

|  |
| --- |
| Projekt SBB ÜK Modul 318 |
|  |
| 18. Dezember  Autor  Nikolic Nikola  17.12.2019 |

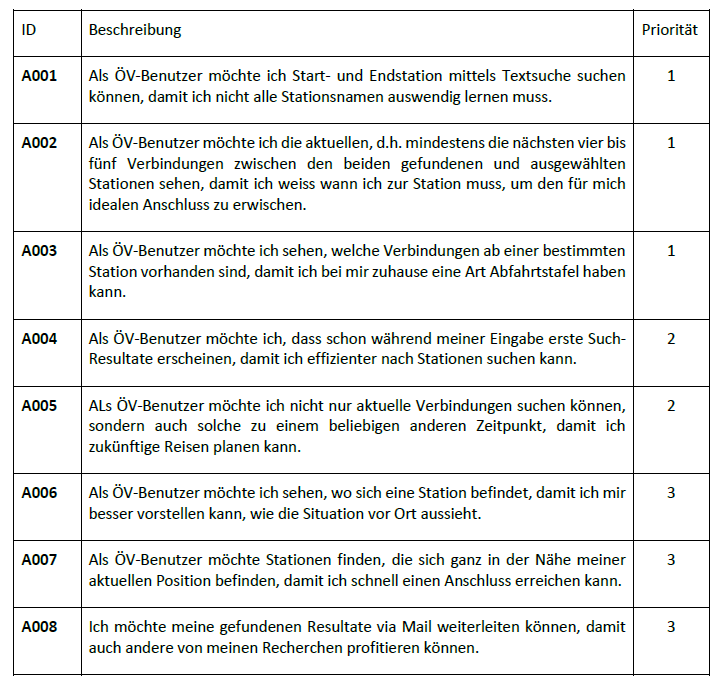
[](https://www.google.com/imgres?imgurl=https%3A%2F%2Fcdn.handelszeitung.ch%2Fsites%2Fdefault%2Ffiles%2Fstyles%2F16x9_280%2Fpublic%2Fjapan_transport_bullet_train.jpg&imgrefurl=https%3A%2F%2Fwww.handelszeitung.ch%2Funternehmen%2Fjapan-testet-den-schnellsten-zug-der-welt&docid=11VX9DJKiP87FM&tbnid=5729cmetGWQlBM%3A&vet=10ahUKEwjWnZaD77zmAhVT6qYKHR1iD2sQMwiEASgpMCk..i&w=420&h=236&bih=751&biw=1536&q=zug&ved=0ahUKEwjWnZaD77zmAhVT6qYKHR1iD2sQMwiEASgpMCk&iact=mrc&uact=8)

# Index

* Anforderungen
* Davon Ausgefüllte Anforderungen
* Use Case Diagramm
* Use Cases Beschreibung
* Aktivitätsdiagramm
* Einleitung
* Zweck
* Mockups
* Testfälle
* Testprotokoll
* Installationsanleitung
* Programmierrichtlinien
* EVT Bilder

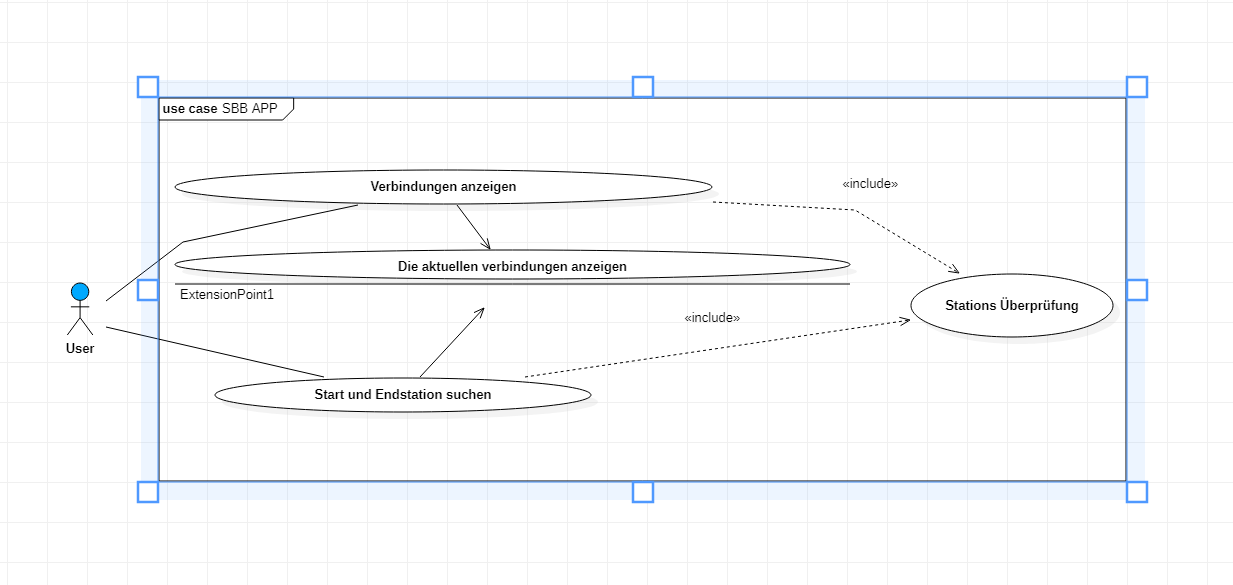
Anforderungen

1 = Muss 2 = Soll 3 = nice to have



Davon Ausgefüllte Anforderungen

Use Case Diagramm

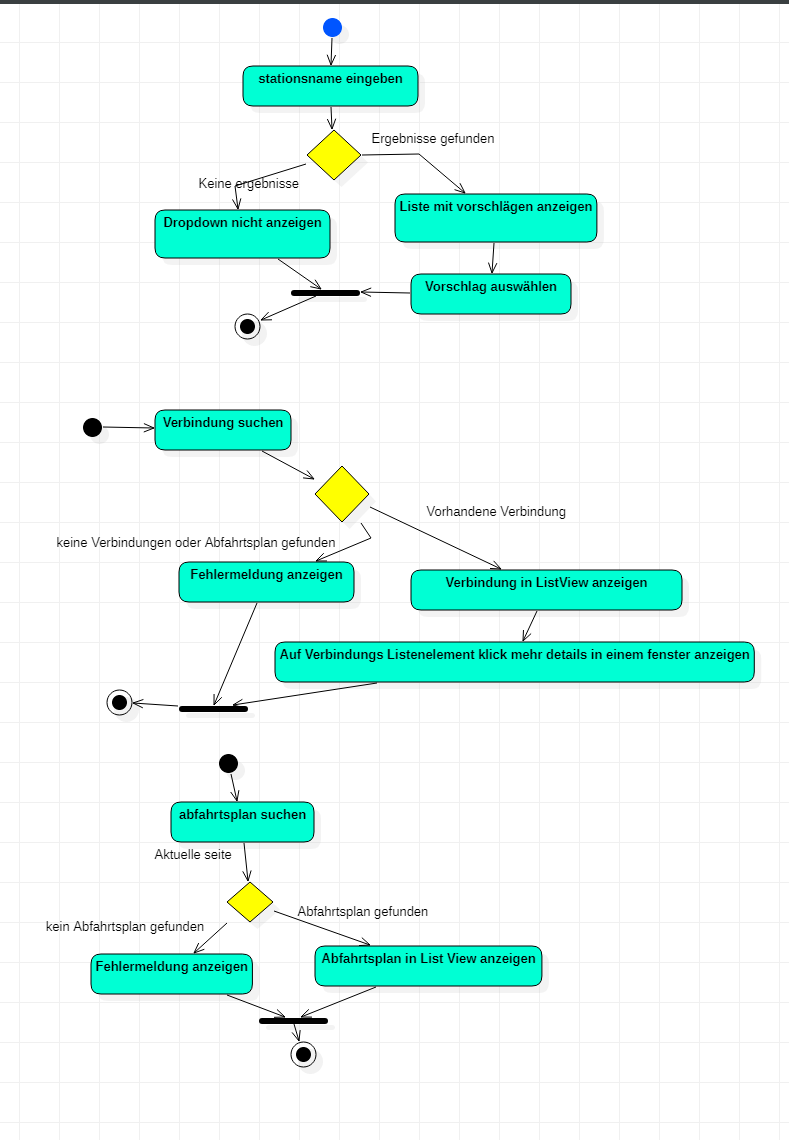


Use Case Beschreibung

|  |  |
| --- | --- |
| Use Case | 1. Verlängerung erfassen |

|  |  |
| --- | --- |
| Beschreibung | Ein Kunde / Mitarbeiter der SBB «Die Schweizerische Bundesbahnen AG» möchte wissen wo es in der Nähe Bus / Zugverbindungen gibt und welche Verbindungen wohin führen |
| Akteur | Kunde der SBB sowie jede öffentliche Person und Mitarbeiter der SBB |
| Auslöser | Interesse am abfahrtsplan und Verbindungen zwischen strecken und ab einer Station. |
| Vorbedingungen | Der Benutzer muss lesen und schreiben können |
| Ablauf | 1. Programm installieren 2. Programm öffnen 3. Station suchen 4. Verbindungen ansehen 5. Hingehen 6. In den Zug / Bus steigen und losfahren |
| Alternativer Ablauf | Kein |
| Ergebnis | Fahrkarte wurde gekauft und man kann sich in den Bus oder Zug setzen und losfahren |

Aktivitätsdiagramm



Einleitung

Über dieses Projekt

In diesem Projekt geht es um ein Programm das den Benutzern ermöglicht eine Verbindung zwischen Punkt A und punkt B, ab einer bestimmten Station oder in der nähe des Benutzers eine Station und deren aus- und eingehenden Verbindungen zu suchen.

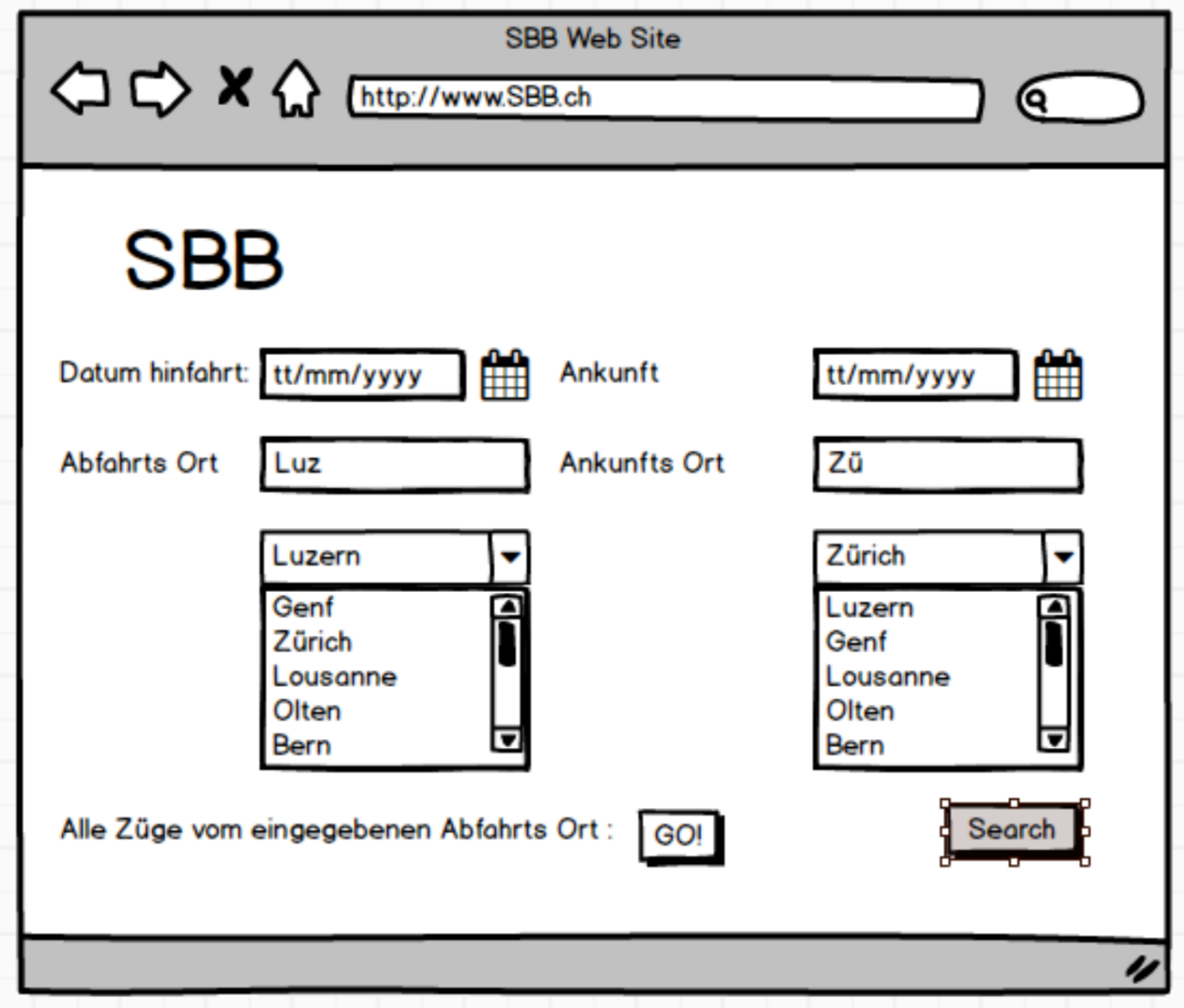
Eigenschaften

Es können zwei Stationen oder Orte (Startort und Ankunftsort) angegeben werden. Standardmäßig werden das aktuelle Datum und Zeit eingefüllt, jedoch kann das Abfahrtsdatum und die Abfahrtszeit beliebig angegeben werden. Wird nach Verbindungen gesucht werden von der API die nächsten Verbindungen anhand der Eingaben zurückgegeben, welche in die Liste gefüllt werden. Es werden pro Verbindung die Startort Station und Abfahrtszeit sowie der Ankunftsort und Ankunftszeit angezeigt. Wird ein Listenelement angewählt, erscheint ein neues Fenster mit mehr Details (Der Start Station die genaue Abfahrtszeit sowie das Gleis, die Reisezeit und der Ankunftsort mit genauer Ankunftszeit und Angaben des Ankunftsgleises). Bei Falschen Angaben erscheint eine Nachricht, welche die fehlerhaften Angaben zurückmeldet.

Zweck

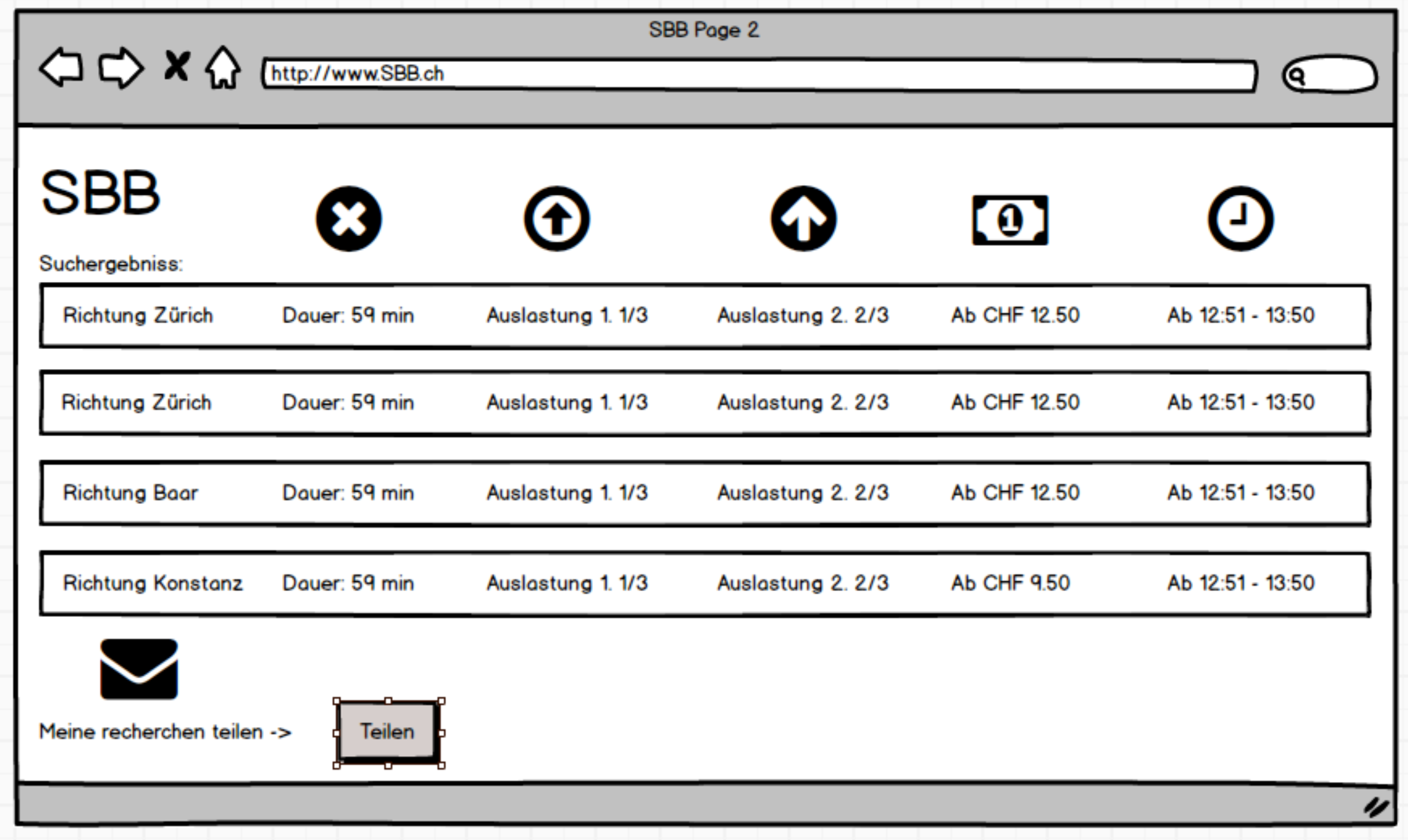
Mockups

Main Site



Mockups

Second Site



Mockups

Nearest Station



Testfälle

Testprotokoll

Installationsanleitung

Programmierrichtlinien

* Verwenden der üblichen Vorgehensweisen im gewählten Programmierparadigma
* Festlegung von Namenskonventionen: Wie sind Bezeichner zu wählen?
* Anwendung von Entwurfsmustern
* Verwendung von Compilerdirektiven und -Schaltern
* Initialisieren von Variablen
* Zugriff auf Variable fremder Objekte / Prozeduren
* Gestaltung von Funktionsaufrufen (Parameterübergaben, Rückgabewerte)
* pflichtgemäß zu verwendende Standardkomponenten, wie Unterprogramme, APIs etc.
* Vermeidung von Redundanz und möglichst breite Wiederverwendbarkeit – durch Modularisierung
* Unabhängigkeit verschiedener Programmteile (Modularität)
* Einheitlichkeit bei der Lösung gleichartiger Probleme, z. B. durch Normierte Programmierung
* Robustheit durch ausführliche Fehler- und Ausnahmebehandlung
* Umfang und Form der Dokumentation: Je Prozedur, je Zeile abgestimmt auf weitere Dokumente