30	3	Sursee	16:47	3	17 min	06.12.2019

Get Departure-Table of location, A003

Schritt	Aktivität	Erwartetes Resultat
1	Auf "StationBoard" klicken.	Das StationBoard Fenster ist offen.
2	In der TextBox für die "End-Location" "Cha" eingeben. Und "Cham" aus der List auswählen.	Cham wird zuoberst in der Liste angezeigt und wenn es ausgewählt wird, wird es in die TextBox abgefüllt und die Liste mit den Vorschlägen verschwindet.
3	Dann klickt man auf den Date Picker geklickt und wählt das aktuelle Datum in 2 Tagen.	Ein Fenster mit dem aktuellen Datum selektiert erscheint. Das gewünschte Datum ist wählbar.
4	Darauf wird noch die Zeit auf 16:00 gesetzt.	Die Zeit ist nun auf 16:00.
5	Es wird auf Suchen geklickt.	Nun werden 4 Datensätze angezeigt, die von Luzern nach Cham fahren. Alle haben eine Abfahrtszeit nach "16:00"

Dieser Test wurde am 04.12.2019 von Raphael Härtel ausgeführt und gab folgendes Resultat zurück. Das Resultat stimmte mit dem Erwarteten Resultat überein.



Category	Name	Number	Operator	Last departure	End-Location
S	S4 21458	4	ZB	12/4/2019 3:27:00 PM	Wolfschuessen
IR	IR 2476	27	SBB	12/4/2019 3:30:00 PM	Basel SBB
S	S9 21958	9	SBB	12/4/2019 3:32:00 PM	Lenzburg
IR	IR 2658	75	SBB	12/4/2019 3:35:00 PM	Konstanz
S	S61 22058	61	BLS-bls	12/4/2019 3:37:00 PM	Schachen LU
VAE	VAE 2583	VAE 2583	SBB-icb	12/4/2019 3:39:00 PM	St. Gallen
S	S5 21558	5	ZB	12/4/2019 3:42:00 PM	Giswil
S	S1 21151	1	SBB	12/4/2019 3:44:00 PM	Sursee
S	S1 21152	1	SBB	12/4/2019 3:51:00 PM	Baar
IC	IC 680	21	SBB	12/4/2019 3:54:00 PM	Basel SBB
RE	RE 4478	RE 4478	BLS-bls	12/4/2019 3:57:00 PM	Bern
RE	RE 4478	RE 4478	BLS-bls	12/4/2019 3:57:00 PM	Willisau
S	S4 21460	4	ZB	12/4/2019 3:57:00 PM	Stans
IR	IR 2528	15	SBB	12/4/2019 4:00:00 PM	Genève-Aéroport
S	S9 21960	9	SBB	12/4/2019 4:02:00 PM	Lenzburg
RE	RE 4728	RE 4728	SBB	12/4/2019 4:05:00 PM	Olten
IR	IR 2932	IR 2932	ZB	12/4/2019 4:06:00 PM	Interlaken Ost
S	S3 21361	3	SBB	12/4/2019 4:06:00 PM	Brunnen
IR	IR 2660	70	SBB	12/4/2019 4:10:00 PM	Zürich HB
IR	IR 2978	IR 2978	ZB	12/4/2019 4:10:00 PM	Engelberg
S	S5 21560	5	ZB	12/4/2019 4:12:00 PM	Giswil
S	S1 21153	1	SBB	12/4/2019 4:14:00 PM	Sursee
S	S6 21660	6	BLS-bls	12/4/2019 4:16:00 PM	Langnau i.E.

Installation

1. Zuerst muss man Git installieren.
2. Danach muss man zu dem Pfad gehen, an dem man das Programm möchte.
Zu diesem Pfad muss man im CMD wechseln und dann
"Git clone <https://github.com/LostLegend/modul-318-student.git>" eingeben.
3. Dann das Projekt "modul-318-student" öffnen.
4. Dann in den "TransportApp" Ordner gehen.
5. Und zu guter Letzt in den "bin\Debug" Ordner und die "TransportApp.exe" starten.

Gesamter Pfad: "C:**your own path**\modul-318-student\TransportApp\bin\Debug\TransportApp.exe"

Zum Deinstallieren einfach den Ordner "modul-318-studen" löschen. An dem installierten Pfad.

Programmierrichtlinien

Naming Conventions

Alle Namen sind Englisch oder Deutsch, kein Mix.

Keine Abkürzungen bei der Benennung.

Variablen sind klein geschrieben.

Properties, Methoden, Klassen und Gui-Controls werden grossgeschrieben.

Bei mehreren Wörtern sollen die nächsten Wörter gross sein.

Beispiel Variable: "_numberOne" und nicht "NumberOne", bei lokalen variablen ohne bodenstrich.

Also z.B. "numberOne".

Bei einer Methode: "CalculateSinus()" und nicht "calculatesinus()".

"Exception" und nicht "ex".

Declaration

Zuerst sollte der Konstruktor sein, dann Properties und zuletzt Methoden.

Deklaration von Properties sollen, wenn möglich im Konstruktor vorgenommen werden.

Comments

Methoden sowie auch Properties sollen kommentiert sein. Innerhalb von Methoden sollten keine Kommentare vorhanden sein, da der Code selbsterklären sein sollte.

Statements

Bei Statements die mehr als eine Zeile beinhalten sollten immer geschweifte Klammern verwendet werden.

Falls aber z.B. ein If/Else nur eine Zeile Code beinhaltet dürfen die geschweifte Klammern weggelassen werden.

Keine Verschachtelung von Statements. Am besten invertiert man ein Statement, um dies zu vermeiden.

Projektinfo

Mein Projekt habe ich mit WPF erstellt. Ich habe dabei das MVVM Model benutzt. Für die Events mit MVVM habe ich noch da NuGet "Expression.Blend.Sdk" heruntergeladen. Zusätzlich habe ich für die Style das NuGet "Material Design Toolkit" gebraucht. Ich habe angefangen meine Style selbst zu machen, hatte leider zu wenig Zeit zum alles selbst zu designen. Deswegen habe ich diese NuGet gebraucht.

