**ÖV-Applikation**

Projektarbeit ÜK 318

|  |  |
| --- | --- |
| Inhalt | **Projektarbeit ÜK 318** |
| Erstelldatum | **Montag, 3. Dezember 2018** |
| Änderungsdatum | **Montag, 3. Dezember 2018** |
| Autor | **Bucher Luca [Bison Schweiz AG]** |

**Inhaltsverzeichnis**

[1 Einleitung 3](#_Toc531641514)

[2 Vorhersage 4](#_Toc531641515)

[2.1 Klassendiagramm 4](#_Toc531641516)

[2.2 Mockup 5](#_Toc531641517)

[3 Code Guidelines 7](#_Toc531641518)

[3.1 Klassennamen 7](#_Toc531641519)

[3.2 Variabel-Namen 7](#_Toc531641520)

[3.3 Methoden 8](#_Toc531641521)

[3.4 Geschweifte Klammern 8](#_Toc531641522)

[3.5 Files 8](#_Toc531641523)

[3.6 WinForms Controls 9](#_Toc531641524)

[4 Umgesetzte Anforderungen 10](#_Toc531641525)

[4.1 A001 10](#_Toc531641526)

[4.2 A002 10](#_Toc531641527)

[4.3 A003 10](#_Toc531641528)

[4.4 A005 10](#_Toc531641529)

[5 Use Cases 11](#_Toc531641530)

[5.1 Diagramm 11](#_Toc531641531)

[5.2 Beschreibungen 11](#_Toc531641532)

[5.2.1 Station auswählen 11](#_Toc531641533)

[5.2.2 Verbindung suchen 12](#_Toc531641534)

[5.2.3 Abfahrten anzeigen 12](#_Toc531641535)

[6 Testfälle 13](#_Toc531641536)

[6.1 UnitTests 13](#_Toc531641537)

[6.1.1 searchConnections() 13](#_Toc531641538)

[6.1.2 searchStationboards() 13](#_Toc531641539)

[6.2 Manuelle Tests 13](#_Toc531641540)

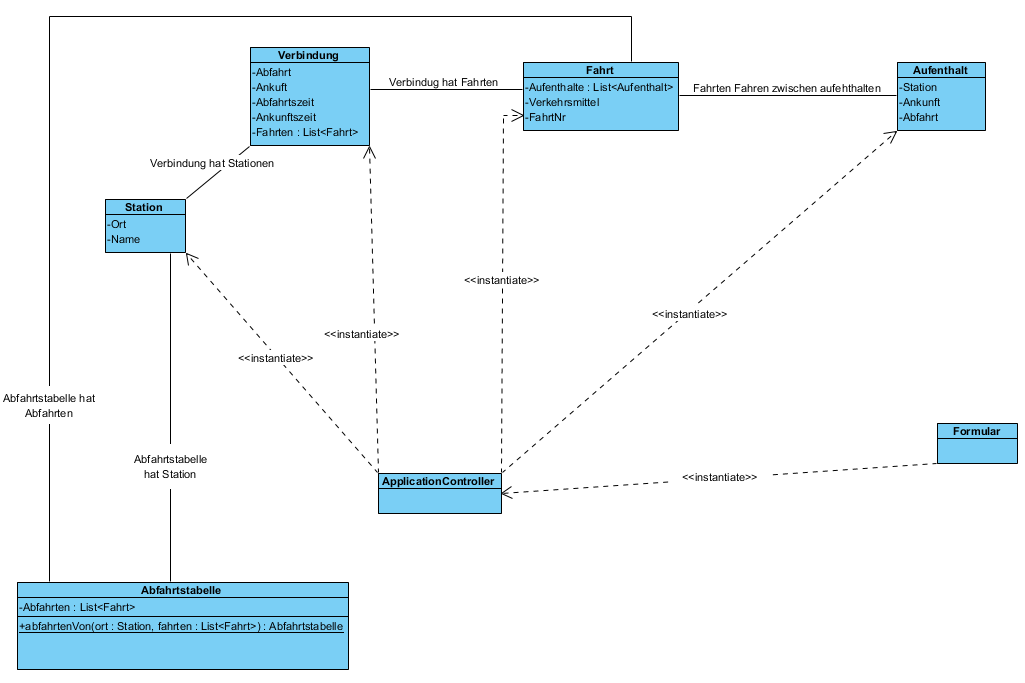
[7 Installationsanleitung 14](#_Toc531641541)

# Einleitung

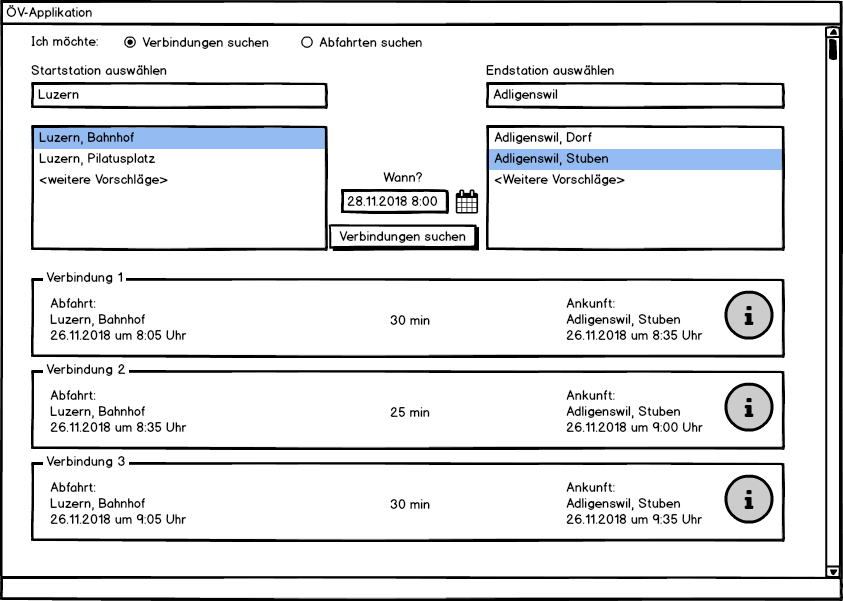
In diesem Projekt geht es darum, ein Projekt basierend auf gegebenen Anforderungen zu führen. Das Ziel ist es, ein Windows-Programm zu entwickeln, das mindestens die ersten 3 Anforderungen erfüllt. Zählend ist aber nicht nur das Programm am Schluss sondern auch die Punkte Programmcode, Testing und Dokumentation.

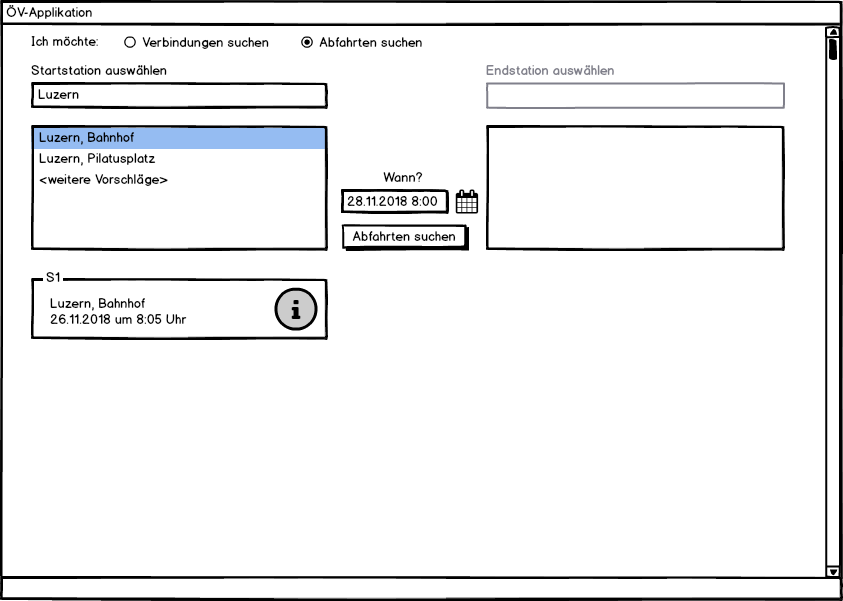
# Vorhersage

## Klassendiagramm



## Mockup



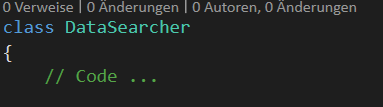


# Code Guidelines

Alles was nicht mit Ausgaben an den Benutzer zu tun hat, ist in Englisch zu halten. (Klassen, Variablen, Methoden, …)

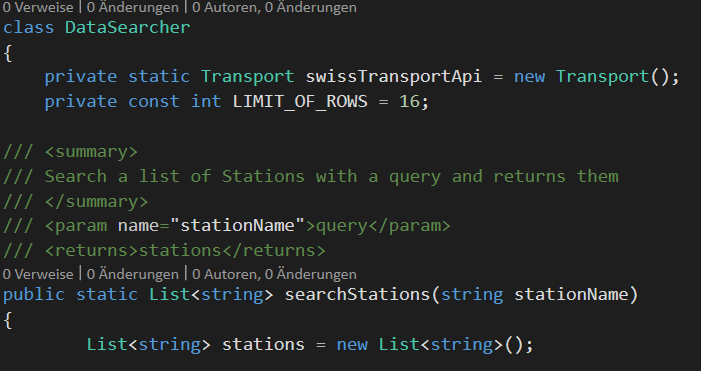
## Klassennamen

Klassennamen werden in CamelCase geschrieben und beginnen mit einem Grossbuchstaben. Der Name der Klasse sollte ein Nomen sein.



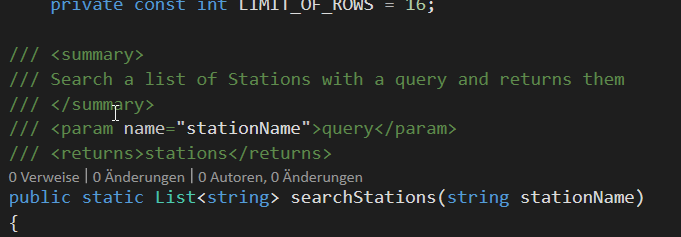
## Variabel-Namen

Variabel-Namen sind ebenfalls in CamelCase geschrieben, beginnen jedoch mit einem kleinen Buchstaben. Hierbei spielt es keine Rolle, ob die Variable global, lokal oder ein Attribut einer Methode ist. Die einzige Ausnahme bildet die Konstante, welche nur in Grossbuchstaben geschrieben wird. In einer Konstante werden die einzelnen Wörter mit Underscore (\_) getrennt.



## Methoden

Auch Methoden-Namen werden in CamelCase geschrieben. Auch hier beginnt der Name mit einem kleinen Buchstaben. Eine Methoden-Name sollte ein Zusammenspiel zwischen einem Verb und einem Nomen sein. Wenn die Aktion in der Methode etwas mit dem Objekt selber macht, kann auch nur ein Verb verwendet werden (z.B. Duration.parse(…)). Zwischen zwei Methoden ist immer eine Linie Abstand. Alle public-Methoden haben oberhalb der Methodendeklaration einen XML-Kommentar (///).



## Geschweifte Klammern

Die geschweiften Klammern kommen immer unterhalb des Kopfes. Ein else und else if steht auf einer Linie zwischen zwei geschweiften Klammern. Die geschweiften klammern werden auch dann verwendet, wenn danach nur eine Zeile Code ausgeführt wird.

## Files

Jedes File soll gleich heissen, wie die Klasse darin (z.B. bei der Klasse Transport soll das File Transport.cs heissen).

## WinForms Controls

Auch die WinForms Controls werden in CamelCase beginnend mit kleinen Buchstaben geschrieben. Voraus kommt aber immer noch ein dreibuchstabiger identifier:

|  |  |
| --- | --- |
| Identifier | WinForms Control |
| btn | Button |
| clm | Spalte einer DataGridView |
| dtp | DateTimePicker |
| grb | GroupBox |
| lbl | Label |
| lsb | ListBox |
| pnl | Panel |
| rdb | RadioButton |
| tbl | DataGridView (Tabelle) |
| txb | TextBox |
| wbb | WebBrowser |

# Umgesetzte Anforderungen

## A001

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

## 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.

## A003

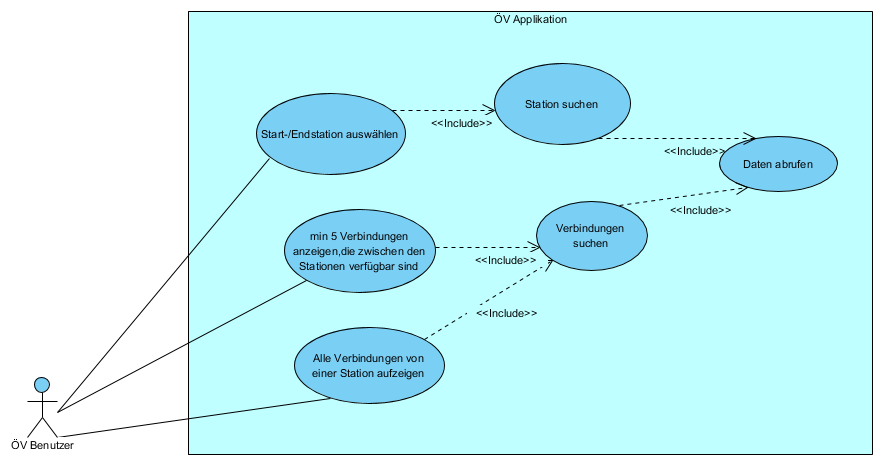
Als ÖV-Benutzer möchte ich sehen, welche Verbindungen ab einer bestimmten Station vorhanden sind, damit ich bei mir zuhause eine Art Abfahrtstafel haben kann.

## 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.

# Use Cases

## Diagramm



## Beschreibungen

### Station auswählen

|  |  |
| --- | --- |
| Use Case Beschreibung | |
| Beschreibung | Der ÖV-Benutzer möchte eine Station auswählen |
| Voraussetzungen | Internetverbindung |
| Aktor | ÖV-Benutzer |
| Ablauf | 1. Programm öffnen 2. Textsuche nach Station 3. Station aus Vorschlägen anzeigen |
| Varianten | Start/Endstation |
| Ergebnis | Station wurde ausgewählt |

### Verbindung suchen

|  |  |
| --- | --- |
| Use Case Beschreibung | |
| Beschreibung | Der ÖV-Benutzer möchte eine Verbindung suchen, um von einer Station zur nächsten zu kommen. |
| Voraussetzungen | Internetverbindung |
| Aktor | ÖV-Benutzer |
| Ablauf | 1. Stationen auswählen (siehe Use Case Station auswählen) 2. Befehl zum suchen geben |
| Varianten | - |
| Ergebnis | Liste von mindestens 5 Verbindungen zwischen den jeweiligen Stationen |

### Abfahrten anzeigen

|  |  |
| --- | --- |
| Use Case Beschreibung | |
| Beschreibung | Der ÖV-Benutzer möchte alle abfahren von einer Station aufgelistet haben |
| Voraussetzungen | Internetverbindung |
| Aktor | ÖV-Benutzer |
| Ablauf | 1. Startstation auswählen (siehe Use Case Station auswählen) 2. Befehl zum suchen geben |
| Varianten | - |
| Ergebnis | Liste von allen Verbindungen, die als Abfahrtsort die Startstation haben. |

# Testfälle

## UnitTests

### TransportTest

#### searchConnections()

Getestete Methode: Transport.searchConnections()

Als ÖV-Benutzer suche ich 4 Verbindungen von Sursee nach Luzern.

Ich erwarte, dass ich 4 Verbindungen bekomme und diese die Startstation Sursee und die Endstation Luzern haben.

#### searchStationboards()

Getestete Methode: Transport.searchStationboards();

Als ÖV-Benutzer möchte ich mich über die nächsten Abfahrten von Sursee informieren.

Ich erwarte, dass alle Abfahrten von Sursee her gehen.

### DurationTest

Die Klasse Duration wandelt den Duration String der Api um und hat eine Formatierte Ausgabe.

## Manuelle Tests

# Installationsanleitung

# Schlusswort