|  |  |
| --- | --- |
| C:\Users\Djordje\Desktop\Download.jpg  SBB Fahrplan  Desktopapplikation | Djordje Aksentijevic  Modul 318 - Analysieren und objektbasiert programmieren (C#) |

Inhaltsverzeichnis

[1. Einleitung 2](#_Toc533151412)

[2. Zweck der Dokumentation 2](#_Toc533151413)

[Vorgehen 2](#_Toc533151414)

[2.1 Code-Stil (Richtlinien) 2](#_Toc533151415)

[3. Use case 3](#_Toc533151416)

[3. Aktivitätsdiagramm 3](#_Toc533151417)

[5. GUI 4](#_Toc533151418)

[6. Mockup / Diagramme 6](#_Toc533151419)

[6.1 Anforderung A001 6](#_Toc533151420)

[6.1 Anforderung A002 7](#_Toc533151421)

[7.3 Anforderung A003 8](#_Toc533151422)

[6.1 Anforderung A004 9](#_Toc533151423)

[6.1 Anforderung A005 10](#_Toc533151424)

[6.1 Anforderung A008 11](#_Toc533151425)

[7.0 Installations-Anleitung 13](#_Toc533151426)

# Einleitung

Am Mittwoch, dem 19.12 haben wir den Auftrag bekommen eine ÖV Applikation umzusetzen und sauber zu dokumentieren.

In dieser Dokumentation möchte ich ihnen aufzeigen wie ich bei meiner Arbeit vorgegangen bin. Unteranderem will ich meine Erfahrung mit euch teilen. Ebenso will ich euch beschreiben welche Probleme ich während der Arbeit hatte, was ich tat, um diese zu lösen und wie am Schluss das gesamt Resultat geworden ist.

# Zweck der Dokumentation

Dieses Dokument soll dazu dienen, dass man den Verlauf und die Entwicklung der Applikation nachvollziehen kann. Man sieht wie die Applikation strukturell aufgebaut ist. Der Hauptzweck dient dazu, Punkte für die Bewertung zu sammeln, da eine Dokumentation eine Anforderung ist.

# Vorgehen

## 2.1 Code-Stil (Richtlinien)

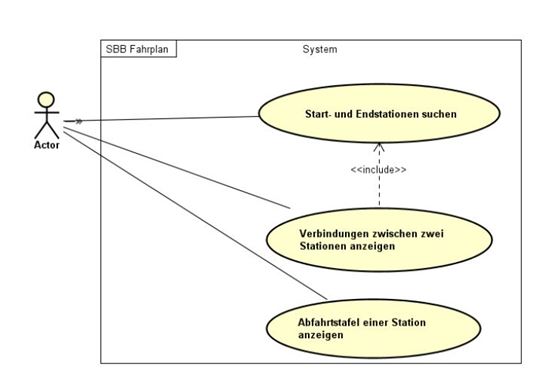
Beim Programmieren versuche ich meinen Code so zu schreiben, dass er ohne Kommentare klar und verständlich ist. Die Namen der Objekte und Variablen sind alle einheitlich im «Camel-Case» Stil geschrieben. (was camel case ist)

Code der mehrmals gebraucht wird habe ich jeweils in Methoden gepackt um Redundanzen zu verhindern.

# 

# 3. Use case

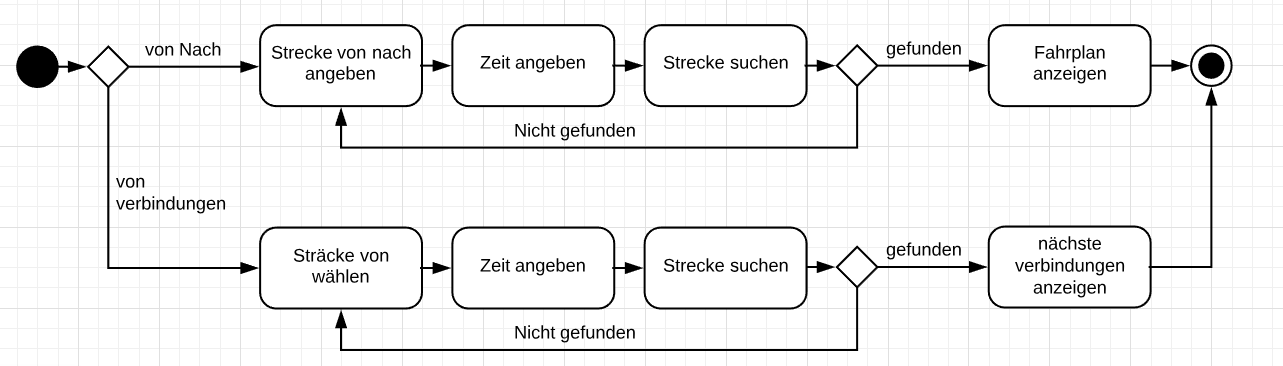
Nebst dem GUI-Design habe ich auch noch ein UseCase kreiert. Man sieht also welche Aktion was ganz genau macht.



# 

# Aktivitätsdiagramm

Mein Aktivitätsdiagramm ist ebenfalls recht simpel, da das was es jetzt macht, meiner Meinung nach genug ist. Sollte mir auffallen das mir etwas fehlt oder sollte ich am Schluss noch genug Zeit haben, so werde ich bestimmt noch weitere Features in das Programm einfügen.



# 5. GUI

Beim Designen von meinem Graphik User Interface hatte ich ein Motto: Schlicht und möglichst selbsterklärend. Meiner Meinung nach ist mir dies auch gelungen.



**1.Die GUI-Controls werden folgendermassen benannt:**

**2.Die Klassennamen werden mit Grossbuchstaben begonnen.**

Zeit -> dateTimePicker1

Resultat mit Mail senden -> button\_Mail

Abfahrtsort -> comboBox\_Abfahrtsort

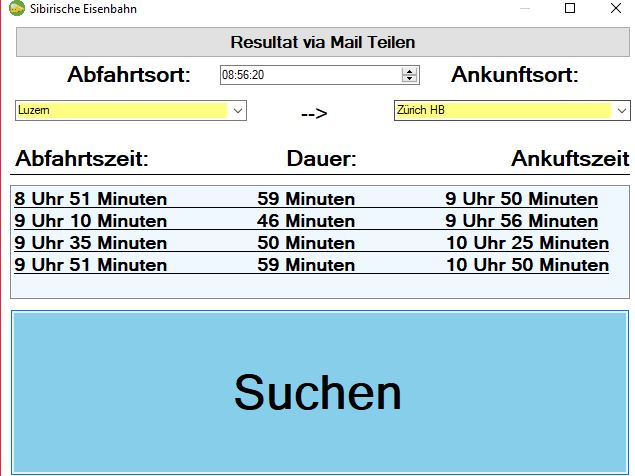
Ankunftsort -> comboBox\_Ankunftsort

listBox für die Ausgabe -> listBox\_Ausgabe

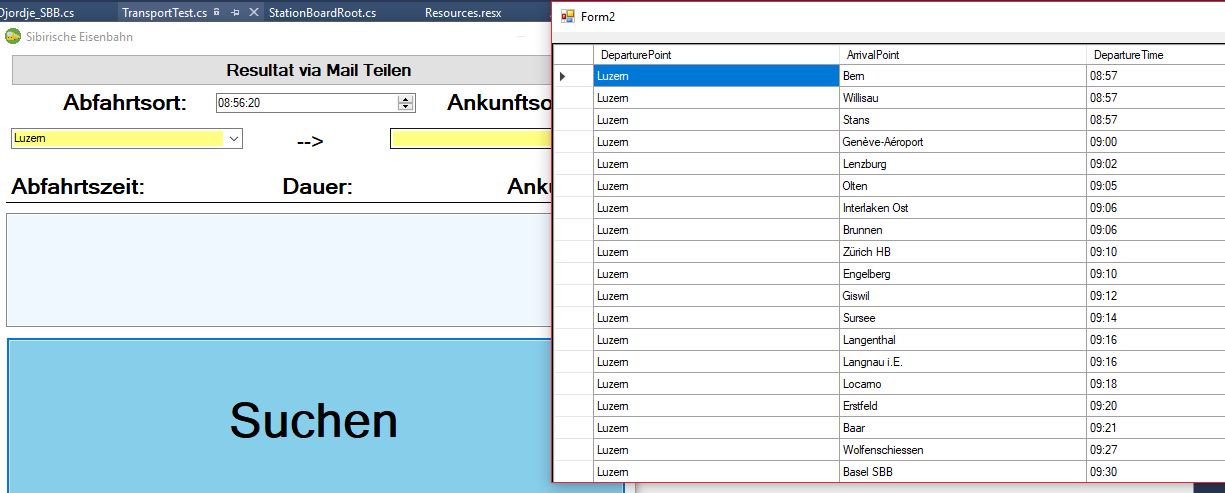
Suchen -> btn\_Suchen

hilfe button ->button\_Help

Sponsoring button -> button\_Sponoring

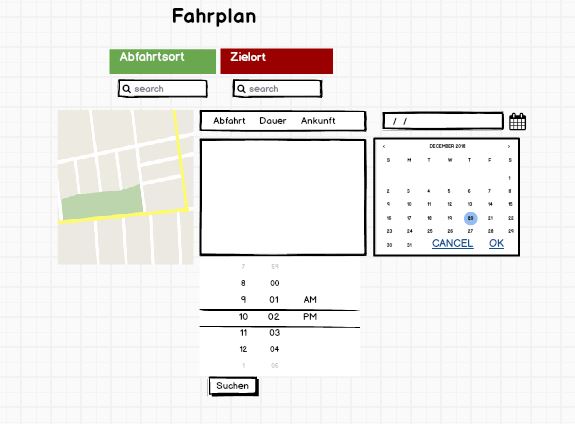


**Wenn man nur den Abfahrtsort angibt bekommt man als Ausgabe die Abfahrtsliste**



## 

## 6. Mockup / Diagramme

Bevor ich mit dem programmieren beginne, muss ich noch ein paar Diagramme erstellen, damit ich sehe was ich alles machen kann. Als erstes erstellte ich ein GUI-Design an welchem ich mich orientieren werde. 

Ich habe versucht, das Design schlicht und einfach zu halten, sodass man ohne gross zu überlegen weiss, was man machen muss.

# Anforderung A001

|  |  |  |
| --- | --- | --- |
| Schritt | Aktivität | Erwartetes Resultat |
| **1** | Der Benutzer klickt auf die Combobox «Abfahrtsort» und gibt «Lu» ein. | Es sollen Stationen welche «Lu» enthalten vorgeschlagen werden.  Es muss so aussehen: |
| **2** | Der Benutzer klickt auf die Combobox «Ankunftsort» und gibt «Zu» ein. | Es sollen Stationen welche «Zu» enthalten vorgeschlagen werden.  Es muss so aussehen: |

# Anforderung A002

|  |  |  |
| --- | --- | --- |
| Schritt | Aktivität | Erwartetes Resultat |
| **1** | Der Benutzer klickt auf die Combobox «Abfahrtsort» und gibt «Lu» ein. | Es sollen Stationen welche «Lu» enthalten vorgeschlagen werden.  Es muss so aussehen: |

|  |  |  |
| --- | --- | --- |
| **2** | Der Benutzer klickt auf die Combobox «Ankunftsort» und gibt «Zu» ein. | Es sollen Stationen welche «Zu» enthalten vorgeschlagen werden.  Es muss so aussehen: |
| **3** | Der Benutzer muss auf die «Suchen»-Taste klicken, und ihm werden die Nächsten vier Verbindungen zwischen seinem Abfahrtsort und Ankunftsort angezeigt. | In der List Box über der «Suchen»-Taste werden die nächsten vier Verbindungen angezeigt (in diesem Beispiel die Verbindungen zwischen Luzern und Zürich) |

# 7.3 Anforderung A003

|  |  |  |
| --- | --- | --- |
| **1** | Der Benutzer klickt auf die Combobox «Abfahrtsort» und gibt «Lu» ein. | Es sollen Stationen welche «Lu» enthalten vorgeschlagen werden.  Es muss so aussehen: |
| **2** | Der Benutzer kann direkt auf «Suchen» klicken, ohne einen Ankunftsort eingegeben zu haben. | Es muss ein neues Fenster aufgehen, wo sich alle Verbindungen ab der Abfahrtstation anzeigen lassen.  Hier mit Beispiel als Abfahrtort Luzern: |

# Anforderung A004

|  |  |  |
| --- | --- | --- |
| **1** | Der Benutzer klickt auf die Combobox «Abfahrtsort» und gibt «L» ein. Die Suche beginnt ab dem ersten Char. | Autofill wird direkt gestartet. Es werden alle möglichen Vorschläge, welche zur Suche passen angezeigt.  Es muss so aussehen: |
| **2** | Der Benutzer klickt auf die Combobox «Ankunftsort» und gibt «Z» ein. Die Suche beginnt ab dem ersten Char. | Es sollen Stationen welche «Z» enthalten vorgeschlagen werden.  Es muss so aussehen: |

|  |  |  |
| --- | --- | --- |
| **3** | Der Benutzer muss auf die «Suchen»-Taste klicken, und ihm werden die Nächsten vier Verbindungen zwischen seinem Abfahrtsort und Ankunftsort angezeigt. | In der List Box über der «Suchen»-Taste werden die nächsten vier Verbindungen angezeigt (in diesem Beispiel die Verbindungen zwischen Luzern und Zürich) |

# Anforderung A005

|  |  |  |
| --- | --- | --- |
| **1** | Der Benutzer klickt auf die Combobox «Abfahrtsort» und gibt «L» ein. Die Suche beginnt ab dem ersten Char. | Autofill wird direkt gestartet. Es werden alle möglichen Vorschläge, welche zur Suche passen angezeigt.  Es muss so aussehen: |
| **2** | Der Benutzer klickt auf die Combobox «Ankunftsort» und gibt «Z» ein. Die Suche beginnt ab dem ersten Char. | Es sollen Stationen welche «Z» enthalten vorgeschlagen werden.  Es muss so aussehen: |
| **3** | Der Benutzer kann auch die Abfahrtszeit Beliebig ändern. Er kann also die ersten zwei stellen mit der Stunde befüllen und die nächsten zwei mit den Minuten befüllen.  Der Benutzer muss auf die «Suchen»-Taste klicken, und ihm werden die Nächsten vier Verbindungen zwischen seinem Abfahrtsort und Ankunftsort angezeigt. | Es sollen nur Verbindungen angezeigt werden, welche innerhalb der Zeiteingabe des Benutzers liegen. (Hier ein Beispiel falls der Benutzer als Abfahrtszeit «09.35» eingegeben hat) |

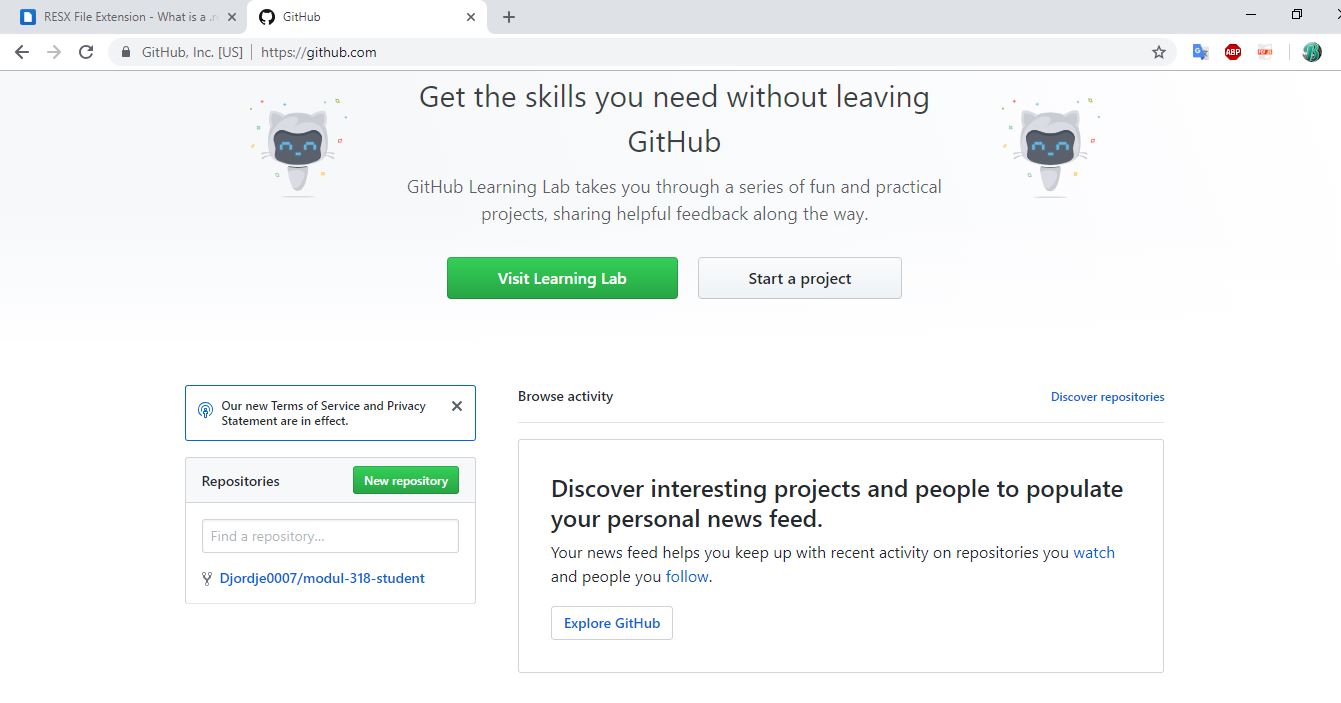
# Anforderung A008

|  |  |  |
| --- | --- | --- |
| **1** | Der Benutzer klickt auf die Combobox «Abfahrtsort» und gibt «L» ein. Die Suche beginnt ab dem ersten Char. | Autofill wird direkt gestartet. Es werden alle möglichen Vorschläge, welche zur Suche passen angezeigt.  Es muss so aussehen: |
| **2** | Der Benutzer klickt auf die Combobox «Ankunftsort» und gibt «Z» ein. Die Suche beginnt ab dem ersten Char. | Es sollen Stationen welche «Z» enthalten vorgeschlagen werden.  Es muss so aussehen: |
| **3** | Der Benutzer kann auch die Abfahrtszeit Beliebig ändern. Er kann also die ersten zwei stellen mit der Stunde befüllen und die nächsten zwei mit den Minuten befüllen.  Der Benutzer muss auf die «Suchen»-Taste klicken, und ihm werden die Nächsten vier Verbindungen zwischen seinem Abfahrtsort und Ankunftsort angezeigt. | Es sollen nur Verbindungen angezeigt werden, welche innerhalb der Zeiteingabe des Benutzers liegen. (Hier ein Beispiel falls der Benutzer als Abfahrtszeit «09.35» eingegeben hat) |
| **4** | Danach kann der Benutzer seine gefundene Resultate via Mail teilen. | Diese Funktion ist mir zuhause bereits einmal gelungen. Ich habe die Zieladresse und den Absender HardCoden müssen. |

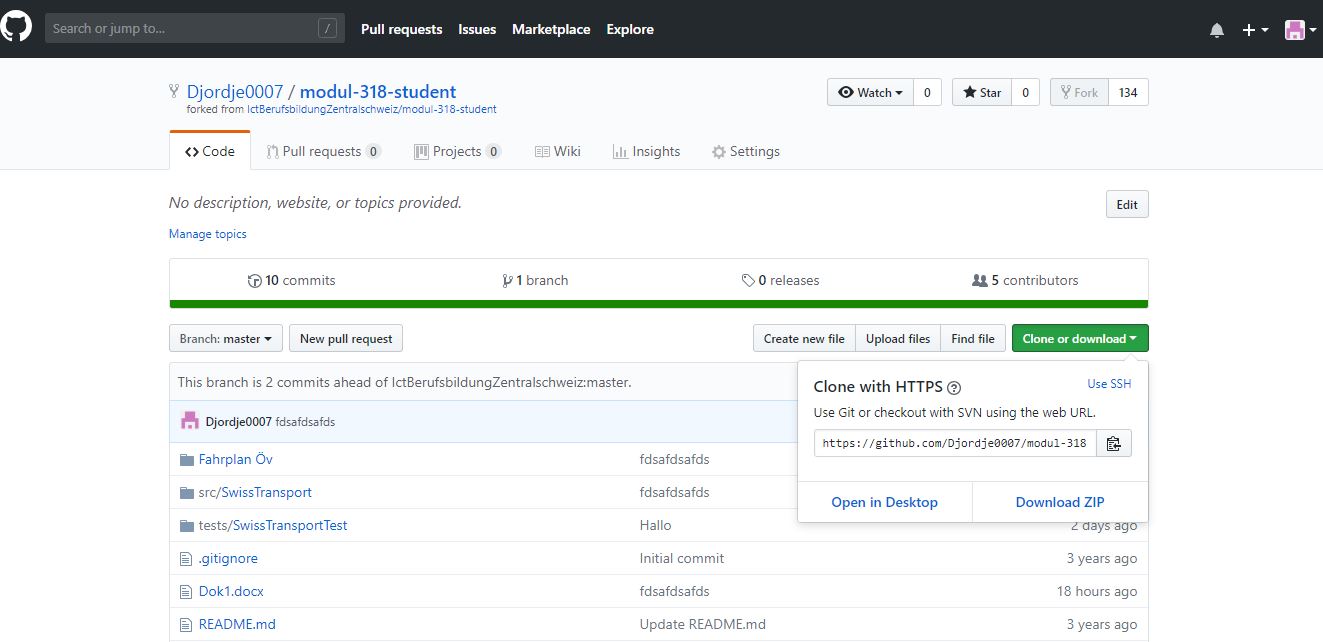
# Installations-Anleitung

Hier eine «Schritt für Schritt» Anleitung, wie man sich meine «SBB»-App lokal herunterladen kann:

Man geht auf github.com und klickt auf «Djordje/modul-318-student

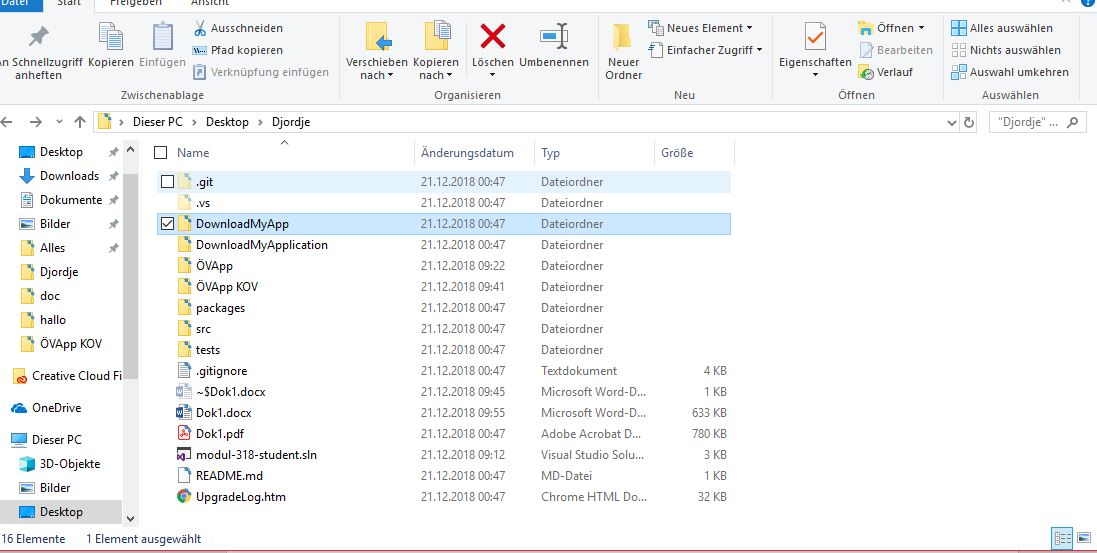


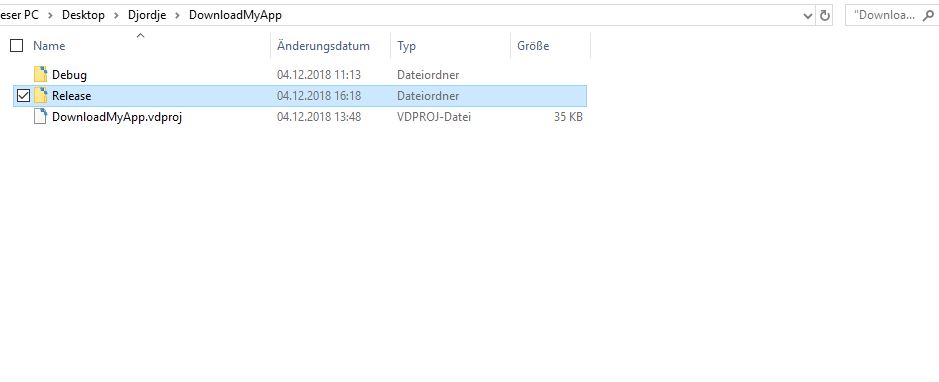
Danach muss man es einfach nur noch als Zip-Datei runterladen



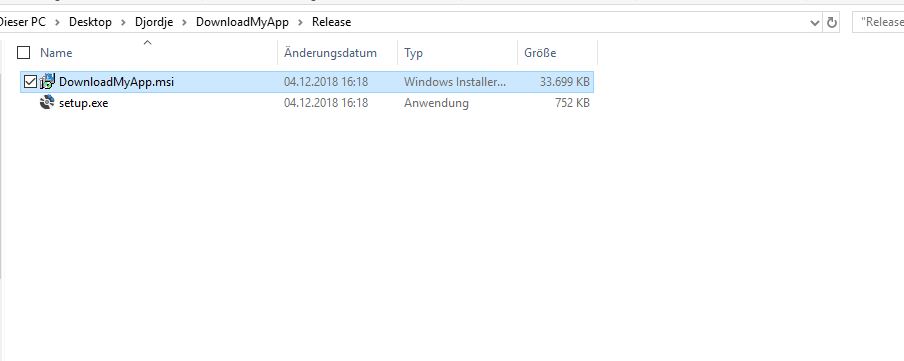
Man öffnet den Speicherort, in dem von der Dtei welche man als zip heruntergeladen hat

Dort öffnet man den Ordner «DownloadMyApp»

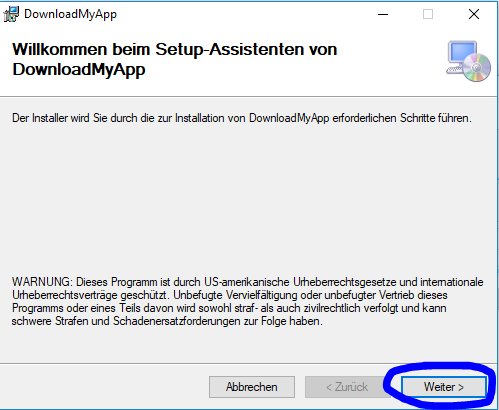


Danach öffnet man den Ordner «Release» 

Als nächstes doppelklickt man die Datei «DownloadMyApp»

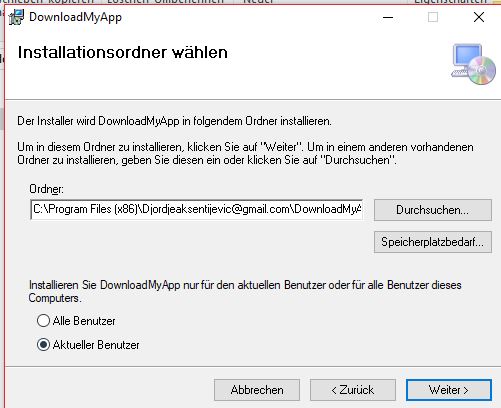


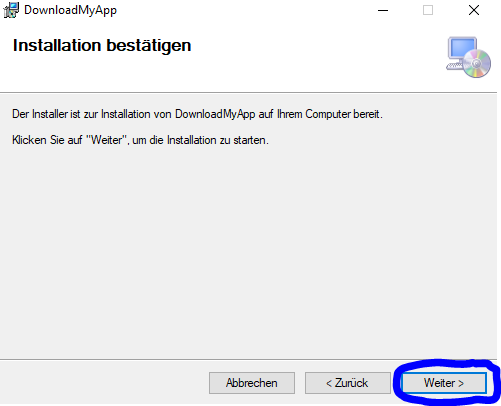
Klicken sie auf weiter



Entweder man kann seinen eigenen Speicher-Pfad Entweder man kann seinen eigenen Speicher-Pfad angeben oder den vom

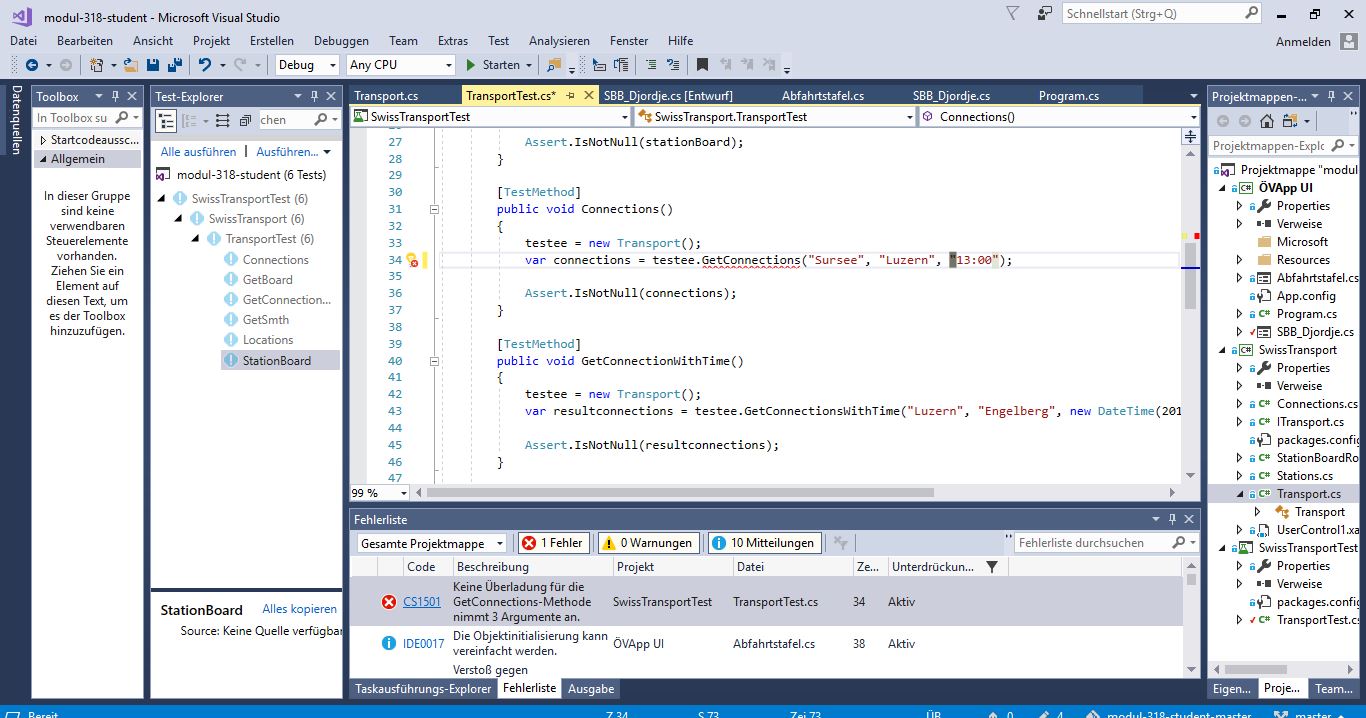
Klicken sie auf weiter



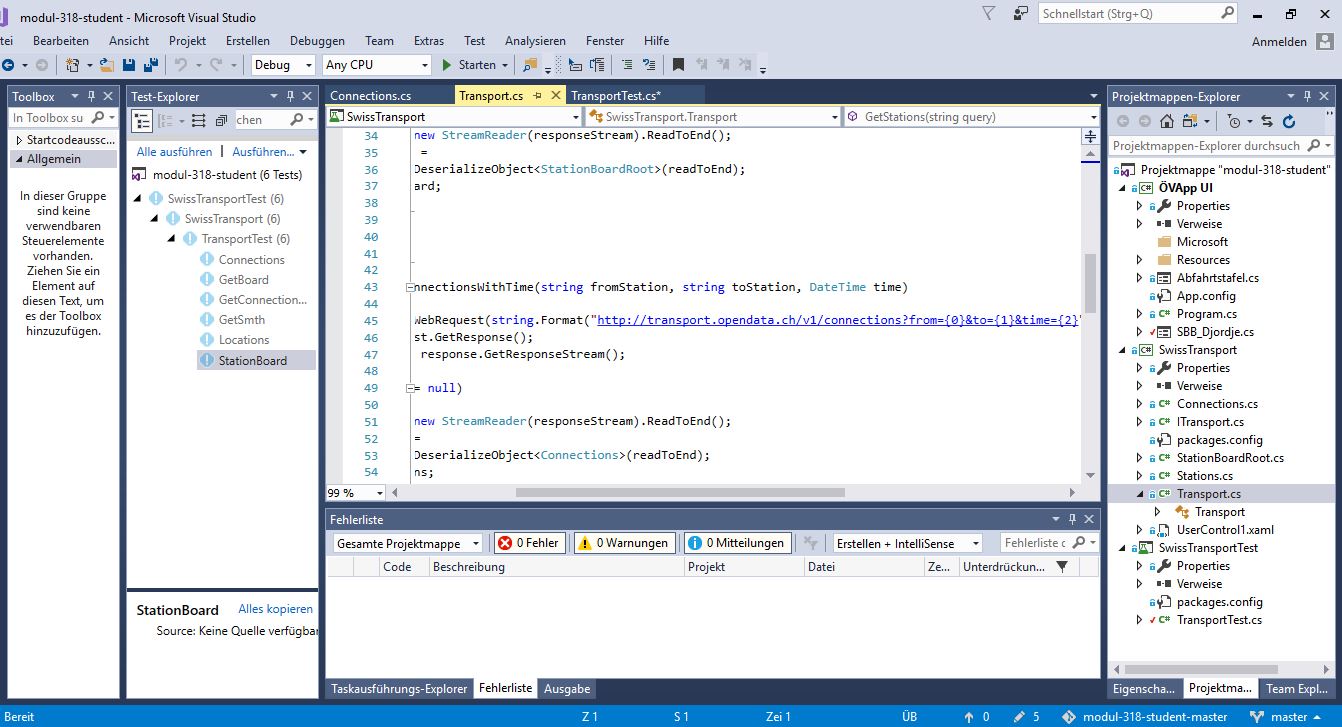
weiter drücke und dann auf schliessen (wurde erfolgreich heruntergeladen).

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Beschreibung** | **Umgesetzt? (Ja/nein)** | **Wie umgesetzt?** |
| A001 | Als ÖV-Benutzer möchte ich Start- und Endstation mittels Textsuche suchen können, damit ich nicht alle Stationsnamen auswendig lernen muss. | Ja | Ich habe ein Auto Fill programmiert |
| 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. | Ja | Ich habe eine List Box im GUI, auf welcher die nächsten vier zutreffenden Verbindungen erscheinen. |
| 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. | Ja | Ich habe ein 2. Form mit nur einem DataGrid erstellt. Sollte der Benutzer nur eine Abfahrt aber keinen Ankunftsort ausgewählt haben, so erscheint Form 2 mit allen Verbindungen ab dem eingegebenen Abfahrtsort. |
| A004 | Als ÖV-Benutzer möchte ich, dass schon während meiner Eingabe erste Such-Resultate erscheinen, damit ich effizienter nach Stationen suchen kann. | Ja | Im Text Change Event rufe ich eine Methode auf, welche für den Auto Fill zuständig ist. |
| 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. | Ja | Ich habe ein DateTimePicker in mein GUI hinzugefügt. So kann der Benutzer die Abfahrtszeit eingeben und es werden Ihm nur die Resultate ab seiner eingegebenen Zeit angezeigt. |
| A006 | Als ÖV-Benutzer möchte ich sehen, wo sich eine Station befindet, damit ich mir besser vorstellen kann, wie die Situation vor Ort aussieht. | Nein |  |
| A007 | Als ÖV-Benutzer möchte Stationen finden, die sich ganz in der Nähe meiner aktuellen Position befinden, damit ich schnell einen Anschluss erreichen kann. | Nein |  |
| A008 | Ich möchte meine gefundenen Resultate via Mail weiterleiten können, damit auch andere von meinen Recherchen profitieren können. | teilweise | Ich habe oben einen Button hinzugefügt, welcher eine mail an, die von mir hart gecodete Mail mit den Resultaten aus der List box schickt. |

Testing:



Kann kein Datum einfügen weil ich nur das Date time habe :



Wenn ich die uhrzeit weg lasse klappt mein testing ohne fehler doch es wird 0 mal ausgeführt .