|  |
| --- |
| SwissTransport |
| Modul 318 – Analysieren und objektbasiert programmieren mit Komponenten |
|  |

Neela Kazi

Gruppe 1, Kursraum 4

ICT-Berufsbildung

1.6.2016

Inhalt

[1. Einleitung 3](#_Toc452561247)

[2. Anforderungen 3](#_Toc452561248)

[3. Diagramme 4](#_Toc452561249)

[3.1 Use Case Diagramm 4](#_Toc452561250)

[3.2 Aktivitätsdiagramm 4](#_Toc452561251)

[3.3 Klassendiagramm 5](#_Toc452561252)

[4. Mockups 6](#_Toc452561253)

[4.1 Verbindungen Von-Nach 6](#_Toc452561254)

[4.2 Abfahrtstafel 7](#_Toc452561255)

[4.3 Email senden 8](#_Toc452561256)

[5. Testing 8](#_Toc452561257)

[5.1 Testfälle 8](#_Toc452561258)

[5.1.1 Test zu A001: Autocomplete Funktion 8](#_Toc452561259)

[5.1.2 Test zu A002: Verbindungen zwischen zwei Stationen anzeigen 8](#_Toc452561260)

[5.1.3 Test zu A003: Verbindungen ab einer Station 9](#_Toc452561261)

[5.2 Testprotokoll 9](#_Toc452561262)

[5.2.1 Test TA001 9](#_Toc452561263)

[5.2.2 Test TA002 9](#_Toc452561264)

[5.2.3 Test TA003 9](#_Toc452561265)

[6. Install/Deinstall 10](#_Toc452561266)

[6.1 Install: Visual Studio 10](#_Toc452561267)

[6.2 Gmail-Account 10](#_Toc452561268)

[7. Endprodukt 10](#_Toc452561269)

[7.1 A001 + A004 10](#_Toc452561270)

[7.2 A002 12](#_Toc452561271)

[7.3 A003 13](#_Toc452561272)

[7.4 A004 14](#_Toc452561273)

[7.5 A005 14](#_Toc452561274)

[7.6 A006 15](#_Toc452561275)

[7.7 A007 16](#_Toc452561276)

[7.8 A008 16](#_Toc452561277)

[8. Reflexion 17](#_Toc452561278)

[8.1 Was war gut? 17](#_Toc452561279)

[8.2 Verbesserungsmöglichkeiten 18](#_Toc452561280)

[8.2.1 AutoComplete verschnellern 18](#_Toc452561281)

[8.2.2 Mail Tabellendarstellung verschönern 18](#_Toc452561282)

[8.2.3 DataGridView in Abfahrtstafel anpassen 18](#_Toc452561283)

[8.2.4 Zufällige Eingabe 19](#_Toc452561284)

[8.3 Erkenntnisse 19](#_Toc452561285)

# Einleitung

Diese Applikation wurde im ÜK Modul 318 entwickelt und dient zur Übersicht des Schweizer-ÖV-Fahrplans. Sie wurde mit C# WinForms programmiert.

# Anforderungen

|  |  |
| --- | --- |
| ID | Beschreibung |
| A001 | Ich als ÖV-Benutzer möchte die Start und Endstation mittels Textsuche suchen können, damit ich nicht alle Stationsnamen auswendig lernen muss. |
| A002 | Ich als ÖV-Benutzer möchte die aktuellen Verbindungen zwischen den beiden gefundenen und ausgewählten Stationen sehen, damit ich weiss wann ich zur Station muss. |
| A003 | Ich als ÖV-Benutzer möchte sehen welche Verbindungen ab einer bestimmten Station vorhanden sind. Damit ich eine Art Station Abfahrtstafel Zuhause haben kann. |
| A004 | Ich als ÖV-Benutzer möchte, dass schon während meiner Eingabe erste Such-Resultate erscheinen, damit ich effizienter nach Stationen suchen kann. |
| A005 | Ich als ÖV-Benutzer möchte Verbindungen zu einem anderen Zeitpunkt suchen können, damit ich zukünftige Reisen planen kann. |
| A006 | Ich als ÖV-Benutzer möchte sehen wo sich eine Station befindet, damit ich mir die vor Ort-Situation besser vorstellen kann. |
| 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. |
| A008 | Ich als ÖV-Benutzer möchte meine gefundenen Resultate via Mail weiter schicken können, damit ich oder andere von meinen Recherchen profitieren können. |

# Diagramme

## Use Case Diagramm

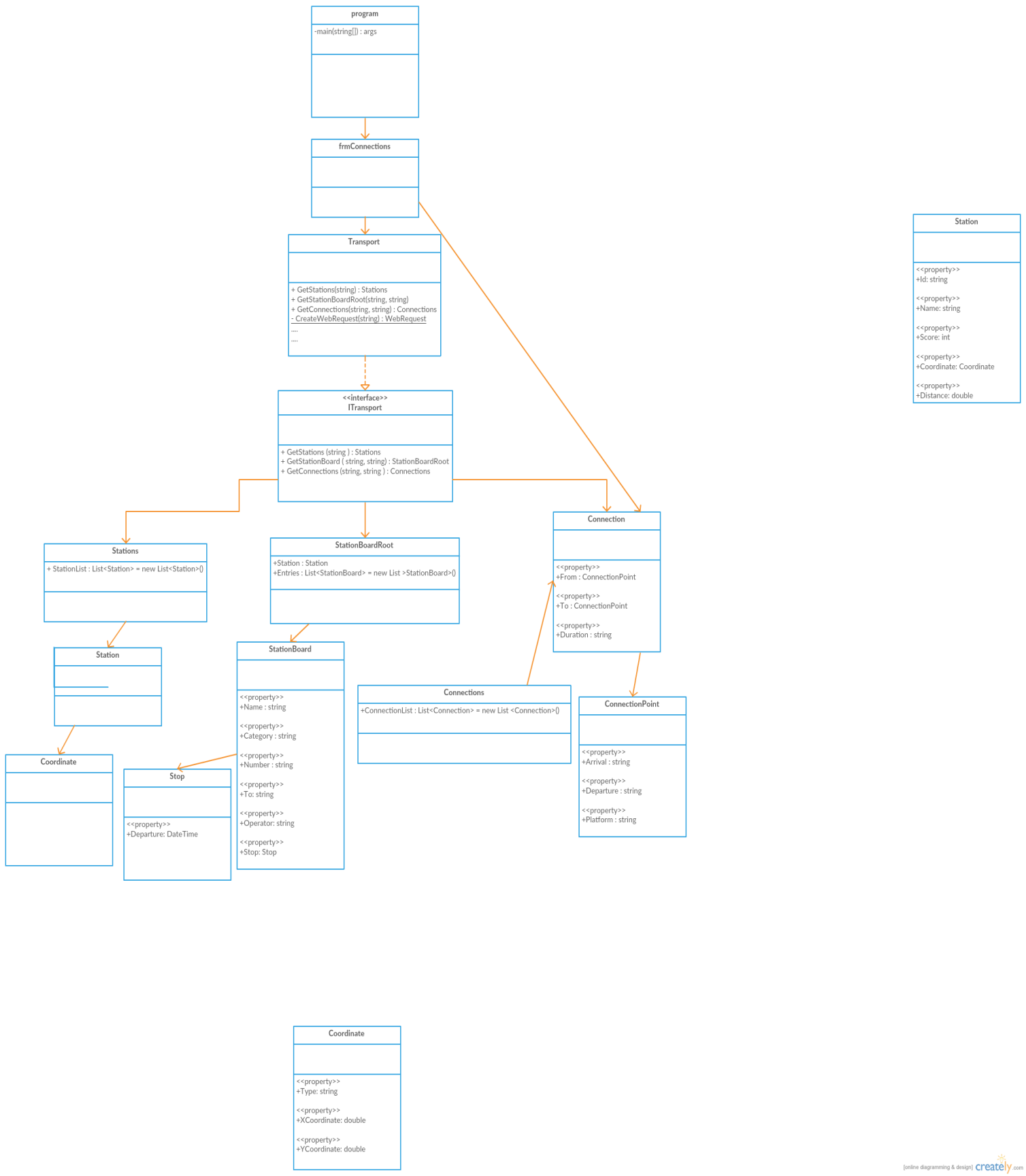


|  |  |
| --- | --- |
| **Beschreibung** | Beim Eingeben der Start- und Endstation sollen nach dem Klick auf den Button die Verbindungen zwischen den eingegebenen Stationen angezeigt werden. |
| **Akteure** | Der Benutzer, welcher die Verbindungen zwischen zwei Stationen einsehen möchte. |
| **Bedingung** | Es müssen zwei gültige Stationen eingegeben werden. |
| **Ergebniszustand** | Der Benutzer kann die Verbindungen zwischen den eingegebenen Stationen sehen. |
| **Standartablauf** | Verbindungen zwischen den zwei eingegebenen Stationen werden in einer Tabelle angezeigt. |
| **Alternative Abläufe** | Falls ungültige oder keine Stationen eingegeben werden, erscheint eine Fehlermeldung, die den Benutzer dazu auffordert gültige Stationen einzugeben. |

## Aktivitätsdiagramm

Die Eingabe der Start- und Endstation wird erst überprüft, sobald der „Verbindungen anzeigen“ Button gedrückt wird.

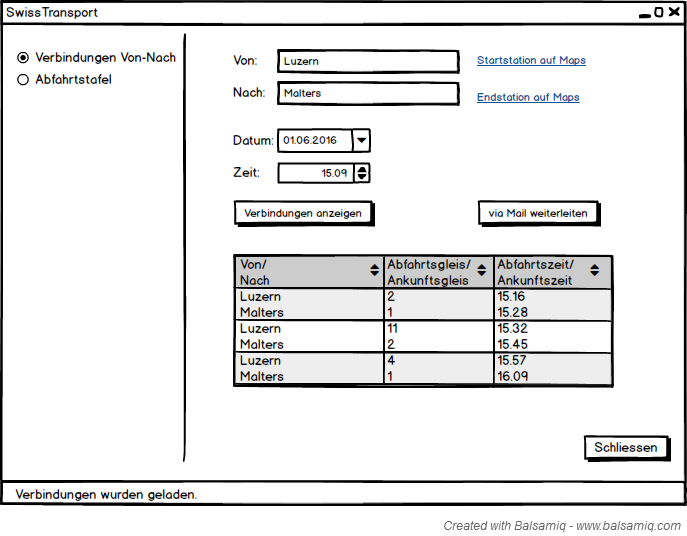
## Klassendiagramm



UML ist noch auf dem Repository um genauer anzuschauen.

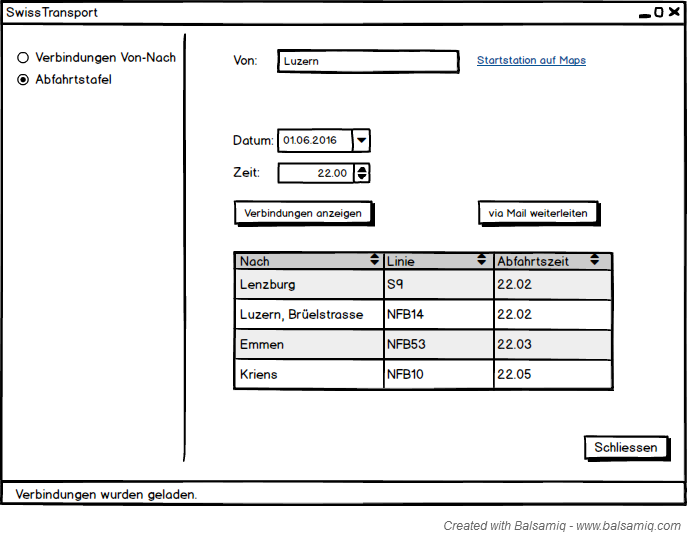
# Mockups

## Verbindungen Von-Nach



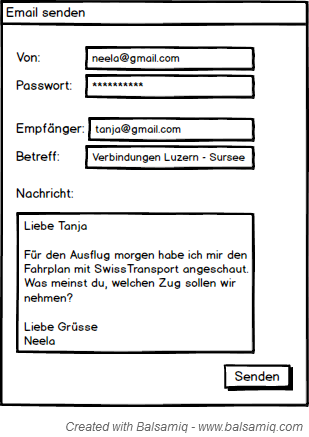
Ansicht wie das Form aussieht, wenn der „Verbindungen Von-Nach“ Radiobutton gewählt ist. Wenn auf den „via Mail weiterleiten“ Button gedrückt wird öffnet sich ein neues Form ([4.3 Email senden](#_Email_senden)).

## Abfahrtstafel



Wenn der „Abfahrtstafel“ Radiobutton gewählt ist, verschwinden beim Form das „Nach“ Label, die Endstation Textbox und der Link für „Endstation auf Maps“.

## Email senden



# Testing

## Testfälle

### Test zu A001: Autocomplete Funktion

|  |  |
| --- | --- |
| **Test-ID** | TA001 |
| **Vorbedingung** | Die sln-Datei muss im VisualStudio geöffnet sein. Das Programm muss gestartet sein. |
| **Gegeben** | Der User befindet sich auf der Verbindungen- oder Abfahrtstafel-Maske. |
| **Wenn** | Der User tippt eine Station ein. |
| **Erwartetes Ergebnis** | Ab dem 4. eingegebenen Zeichen erscheint ein Autocomplete-Dropdown, worin die dazu passenden Stationen aufgeführt sind. |
| **Was darf nicht passieren** | Das Autocomplete-Dropdown erscheint nicht oder es werden falsche Stationen zur Vervollständigung angezeigt. |

### Test zu A002: Verbindungen zwischen zwei Stationen anzeigen

|  |  |
| --- | --- |
| **Test-ID** | TA002 |
| **Vorbedingung** | Die sln-Datei muss im VisualStudio geöffnet sein. Das Programm muss gestartet sein. |
| **Gegeben** | Der User befindet sich auf der Verbindungen -Maske. |
| **Wenn** | Der User gibt eine Start- und Endstation ein (Stationen müssen unterschiedlich sein) und drückt auf den „Verbindungen anzeigen“-Button. |
| **Erwartetes Ergebnis** | Es werden Verbindungen zwischen den zwei Stationen angezeigt. Dabei werden zusätzlich die Abfahrts-/Ankunftsgleise und Abfahrts-/Ankunftszeiten in einer Tabelle ausgegeben. |
| **Was darf nicht passieren** | Es werden falsche Verbindungen, also solche die nicht zwischen diesen zwei Stationen existieren, angezeigt. |

### Test zu A003: Verbindungen ab einer Station

|  |  |
| --- | --- |
| **Test-ID** | TA003 |
| **Vorbedingung** | Die sln-Datei muss im VisualStudio geöffnet sein. Das Programm muss gestartet sein. |
| **Gegeben** | Der User befindet sich auf der Abfahrtstafel-Maske. |
| **Wenn** | Der User tippt eine Station ein und drückt auf den „Verbindungen anzeigen“-Button. |
| **Erwartetes Ergebnis** | Es werden alle Abfahrten ab dieser Station, mit der Linie und Abfahrtszeit in einer Tabelle ausgegeben. |
| **Was darf nicht passieren** | Es werden falsche Abfahrten von der eingegebenen Station angezeigt. |

## Testprotokoll

### Test TA001

|  |  |
| --- | --- |
| **Test-ID** | TA001 |
| **Tatsächliches Ergebnis** | Ab dem 4. Eingegebenen Zeichen erscheint ein Autocomplete-Dropdown mit den dazu passenden Stationen. |
| **Fehler/Meldungen** | Keine Fehler/Meldungen. |

### Test TA002

|  |  |
| --- | --- |
| **Test-ID** | TA002 |
| **Tatsächliches Ergebnis** | Die Verbindungen zwischen den zwei eingegebenen Stationen werden mit den Gleisen und Zeiten in einer Tabelle angezeigt. |
| **Fehler/Meldungen** | Keine Fehler/Meldungen. |

### Test TA003

|  |  |
| --- | --- |
| **Test-ID** | TA003 |
| **Tatsächliches Ergebnis** | Die Abfahrten von der eingegebenen Station werden mit der Linie und Abfahrtszeit in einer Tabelle angezeigt. |
| **Fehler/Meldungen** | Keine Fehler/Meldungen. |

# Install/Deinstall

## Install: Visual Studio

Zum Ausführen des Programms muss Visual Studio 20xx (mind. Visual Studio 2010) installiert sein. Zum Starten des Programms kann im Ordner …\SwissTransportApplication\bin\Debug die exe-Datei ausgeführt werden oder man kann auch die sln-Datei im Visual Studio öffnen und mit F5 starten.

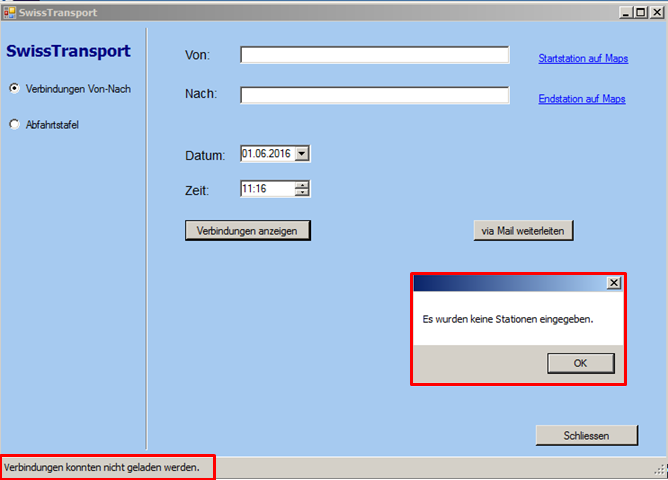
## Gmail-Account

Um Verbindungen per Email (A008) zu versenden, ist ein Gmail-Account erforderlich, da der SMTP-Server von Gmail genommen wurde.

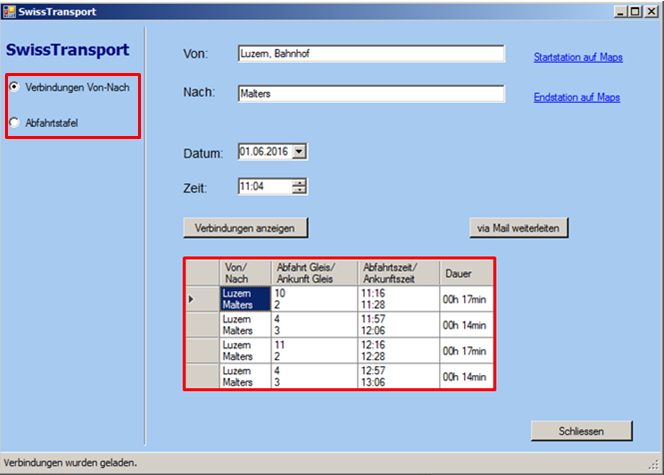
# Endprodukt

## A001 + A004

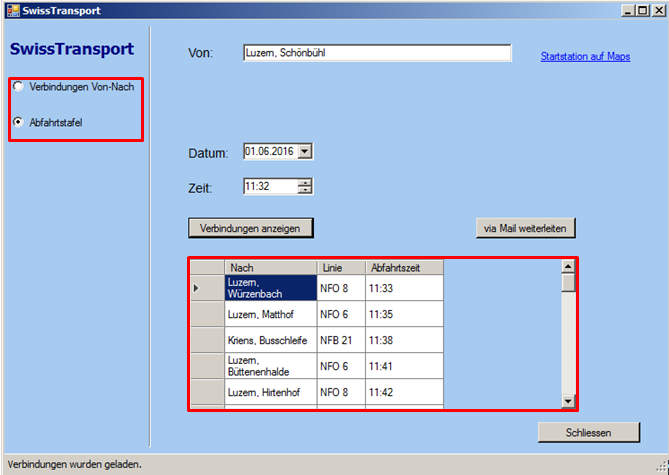
Die Anforderung A001 wurde mit der Anforderung A004 kombiniert. Sobald der Benutzer mehr als 3 Zeichen eingibt, erscheint ein Autocomplete-Dropdown. Somit kann er die verfügbaren Stationen zu seiner Eingabe sehen und muss sie nicht auswendig lernen.

Falls keine Stationen bzw. nur eine Station eingegeben wurden, erscheint eine Fehlermeldung.

## A002

Um die Verbindungen zwischen zwei Stationen zu sehen, muss der „Verbindungen Von-Nach“ Radiobutton selektiert sein. Der User kann eine Start- und Endstation eingeben. Sobald auf „Verbindungen anzeigen“ gedrückt wird, erscheint eine Tabelle mit den aktuellen Verbindungen. Dabei wird zusätzlich das Abfahrts- und Ankunftsgleis, die Abfahrts- und Ankunftszeit und die Dauer der Fahrt angegeben.

## A003

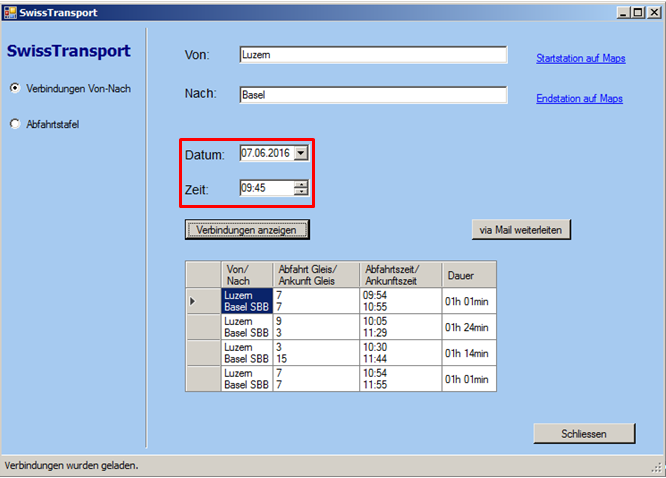


Um die Verbindungen ab einer bestimmten Station anzuzeigen, muss der „Abfahrtstafel“ Radiobutton selektiert sein. Dann kann die Startstation eingegeben werden, von welcher man die Abfahrten sehen möchte. Sobald auf „Verbindungen anzeigen“ gedrückt wird, erscheint eine Tabelle wobei noch die Linie und Abfahrtszeit angegeben wird.

## A004

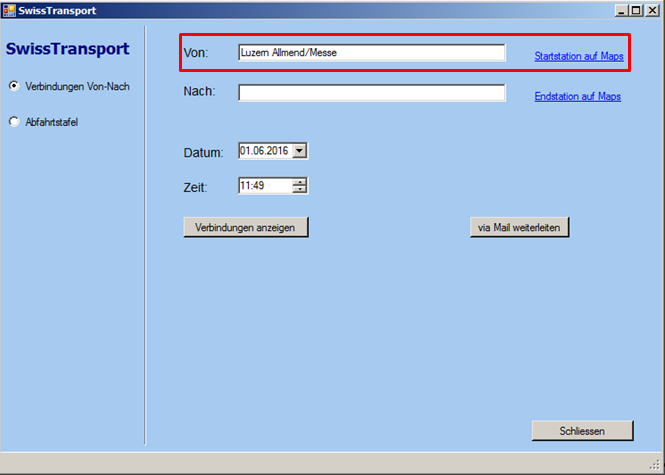
Die Autocomplete-Funktion wurde bereits im [Kapitel 7.1 A001 + A004](#_A001_+_A004) beschrieben.

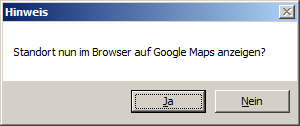
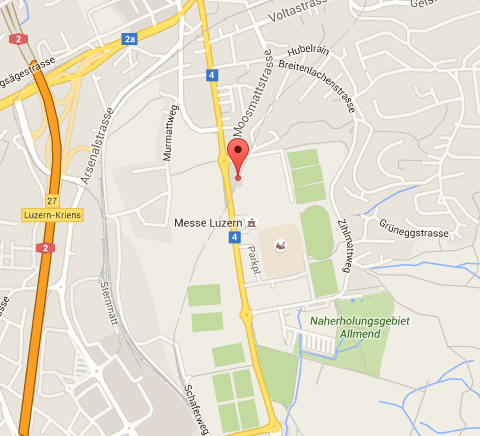
## A005



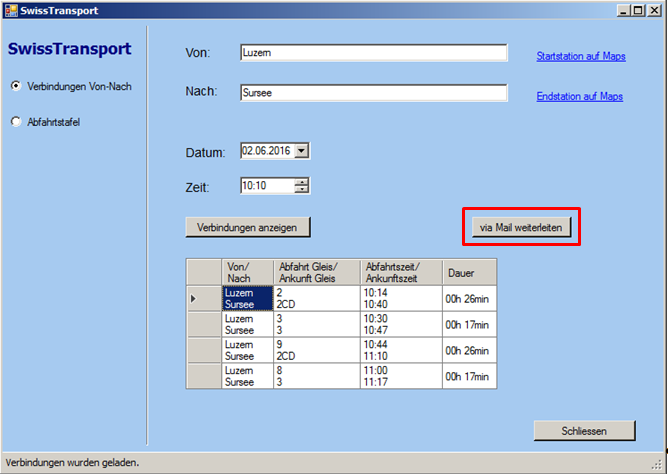
Um zukünftige Reisen zu planen, können nach der Eingabe der Stationen das Datum und die Zeit bestimmt werden. Es sind nur zukünftige Daten möglich, die Zeit ist beliebig wählbar.

## A006



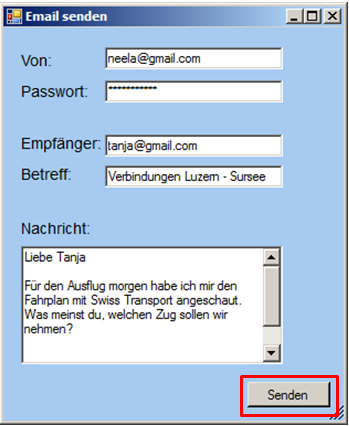
Um zu sehen wo sich eine Station befindet, muss eine Station eingegeben werden. Danach kann je nach dem auf „Start-„ oder „Endstation auf Maps“ gedrückt werden. Es öffnet sich ein Dialogfenster, worin der Benutzer gefragt wird, ob er den Standort im Standartbrowser auf Google Maps öffnen möchte. So wird der Standort im Google Maps mit dem Standartbrowser geöffnet.

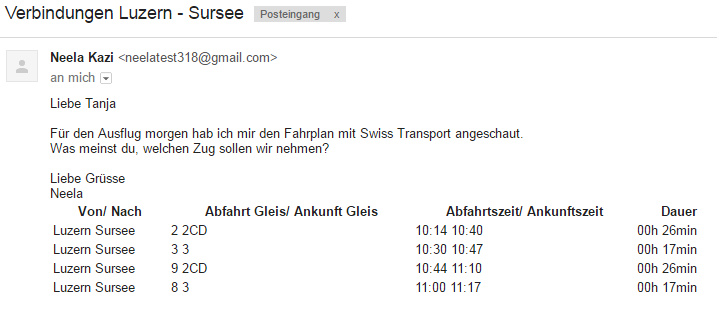
## A007

Die Anforderung A007 wurde aus zeitlichen Gründen nicht mehr umgesetzt. Dafür wurde A008 programmiert, da diese weniger Zeit in Anspruch nahm.

## A008

Wenn Verbindungen zwischen zwei Stationen, oder Abfahrten ab einer Station, kann der „via Mail weiterleiten“ Button gedrückt werden. Somit öffnet sich ein neues Fenster womit man sich mit seinem Gmail Account anmelden kann und einen Empfänger und Betreff eingeben kann. Zusätzlich kann dem Empfänger noch eine Nachricht hinterlassen werden.



Sobald die Mail verschickt wurde, erscheint eine Meldung, dass die Mail erfolgreich versandt wurde. Andernfalls erscheint eine Fehlermeldung (z.B. falsches Passwort).

# Reflexion

## Was war gut?

Die ersten 4 Anforderungen konnten schnell umgesetzt werden. Die API Dokumentation war gut verständlich. Im allgemeinen war es ein gut machbares Projekt und für unser Niveau ideal.

## Verbesserungsmöglichkeiten

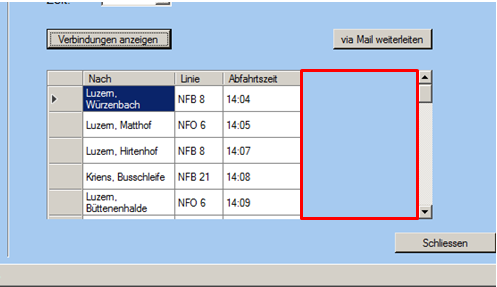
### AutoComplete verschnellern

Die AutoComplete-Funktion ist manchmal etwas langsam. Man könnte dies verbessern indem man die API mit async umschreibt. Jedoch wurde dies aus zeitlichen Gründen nicht mehr umgesetzt.

### Mail Tabellendarstellung verschönern

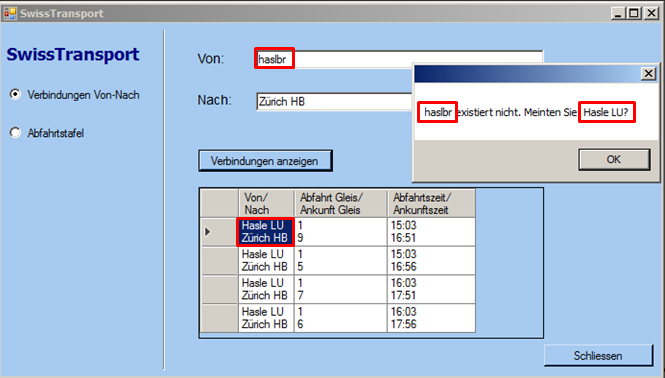
Die Tabelle mit den Verbindungen wird nicht sehr schön dargestellt und ist zum Teil etwas schwierig zu lesen. Dies könnte man mit HTML noch anpassen, wurde jedoch auch aus zeitlichen Gründen nicht gemacht.

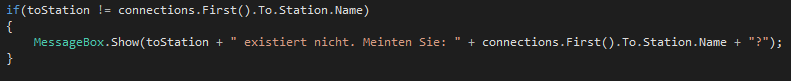
### DataGridView in Abfahrtstafel anpassen

Der Border bei der DataGridView könnte angepasst werden, zwischen der Scrollbar und den Daten ist ein gewisser Abstand. Versucht wurde es mit AutoSize-Befehlen, jedoch erfolglos. Eine andere Möglichkeit wäre gewesen, die Breite der DataGridView manuell anzupassen.

### Zufällige Eingabe

Wenn irgendwelche Buchstaben z.B. in die Startstation Textbox geschrieben werden, wird diese Station die am meisten zur Eingabe zutrifft genommen, was eigentlich eine praktische Funktion ist. Ich wollte, falls dies passiert, eine Dialogbox erscheinen lassen wie z.B:





Jedoch war das Problem dabei, dass teilweise Fehler erkannt wurden, wenn man z.B. „Zürich“ statt „Zürich HB“ eintippte, was das ganze wieder benutzerunfreundlich machte.

## Erkenntnisse

Bei diesem Projekt habe ich allgemein mehr über objektorientiertes Programmieren, vor allem aber auch über die Programmierung von WinForms gelernt. Ich kenne jetzt verschiedene Komponenten und weiss wie und wo man sie einsetzen kann.