Dokumentation ÖV-Info Programm

**2016**

ÜK 318

Manuel Christen

01.06.2016

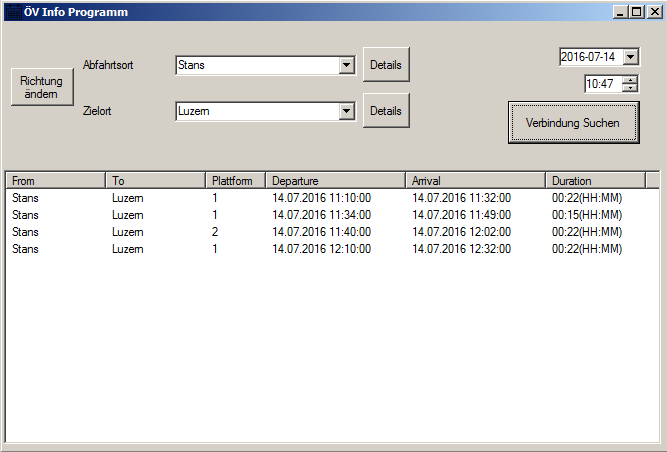


Bild 1 Hauptfenster

Inhalt

[Einleitung 2](#_Toc452558625)

[Aufgabenstellung 2](#_Toc452558626)

[Umsetzung 2](#_Toc452558627)

[Use Case Diagramm 3](#_Toc452558628)

[Aktivitätsdiagramme 3](#_Toc452558629)

[Aktivitätsdiagramm 1 3](#_Toc452558630)

[Aktivitätsdiagramm 2 3](#_Toc452558631)

[Klassendiagramm 4](#_Toc452558632)

[Install/ Uninstall 5](#_Toc452558633)

[Testfälle der Anforderung eins bis drei 6](#_Toc452558634)

[A001: 6](#_Toc452558635)

[Testfall: 6](#_Toc452558636)

[A002: 7](#_Toc452558637)

[Testfall: 7](#_Toc452558638)

[A003: 8](#_Toc452558639)

[Testfall: 8](#_Toc452558640)

[Reflexion 9](#_Toc452558641)

[Gruppe 9](#_Toc452558642)

[Verbesserungsmöglichkeiten 9](#_Toc452558643)

[Fazit 9](#_Toc452558644)

[ÜK Rückblick 10](#_Toc452558645)

[Abbildungsverzeichnis 11](#_Toc452558646)

# Einleitung

Dieses Programm haben wir im ÜK 318 geschrieben. Es wurde im Visual Studio programmiert in C# und mithilfe vom Win-Forms. Uns wurden verschiedene Anforderungen gestellt, welche unser Programm erfüllen muss. Insgesamt haben wir fünf Tage Zeit um unser Programm fertig zu stellen.

# Aufgabenstellung

Das Programm sollte eine Win-Forms Anwendung sein welche mit C# programmiert ist. Von acht Anforderungen mussten die erste drei erfüllt sein. Diese waren:

* Ich als ÖV-Benutzer möchte die Start und Endstation mittels Textsuche suchen können, damit ich nicht alle Stationsnamen auswendig lernen muss.
* Ich als ÖV-Benutzer möchte die aktuellen Verbindungen zwischen den beiden gefunden und ausgewählten Stationen sehen, damit ich weiss wann ich zur Station muss.
* Ich als ÖV-Benutzer möchte sehen welche Verbindungen ab einer bestimmten Station vorhanden sind. Damit ich eine Art Station Abfahrtstafel Zuhause haben kann.

Den Rest der Anforderungen waren Zusatzanforderungen. In diesen war enthalten:

* Erste Suchresultate erscheinen schon bei der Eingabe
* Dass Verbindungen zu einem anderen Zeitpunkt gesucht werden können
* Sehen wo sich eine Station befindet
* Stationen finden welche in der Nähe meiner Position sind
* Gefunden Resultate sollen via Mail weiter geschickt werden

# Umsetzung

Die obligatorischen Anforderungen werden alle erfüllt. Von den Zusatzanforderungen sind nur wenige nicht implementiert. Der Benutzer kann das Programm starten und eine Start und End Station eingeben. Diese werden in einem Dropdown-Menü dem Benutzer schon vorgeschlagen während dem er die Station eintippt. Danach kann der Benutzer nach der gewünschten Verbindung suchen. Diese nächsten vier möglichen Verbindungen werden angezeigt. Zusätzlich kann das Abfahrtsdatum und die Uhrzeit ausgesucht werden, damit der Benutzer die Möglichkeit hat Reisen zu planen. Auch kann der Benutzer nur eine Station eingeben und mithilfe vom Button „Details“ die Abfahrtstafel dieser Station sehen. Dies Funktioniert für beide Stationen(Abfahrtsort und Ankunftsort). Mit einem Klick auf „Richtung ändern“ können der Abfahrtsort und der Ankunftsort umgetauscht werden, sodass gleich auch eine Verbindung zurück nachgeschaut werden kann. Es ist möglich eine Adresse einzugeben. Es wird dann nach der nächsten Haltestelle in der Nähe gesucht. Bei der Dauer, wie lange die Fahrt dauern wird, ist auch schon der Fussweg berechnet, also wie lange man etwa braucht um zur Haltestelle zu laufen. Das Icons des Programmes wurde zu einem Kalender geändert.

# Use Case Diagramm

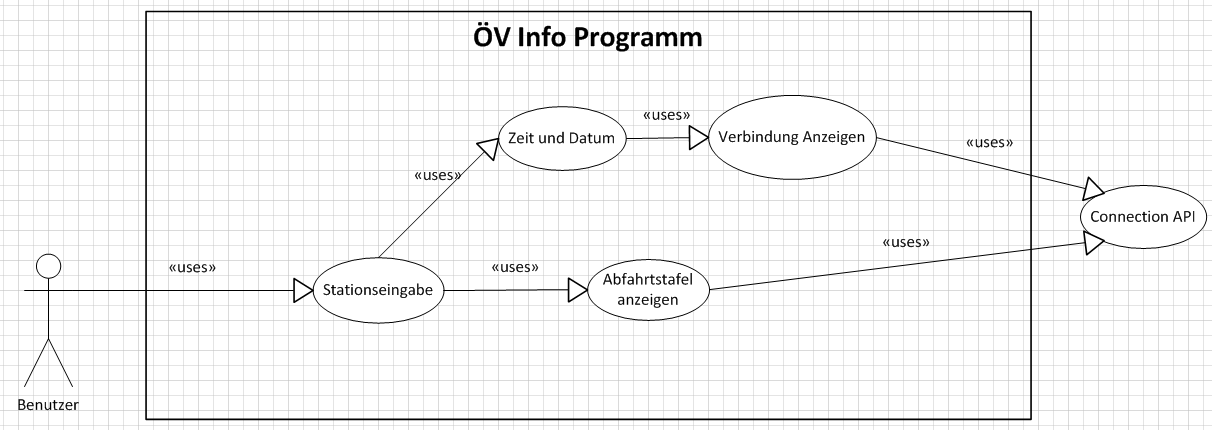
Das Use Case Diagramm zu unserem Programm sieht folgender massen aus:

Bild 2 Use Case Diagramm

# Aktivitätsdiagramme

## Aktivitätsdiagramm 1

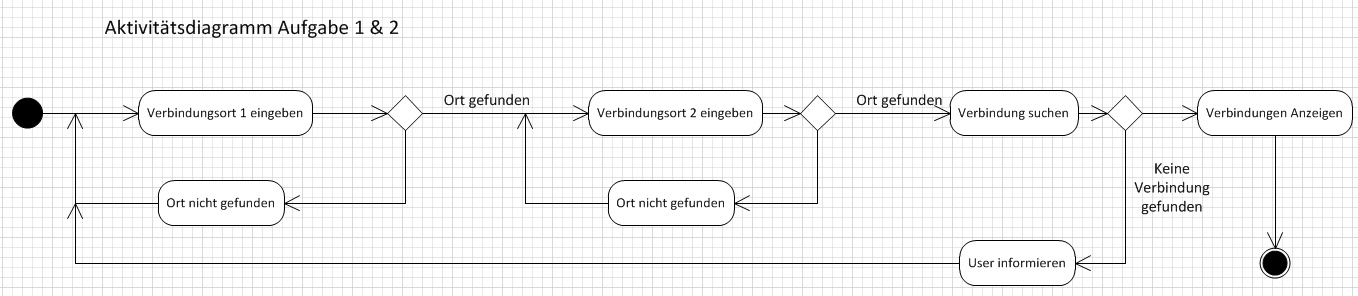


Bild 3 Aktivitätsdiagramm 1

## Aktivitätsdiagramm 2

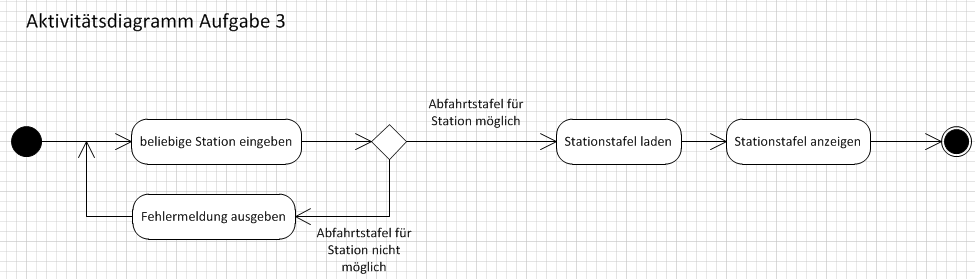


Bild 4 Aktivitätsdiagramm 2

# Klassendiagramm

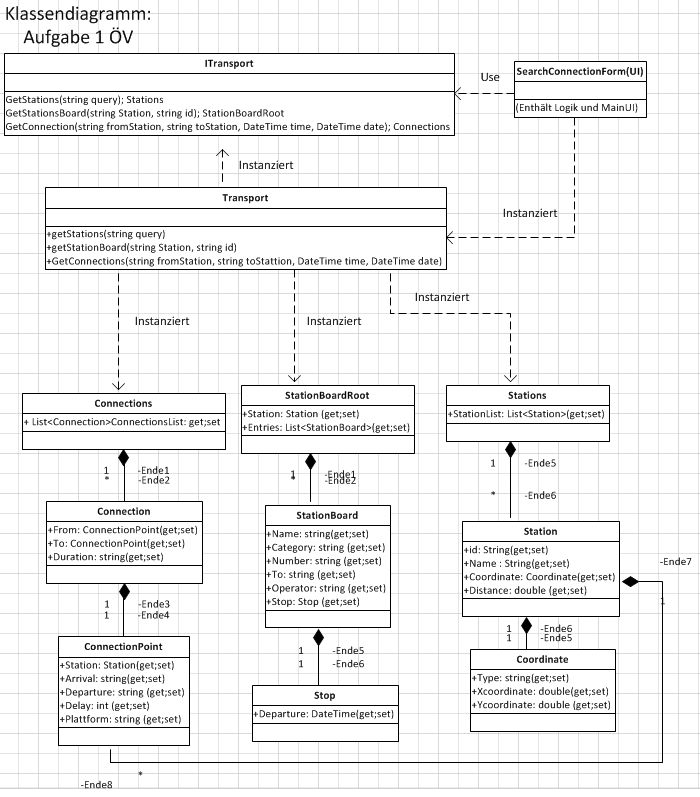
Zur Applikationsplanung ist ein Klassendiagramm sehr wichtig. Durch dieses werden die Verbindungen unter den einzelnen Klassen gut sichtbar. Beim Programmieren ist dies sehr Hilfreich. Und als Programmieren kann man sich danach immer ein wenig auf diesem „Abstützen“. Dieses Klassendiagramm wurde erst fertig gestellt als die Applikation schon fertig war. Dadurch sind die einzelnen Klassen sehr genau beschrieben.

Bild Klassendiagramm

# Install/ Uninstall

Dieses Programm kann unter folgendem Link gefunden werden:

<https://github.com/tyroklan/modul-318-student>

Um das Programm zu starten wird Visual Studio 2015 benötigt. Es muss die Datei mit der „.sln“ Dateiendung ausgeführt werden. Es startet sich Visual Studio und das Projekt automatisch. Dort kann dann oben in der Mitte „Start“ gedrückt werden oder F5 und das Programm startet sich.

Eine Deinstallation ist nicht nötig.

# Testfälle der Anforderung eins bis drei

Natürlich muss das Programm getestet werden, um es auf seine Fähigkeiten zu testen, aber vor allem um Fehler zu entdecken und diese beheben zu können. Bekommt man einen Erfolgreichen Test zurück oder ist das Resultat wie erwartet, ist das Programm korrekt. Jedoch wenn etwas Unerwartetes Passiert oder das Programm sogar abstützt muss man sich nochmals mit dem Code befassen um den Fehler beheben zu können.

Bevor wir mit dem Testen beginnen können, muss natürlich zuerst Visual Studio gestartet werden und das Programm.

Danach kann mit dem Testing begonnen werden.

## A001:

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

### Testfall:

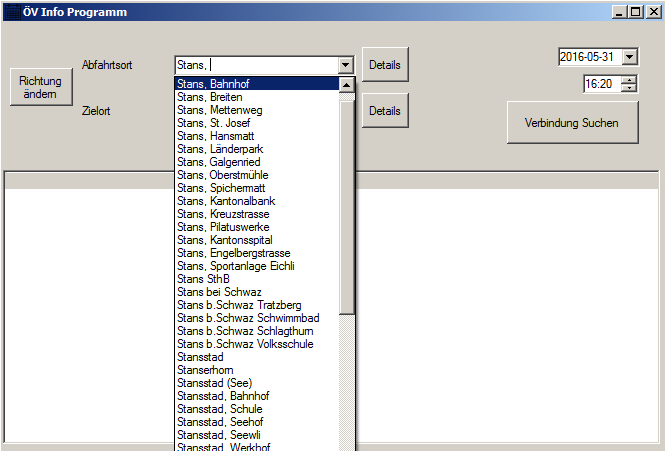
* Programm starten
* Stationen können eingetippt werden (Abfahrtsort und Ankunftsort).
* Diese können automatisch vervollständigt werden (Dropdown).
  + Wenn ein Name einer Station eingegeben wird, soll mit dem Dropdown Menü das Feld vervollständigt werden können.
* Keine Sonderzeichen oder reine Zahlen im Namen.
  + Ansonsten kommt kein Ergebnis.

Bild 6 Testfall A001

## A002:

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

### Testfall:

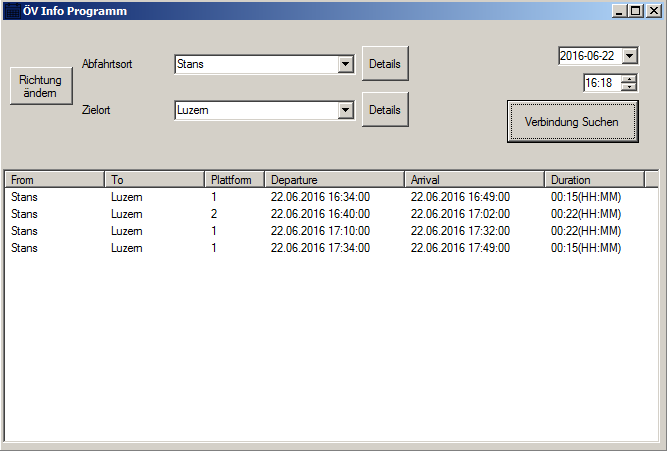
* Programm starten
* Beide Felder müssen ausgefüllt werden.
  + Wieder mithilfe des Dropdowns, welches Vorschläge bringt.
* Korrekte Verbindungen werden angezeigt, mit den vorher eingegebenen Stationen.
  + Wird auf „Verbindung suchen“ geklickt, sollen unten in der Tabelle die nächsten vier möglichen Verbindungen angezeigt werden.
  + Dabei soll die Anfangsstation, die Endstation, das Gleis, die Abfahrtszeit, die Ankunftszeit und die Dauer der Fahrt angezeigt werden.
* Werden die Felder leer gelassen wird keine Verbindung angezeigt sondern es erscheint die Fehlermeldung „Bitte füllen Sie beide Felder aus“.
* Die Stationen können mit dem Button „Richtung ändern“ getauscht werden.
  + Dann wird der Abfahrtsort zum Zielort und der Zielort wird zum Abfahrtsort.
* Das Datum und die Uhrzeit können für die Verbindungen gewählt werden. (Z.B. zum planen einer Reise an einem späteren Zeitpunkt)
  + Es sollen wieder vier Verbindungen angezeigt werden, welche jedoch an dem gewählten Datum und nach der gewählten Zeit Verfügbar sind.
* Wird keine Verbindung gefunden, wird eine Fehlermeldung ausgegeben.
* Es können auch Adressen eingegeben werden
  + Dann wird automatisch die nächstgelegene Station ausgewählt.

Bild 7 Testfall A002

## A003:

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

### Testfall:

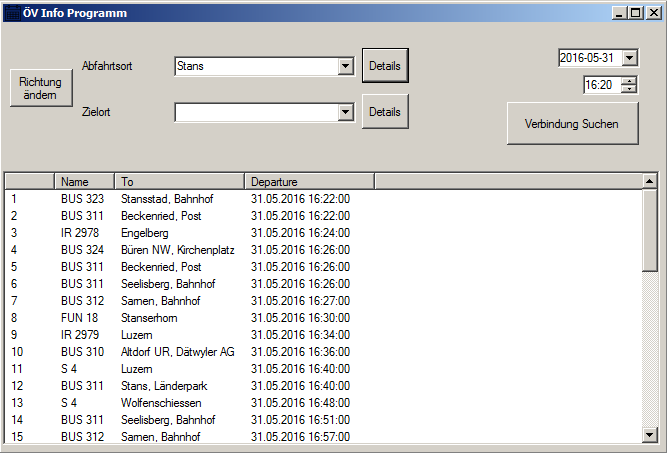
* Programm starten
* Eines der beiden Felder muss ausgefüllt sein.
  + Es muss der „Details“-Button neben dem Ausgefüllten Feld geklickt werden.
* Alle möglichen Verbindungen werden angezeigt.
  + Ist das Feld korrekt ausgefüllt werden alle Verbindungen von diesem Ort angezeigt die in der nächsten Zeit möglich sind. Es wird angezeigt wo diese Verbindung sein wird, also die Nummer( Bus, Zug, etc.). Es ist zu sehen wann die Verbindung sein wird und wohin.
* Dies Funktioniert für beide Stationen, je nach dem welcher Button geklickt wird.
* Wird das Feld leer gelassen wird eine Fehlermeldung ausgegeben.
* Das auswählen des Datums und der Uhrzeit funktioniert für die Abfahrtstafel **nicht**.

Bild 8 Testfall A003

# Reflexion

## Gruppe

Auch wenn dieses Projekt keine Gruppenarbeit war, haben mein Pultnachbar(Leander) und ich uns sehr oft helfen können bei Problemen. Auch sonst fand ich unsere Gruppe sehr gut und bei Fragen fand man immer Hilfe.

## Verbesserungsmöglichkeiten

Natürlich hat mein Programm verschiedenste Verbesserungsmöglichkeiten. Einige davon sind z.B.:

* Dass die Liste noch per Mail versendet werden kann.
* Dass der Standort der Stationen in Google Maps angezeigt werden kann.
* Dass der Benutzer nicht nach jedem Tastendruck in der Eingabebox warten muss.
* Man könnte ein „Via“ einbauen um Zwischenstationen anzugeben.
* Redundanter Code minimieren.

## Fazit

Auch wenn schon viel eingebaut ist, gibt es unendlich viele Möglichkeiten das Programm noch nach seinen Wünschen zu erweitern und auszubauen.

# ÜK Rückblick

Dieser ÜK hat mir sehr gefallen. Ich habe viel Neues gelernt und wir konnten viel rumprobieren. Wir hatten ein gutes Klima im Unterricht und es war immer interessant und spannend.

Ich persönlich hatte sehr wenig Erfahrung mit Win-Forms und für mich waren die Aufgaben sehr anspruchsvoll, jedoch wurde alles gut erklärt und wir bekamen bei Fragen immer Hilfe.

Leider habe ich im Betrieb nicht viel mit C# Programmiert, und kenne es vor allem von der Schule. Im Betrieb programmiere ich nur mit ABAP(SAP).

# Abbildungsverzeichnis

[Bild 1 Hauptfenster 0](file:///W:\Teilnehmer\Manuel%20Christen\Repo\Dokumentation_ÖV-Info.docx#_Toc452558673)

[Bild 2 Use Case Diagramm 3](file:///W:\Teilnehmer\Manuel%20Christen\Repo\Dokumentation_ÖV-Info.docx#_Toc452558674)

[Bild 3 Aktivitätsdiagramm 1 3](#_Toc452558675)

[Bild 4 Aktivitätsdiagramm 2 3](file:///W:\Teilnehmer\Manuel%20Christen\Repo\Dokumentation_ÖV-Info.docx#_Toc452558676)

[Bild 5 Klassendiagramm 4](file:///W:\Teilnehmer\Manuel%20Christen\Repo\Dokumentation_ÖV-Info.docx#_Toc452558677)

[Bild 6 Testfall A001 6](file:///W:\Teilnehmer\Manuel%20Christen\Repo\Dokumentation_ÖV-Info.docx#_Toc452558678)

[Bild 7 Testfall A002 7](file:///W:\Teilnehmer\Manuel%20Christen\Repo\Dokumentation_ÖV-Info.docx#_Toc452558679)

[Bild 8 Testfall A003 8](file:///W:\Teilnehmer\Manuel%20Christen\Repo\Dokumentation_ÖV-Info.docx#_Toc452558680)