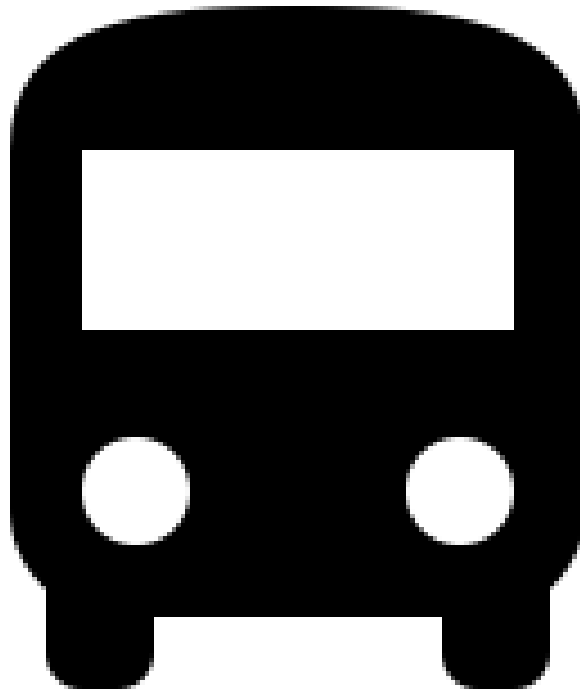


Öv

Fahrplan

Modul 318



Projekt Dokumentation

Beat Huwyler

Inhaltsverzeichnis

1. Einleitung.....	3
2. Mockup.....	3
3. Mein UseCaseDiagram	4
4. Zweck dieserDokumentation	4
5. Mein ActivityDiagram	5
6. Problem Analyse	6
7. Mein EndDesign	7
8. Funktionen	8
9. Testing	9
10. Code Richtlinie	12
10.1 Naming Conventions	12
10.2 Declaration	12
10.3 Comments	12
10.4 Statements	12
11. Installation	12
12. Deinstallation	12
12. Schlusswort	13

1. Einleitung

Wir bekamen denn Auftrag ein Fahrplan Applikation zu erstellen. Die Applikation muss viele verschiedene Funktionen aufweisen wie zum Beispiel Verbindungsanzeigen oder eine Abfahrtstafel anzeigen. Zudem sind nebenbei noch weitere Faktoren zu beachten wie ein Sauberes und fehlerfreies Code zu erstellen. Diese Dokumentation führt Sie durch mein Projekt, dass ich gemacht habe. Ich wünsche Ihnen viel Spass.

2. Mockup

Das ist mein Mockup von meinem Programm. Ich habe das Design sehr schlicht und einfach gehalten um eine sehr übersichtliche Benutzeroberfläche zu erstellen. Unten rechts habe ich einen Button erstellt der auf eine andere Seite führt in der man die Abfahrtstafel angezeigt bekommt.

SBB

Von: Nach:

Abfahrzeit	Ankunftszeit	Reisedauer
12.00	13.00	1.00
13.00	14.00	1.00
14.00	15.00	1.00
15.00	16.00	1.00
16.00	17.00	1.00

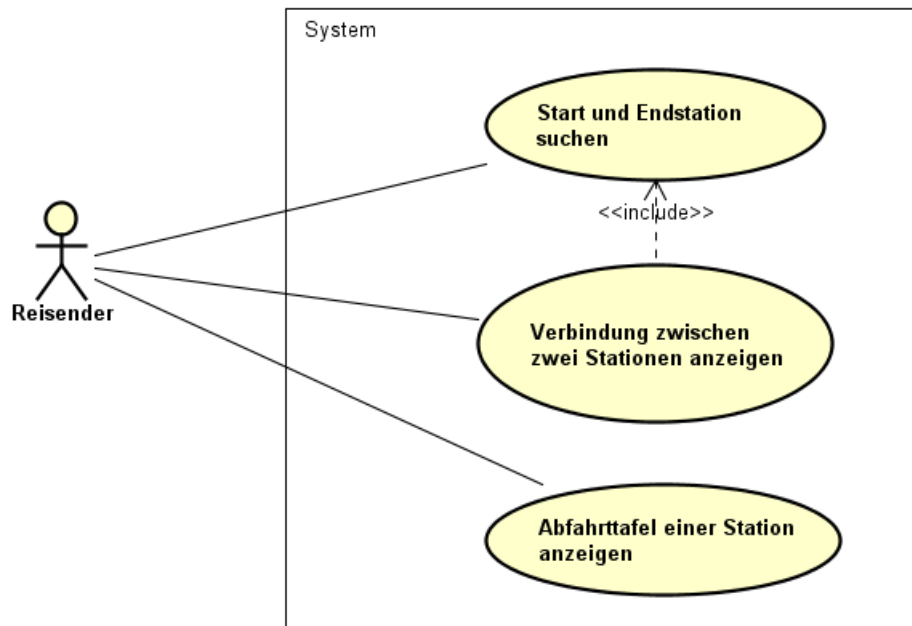
SBB Abfahrtstafel

Mein Standort:

Von	Nach	Abfahrt
Luzern	Bern	14.30
Luzern	Zürich	14.45
Luzern	Basel	14.55
Luzern	Sursee	15.00
Luzern	Emmen	15.15
Luzern	Sempach	15.45
Luzern	Oberkirch	16.00
Luzern	Bern	16.05
Luzern	Zürich	16.10
Luzern	Basel	16.20
Luzern	Sursee	16.45
Luzern	Emmen	16.55

3. Mein UseCaseDiagram

Das ist mein UseCase Diagramm das ich für meine Applikation erstellt habe. In diesem Diagramm sind die Funktionen der Applikation angegeben wie zum Beispiel «Abfahrtstafel einer Station anzeigen». Der Reisende ist ausserhalb des System.

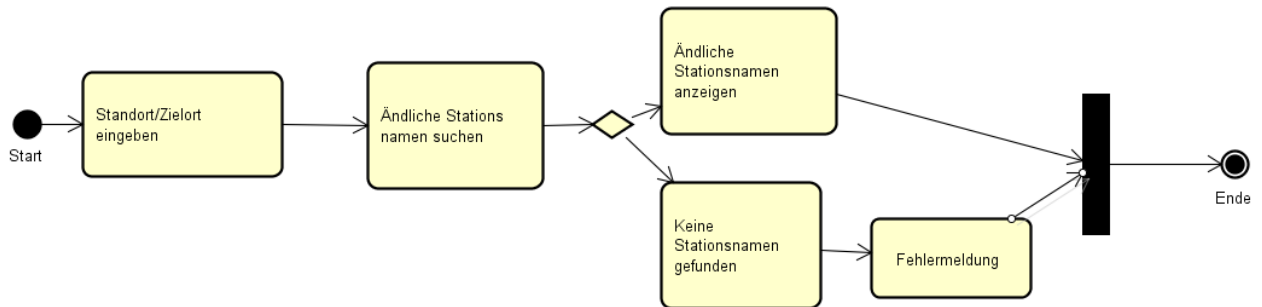


4. Zweck dieser Dokumentation

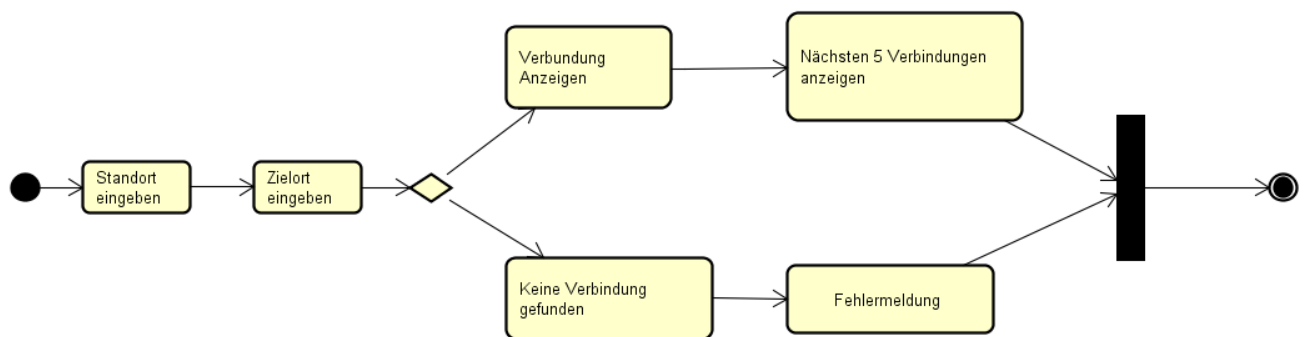
Der Zweck dieser Dokumentation dient in erster Linie einer Guten Note für mich im ÜK 318. Zusätzlich ist es für mich eine gute Vorbereitung für mein Praktikum.

5. Mein ActivityDiagram

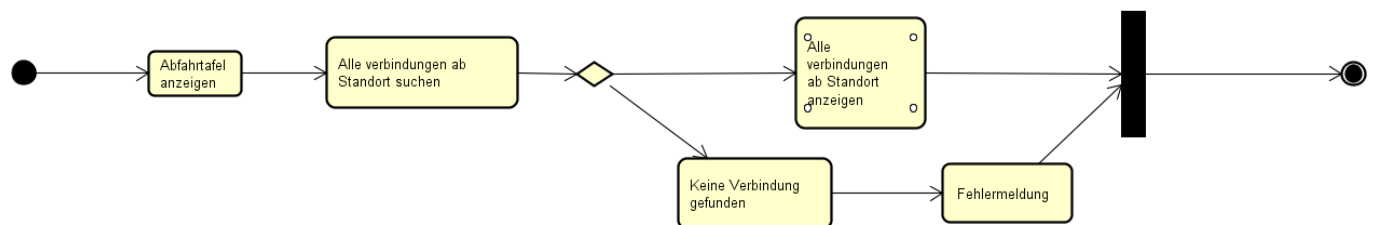
A001



A002



A003



6. Problem Analyse

Ich kam zu Beginn an viele Probleme an die ich nicht gedacht habe. Hier habe ich Zwei Beispiele mit Lösung.

Problem 1.

Nach einer Abfrage stürzt das Programm ab.


Ich habe nicht beachtet, dass eine riesen Anzahl von antworten zurückkommt und mein Programm deswegen abstürzt. Also habe ich die Antworten auf 10 reduziert somit kam mein Programm nach.

Problem 2.

Wenn ich eine Bus Verbindung suche stürzt das Programm ab.

Ich habe nach langem Suchen gesehen, wenn die Verbindung einen Bus beinhaltet dann gibt es keine Gleis Angabe. Daher war die Variable leer und konnte den Wert nicht weiter in die Tabelle geben. Deswegen kam es zu einem Absturz. Um das Problem zu lösen frage ich die Variable am Schluss ab ob sie leer ist oder nicht, wenn sie nicht leer ist dann werden die Daten in die Tabelle gegeben, wenn nicht wird ein Wert eingetragen so dass es zu keinem Absturz kommt.

7. Mein EndDesign

 Fahrplan

Abfahrts- und Ankunftsplan

Von: Tauschen Nach:

	Datum	Abfahrtszeit	Ankunftszeit	Gleis
▶	21.12.2018	11:12	11:38	14
	21.12.2018	11:42	12:08	14
	21.12.2018	12:06	12:24	12
	21.12.2018	12:12	12:38	14
	21.12.2018	12:42	13:08	14
	21.12.2018	13:06	13:24	12

21.12.2018 11:30:55

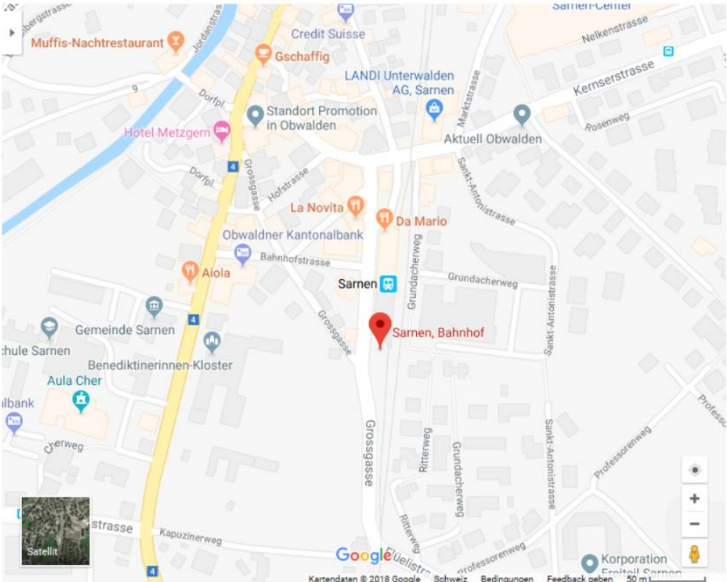
E_mail

Email Adresse:

Abfahrts- und Ankunftsplan

Mein Standort:

Nach	Abfahrt
Giswil	11:39
Luzern	11:49
Giswil	12:9
Luzern	12:19
Interlaken Ost	12:24
Luzern	12:35
Giswil	12:39
Luzern	12:49
Giswil	13:9
Luzern	13:19



8. Funktionen

ID	Beschreibung	Priorität
A001	Als ÖV-Benutzer möchte ich Start- und Endstation mittels Textsuche suchen können, damit ich nicht alle Stationsnamen auswendig lernen muss.	1 ✓
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.	1 ✓
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.	1 ✓
A004	Als ÖV-Benutzer möchte ich, dass schon während meiner Eingabe erste Such-Resultate erscheinen, damit ich effizienter nach Stationen suchen kann.	2 ✗
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.	2 ✓
A006	Als ÖV-Benutzer möchte ich sehen, wo sich eine Station befindet, damit ich mir besser vorstellen kann, wie die Situation vor Ort aussieht.	3 ✓
A007	Als ÖV-Benutzer möchte Stationen finden, die sich ganz in der Nähe meiner aktuellen Position befinden, damit ich schnell einen Anschluss erreichen kann.	3 ✗
A008	Ich möchte meine gefundenen Resultate via Mail weiterleiten können, damit auch andere von meinen Recherchen profitieren können.	3 ✓

Da ich zu Beginn schon wusste das ich nicht alle Anforderungen erreichen werde habe ich mir Ziele gesetzt. Zuerst waren die ersten Drei Funktionen wichtig A001, A002 und A003. Nachdem die erreicht waren konnte ich mir neue Ziele setzen. Da ich nicht mehr viel Zeit hatte habe ich mir die Funktionen ausgesucht die mir am leichtesten fallen das waren A005, A006 und A008.

9. Testing

Testfall «Endliche Stationen » A001

Vorbedingungen:

Ein Benutzer muss ein Zielort Suchen

Schritt	Aktivität	Erwartetes Resultat
1.	Klickt auf combobox .	Programm erwartet Eingabe.
2.	Ortsname eintragen. • Sursee	Keine Reaktion.
3.	Drop down Taste drücken.	Endliche Ortsnamen werden aufgelistet. • Sursee Bahnhof • Sursee Park • Usw.

Testfall «Verbindung» A002

Vorbedingungen:

Ein Benutzer muss ein Standort und Zielort haben.

Schritt	Aktivität	Erwartetes Resultat
1.	Klickt auf combobox .	Programm erwartet Eingabe.
2.	Ortsname eintragen. • Sursee • Luzern	Keine Reaktion.
3.	Drop down Taste drücken.	Endliche Ortsnamen werden aufgelistet. • Sursee Bahnhof • Sursee Park • Usw.
4.	Klick auf Verbindung suchen.	5 Verbindungen werden zurückgegeben. • Ab09.18An09.45 • Ab09.32An09.55

Testfall «Abfahrtstafel» A003

Vorbedingungen:

Ein Benutzer muss ein Standort haben.

Schritt	Aktivität	Erwartetes Resultat	
1.	Klickt auf combobox .	Programm erwartet Eingabe.	
2.	Ortsname eintragen. Sursee	Keine Reaktion.	
3.	Drop down Taste drücken.	Endliche Ortsnamen werden aufgelistet. Sursee Bahnhof Sursee Park Usw.	
4.	Klick auf Suchen.	Alle abgehenden Verbindungen werden angezeigt. Olten 09.26 Luzern 09.32 Luzern 09.41 Usw.	

Testfall «Andere Zeit/Datum» A005

Vorbedingungen:

Ein Benutzer muss ein Zielort Suchen

Schritt	Aktivität	Erwartetes Resultat	
1.	Klickt auf combobox .	Programm erwartet Eingabe.	
2.	Ortsname eintragen. <ul style="list-style-type: none"> • Sursee • Luzern 	Keine Reaktion.	
3.	Drop down Taste drücken.	Endliche Ortsnamen werden aufgelistet. <ul style="list-style-type: none"> • Sursee Bahnhof • Sursee Park • Usw. 	
	Datum aussuchen. <ul style="list-style-type: none"> • 25.12.2018 	Datum wird angenommen.	
	Zeit aussuchen. <ul style="list-style-type: none"> • 20.00 Uhr 	Zeit wird angenommen.	
	Klick auf Verbindung suchen.	Die richtigen Verbindungen wurden ausgegeben. <ul style="list-style-type: none"> • 21.12.18 Ab 19.48 An 20.15 • 21.12.18 Ab 20.11 An 20.30 • 21.12.18 Ab 20.18 An 20.45 	

Testfall «Google Maps» A006

Vorbedingungen:

Ein Benutzer muss ein Zielort Suchen

Schritt	Aktivität	Erwartetes Resultat
1.	Klickt auf combobox .	Programm erwartet Eingabe.
2.	Ortsname eintragen. <ul style="list-style-type: none"> • Sursee • Luzern 	Keine Reaktion.
3.	Drop down Taste drücken.	Endliche Ortsnamen werden aufgelistet. <ul style="list-style-type: none"> • Sursee Bahnhof • Sursee Park • Usw.
4.	Klick auf Suchen.	Alle abgehenden Verbindungen werden angezeigt. Standort wird in Google maps angezeigt

Testfall «Verbindungsdaten per Mail Senden» A008

Vorbedingungen:

Ein Benutzer muss ein Zielort Suchen.

Schritt	Aktivität	Erwartetes Resultat
1.	Klickt auf combobox .	Programm erwartet Eingabe.
2.	Ortsname eintragen. <ul style="list-style-type: none"> • Sursee • Luzern 	Keine Reaktion.
3.	Drop down Taste drücken.	Endliche Ortsnamen werden aufgelistet. <ul style="list-style-type: none"> • Sursee Bahnhof • Sursee Park • Usw.
4.	Klick auf Verbindung suchen.	5 Verbindungen werden zurückgegeben. <ul style="list-style-type: none"> • Ab09.11An09.30 • Ab09.18An09.45 • Ab09.32An09.55 • Usw.
	Klick auf Weiterleiten	Neues Fenster wird geöffnet
	Email-Adresse eintragen <ul style="list-style-type: none"> • beat.huwyler@eduone.ch 	Email wird an angegebene Adresse versendet.
	Email Checken	Email Erfolgreich erhalten.

10. Code Richtlinie

Folgende Richtlinien versuche ich beim Programmieren einzuhalten. Jedoch gelingt es mir nicht immer.

10.1 Naming Conventions

Ich versuch immer die Namensgebung möglichst logisch zu machen. Die Variablen platziere ich am Ort wo sie verwendet wird. Auch Methoden und Klassen gebe ich möglichst logische Namen.

10.2 Declaration

Ich versuche Globale Variablen zu vermeiden nur wenn es nötig ist.

10.3 Comments

Ich Kommentiere in meinem Code nur das nötigste. So das ich weiss was in dem Code teil gemacht wird.

10.4 Statements

Ich verwende am meisten ein (if-else) da es mir am logischen vorkommt.

11. Installation

1. ZIP Datei entpacken
2. Im entpackten Ordner in den Pfad Modul 318\Projektarbeit\modul-318-student-master\Fahrplan wechseln
3. Die Datei Fahrplan.msi ausführen
4. Den Angaben im Installationsfenster folgen
5. Auf dem Desktop Fahrplan ausführen

12. Deinstallation

1. Unter Start/ Einstellungen auf Apps gehen
2. In der Auflistung nach der App Fahrplan suchen
3. Auf Deinstallieren klicken

12. Schlusswort

Mir hat dieses Projekt sehr gefallen obwohl ich am Anfang ein wenig überfordert war. Nach einigem Ausprobieren wurde mir sehr schnell klar wie man manche Anforderungen lösen kann. Da ich meine Ausbildung auf schulischem Wege mache war es mal was Anderes eine ganze Woche nur an einem Thema zu arbeiten.