Polgar Daniel

[Firmenname]

[Datum]

Dokumentation SBB App

Inhalt

[Anforderungen 2](#_Toc34916341)

[GUI-Entwurf 2](#_Toc34916342)

[Use Case Diagramm 4](#_Toc34916343)

[Use Case Beschreibung 4](#_Toc34916344)

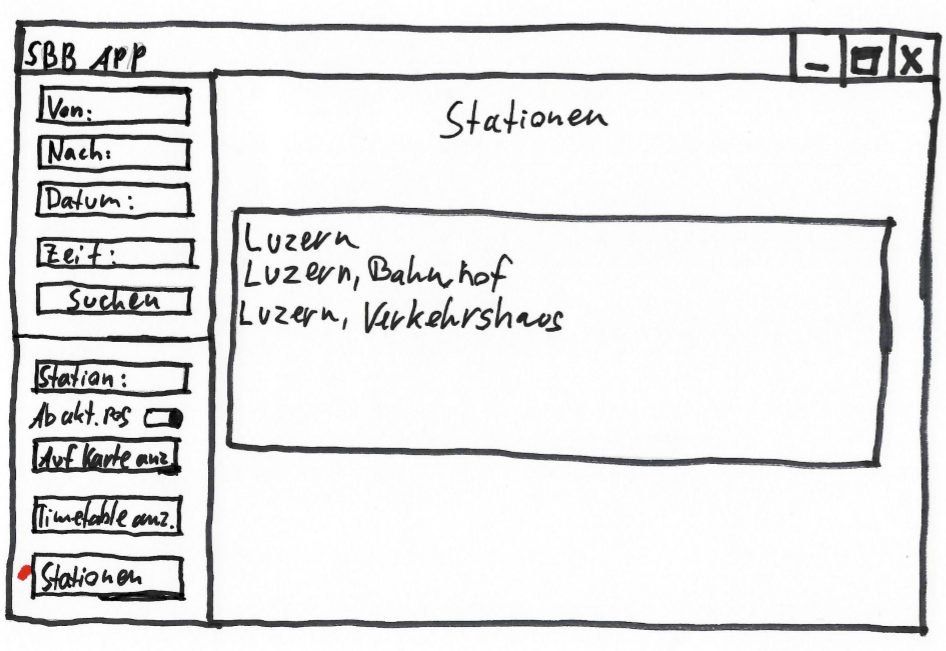
[Testfälle 8](#_Toc34916345)

[Programm Richtlinien 10](#_Toc34916346)

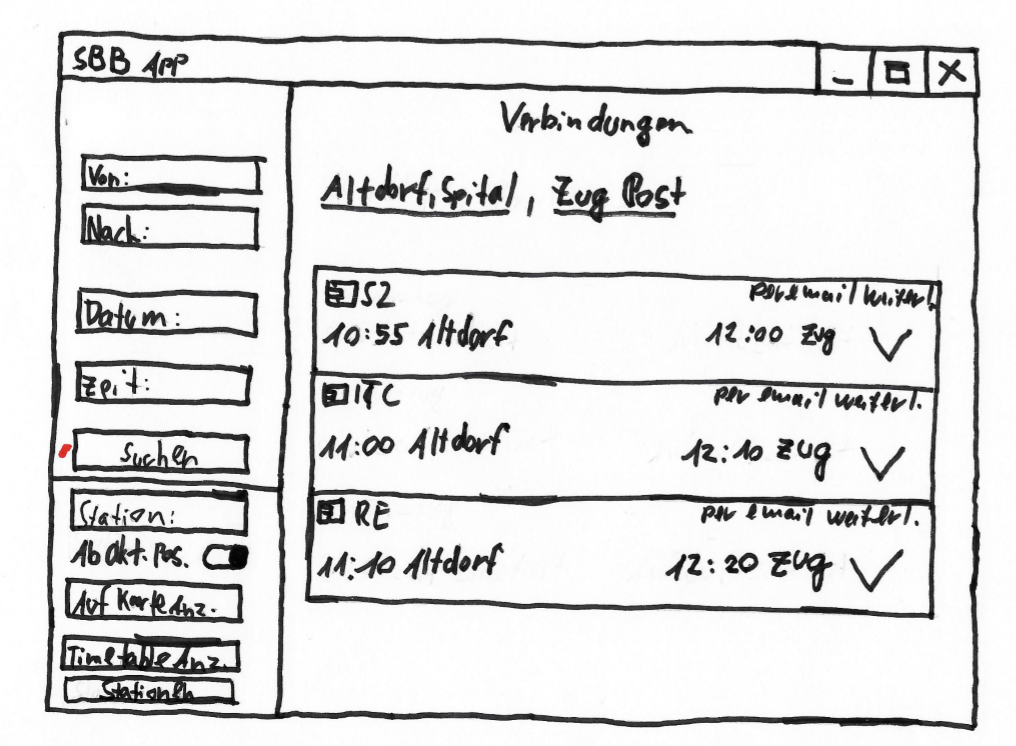
|  |  |  |  |
| --- | --- | --- | --- |
| ID | Beschreibung | Priorität | Umgesetzt? |
| A001 | Als ÖV-Benutzer möchte ich Start- und Endstation mittels Textsuche suchen können, damit ich nicht alle Stationsnamen auswendig lernen muss. | 1 | ja |
| A002 | Als ÖV-Benutzer möchte ich die aktuellen, d.h. 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 | ja |
| 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 | ja |
| A004 | Als ÖV-Benutzer möchte ich, dass schon während meiner Eingabe erste Suchresultate erscheinen, damit ich effizienter nach Stationen suchen kann. | 2 | nein |
| 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 | ja |
| 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 | ja |
| 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 | nein |
| A008 | Ich möchte meine gefundenen Resultate via Mail weiterleiten können, damit auch andere von meinen Recherchen profitieren können. | 3 | ja |

## Anforderungen

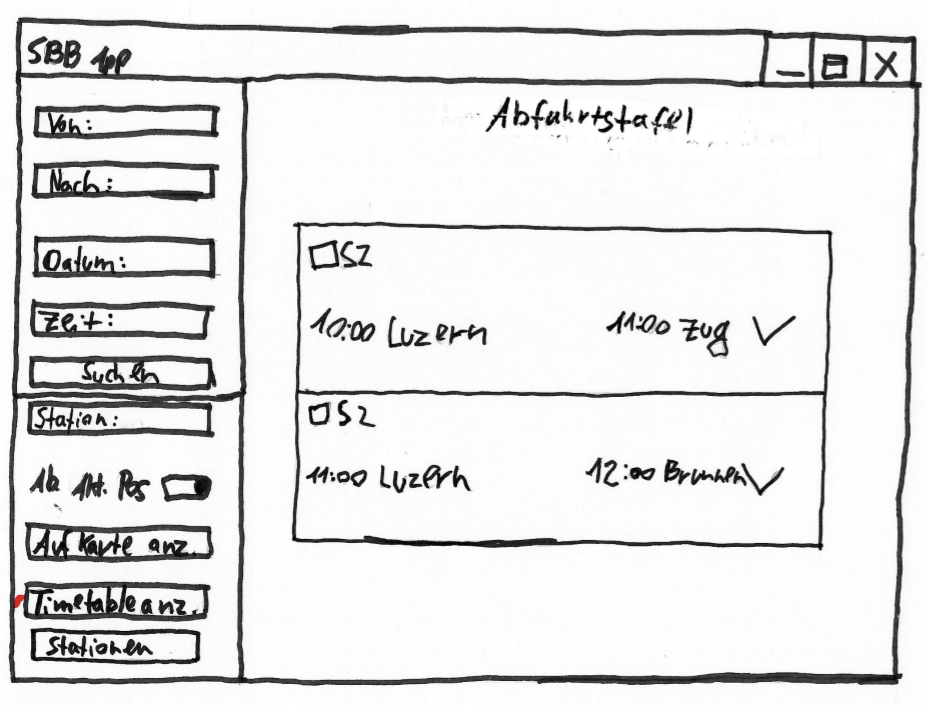
## GUI-Entwurf



A001

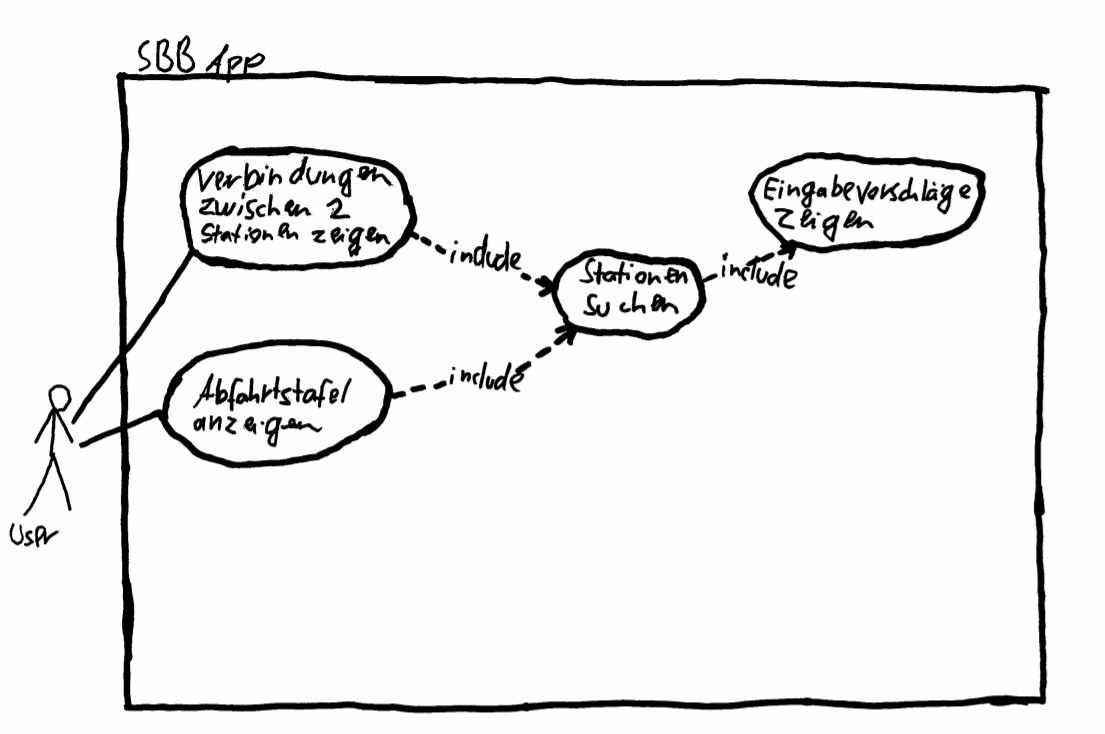


A002



A003

## Use Case Diagramm



## Use Case Beschreibung

|  |  |
| --- | --- |
| Use Case | 1, Verbindungen zwischen 2 Stationen zeigen |
| Beschreibung | 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. |
| Akteur(e) | Der Benutzer |
| Auslöser | Die «Suchen» Button |
| Vorbedingungen | Die «Suchen» Button klicken |
| Ablauf | 1, SBB App Starten  2, in «Von» oder/und «Nach» ein gültiger Text eingeben  3, Die «Suchen» Button klicken |
| Ergebnis | Der ÖV-Benutzer sieht 4-5 Verbindungen zwischen den 2 eingegebenen Stationen |

|  |  |
| --- | --- |
| Use Case | 2, Abfahrtstafel anzeigen |
| Beschreibung | Als ÖV-Benutzer möchte ich sehen, welche Verbindungen ab einer bestimmten Station vorhanden sind, damit ich bei mir zuhause eine Art Abfahrtstafel haben kann. |
| Akteur(e) | Der Benutzer |
| Auslöser | Die «Suchen» Button |
| Vorbedingungen | Die «Suchen» Button klicken |
| Ablauf | 1, SBB App Starten  2, in «Von» oder/und «Nach» ein gültiger Text eingeben  3, Die «Suchen» Button klicken |
| Ergebnis | Der ÖV-Benutzer sieht die Abfahrtstafel der eingegebenen Station |

|  |  |
| --- | --- |
| Use Case | 3, Stationen Suchen |
| Beschreibung | Als ÖV-Benutzer möchte ich alle Stationen finden können, die zu meiner Verbindung nötig sind. |
| Akteur(e) | Der Benutzer |
| Auslöser | Die «Suchen» Button |
| Vorbedingungen | Vollständige Eingaben in den Beiden Textfelder für die Stationen und «Suchen» Button Klick. |
| Ablauf | 1, SBB App Starten  2, in «Von» oder/und «Nach» ein gültiger Text eingeben  3, Die «Suchen» Button klicken |
| Ergebnis | Die Stationen, die für die Verbindungen nötig sind, werden gesucht. |

|  |  |
| --- | --- |
| Use Case | 4, Eingabevorschläge zeigen |
| Beschreibung | Als ÖV-Benutzer möchte ich Start- und Endstation mittels Textsuche suchen können, damit ich nicht alle Stationsnamen auswendig lernen muss. |
| Akteur(e) | Der Benutzer |
| Auslöser | Die Textfelder, wo die Eingaben stattfinden |
| Vorbedingungen | Eingabe im Textfeld |
| Ablauf | 1, SBB App Starten  2, in «Von» oder/und «Nach» ein Text eingeben |
| Ergebnis | Der ÖV-Benutzer sieht Eingabevorschläge für die Stationen |



A001:

Der Benutzer muss eine Station eingeben, dann auf dem «Station Optionen» drücken. Nachdem wird überprüft, ob die Eingabe richtig ist (z.B.: keine Sonderzeichen). Falls nein, wird eine Fehlermeldung kommen, was den Benutzer für die richtige Eingabe aufmerksam macht. Falls ja werden mit Hilfe von einem API die Daten gesucht. Wenn die API keine Treffer findet, wird eine Fehlermeldung kommen, was mit dem Benutzer mitteilt, dass es keine Treffer gibt. Wenn die API Treffer findet, werden sie ausgegeben.



A002:

Der Benutzer muss ein Start, ein Ziel, ein Datum und eine Zeit eingeben, dann auf dem «Suchen» drücken. Nachdem wird überprüft, ob die Eingaben richtig sind (z.B.: Zeit besteht aus Zahlen und hat ein Doppelpunkt). Falls nein, wird eine Fehlermeldung kommen, was den Benutzer für die richtige Eingabe aufmerksam macht. Falls ja werden mit Hilfe von einem API die Daten gesucht. Wenn die API keine Treffer findet, wird eine Fehlermeldung kommen, was mit dem Benutzer mitteilt, dass es keine Treffer gibt. Wenn die API Treffer findet, sie werden für die Ausgabe vorbereitet, dann ausgegeben.



A003:

Der Benutzer muss ein Start, ein Ziel, ein Datum und eine Zeit eingeben, dann auf dem «Suchen» drücken. Nachdem wird überprüft, ob die Eingabe richtig ist (z.B.: keine Sonderzeichen). Falls nein, wird eine Fehlermeldung kommen, was den Benutzer für die richtige Eingabe aufmerksam macht. Falls ja werden mit Hilfe von einem API die Daten gesucht. Wenn die API keine Treffer findet, wird eine Fehlermeldung kommen, was mit dem Benutzer mitteilt, dass es keine Treffer gibt. Wenn die API Treffer findet, sie werden für die Ausgabe vorbereitet, dann ausgegeben.

## Testfälle

A001test

|  |  |  |
| --- | --- | --- |
| Schritt | Wenn… | Dann… |
| 1 | das App gestartet wird mit verfügbarem Internet. | Die GUI soll erscheinen |
| 2 | «Luzern» in «Station» eingegeben wird und auf «Station Optionen» gedrückt wird | sollen mehrere Treffer, kommen, die «Luzern» beinhalten. Output:  **Luzern Luzern, Bahnhof**  **Luzern, Pilatusplatz**  **Luzern, Schwahnenplatz**  **Luzern, Luzernerhof**  **Luzern Allmend/Messe**  **Luzern, Gütsch**  **Luzern, Kasernenplatz**  **Luzern, Eichhof** |

A002test

|  |  |  |
| --- | --- | --- |
| Schritt | Wenn… | Dann… |
| 1 | das App gestartet wird mit verfügbarem Internet. | Die GUI soll erscheinen |
| 2 | «Luzern» in «von», «Sursee» in «bis», «12.03.2020» in «Datum» und «14:00» in «Zeit» eingegeben wird und auf «Suchen» gedrückt wird | sollen mehrere Verbindungen kommen, die von Luzern bis Sursee fahren. Output:  **2020-03-12 | 14:00 Luzern --> Sursee 14:17**  **2020-03-12 | 14:05 Luzern --> Sursee 14:24**  **2020-03-12 | 14:14 Luzern --> Sursee 14:40**  **2020-03-12 | 14:30 Luzern --> Sursee 14:47** |

A003test

|  |  |  |
| --- | --- | --- |
| Schritt | Wenn… | Dann… |
| 1 | das App gestartet wird mit verfügbarem Internet. | Die GUI soll erscheinen |
| 2 | «Luzern» in «Station» eingegeben wird und auf «Abfahrtstafen» gedrückt wird | sollen mehrere Verbindungen kommen, die gemäss dem aktuellen Datum und Uhrzeit von Luzern fahren. Output ist immer etwas anderes, kann man beschränkt beeinflussen. |

A005test

|  |  |  |
| --- | --- | --- |
| Schritt | Wenn… | Dann… |
| 1 | das App gestartet wird mit verfügbarem Internet. | Die GUI soll erscheinen |
| 2 | «Luzern» in «von», «Sursee» in «bis», «12.03.2020» in «Datum» und «14:00» in «Zeit» eingegeben wird und auf «Suchen» gedrückt wird | sollen mehrere Verbindungen kommen, die von Luzern bis Sursee fahren. Output:  **2020-03-12 | 14:00 Luzern --> Sursee 14:17**  **2020-03-12 | 14:05 Luzern --> Sursee 14:24**  **2020-03-12 | 14:14 Luzern --> Sursee 14:40**  **2020-03-12 | 14:30 Luzern --> Sursee 14:47** |

A006test

|  |  |  |
| --- | --- | --- |
| Schritt | Wenn… | Dann… |
| 1 | das App gestartet wird mit verfügbarem Internet. | Die GUI soll erscheinen |
| 2 | «Luzern» in «Station» eingegeben wird und auf «online Karte anzeigen» gedrückt wird | die Standard Browser soll öffnen in Google Maps bei Luzern, also die Koordinaten: **47.050174, 8.310185** |

A008test

|  |  |  |
| --- | --- | --- |
| Schritt | Wenn… | Dann… |
| 1 | das App gestartet wird mit verfügbarem Internet. | Die GUI soll erscheinen |
| 2 | «Luzern» in «von», «Sursee» in «bis», «12.03.2020» in «Datum» und «14:00» in «Zeit» eingegeben wird und auf «Suchen» gedrückt wird | sollen mehrere Verbindungen kommen, die von Luzern bis Sursee fahren. Output:  **2020-03-12 | 14:00 Luzern --> Sursee 14:17**  **2020-03-12 | 14:05 Luzern --> Sursee 14:24**  **2020-03-12 | 14:14 Luzern --> Sursee 14:40**  **2020-03-12 | 14:30 Luzern --> Sursee 14:47** |
| 3 | «acb@abc.com» in «Empfänger Email» eingegeben wird und auf «in Email weiterleiten» gedrückt wird | die Standard Mail App soll öffnen mit dem Empfänger: [acb@abc.com](mailto:acb@abc.com) und Body: **2020-03-12 | 14:00 Luzern --> Sursee 14:17**  **2020-03-12 | 14:05 Luzern --> Sursee 14:24**  **2020-03-12 | 14:14 Luzern --> Sursee 14:40**  **2020-03-12 | 14:30 Luzern --> Sursee 14:47** |

## Test Protokoll

|  |  |
| --- | --- |
| Testfall | Ergebnis |
| A001test | Erfolgreich |
| A002test | Erfolgreich |
| A003test | Erfolgreich |
| A005test | Erfolgreich |
| A006test | Erfolgreich |
| A008test | Erfolgreich |

## Programm Richtlinien

Naming Conventions:

-Pascal Case: Klassen

- Camel Case: Variablen, Methoden, Eigenschaften, GUI-Controls

- Für jede Klasse gibt es ein separater File.

- Die Methoden sind nach ihrer Funktion genannt.

Declaration

- Die Membervariablen sind ganz oben unter dem Konstruktor.

- Die Unit Testrelevante Methoden sind sicher Public.

- Die Kontroller Methoden sind Private.

- Die Methoden von Klassen, die im Kontroller aufgerufen werden, sind Public.

- Die geschweiften Klammern sind in eine neue Zeile bei der Methode Deklaration.

Pro Methode möglichst wenig Zeilen.

Comments

- Wird, für die Unterscheiden von verschiedenen Bereichen benötigt (z.B.: Membervariablen, Kontrol Methoden, Validierung, usw.).

Statements

- Bei for Scheifen ein Buchstabe brauchen

- Innerhalb von Statements sind die Zeilen eingerückt

- Die geschweiften Klammern sind in eine neue Zeile bei den Statements.