Jonas Hodel, Kursraum 4

Dies ist die Dokumentation von Jonas Hodel vom ÜK 318 Analysieren und objektbasierte programmieren mit Komponenten. Während diesem ÜK haben wir die Applikation ÖV Applikation entwickelt. In der Dokumentation ist der Beschrieb der Applikation, Diagramme und die Testfälle vorhanden.

ÖV Applikation Dokumentation

[Geben Sie den Untertitel des Dokuments ein]

Inhalt

[Anforderungen 3](#_Toc452558951)

[Verbindungssuche 3](#_Toc452558952)

[Abfahrtsplan 3](#_Toc452558953)

[Zusatz Funktionen 3](#_Toc452558954)

[Suche Per Adresse 3](#_Toc452558955)

[Station auf Karte Anzeigen 3](#_Toc452558956)

[Combobox Auswahl 3](#_Toc452558957)

[Was wurde Umgesetzt 4](#_Toc452558958)

[Installationsanleitung 4](#_Toc452558959)

[Use Case Diagramm 5](#_Toc452558960)

[Verbindung Suchen 5](#_Toc452558961)

[Aktivitäts Diagramm 6](#_Toc452558962)

[Verbindungen Laden 6](#_Toc452558963)

[Station auf Map anzeigen 7](#_Toc452558964)

[Klassen Diagramm 8](#_Toc452558965)

[Analyse Klassen Diagramm 8](#_Toc452558966)

[Programmbeschrieb 8](#_Toc452558967)

[Verbindungssuche 8](#_Toc452558968)

[Abfahrtsplan 9](#_Toc452558969)

[Erkannte Fehler 10](#_Toc452558970)

[Listview 10](#_Toc452558971)

[Window Size 10](#_Toc452558972)

[Abfahrtszeit und Abfahrtsdatum 10](#_Toc452558973)

[Abfahrtsplan Kategorie und Anbieter 10](#_Toc452558974)

[Email 10](#_Toc452558975)

[Unschöner Code 11](#_Toc452558976)

[HartCodierter String 11](#_Toc452558977)

[Validation Error Messages 11](#_Toc452558978)

[Methoden 11](#_Toc452558979)

[Validation in View 11](#_Toc452558980)

[Methoden ohne Aufrufe 11](#_Toc452558981)

[OeV\_Application Klasse 11](#_Toc452558982)

[Variabel, Properties und Controls Namen 11](#_Toc452558983)

[Testfälle 12](#_Toc452558984)

[Voraussetzungen 12](#_Toc452558985)

[A001 12](#_Toc452558986)

[Anforderung Beschrieb: 12](#_Toc452558987)

[Testfälle: 12](#_Toc452558988)

[A002 14](#_Toc452558989)

[Anforderung Beschrieb: 14](#_Toc452558990)

[Testfälle: 14](#_Toc452558991)

[A003 15](#_Toc452558992)

[Anforderung Beschrieb: 15](#_Toc452558993)

[Testfälle: 15](#_Toc452558994)

[A005 16](#_Toc452558995)

[Anforderung Beschrieb: 16](#_Toc452558996)

[Testfälle: 16](#_Toc452558997)

[A008 18](#_Toc452558998)

[Anforderung Beschrieb: 18](#_Toc452558999)

[Testfälle: 18](#_Toc452559000)

[Validation Testfälle 20](#_Toc452559001)

[Weitere Testfälle 35](#_Toc452559002)

[Reflexion 35](#_Toc452559003)

# Anforderungen

## Verbindungssuche

Das Programm besteht aus zwei Grundfunktionen die einte Funktion ist, dass heraussuchen von Verbindungen zwischen zwei Stationen. Dabei kann der Benutzer die Auswahl der Stationen über die Comboboxen ausführen. Die Suchanfrage kann nur ausgeführt werden, wenn eine Station mit dem gleichen Namen vorhanden ist. Ist der Name nicht vorhanden, muss der Benutzer eine Station aus der Auswahl auswählen. Nach dem die Stationen erfasst wurde, kann der Benutzer definieren, an welchem Datum er reisen möchte. Dies macht einen Unterschied, da im Wochenende und an Feiertagen die Öffentlichen Verkehrsmittel anders fahren als an normalen Wochentagen. Dazu kann der Benutzer definieren, um welche Uhrzeit er Reisen möchte. Dazu kann er bestimmen, ob er um eine Bestimmte Uhrzeit am Zielort ankommen möchte oder zu einem bestimmten Zeitpunkt am Abfahrtsort Abfahren. Die gefundenen Verbindungen werden dann angezeigt. Dem Benutzer sollte klar ersichtlich sein, wenn eine Verbindung verspätet ist.

## Abfahrtsplan

Die zweite Grundfunktion ist, dass man alle Abfahrten von einem Standort abfragen kann. Dazu muss der Benutzer wieder eine Station aussuchen. Er kann dies per Texteingabe in der Combobox oder per Auswahl der Combobox tun. Ist der eingegebene Text in der Auswahl nicht vorhanden, muss der Benutzer einen Wert aus der Combobox auswählen. Der Benutzer sollte sehen, auf welchem Peron das Öffentliche Verkehrsmittel abfährt, welche Art es ist. Hier sollte auch wieder klar ersichtlich sein, wenn die Verbindung Verspätung hat.

# Zusatz Funktionen

## Suche Per Adresse

Bei der Suche in der Combobox ist es mögliche eine Adresse einzugeben. Dabei ist zu beachten, man Ortschaft, Strasse und Hausnummer angeben muss. Wird die Adresse bei der direkten Eingabe nicht erkannt, so muss man noch die Auswahl in der Combobox treffen. Bei der Eingabe der Adresse, wird vom API direkt die nächst gelegenen Stationen, für die Suche ausgewählt. So können Resultate von mehreren Stationen kommen.

## Station auf Karte Anzeigen

Nach dem man bei der Combobox eine Auswahl getroffen hat, kann der Benutzer mit dem Klick auf den Button “i“, die Station auf der Karte anzeigen lassen. Es wäre noch schön, wenn der Benutzer z.B noch drei unterschiedliche Stationen in der Combobox hat, dass dann diese drei auf der Karte angezeigt werden. Somit kann der Benutzer eine bessere Auswahl der einzelnen Stationen treffen. Sehr toll wäre es noch, wenn die Aktuelle Position der IP Adresse auf der Karte auch noch dargestellt werden würde.

## Combobox Auswahl

Wenn man in der Combobox einen Suchtext eingibt, werden die Vorschläge in die Auswahl der Combobox geladen. Es wäre noch schön, wenn immer der beste Vorschlag als AutoComplete Vorschlage genommen werden würde. Dies würde die Eingabe des Benutzers noch vereinfachen.

# Was wurde Umgesetzt

Die Grundanforderungen, Aufgabe A001, A002 und A003 wurden umgesetzt. Aufgabe A004, die AutoComplete Funktion wurde nicht umgesetzt, da diese Implementation die Performance des Programmes noch mehr beeinträchtigt hat. Darum wurde diese Implementation wieder entfernt. Die Aufgabe A005, das Suchen von Verbindungen zu einem anderen Zeitpunkt wurde umgesetzt. Es ist nun möglich Verbindungen und Abfahrtstafeln zeitverschoben zu suchen. Die Aufgaben A006 und A007 konnte ich nicht umsetzten, da mir für diese Aufgaben das KnowHow und die Zeit fehlte. Es ist ein gewisser Teil im Programm vorhanden. Dazu kann man hinter der Station auf den Button “i“ Klicken. Dann wird ein neues Fenster geöffnet, das den Standort der Station anzeigen würde. Die Aufgabe A007 wurde nicht vollständig umgesetzt. Das Programm erkennt nicht von alleine deine Position und sucht somit auch nicht direkt nach der nächsten Station. Es ist aber möglich eine Adresse in das Suchfeld einzugeben. Dann gibt das API die Stationen zurück, die am nächsten sind. Die Aufgabe A008 wurde umgesetzt. Es ist möglich eine Verbindung an mehrere Personen per Mail zu senden. Zu beachten ist, dass die SmtpClient Verbindung hart codiert ist.

|  |  |  |
| --- | --- | --- |
| **Aufgabe** | **Implementiert** | **Kommentar** |
| A001 | X | Wurde vollständig implementiert. |
| A002 | X | Wurde vollständig implementiert. |
| A003 | X | Wurde vollständig implementiert. |
| A004 | 0 | Wurde nicht implementiert. |
| A005 | X | Wurde vollständig implementiert. |
| A006 | - | Arbeiten konnten nicht abgeschlossen werden, da die Zeit und das KnowHow fehlen. |
| A007 | - | Arbeiten konnten nicht abgeschlossen werden, da die Zeit und das KnowHow fehlen. |
| A008 | X | Wurde vollständig implementiert. |

# Installationsanleitung

Die Exe Datei ist auf dem Repository unter dem Namen OeV\_Application.exe vorhanden. Diese Exe Datei kann ausgeführt werden.



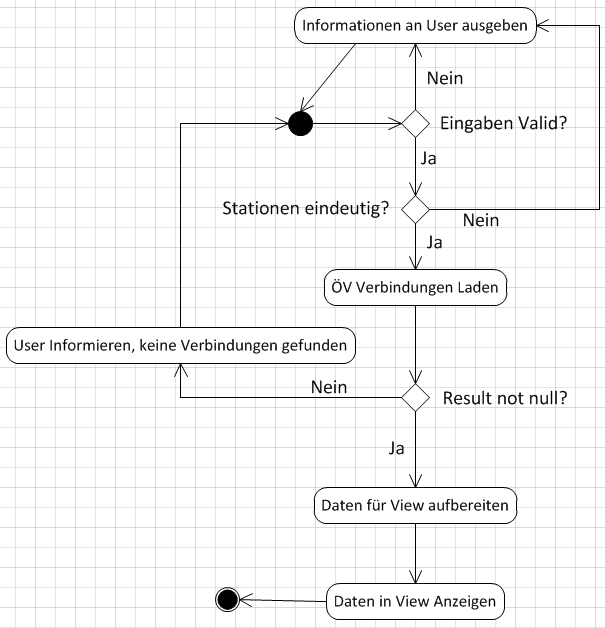
# Use Case Diagramm

## Verbindung Suchen

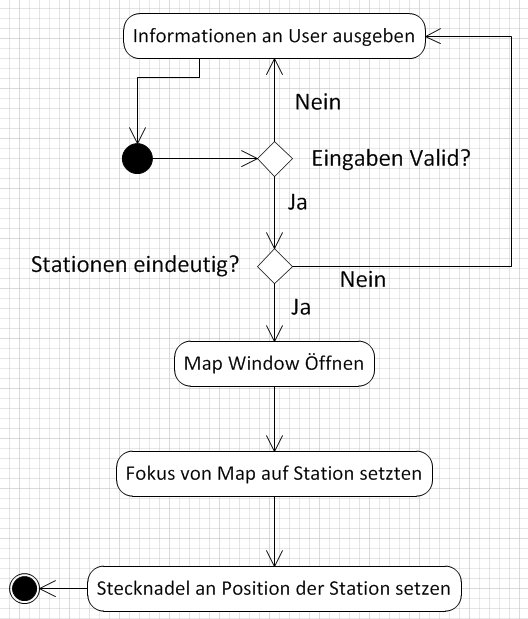
## Station auf Map anzeigen

# Aktivitäts Diagramm

## Verbindungen Laden



## Station auf Map anzeigen



# Klassen Diagramm

Link zum Klassen Diagramm: <https://creately.com/diagram/iowrgxpn1> .   
Sollte es nicht möglich sein, den Link zu öffnen, ist noch ein Bild vom Diagramm im Ordner vorhanden.

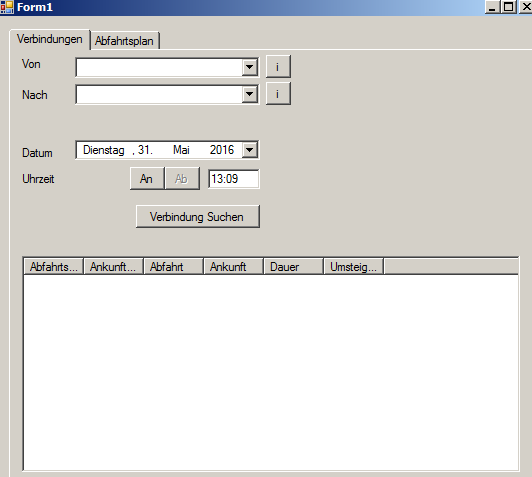
## Analyse Klassen Diagramm

Beim Klassen Diagramm sieht man bei den Klassen für das Senden der Mail, dass wir “eine Kreisstruktur“ haben. Wenn man das Diagramm im Voraus erstellt hätte, so hätte man diese Klassen anders aufbauen können. Es macht keinen Sinn, dass von der View Klasse zuerst einen Aufruf an den MailBuilder geht und danach noch ein Aufruf an den Mailsender. Ich würde den Programmcode so abändern, dass der Aufruf von der View nur an den Sender geht. Die Sender Klasse macht danach einen Aufruf an die MailBuilder Klasse. Mit diesem Konstrukt hätte man einen viel schöneren Aufbau.

# Programmbeschrieb

## Verbindungssuche

Stationsname Abfahrt



Verbindungsresultate Listview

Ist Datum und Uhrzeit für Abfahrt oder Ankunft

Uhrzeit der Verbindungssuche

Datum der Verbindungssuche

Stationsname Ankunft

Station auf Karte anzeigen

#### An / Ab Button

Es sind auf der View zwei Buttons vorhanden, die noch weitere Informationen benötigen. Dies ist der An und Ab Button. Als erste Information, die beiden Buttons stehen in einer Verbindung zu einander. Es kann nur immer ein Button von beiden enabled sein. Wenn der Button Ab disabled ist, ist definiert, dass das Datum und die Uhrzeit für den Abfahrtszeitpunkt definiert sind. Klickt man nun auf den Button an, ist nun der Button Ab enabled und der Button An disabled und das Datum und die Uhrzeit ist nun für den Abfahrtzeitpunkt definiert.

#### I Button

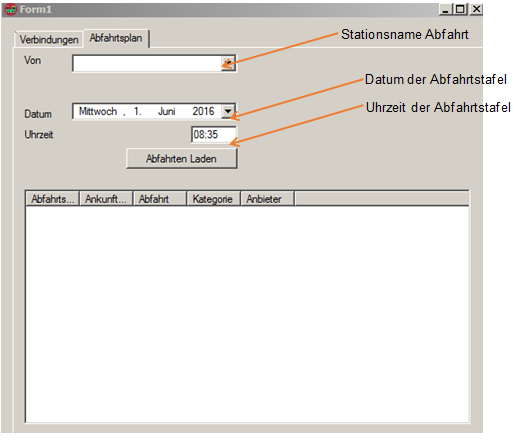
Mit dem Klick auf den i Button sollte die Station auf der Karte angezeigt werden. Dies wurde aber noch nicht vollständig ausgeführt. Zum jetzigen Zeitpunkt, wird der Fokus(rotes Kreutz) der Karte auf die Station gesetzt. Ich möchte aber noch eine Stecknadel Setzen. Damit man sieht, an welcher Position sich die Station befindet. Auch wenn man den Focus der Karte verändert. Zum jetzigen Zeitpunkt(31.05.2016 13:15) ist dies aber noch nicht implementiert.

## Abfahrtsplan

Uhrzeit der Abfahrtstafel

Datum der Abfahrtstafel

Stationsname Abfahrt



# Erkannte Fehler

## Listview

Die ListviewColumn Width ist “dynamisch“ aufgebaut. Die Width wird der breite des Headers jeweils angepasst. Dabei ist aber das Problem, dass die letzte Column die Width “\*“ hat. Somit ist die letzte Column zu breit. Ein weiteres Problem ist, wenn wir nun einen langen Abfahrtsort oder Ankunftsort haben, ist dieser nicht vollkommen sichtbar. Es würde die Möglichkeit geben, dass man die Width den Items anpasst, dann ist aber bei der Abfahrtszeit der Header nicht mehr vollständig sichtbar. Es würde sicherlich eine Einstellung geben, bei der man definieren kann, dass alle Werte schön Dargestellt werden. Dafür reicht mir aber die Zeit nicht mehr.

## Window Size

Wenn man die Grösse des Fensters verändert, sieht die Applikation nicht mehr schön aus.

## Abfahrtszeit und Abfahrtsdatum

Die Abfahrtszeit und das Abfahrtsdatum werden nicht mehr aktualisiert. Diese Werte werden beim Programmstart in die Textbox geschrieben. Wenn man die Applikation nun länger offen hat, kann es zu sehr grossen Zeitdifferenzen kommen. Dieses Problem wäre am einfachsten mit einem Timmer zu lösen. Dazu reichte mir aber die Zeit nicht mehr.

## Abfahrtsplan Kategorie und Anbieter

Beim Abfahrtsplan werden die Kategorie und der Anbieter mit den “Codes“ angezeigt. Es wäre schön gewesen, wenn man den Voll ausgeschriebenen Namen anzeigen könnte. Bei der Kategorie wäre dies sehr gut möglich gewesen. Dazu hätte man eine List vom Typen KeyValuePair erstellen können. Bei den Anbietern wäre es schwieriger geworden, da es sehr viele unterschiedliche Anbieter gibt. Eventuell würde es dazu einen weiteren WebService geben, den man ansprechen kann um den Anbieter auszulesen. Dazu reichte aber die Zeit nicht mehr.

## Email

Es ist eine Hardcodierte Email Adresse von der Applikation([OeV.Application@gmx.ch](mailto:OeV.Application@gmx.ch)) implementiert. Alle Mails werden über diese Mail Adresse versendet. Es wäre schön, wenn der User seinen eigenen Mail Account angeben könnte. Dies könnte man über einen Menu Punkt machen, bei dem der User seine Settings eingeben könnte. Oder über ein Conifg File.

## Unschöner Code

### HartCodierter String

Es sind im Code hart Codierte String vorhanden. Zum einten sind Strings eingebaut, mit Hilfe von denen das Programm herauslesen kann, um welchen Button oder Listview es sich handelt. Diese Codierung ist sehr unschön und sehr Fehler anfällig. Denn ich prüfe den Namen des Controls mit dem hart Codiertem String. Ändert sich nun den Namen des Controls, müsste man den hart codierten string auch anpassen. Dies muss man aber wissen. Ob man dies in einem Monat noch weiss, wenn man wieder Anpassungen am Code mach ist aber fraglich. Aus zeitlichen Gründen habe ich aber diese Methode, zum definieren des Controls gewählt.

Zum anderen habe ich hart codierte Strings beim Erstellen des Mails. Mit Hilfe von diesen Strings, erstelle ich den Text und den Betreff des Mails. Es wäre schöner gelöst, wenn man diesen String in einem Ressource File hinterlegen würde. Dann könnte man den String einfacher verändern.

### Validation Error Messages

Beim Ausgeben der Meldungen, wenn invalide Daten eingegeben wurden, hat man mehrmals den gleichen String im Code. Dieses Strings könnte man auch in ein Ressource File auslagern. Damit der string nur einmal vorhanden ist.

### Methoden

Der Aufruf der Daten ist nicht immer gleich aufgebaut. Manchmal wird der Aufruf der Daten direkt aus einem EventHandler gemacht. An einem anderen Ort passiert der Aufruf über eine Methode. Das gleiche ist beim Abfüllen der Daten in die View. Manchmal füllt der EventHandler die Daten in die View ab. Manchmal übernimmt dieser Task eine Methode.

Der Aufbau ist nicht immer gleich, somit wird es auch schwieriger zum verstehen des Codes.

### Validation in View

Die Validation der Eingaben erfolgt in der View. Dies ist nicht schön gelöst, da in der View keine Logik sein sollte. Bei einer WindowsForms Applikation ist aber meistens gewisse Logik in der View.

### Methoden ohne Aufrufe

Es kann sein, dass im Programmcode Methoden vorhanden sind, die nicht aufgerufen werden. Leider hatte ich keinen Resharper oder Ähnliches installiert. Somit ist es nicht so einfach, zu erkennen, welche Methoden benutzt werden und welche nicht.

### OeV\_Application Klasse

In der OeV\_Application Klasse(ab jetzt OeV Klasse genannt) ist nicht sauber codiert. Da wir in dieser Klasse zwei unterschiedliche Jobs ausführt. Zum einten ist sie für die Funktionalität der ConnectionSearchFunktion und der StationBoardFunktion vorhanden. Dies nicht sauber codiert, da jede Klasse nur eine bestimmte Funktion übernehmen sollte.

### Variabel, Properties und Controls Namen

Die Namen von den Variabeln, Properties und den Controls sind nicht immer gleich aufgebaut. So kann es zu Missverständnissen zu einem späteren Zeitpunkt kommen.

Alle diese Erkenntnisse sind vom 31.05.2016 22:30. In der Zwischenzeit kann sich wieder etwas verändert haben.

# Testfälle

## Voraussetzungen

Die Applikation wird gestartet. Dazu kann die .exe Datei aus dem Ordner ausgeführt werden. Zudem muss man eine Verbindung zur API von <transport.opendata.ch> haben. Ist diese Verbindung nicht vorhanden, können keine Daten geladen werden. Sonst können die Tests nicht durchgeführt werden.

## A001

### Anforderung Beschrieb:

Ich als ÖV-Benutzer möchte die Start und Endstation mittels Textsuche suchen können, damit ich nicht alle Stationsnamen auswendig lernen muss.

### Testfälle:

#### Testfall 1 : Bekannte Ortschaft

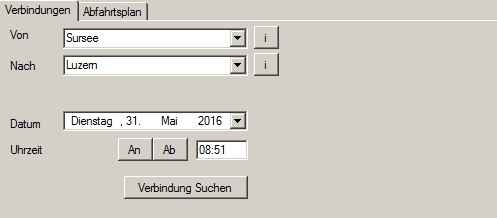
Geben sie in die Combobox für den Abfahrtsort, den Wert “Sursee“ ein.

Geben sie in die Combobox für den Ankunftsort, den Wert “Luzern“ ein.

Die Uhrzeit und Datum für den Abfahrtszeitpunkt, kann standartmässig übernommen werden.

Betätigen sie nun den Button Verbindung suchen.

**Erwartetes Resultat**: Die beiden Stationen Luzern und Sursee wurde gefunden, da diese Stationssuchen eindeutig sind. Bei Ortschaften mit einem Bahnhof wird standartmässig beim Ortsnamen der Bahnhof als Auswahl getroffen.



#### Testfall 2: Unbekannte Ortschaften

Geben sie in die Combobox für den Abfahrtsort, den Wert “Ettiswil“ ein.

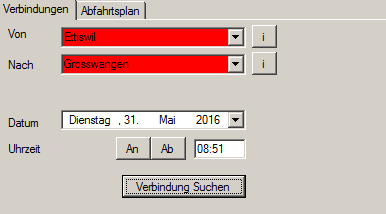
Geben sie in die Combobox für den Ankunftsort, den Wert “Grosswangen“ ein.

Die Uhrzeit und Datum für den Abfahrtszeitpunkt, kann standartmässig übernommen werden.

Betätigen sie nun den Button Verbindung suchen.

**Erwartetes Resultat**: Die beiden Comboboxen werden rot markiert und eine Fehlermeldung wird ausgegeben, da die Eingabe nicht eindeutig ist. Der Benutzer hat die Möglichkeit eine Station aus der Combobox Auswahl auszuwählen

.



#### Testfall 3: Unbekannte Ortschaften mit Haltestelle

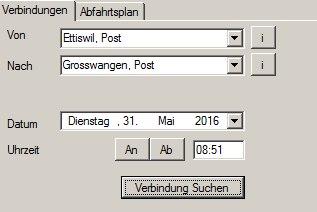
Geben sie in die Combobox für den Abfahrtsort, den Wert “Ettiswil, Post“ ein.

Geben sie in die Combobox für den Ankunftsort, den Wert “Grosswangen, Post“ ein.

Die Uhrzeit und Datum für den Abfahrtszeitpunkt, kann standartmässig übernommen werden.

Betätigen sie nun den Button Verbindung suchen.

**Erwartetes Resultat**: Die beiden Stationen Ettiswil, Post und Grosswangen, Post wurde gefunden, da diese Stationssuchen eindeutig sind.



## A002

### Anforderung Beschrieb:

Ich als ÖV-Benutzer möchte alle aktuellen Verbindungen zwischen den beiden gefundenen und ausgewählten Stationen sehen, damit ich weiss wann ich zur Station muss.

### Testfälle:

#### Testfall 1 : Verbindung Suche

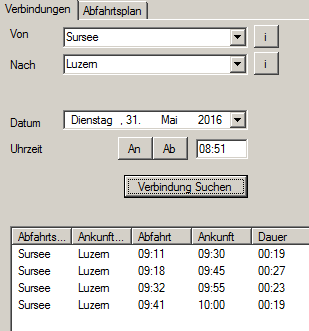
Geben sie in die Combobox für den Abfahrtsort, den Wert “Sursee“ ein.

Geben sie in die Combobox für den Ankunftsort, den Wert “Luzern“ ein.

Die Uhrzeit und Datum für den Abfahrtszeitpunkt, kann standartmässig übernommen werden.

Betätigen sie nun den Button Verbindung suchen.

**Erwartetes Resultat**: Es werden die ersten Vier Verbindungen zwischen Sursee und Luzern angezeigt.



## A003

### Anforderung Beschrieb:

Ich als ÖV-Benutzer möchte sehen welche Verbindungen ab einer bestimmten Station vorhanden sind. Damit ich eine Art Station Abfahrtstafel Zuhause haben kann.

### Testfälle:

#### Testfall 1 : Abfahrtstafel Suche

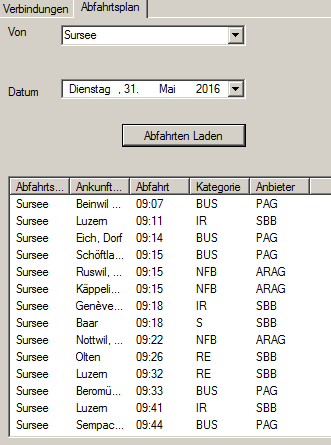
Wechseln sie in das Register Abfahrtsplan.

Geben sie in die Combobox für den Abfahrtsort, den Wert “Sursee“ ein.

Betätigen sie nun den Button Abfahrten Laden.

Das Datum für den Abfahrtszeitpunkt, kann standartmässig übernommen werden.

**Erwartetes** **Resultat:** Es werden die nächsten Verbindungen ab Sursee angezeigt. Sie sind nach der Abfahrtszeit sortiert, damit der Benutzer eine gute Übersicht hat.



## A005

### Anforderung Beschrieb:

Ich als ÖV-Benutzer möchte Verbindungen zu einem anderen Zeitpunkt suchen können, damit ich zukünftige Reisen planen kann.

### Testfälle:

#### Testfall 1 : Abfahrt 26.05.2016 um 8:15

Geben sie in die Combobox für den Abfahrtsort, den Wert “Sursee“ ein.

Geben sie in die Combobox für den Ankunftsort, den Wert “Luzern“ ein.

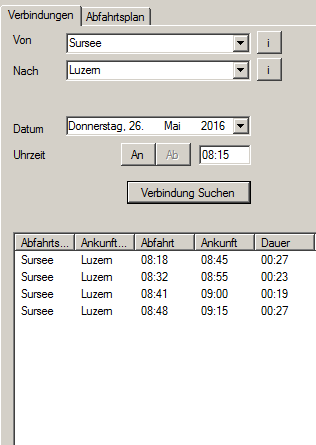
Wählen sie als Datum den 26.05.2016

Wählen sie als Uhrzeit 8:15

Betätigen sie den Button “Ab“

Betätigen sie nun den Button Verbindung suchen.

**Erwartetes Resultat**: Es werden die ersten Vier Verbindungen, die nach 8:15 abfahren, zwischen Sursee und Luzern angezeigt.



#### Testfall 2 : Ankunft 26.05.2016 um 8:15

Geben sie in die Combobox für den Abfahrtsort, den Wert “Sursee“ ein.

Geben sie in die Combobox für den Ankunftsort, den Wert “Luzern“ ein.

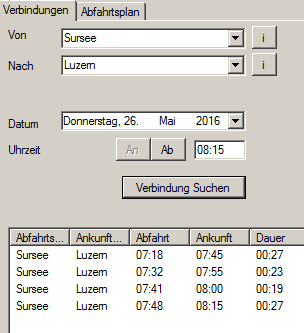
Wählen sie als Datum den 26.05.2016

Wählen sie als Uhrzeit 8:15

Betätigen sie den Button “An“

Betätigen sie nun den Button Verbindung suchen.

**Erwartetes Resultat**: Es werden die ersten Vier Verbindungen, die nach 8:15 in Sursee ankommen, zwischen Sursee und Luzern angezeigt.



## A008

### Anforderung Beschrieb:

Ich möchte meine gefundenen Resultate Via Mail weiter schicken können, damit ich oder andere von meinen Recherchen profitieren können.

### Testfälle:

#### Testfall 1: Verbindung per Mail senden

Geben sie in die Combobox für den Abfahrtsort, den Wert “Sursee“ ein.

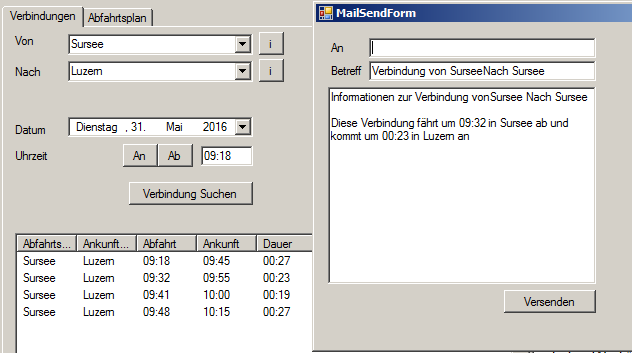
Geben sie in die Combobox für den Ankunftsort, den Wert “Luzern“ ein.

Die Uhrzeit und Datum für den Abfahrtszeitpunkt, kann standartmässig übernommen werden.

Betätigen sie nun den Button Verbindung suchen.

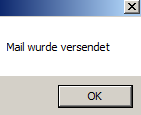
Machen sie einen Doppelklick auf eine Verbindung.

**Erwartetes Resultat**: Ein neues Fenster wird angezeigt. Der Betreff und der Text sind schon einmal abgefüllt.



#### Testfall 2: Mail senden

Geben sie in die Textbox den Wert „jonas.hodel@gmx.ch; TestEmail; jonas.hodel@gmx.ch“ ein.

Klicken sie auf den Button Versenden.

**Erwartetes Resultat**: Sie werden informiert, dass ein Mail versendet wurde.

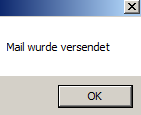
#### Testfall 3: Mail senden

Geben sie in die Textbox den Wert “jonas.hodel@gmx.ch“ ein.

Geben sie in den Betreff den Wert “Verbindung“ ein.

Ändern sie den Text ab.

Klicken sie auf den Button Versenden.

**Erwartetes Resultat**: Sie werden informiert, dass ein Mail versendet wurde. Überprüfen sie, dass der angepasste Betreff und Text gesendet wurde.

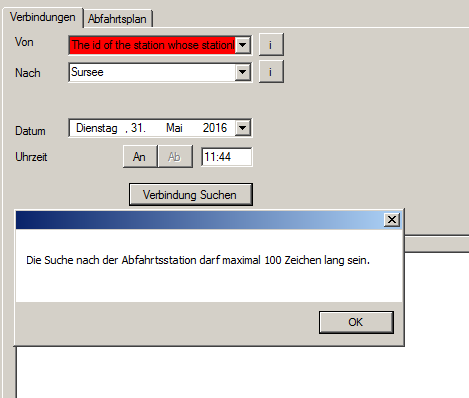
## Validation Testfälle

#### Testfall 1 : Abfahrt 100 Zeichen String Suche

Geben sie in die Combobox für die Abfahrt einen String von über 100 Zeichen ein.

Klicken sie auf den Button Versenden.

**Erwartetes Resultat**: Die Combobox wird rot markiert und sie werden durch ein Dialogfeld darauf hingewiesen, dass die Eingabe zu lang ist.

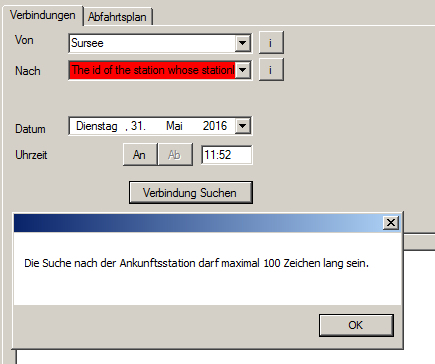


#### Testfall 2 : Ankunft 100 Zeichen String Suche

Geben sie in die Combobox für die Ankunft einen String von über 100 Zeichen ein.

Klicken sie auf den Button Versenden.

**Erwartetes Resultat**: Die Combobox wird rot markiert und sie werden durch ein Dialogfeld darauf hingewiesen, dass die Eingabe zu lang ist.



#### Testfall 4 : Mailsenden Recievers string.Empty

Geben sie in die Combobox für den Abfahrtsort, den Wert “Sursee“ ein.

Geben sie in die Combobox für den Ankunftsort, den Wert “Luzern“ ein.

Die Uhrzeit und Datum für den Abfahrtszeitpunkt, kann standartmässig übernommen werden.

Betätigen sie nun den Button Verbindung suchen.

Machen sie einen Doppelklick auf eine Verbindung.

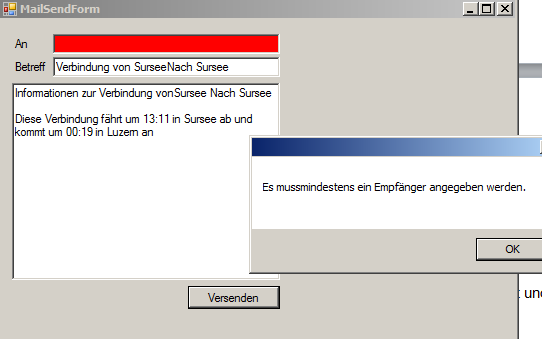
Keine Empfänger eingaben.

Der Betreff kann standartmässig übernommen werden.

Der Text kann standartmässig übernommen werden.

Klicken sie auf den Button Versenden.

**Erwartetes Resultat**: Die Textbox für die Eingabe der Empfänger wird rot markiert. Zudem wird der Benutzer auf den Fehler bei der Eingabe per Dialogfeld hingewiesen.



#### Testfall 5 : Mailsenden Recievers string zu lang

Geben sie in die Combobox für den Abfahrtsort, den Wert “Sursee“ ein.

Geben sie in die Combobox für den Ankunftsort, den Wert “Luzern“ ein.

Die Uhrzeit und Datum für den Abfahrtszeitpunkt, kann standartmässig übernommen werden.

Betätigen sie nun den Button Verbindung suchen.

Machen sie einen Doppelklick auf eine Verbindung.

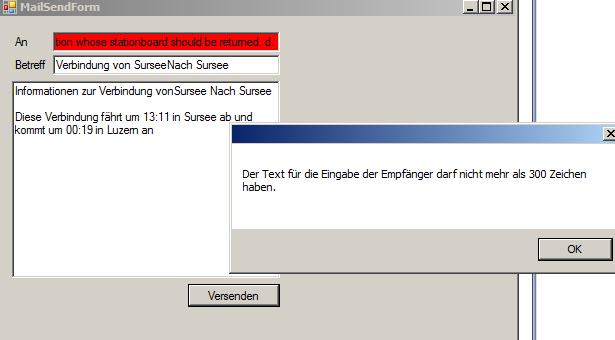
Geben sie in das Recievers Textfeld einen string ein, der länger als 300 Zeichen ist.

Der Betreff kann standartmässig übernommen werden.

Der Text kann standartmässig übernommen werden.

Klicken sie auf den Button Versenden.

**Erwartetes Resultat**: Die Textbox für die Eingabe der Empfänger wird rot markiert. Zudem wird der Benutzer auf den Fehler bei der Eingabe per Dialogfeld hingewiesen.



#### Testfall 6 : Mailsenden Recievers

Geben sie in die Combobox für den Abfahrtsort, den Wert “Sursee“ ein.

Geben sie in die Combobox für den Ankunftsort, den Wert “Luzern“ ein.

Die Uhrzeit und Datum für den Abfahrtszeitpunkt, kann standartmässig übernommen werden.

Betätigen sie nun den Button Verbindung suchen.

Machen sie einen Doppelklick auf eine Verbindung.

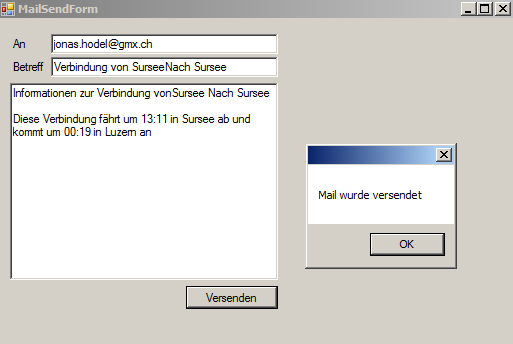
Geben sie in die Textbox der Empfänger den Wert “jonas.hodel@gmx.ch“ ein.

Der Betreff kann standartmässig übernommen werden.

Der Text kann standartmässig übernommen werden.

Klicken sie auf den Button Versenden.

**Erwartetes Resultat**: Das Mail wurde versendet



#### Testfall 7 : Mailsenden Recievers ungültige Email

Geben sie in die Combobox für den Abfahrtsort, den Wert “Sursee“ ein.

Geben sie in die Combobox für den Ankunftsort, den Wert “Luzern“ ein.

Die Uhrzeit und Datum für den Abfahrtszeitpunkt, kann standartmässig übernommen werden.

Betätigen sie nun den Button Verbindung suchen.

Machen sie einen Doppelklick auf eine Verbindung.

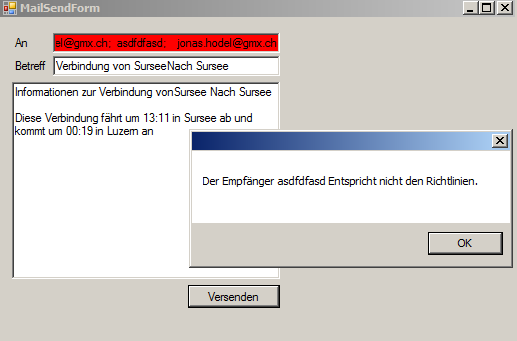
Geben sie in die Textbox der Empfänger den Wert “jonas.hodel@gmx.ch; asdfdfasd; jonas.hodel@gmx.ch“ ein.

Der Betreff kann standartmässig übernommen werden.

Der Text kann standartmässig übernommen werden.

Klicken sie auf den Button Versenden.

**Erwartetes Resultat**: Die Textbox für die Eingabe der Empfänger wird rot markiert. Zudem wird der Benutzer auf den Fehler bei der Eingabe per Dialogfeld hingewiesen.



#### Testfall 8 : Mailsenden Text string.Empty

Geben sie in die Combobox für den Abfahrtsort, den Wert “Sursee“ ein.

Geben sie in die Combobox für den Ankunftsort, den Wert “Luzern“ ein.

Die Uhrzeit und Datum für den Abfahrtszeitpunkt, kann standartmässig übernommen werden.

Betätigen sie nun den Button Verbindung suchen.

Machen sie einen Doppelklick auf eine Verbindung.

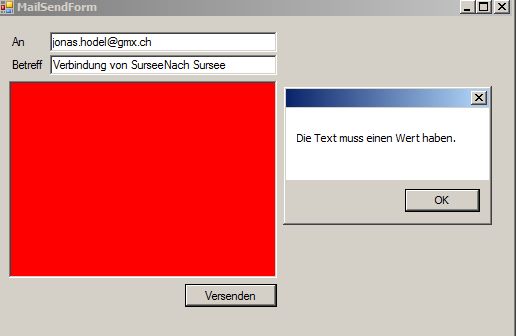
Geben sie in die Textbox der Empfänger den Wert “jonas.hodel@gmx.ch“ ein.

Der Betreff kann standartmässig übernommen werden.

Löschen sie den Text für das Mail.

Klicken sie auf den Button Versenden.

**Erwartetes Resultat**: Die RichTextbox für die Eingabe des Textes wird rot markiert. Zudem wird der Benutzer auf den Fehler bei der Eingabe per Dialogfeld hingewiesen.



#### Testfall 9 : Mailsenden string zu lang

Geben sie in die Combobox für den Abfahrtsort, den Wert “Sursee“ ein.

Geben sie in die Combobox für den Ankunftsort, den Wert “Luzern“ ein.

Die Uhrzeit und Datum für den Abfahrtszeitpunkt, kann standartmässig übernommen werden.

Betätigen sie nun den Button Verbindung suchen.

Machen sie einen Doppelklick auf eine Verbindung.

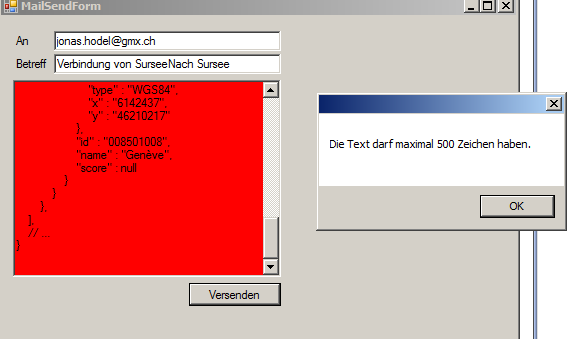
Geben sie in die Textbox der Empfänger den Wert “jonas.hodel@gmx.ch“ ein.

Der Betreff kann standartmässig übernommen werden.

Geben sie einen Text ein, der länger als 500 Zeichen ist.

Klicken sie auf den Button Versenden.

**Erwartetes Resultat**: Die RichTextbox für die Eingabe des Textes wird rot markiert. Zudem wird der Benutzer auf den Fehler bei der Eingabe per Dialogfeld hingewiesen.



#### Testfall 10 : Header string.Empty

Geben sie in die Combobox für den Abfahrtsort, den Wert “Sursee“ ein.

Geben sie in die Combobox für den Ankunftsort, den Wert “Luzern“ ein.

Die Uhrzeit und Datum für den Abfahrtszeitpunkt, kann standartmässig übernommen werden.

Betätigen sie nun den Button Verbindung suchen.

Machen sie einen Doppelklick auf eine Verbindung.

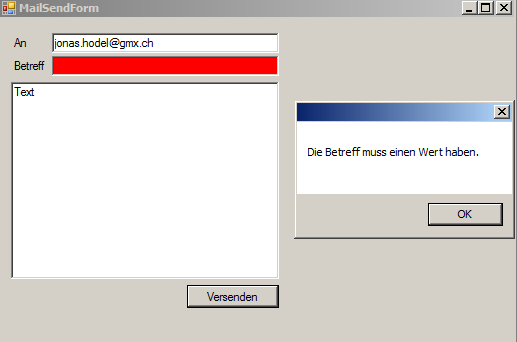
Geben sie in die Textbox der Empfänger den Wert “jonas.hodel@gmx.ch“ ein.

Löschen sie den Betreff.

Der Text kann standartmässig übernommen werden.

Klicken sie auf den Button Versenden.

**Erwartetes Resultat**: Die Textbox für die Eingabe des Header wird rot markiert. Zudem wird der Benutzer auf den Fehler bei der Eingabe per Dialogfeld hingewiesen.



#### Testfall 11: Header zu lang

Geben sie in die Combobox für den Abfahrtsort, den Wert “Sursee“ ein.

Geben sie in die Combobox für den Ankunftsort, den Wert “Luzern“ ein.

Die Uhrzeit und Datum für den Abfahrtszeitpunkt, kann standartmässig übernommen werden.

Betätigen sie nun den Button Verbindung suchen.

Machen sie einen Doppelklick auf eine Verbindung.

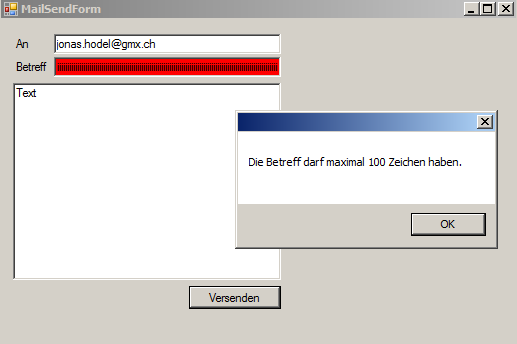
Geben sie in die Textbox der Empfänger den Wert “jonas.hodel@gmx.ch“ ein.

Geben sie einen Header ein, der länger als 100 Zeichen ist.

Der Text kann standartmässig übernommen werden.

Klicken sie auf den Button Versenden.

**Erwartetes Resultat**: Die Textbox für die Eingabe des Header wird rot markiert. Zudem wird der Benutzer auf den Fehler bei der Eingabe per Dialogfeld hingewiesen.



#### Testfall 12: Uhrzeit falsches Format

Geben sie in die Combobox für den Abfahrtsort, den Wert “Sursee“ ein.

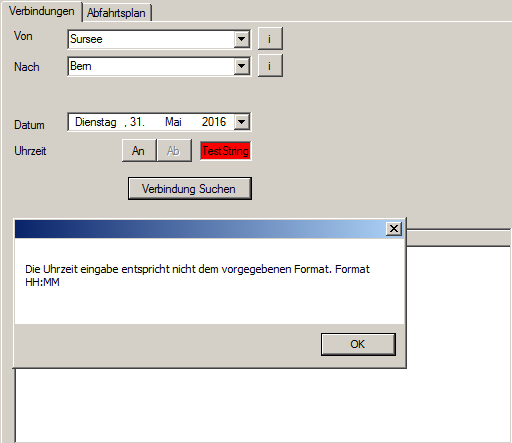
Geben sie in die Combobox für den Ankunftsort, den Wert “Luzern“ ein.

Das Datum kann standartmässig übernommen werden.

Geben sei den Wert “TestString“ für die Uhrzeit ein.

Betätigen sie nun den Button Verbindung suchen.

**Erwartetes Resultat**: Die Textbox für die Eingabe der Uhrzeit wird rot markiert. Zudem wird der Benutzer auf den Fehler bei der Eingabe per Dialogfeld hingewiesen.



#### Testfall 13: Uhrzeit Stunde nicht erlaubt

Geben sie in die Combobox für den Abfahrtsort, den Wert “Sursee“ ein.

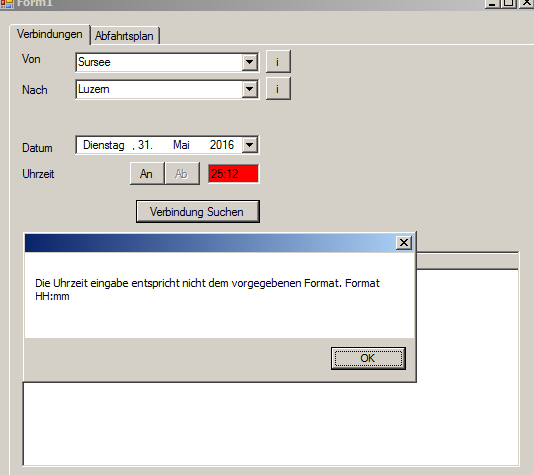
Geben sie in die Combobox für den Ankunftsort, den Wert “Luzern“ ein.

Das Datum kann standartmässig übernommen werden.

Geben sei den Wert “25:12“ für die Uhrzeit ein.

Betätigen sie nun den Button Verbindung suchen.

**Erwartetes Resultat**: Die Textbox für die Eingabe der Uhrzeit wird rot markiert. Zudem wird der Benutzer auf den Fehler bei der Eingabe per Dialogfeld hingewiesen.



#### Testfall 13: Uhrzeit Minute nicht erlaubt

Geben sie in die Combobox für den Abfahrtsort, den Wert “Sursee“ ein.

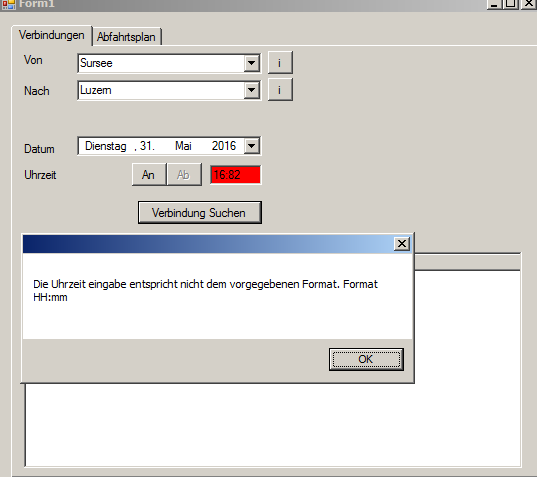
Geben sie in die Combobox für den Ankunftsort, den Wert “Luzern“ ein.

Das Datum kann standartmässig übernommen werden.

Geben sei den Wert “16:82“ für die Uhrzeit ein.

Betätigen sie nun den Button Verbindung suchen.

**Erwartetes Resultat**: Die Textbox für die Eingabe der Uhrzeit wird rot markiert. Zudem wird der Benutzer auf den Fehler bei der Eingabe per Dialogfeld hingewiesen.



#### Testfall 14: Uhrzeit falsches Format

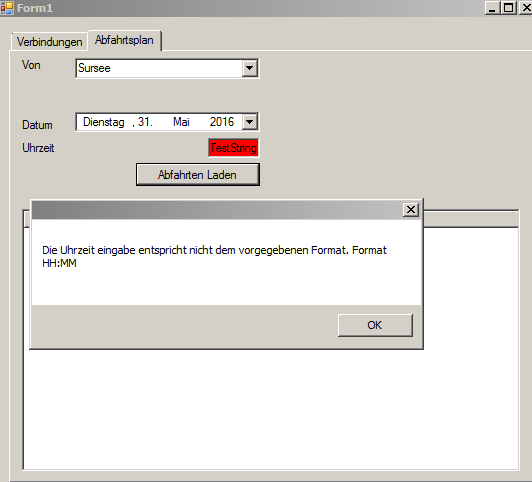
Geben sie in die Combobox für den Abfahrtsort, den Wert “Sursee“ ein.

Das Datum kann standartmässig übernommen werden.

Geben sei den Wert “TestString“ für die Uhrzeit ein.

Betätigen sie nun den Button Verbindung suchen.

**Erwartetes Resultat**: Die Textbox für die Eingabe der Uhrzeit wird rot markiert. Zudem wird der Benutzer auf den Fehler bei der Eingabe per Dialogfeld hingewiesen.



#### Testfall 15: Uhrzeit Stunde nicht erlaubt

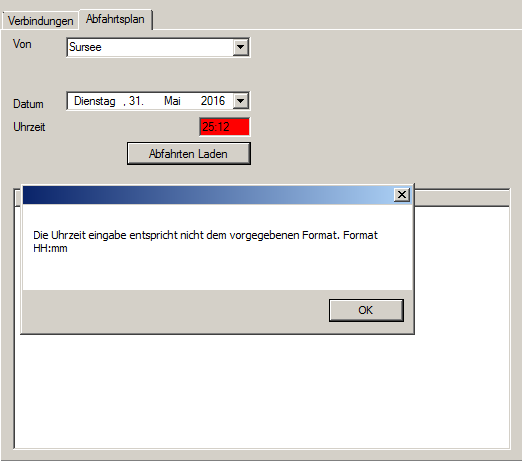
Geben sie in die Combobox für den Abfahrtsort, den Wert “Sursee“ ein.

Das Datum kann standartmässig übernommen werden.

Geben sei den Wert “25:12“ für die Uhrzeit ein.

Betätigen sie nun den Button Verbindung suchen.

**Erwartetes Resultat**: Die Textbox für die Eingabe der Uhrzeit wird rot markiert. Zudem wird der Benutzer auf den Fehler bei der Eingabe per Dialogfeld hingewiesen.



#### Testfall 16: Uhrzeit Minute nicht erlaubt

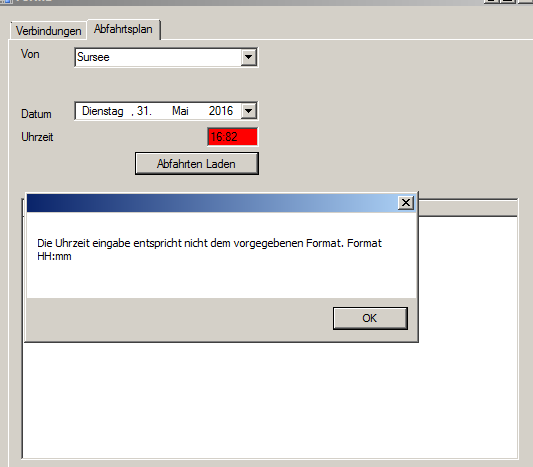
Geben sie in die Combobox für den Abfahrtsort, den Wert “Sursee“ ein.

Das Datum kann standartmässig übernommen werden.

Geben sei den Wert “16:82“ für die Uhrzeit ein.

Betätigen sie nun den Button Abfahrten suchen.

**Erwartetes Resultat**: Die Textbox für die Eingabe der Uhrzeit wird rot markiert. Zudem wird der Benutzer auf den Fehler bei der Eingabe per Dialogfeld hingewiesen.



## Weitere Testfälle

# Reflexion

Das Projekt war eine interessante Arbeit, da es eine nicht alltägliche Aufgabenstellung war. Normalerweise macht man bei solchen Themen immer eine Personenverwaltung oder ähnliches. Ich fand es auf der einten Seite sehr interessant, dass wir mit einer API gearbeitet haben. Da ich dies noch nie gemacht habe. Auf der anderen Seite war es aber etwas umständlich zum Arbeiten, da man immer auf die Antwort des API warten musste. Die Aufgabenstellungen waren sehr gut. Die Grundanforderungen konnte man sehr schnell und einfach lösen. Die weiteren Aufgaben waren dann schon etwas schwieriger. Dies fand ich aber auch gut, denn so hat man die Möglichkeit neue Komponenten kennen zu lernen.

Von den Aufgaben, die ich erledigen konnte, bin ich nicht 100% zufrieden. Ich hätte gerne noch die Aufgabe mit der GMap Komponente gemacht. Denn bis zum jetzigen Zeitpunkt kenne ich mich nicht gut mit der GMap Komponente aus. Dies wäre eine gute Möglichkeit gewesen, in diesem Bereich mein Wissen zu verbessern. Ich denke, dass ich dieses Wissen über die GMap Komponente zu einem Späteren Zeitpunkt sicherlich noch gut gebrauchen kann. Da man diese Komponente gut bei Personenverwaltungen einbinden kann und von diesen macht man in der Schule genug. Zudem hätte ich gerne noch bestimmte Funktionen eingebaut, diese die Arbeit des Benutzers erleichtern. Z.b dass man die Verbindungen mit Verspätung gut erkennt, eine AutoComplete Funktion(die nicht die Performance beeinträchtigt) oder Tastenkombinationen. Leider bin ich zu diesen Funktionalitäten nicht mehr gekommen, da mir dafür die Zeit nicht mehr reichte.

Mit dem geschriebenen Code bin ich auch nicht zufrieden. Ich habe mehrere Fehler im Aufbau der Applikation. Zum einten habe ich für mich, zu viel Logik in der View der Applikation. Zudem habe ich meinen Programmcode nicht immer gleich aufgebaut. Sei es mit dem Aufruf des Datenladen oder die Namen von den Variabeln. Eine andere Fehlentscheidung, die ich getroffen habe, ist dass ich mit zwei Registern gearbeitet habe. So hat man in einer View, Programmcode von zwei verschiedenen UseCases und die entspricht nicht den Definitionen von Clean Code. Das Error Handling finde ich noch ausbaufähig, da wir auf der Transport Klasse Error-Meldungen ausgeben. Ich würde es besser finden, wenn man diese Meldungen erst auf der View ausgeben würde.

Ich fand es sehr gut, dass wir die Test Cases für unser Programm definierten. Es war für mich sehr neu, dass ich die Test Cases selber definiere. In meinem Betrieb, bekomme ich die Test Cases von einem anderen Mitarbeiter zur Verfügung gestellt. Die Test Cases waren sehr praktisch zum Testen der Applikation, so konnten man sehnen, welche Funktionalitäten schon eingebaut sind und welche noch nicht. Ich konnte leider nicht für alle Funktionalitäten, die ich in die Applikation einbauen wollte einen Test Case schreiben. Die Test Cases, die bis jetzt erfasst sind, sollten erfolgreich getestet werden können.