

# M318

Analysieren und objektbasiert programmieren mit Komponenten



## Timetable Application – Dokumentation

Autor: Maurin Jenni  
Instruktor: Markus Estermann  
Datum: 01.06.2016

## Inhaltsverzeichnis

1.0 Einleitung .....	3
2.0 Vorgaben und Anforderungen.....	3
2.1 Vorgaben .....	3
2.2 Anforderungen .....	3
3.0 Analyse und Design .....	4
3.1 Use Case.....	4
3.2 Aktivitätsdiagramm .....	4
3.2.1 Station suchen .....	4
3.2.2 Datum und Zeiteingabe .....	4
3.2.3 Verbindung suchen.....	5
3.2.4 Abfahrtstafel suchen.....	5
3.3 Klassendiagramm .....	6
4.0 Installation und Anwendung .....	7
5.0 Test Case .....	8
5.1 A001 und A004: Station suchen.....	8
5.2 Verbindung suchen.....	9
5.2.1 A002: Verbindung suchen (Search Connection).....	9
5.2.2 A005: Zeitspezifische Verbindungsuche .....	9
5.3 A003: Abfahrtstafel suchen (Departure Table).....	10

## 1.0 Einleitung

In diesem ÜK geht es um die Programmierung einer Applikation, welche auf die Fahrplandaten des Schweizerischen öffentlichen Verkehrs benutzt. Diese Daten erreicht man mit dem frei zur Verfügung stehenden API: <http://transport.opendata.ch/>

Das ganze Projekt findet man im GIT-Repository unter der Adresse:  
<https://github.com/maurinjenni/modul-318-student>

## 2.0 Vorgaben und Anforderungen

### 2.1 Vorgaben

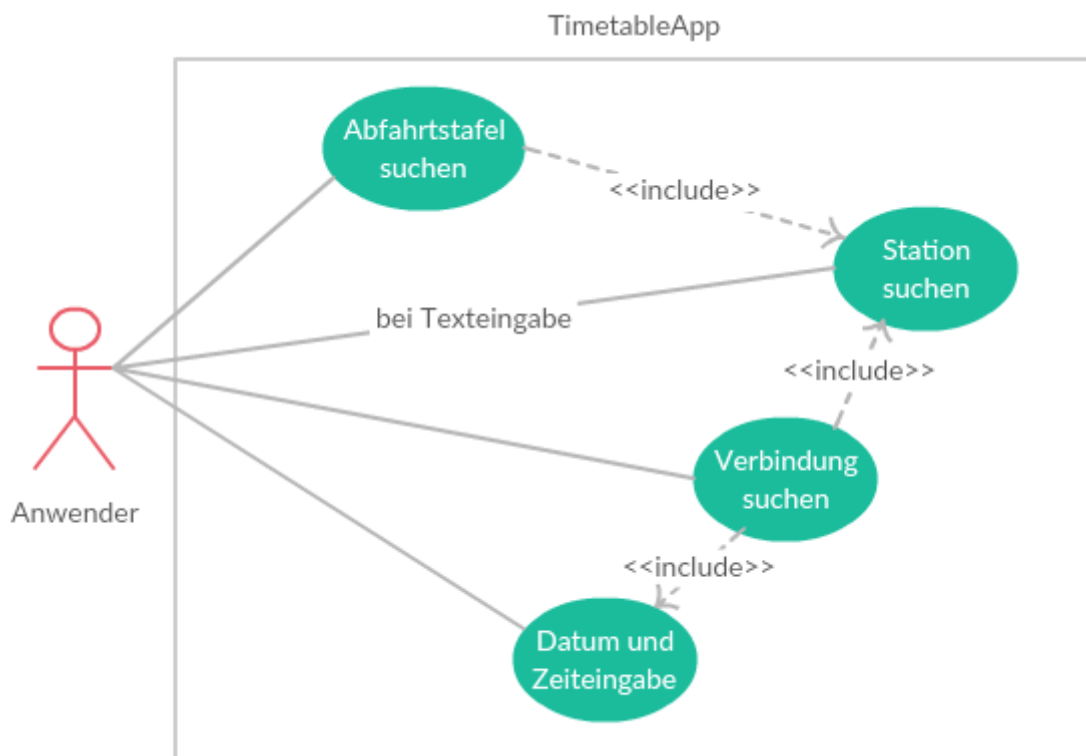
Vorgabe für das Projekt ist die Umsetzung mit WinForms in der Entwicklungsumgebung Visual Studio Community 2015. Die Schnittstelle für das API ist schon ausprogrammiert und kann auf GitHub geklont werden. Der selbergeschriebene Code (UI) ist auch auf GitHub eingchecked.

### 2.2 Anforderungen

ID	Beschreibung	Status/Bemerkung
A001	Ich als ÖV-Benutzer möchte die Start und Endstation mittels Textsuche suchen können, damit ich nicht alle Stationsnamen auswendig lernen muss.	Umgesetzt (zusammen mit A004)
A002	Ich als ÖV-Benutzer möchte die aktuellen Verbindungen zwischen den beiden gefundenen und ausgewählten Station sehen, damit ich weiss wann ich zur Station muss.	Umgesetzt
A003	Ich als ÖV-Benutzer möchte sehen welche Verbindungen ab einer bestimmten Station vorhanden sind. Damit ich eine Art Station Abfahrtstafel Zuhause haben kann.	Umgesetzt
A004	Ich als ÖV-Benutzer möchte, dass schon während meiner Eingabe erste Suchresultate erscheinen, damit ich effizienter nach Stationen suchen kann.	Umgesetzt (zusammen mit A001)
A005	Ich als ÖV-Benutzer möchte Verbindungen zu einem anderen Zeitpunkt suchen können, damit ich zukünftige Reisen planen kann.	Umgesetzt
A006	Ich als ÖV-Benutzer möchte sehen wo sich eine Station befindet, damit ich mir die vor Ort-Situation besser vorstellen kann.	Nicht Umgesetzt
A007	Ich als ÖV-Benutzer möchte Stationen finden die in der Nähe meiner aktuellen Position sind, damit ich schnell auf das ÖV Netz der Schweiz kommen kann.	Teilweise umgesetzt (Durch die Eingabe einer Adresse in das Suchfeld wird die nächste Station gefunden)
A008	Ich möchte meine gefundenen Resultate via Mail weiter schicken können, damit ich oder andere von meinen Recherchen profitieren können.	Nicht Umgesetzt

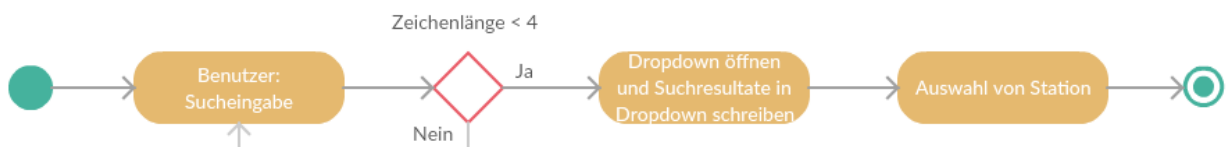
## 3.0 Analyse und Design

### 3.1 Use Case

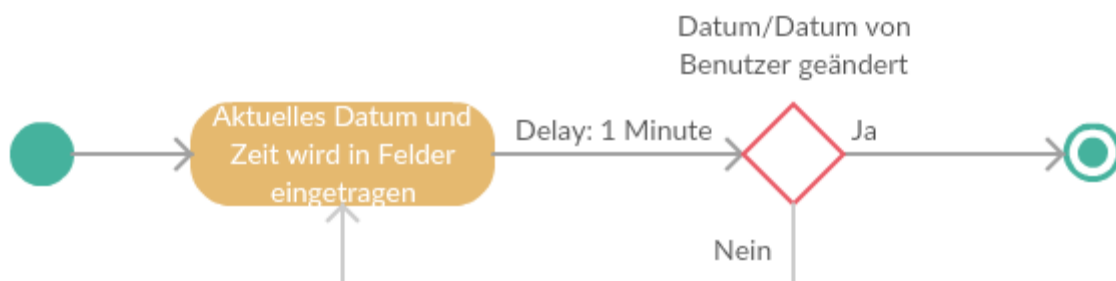


### 3.2 Aktivitätsdiagramm

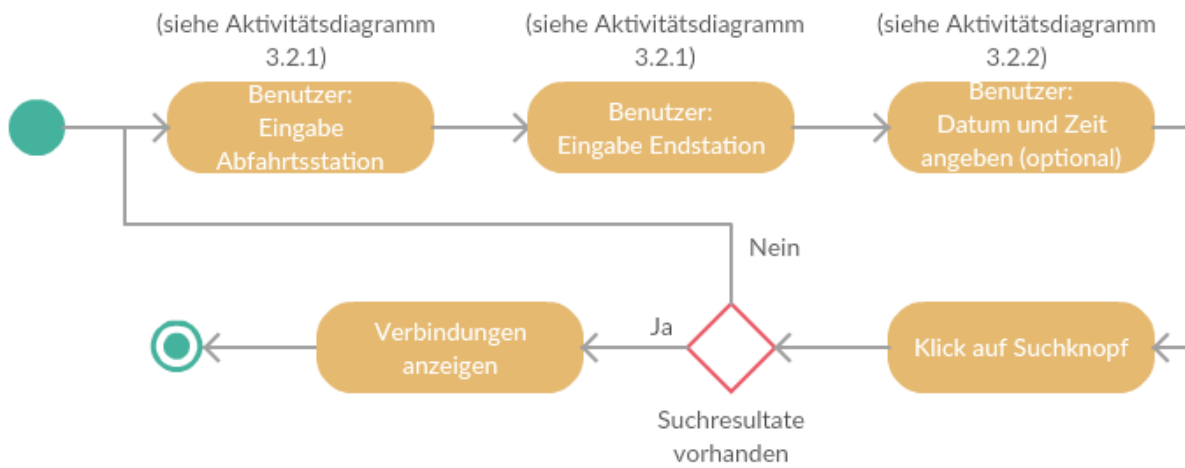
#### 3.2.1 Station suchen



#### 3.2.2 Datum und Zeiteingabe



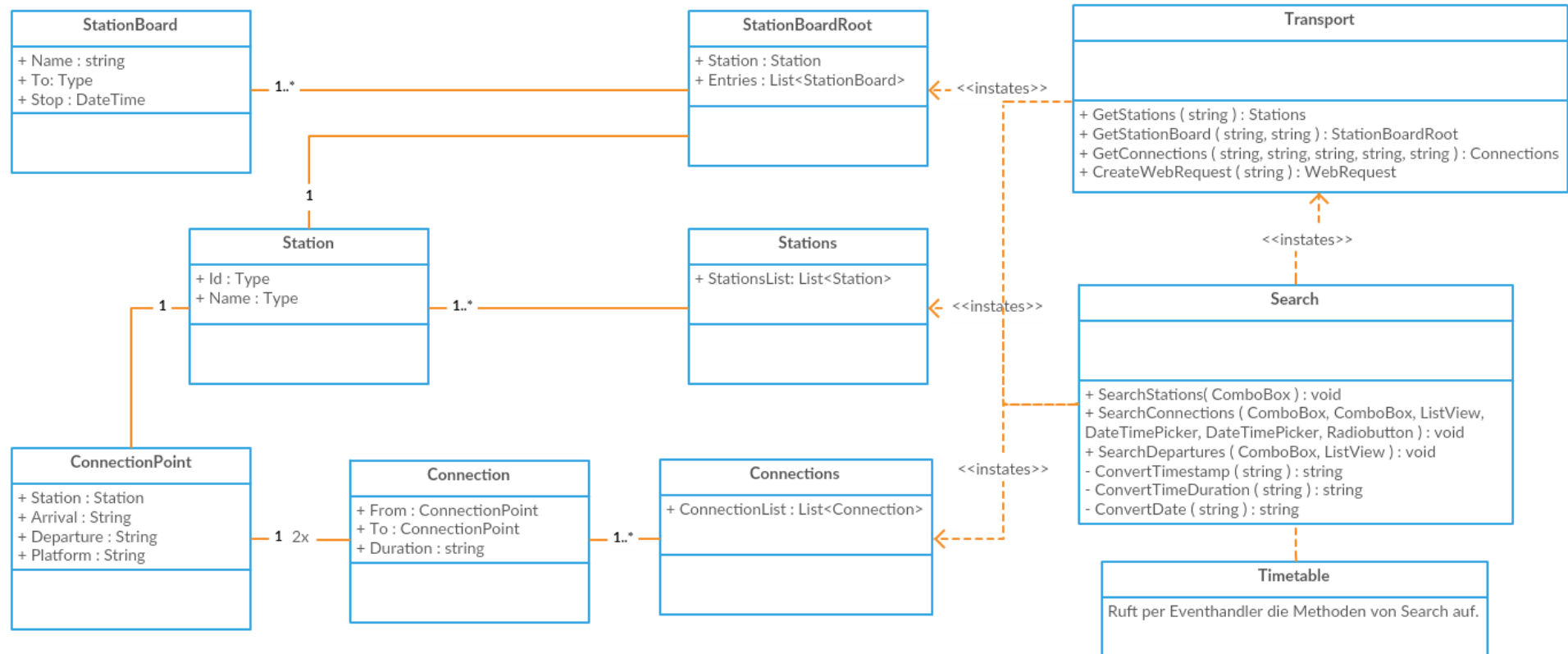
### 3.2.3 Verbindung suchen



### 3.2.4 Abfahrtstafel suchen



### 3.3 Klassendiagramm



## 4.0 Installation und Anwendung

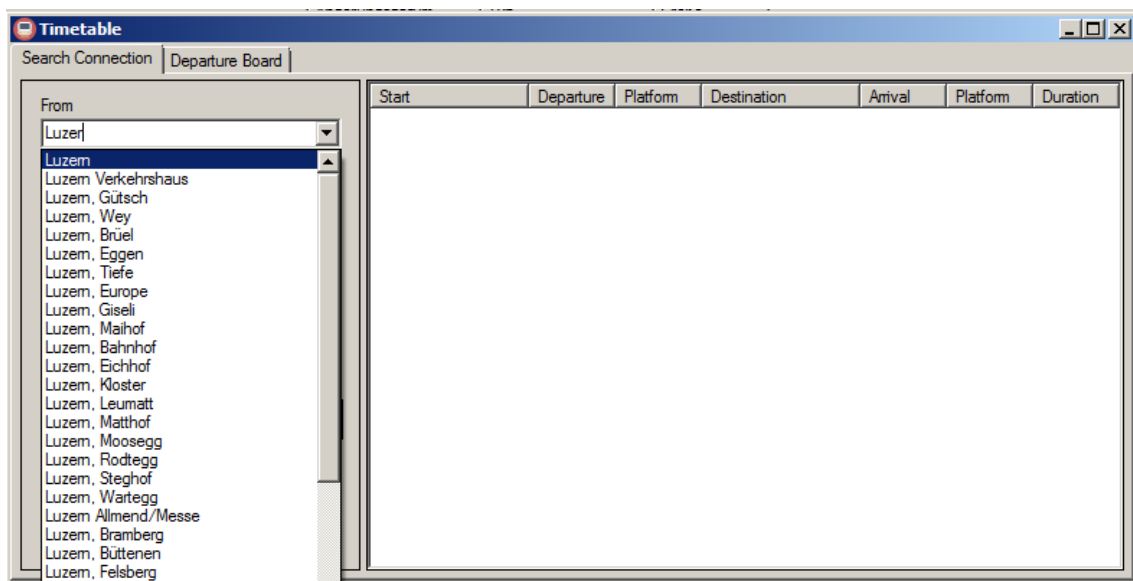
Die Installation ist sehr simpel, da das Programm nur die ausführbare Datei (.exe) und Zusatzdateien benötigt kann der Ordner kopiert und die Datei ausgeführt werden. Diesen findet man unter der Adresse <https://github.com/maurinjenni/modul-318-student> im Unterordner „maurinjenni“. Weitere Infos sind in der Readme-Datei vorhanden.

## 5.0 Test Case

Vorbereitung: Programm öffnen.

### 5.1 A001 und A004: Station suchen

1. Nach der Eingabe des vierten Zeichens in ein Stationen-Suchfeld, soll sich das Dropdown öffnen und die Resultate anzeigen. Mittels Pfeiltasten oder Maus kann nun die gewünschte Station ausgewählt werden. Nach dem Klick/Enter schliesst sich das Dropdown wieder und im Feld befindet sich die Station.
2. Wenn sich die Anzahl Zeichen im Feld wieder unter 4 verringert, soll sich das Dropdown wieder schliessen und die Resultate nicht mehr im Dropdown haben.
3. Die Resultate werden bei neuer Eingabe immer wieder überschrieben.

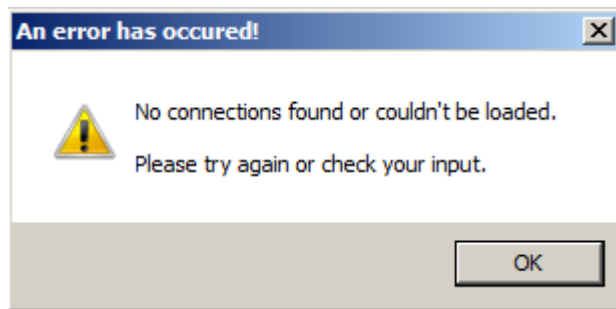




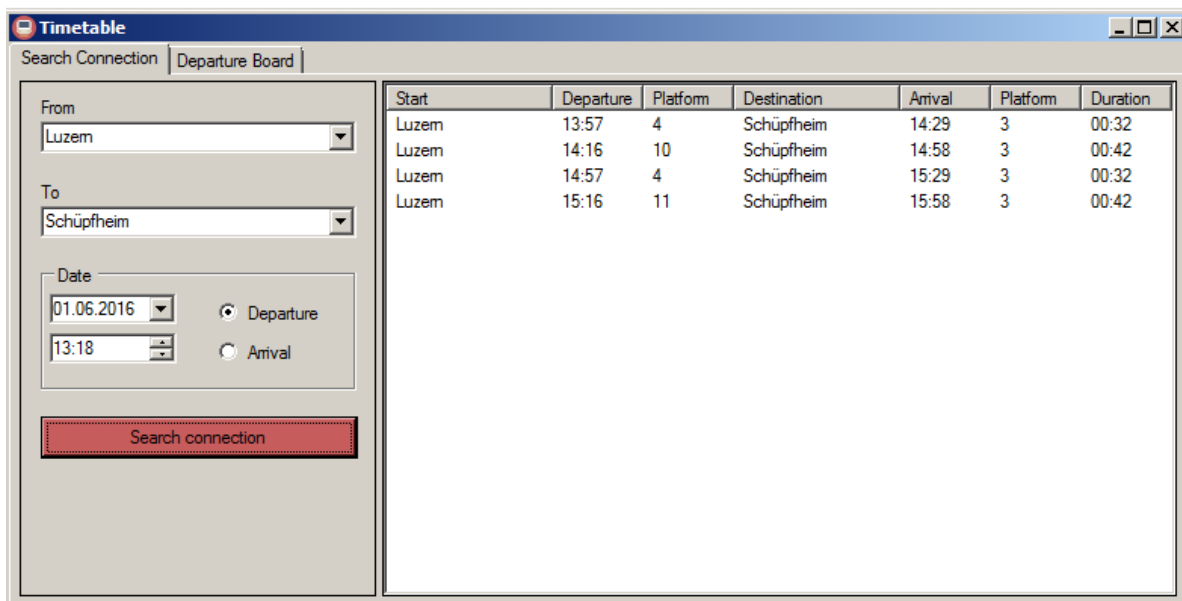
## 5.2 Verbindung suchen

### 5.2.1 A002: Verbindung suchen (Search Connection)

1. Als Voraussetzung für die Verbindungssuche müssen die Stations-Suchfelder (From, To) einen gültigen Wert enthalten. Ansonsten wird eine Meldung angezeigt.



2. Nach erfolgreicher Suche sollen die gefundenen Verbindungen mit dem Abfahrts- bzw. Zielort (Name, Uhrzeit, Gleis) und die Dauer in einer Liste (ListView) angegeben werden.
3. Zeit wird im korrektem Format (Stunden: Minuten) angezeigt.
4. Nach erneuter Suche werden die Resultate überschrieben.



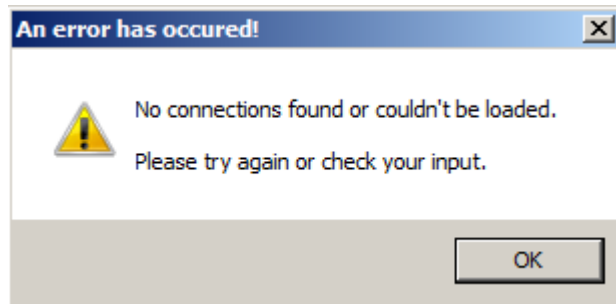
Start	Departure	Platform	Destination	Arrival	Platform	Duration
Luzern	13:57	4	Schüpfheim	14:29	3	00:32
Luzern	14:16	10	Schüpfheim	14:58	3	00:42
Luzern	14:57	4	Schüpfheim	15:29	3	00:32
Luzern	15:16	11	Schüpfheim	15:58	3	00:42

### 5.2.2 A005: Zeitspezifische Verbindungssuche

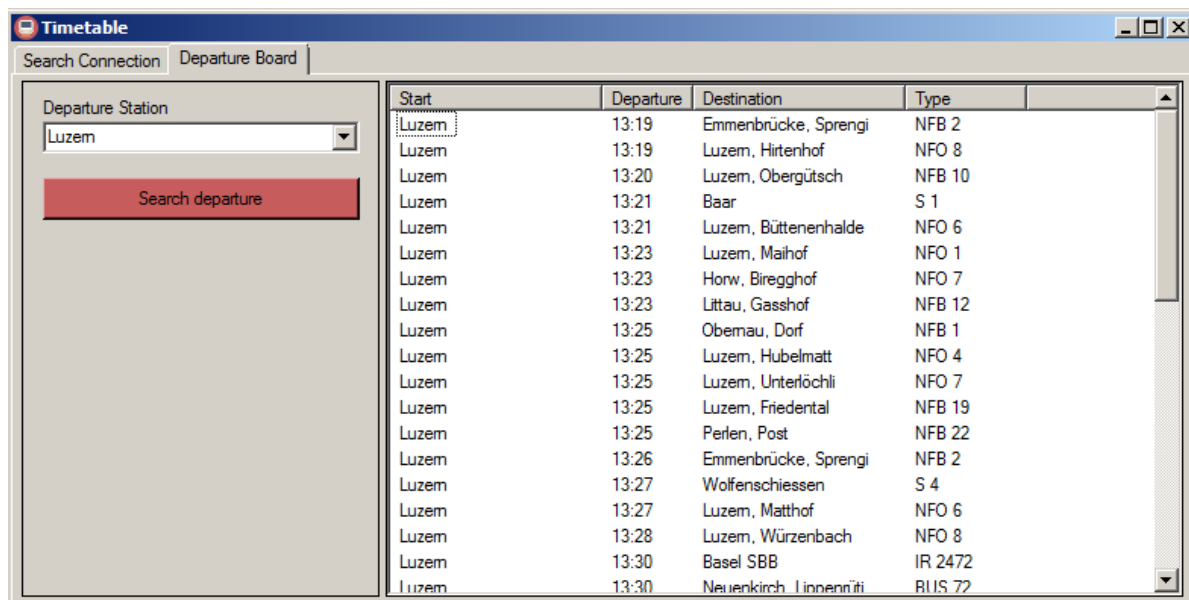
1. Es gelten die gleichen Voraussetzungen, wie bei „A002 Verbindung suchen“
2. Programm schliessen und neu öffnen: Datum und Zeit sollten aktuell sein
3. Nach einer Wartezeit von 1 Minute muss sich die Zeit aktualisieren, wenn diese nicht durch den Benutzer geändert wurde

### 5.3 A003: Abfahrtstafel suchen (Departure Table)

1. Als Voraussetzung für die Verbindungssuche muss das Station-Suchfeld (Departure Station) einen gültigen Wert enthalten. Ansonsten wird eine Meldung angezeigt.



2. Nach erfolgreicher Suche sollen die gefundenen Verbindungen mit dem Abfahrts- bzw. Zielort (Name), die Abfahrtszeit und der Transportmittel in einer Liste (ListView) angegeben werden.
3. Zeit wird im korrektem Format (Stunden: Minuten) angezeigt.
4. Nach erneuter Suche werden die Resultate überschrieben.



Start	Departure	Destination	Type
Luzern	13:19	Emmenbrücke, Sprengi	NFB 2
Luzern	13:19	Luzern, Hirtenhof	NFO 8
Luzern	13:20	Luzern, Obergütsch	NFB 10
Luzern	13:21	Baar	S 1
Luzern	13:21	Luzern, Büttenehalde	NFO 6
Luzern	13:23	Luzern, Maihof	NFO 1
Luzern	13:23	Horw, Bireggshof	NFO 7
Luzern	13:23	Littau, Gasshof	NFB 12
Luzern	13:25	Obemau, Dorf	NFB 1
Luzern	13:25	Luzern, Hubelmatt	NFO 4
Luzern	13:25	Luzern, Unterlöchli	NFO 7
Luzern	13:25	Luzern, Friedental	NFB 19
Luzern	13:25	Perlen, Post	NFB 22
Luzern	13:26	Emmenbrücke, Sprengi	NFB 2
Luzern	13:27	Wolfenschiessen	S 4
Luzern	13:27	Luzern, Matthof	NFO 6
Luzern	13:28	Luzern, Würzenbach	NFO 8
Luzern	13:30	Basel SBB	IR 2472
Luzern	13:30	Neuenkirch, Linnendüti	RLIS 72