Digital ÖV

Projektarbeit ÜK M318

Von Dominik Arnet



Inhaltsverzeichnis

[1 Einleitung 3](#_Toc499040190)

[2 Use Case Diagramm 3](#_Toc499040191)

[2.1 Grundanforderungen 3](#_Toc499040192)

[2.2 Alle Anforderungen 3](#_Toc499040193)

[3 Activity Diagramm 4](#_Toc499040194)

[3.1 Grundanforderungen 4](#_Toc499040195)

[3.2 Alle Anforderungen 4](#_Toc499040196)

[4 GUI Mockup 5](#_Toc499040197)

[5 Code Conventions 7](#_Toc499040198)

[5.1 Naming Conventions 7](#_Toc499040199)

[5.2 Declaration 7](#_Toc499040200)

[5.3 Comments 7](#_Toc499040201)

[5.4 GUI 7](#_Toc499040202)

[6 Was wurde umgesetzt? 7](#_Toc499040203)

[7 Systemtests 7](#_Toc499040204)

[7.1 Vorbedingungen 7](#_Toc499040205)

[7.2 A001 Textsuche 7](#_Toc499040206)

[7.3 A002 Verbindungen suchen 8](#_Toc499040207)

[7.4 A003 Abfahrtstafel 8](#_Toc499040208)

[7.5 A004 Autocomplete 8](#_Toc499040209)

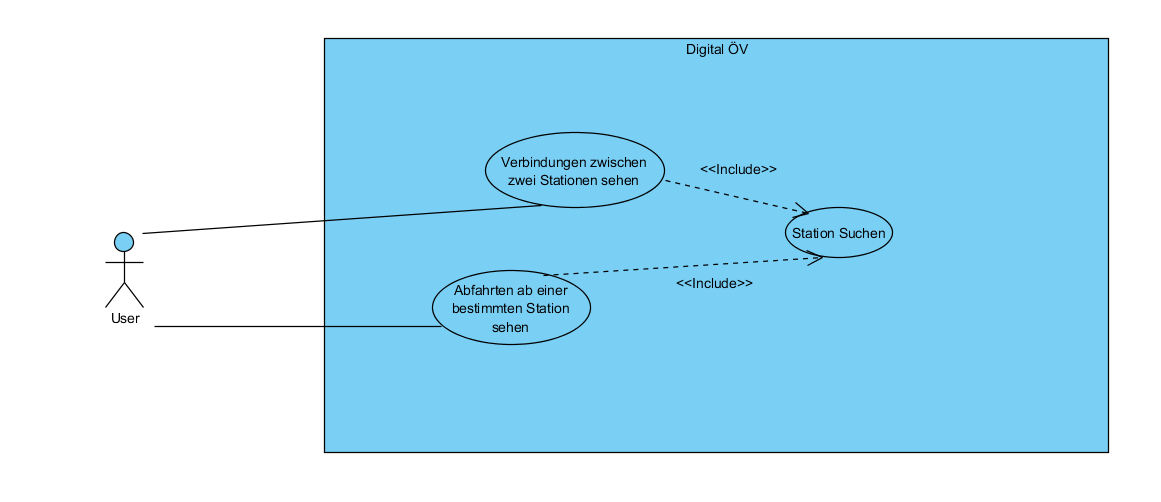
# Einleitung

Im ÜK Modul 318 bestand die Projektarbeit darin, eine Desktop Applikation zu entwickeln, um unteranderem ÖV Verbindungen zu suchen. Dieses Dokument dokumentiert die Entwicklung der Digital ÖV Desktop Applikation und wie sie zu installieren und zu deinstallieren ist in der Anleitung.docx beschrieben.

# Use Case Diagramm

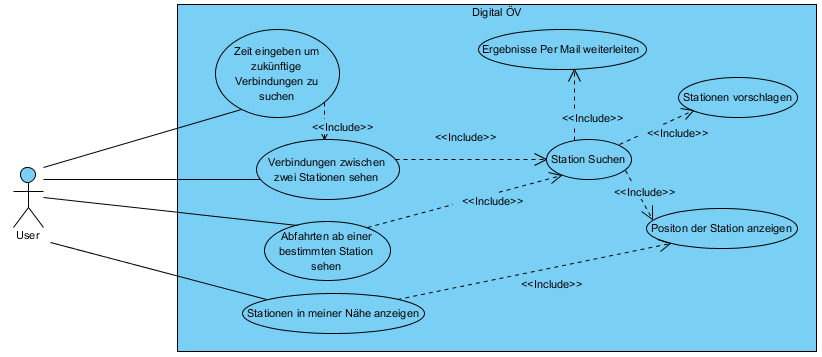
In diesem Kapitel wird die Funktionsweise der Applikation mithilfe von UseCase Diagrammen dargestellt.

## Grundanforderungen



Oben ist das Use Case Diagramm der Grundanforderungen zu sehen.

## Alle Anforderungen

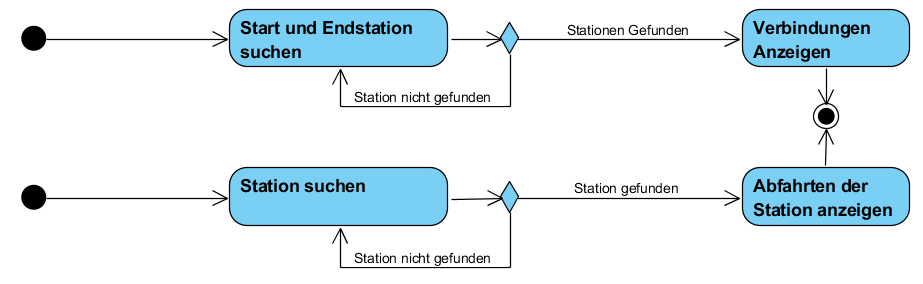


Hier ist das Use Case Diagramm von allen Anforderungen (Priorität 1,2 und 3).

# Activity Diagramm

In diesem Kapitel wird die Funktionsweise der Applikation mithilfe von Activity Diagrammen dargestellt.

## Grundanforderungen



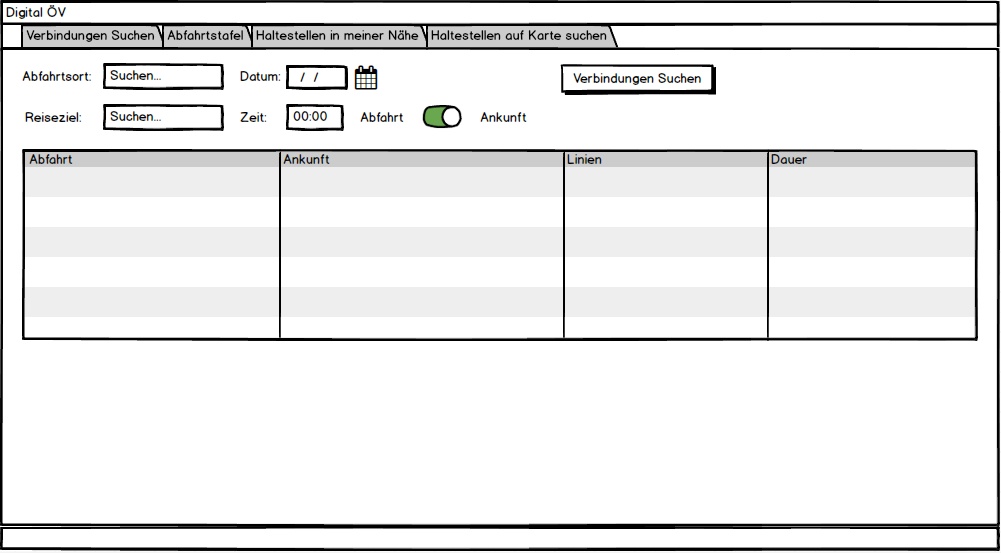
Activity Diagramm der Grundanforderungen.

## Alle Anforderungen

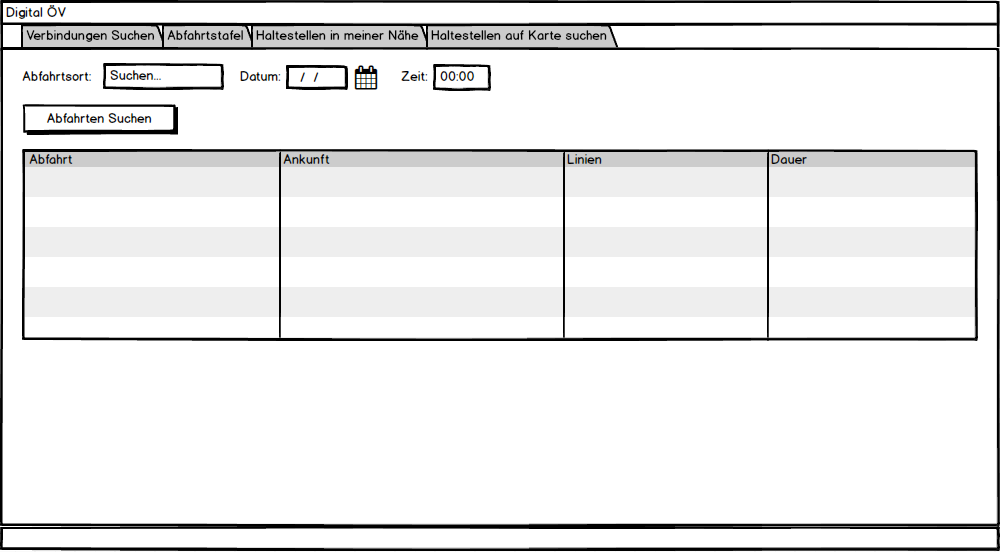
Activity Diagramm von allen Anforderungen.

# GUI Mockup

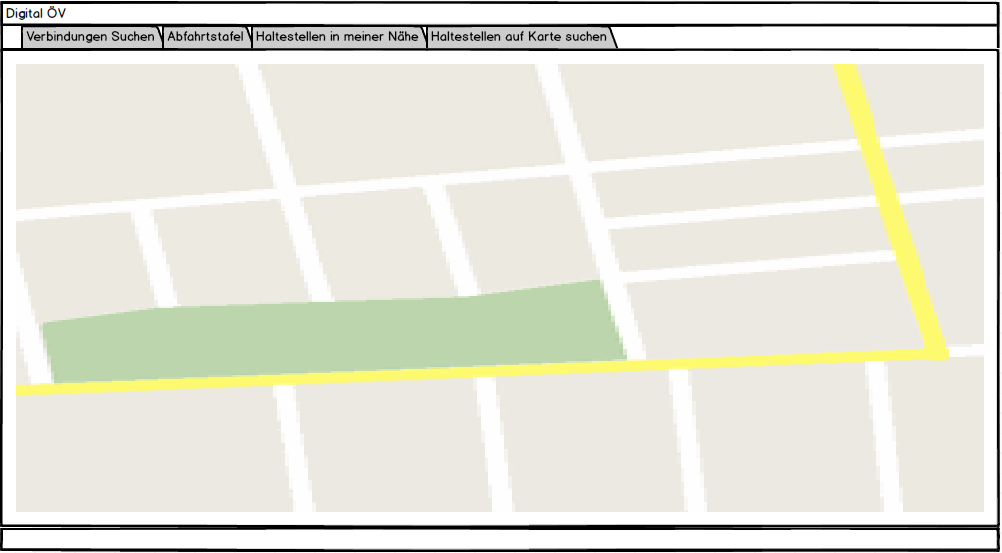
Das GUI wird in vier Registerkarten aufgeteilt. Jede Registerkarte übernimmt eine spezifische Anforderung.



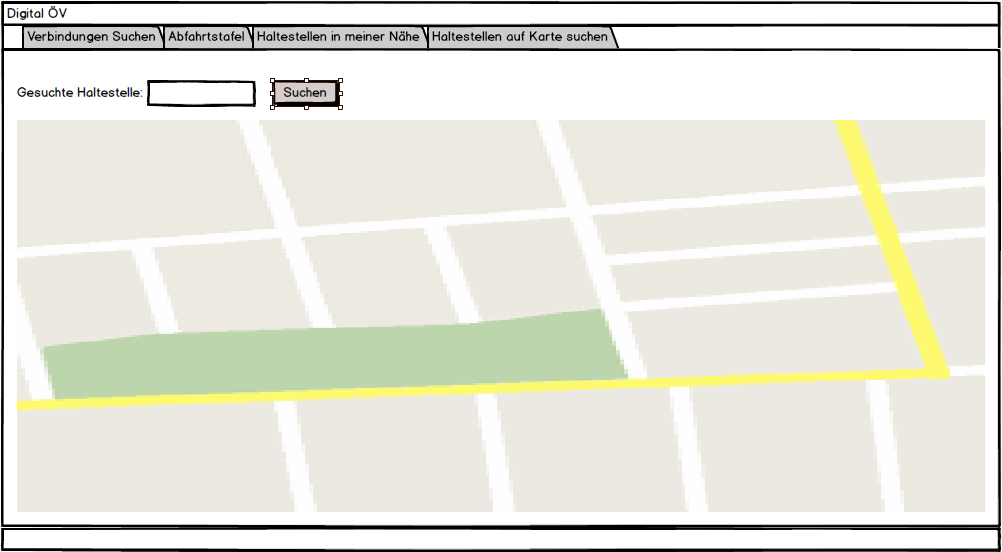
Verbindungen suchen und Anzeigen.



Abfahrtstafel.



Haltestellen in der Nähe anzeigen.



Haltestellen auf Karte suchen.

# Code Conventions

Ein Punkt der Bewertung ist die Einhaltung der selbst gesetzten Code Conventions.

## Naming Conventions

Als Naming Convention verwende ich CamelCase. Variablen, Methoden, Klassen und Controls müssen selbsterklärend benennt werden. Namen müssen alle Englisch sein. Controls müssen vor dem eigentlichen Namen des Controls ein Kürzel oder eine Beschreibung des Typs enthalten (z.B. Button = btn, Radiobutton=rb, Groupbox=gb, DataGrid= table, etc).

## Declaration

Variablen werden da deklariert wo sie zum ersten verwendet werden und immer auch gleich initialisiert.

## Comments

Nur schwierig verständliche Codepassagen werden kommentiert. Kommentare sind auf Deutsch. Vor Funktionen wird immer ein Summary angegeben.

## GUI

Im GUI soll darauf geachtet werden, dass die Abstände immer gleich sind (z.B. Abstand Label und Control).

# Was wurde umgesetzt?

Die Anforderungen A001 – A004 wurden komplett implementiert und funktionieren. Für A005 sind zwar Controls auf dem GUI vorhanden, jedoch keine Funktionalität. Für A007 und A008 wird die Bing Map dargestellt, aber sie haben keine Funktionalität, d.h. es werden keine Haltestellen in der Nähe angezeigt und es kann auch nicht nach Haltestellen gesucht werden auf einer Karte (Controls wären jedoch vorhanden auf dem GUI).

# Systemtests

Hier folgen die Systemtest zu den umgesetzten Anforderungen.

## Vorbedingungen

DigitalÖV Applikation muss gestartet und am laufen sein.

## A001 Textsuche

|  |  |  |
| --- | --- | --- |
| Schritt | Aktivität | Erwartetes Resultat |
| 1 | In Abfahrtsort «Luz» eingeben. | Dropdown erscheint mit Inhalt:  Luzern  Luzern, Bahnhof  Luzern, Kantonalbank  Luzern, Pilatusplatz  Luzern, Schwanenplatz  Luzern, Allmend/Messe  Luzern, Gütsch |

## A002 Verbindungen suchen

|  |  |  |
| --- | --- | --- |
| Schritt | Aktivität | Erwartetes Resultat |
| 1 | Bei Abfahrtsort «Buttisholz, Dorf» eintragen. (Darf mit Autocomplete eingetragen werden) |  |
| 2 | Als Reiseziel «Luzern, Bahnhof» eintragen. (Darf mit Autocomplete eingetragen werden) |  |
| 3 | «Verbindungen suchen» Button drücken | Ca. 4 Verbindungen zwischen Buttisholz, Dorf und Luzern mit Dauer 41:00 oder 48:00. |

## A003 Abfahrtstafel

|  |  |  |
| --- | --- | --- |
| Schritt | Aktivität | Erwartetes Resultat |
| 1 | Tab «Abfahrtstafel» auswählen. |  |
| 2 | Bei Abfahrtsort «Buttisholz, Dorf» eintragen. (Darf mit Autocomplete eingetragen werden) |  |
| 3 | «Abfahrten suchen» Button drücken. | Diverse Abfahrten der Linien 61 und 62 (aber auch andere). |

## A004 Autocomplete

|  |  |  |
| --- | --- | --- |
| Schritt | Aktivität | Erwartetes Resultat |
| 1 | In Abfahrtsort «Luz» eingeben. | Dropdown erscheint mit Inhalt:  Luzern  Luzern, Bahnhof  Luzern, Kantonalbank  Luzern, Pilatusplatz  Luzern, Schwanenplatz  Luzern, Allmend/Messe  Luzern, Gütsch |
| 2 | Mit den Pfeiltasten kann nun eine Station ausgesucht und mit Enter ausgewählt werden. |  |