**Dokumentation**

Swiss Transport

|  |  |
| --- | --- |
| Inhalt | **Dokumentation Swiss Transport** |
| Erstelldatum | **Dienstag, 26. November 2019** |
| Änderungsdatum | **Dienstag, 3. Dezember 2019** |
| Autor | **Janis Kneubühler** |

**Inhaltsverzeichnis**

[1 Aufgabenstellung 3](#_Toc532559986)

[2 Problemanalyse 3](#_Toc532559987)

[3 Grafische Darstellung 4](#_Toc532559988)

[4 Programmablaufplan 5](#_Toc532559989)

[5 Programmtest 6](#_Toc532559990)

[6 Abschluss 6](#_Toc532559991)

Es wird keine Dokumentation im eigentlichen Sinne verlangt. Es braucht jedoch im Minimum ein Dokument,

das die folgenden Informationen enthält.

OK - Autor, Datum.

- Eine Einleitung (Management Summary).

- Zweck des Dokuments.

- Was (d.h. welche Funktionen) wurde umgesetzt?

- Falls bestimmte Funktionen nur teilweise umgesetzt wurden: Welche? Welcher Teil der

Funktionalität fehlt noch? Bekannte Fehler/Bugs?

- Mockups für alle Formulare/Dialoge (GUI) , die benötigt werden um die Anforderungen A001, A002

und A003 abzudecken.

- Use Cases, Use Case Beschreibungen und Aktivitätendiagramme mindestens für die mit Priorität 1

klassifizierten Anforderungen.

- Testfälle (Systemtests), verständlich und eindeutig nachvollziehbar geschrieben, so dass ein Tester

diese ohne weiteren Erklärungen durchführen kann.

- Testprotokoll mit Durchführungsdatum der Tests, Name des Testers und Protokollierung der

tatsächlichen Testresultate.

- Installationsanleitung: Wie wird die Software installiert? Wie wird die Software deinstalliert?

- Andere spannende Informationen für die Bewertung.

Die Dokumentation soll sauber gestaltet, inhaltlich korrekt und übersichtlich strukturiert sein. Ein

Inhaltsverzeichnis soll vorhanden sein. Achte auf Typographie und Rechtschreibung!

# Einleitung

### Worum geht es?

Sagen Sie im Titel und mit den ersten Sätzen, um welches Thema es bei der Entscheidungsvorlage geht. Formulieren Sie möglichst genau, welche Entscheidung getroffen werden soll. Wenn die Entscheider es erwarten, kann hier schon die Handlungsempfehlung benannt sein.

Das Ziel dieser Arbeit ist es, ein Programm mit GUI herzustellen, welches ÖV-Verbindungen anzeigen soll. Diese Verbindungen werden mithilfe der zur Verfügung gestellten API gesucht. Der Kunde hat verschiedene Anforderungen, die das Programm haben soll, aufgeschrieben und priorisiert.

### Warum muss eine Entscheidung getroffen werden?

Erklären Sie kurz, warum das Thema so wichtig ist und warum überhaupt eine Entscheidung getroffen werden soll. Benennen Sie den Grund, der die Entscheidung nötig macht. Das kann ein konkreter Anlass sein, zum Beispiel eine Gesetzesänderung. Es können auch ungenutzte Potenziale sein oder Risiken, die drohen und vermieden werden sollten. Es können Kunden oder Mitarbeiter sein, die etwas erwarten, oder Produktmängel, die beseitigt werden müssen.

Diese Arbeit mitsamt den Anforderungen wurde vom ÜK vorgegeben. Welche Anforderungen eingebaut werden, ist mir selber überlassen. Jedoch sollten diese mit der Priorität 1 sicher vorhanden sein!

### Welche Ziele werden verfolgt?

Benennen Sie die wichtigsten Ziele, die mit der Entscheidung verbunden sind: Was soll verbessert, optimiert, eingespart, verringert, ausgebaut oder erreicht werden? Ergänzen Sie, in welcher Zeit ein Ziel erreicht werden soll oder für welchen Bereich es gilt. Benennen Sie auch die Rahmenbedingungen, die eingehalten werden müssen. Aus diesen Zielen und Rahmenbedingungen werden die Kriterien abgeleitet, die bei der Bewertung der Handlungsoptionen im Management Summary herangezogen werden.

Bis am Dienstag, 3. Dezember, möchte ich sicher mit den Anforderungen, welche die Priorität 1 haben, fertig sein. Danach geht es an die Verschönerung des GUIS, ans Schreiben der Dokumentation und erst dann werden zusätzliche Funktionen ins Programm eingebaut.  
Am Schluss möchte ich ein Programm ohne Bugs haben, falls nicht alle Funktionen eingebaut sind, ist dies kein Problem. Mir ist es wichtiger, dass die Qualität des Codes stimmt und dafür das Programm nicht ganz so umfangsreich ist.

### Welche Optionen gibt es?

Beschreiben Sie die verschiedenen Handlungsoptionen, aus denen die Entscheider eine auswählen können. Eine ausführliche Beschreibung der einzelnen Optionen ist in der Kurzfassung der Entscheidungsvorlage nicht möglich. Beschränken Sie sich deshalb auf die Bezeichnung und die wichtigsten Merkmale, die zum Verständnis notwendig sind. Hilfreich ist es, wenn Sie sich dabei an den W-Fragen orientieren (Was? Wer? Wie? Wann? Wo? Womit?). Möglicherweise gibt es nur eine Handlungsoption, die Sie mit der Vorlage vorschlagen. Doch auch dann gibt es eine zweite Handlungsoption: nichts tun und alles so lassen wie bisher.

### Welche Folgen haben die Optionen?

Machen Sie transparent, welche Folgen mit der jeweiligen Handlungsoption verbunden sind. Benennen Sie Erträge, Kosten, Vorteile, Nachteile, Chancen, Risiken, Stärken, Schwächen, Verbesserungen oder Verschlechterungen für jede einzelne Handlungsoption. Beziehen Sie sich dabei auf die oben genannten Ziele und Kriterien. Sagen Sie gegebenenfalls auch, was passiert, wenn nichts unternommen wird.

### Welche Option wird empfohlen?

Fassen Sie die Ergebnisse der Analyse und Bewertung in einer Empfehlung zusammen. Sagen Sie, welche Option aus Ihrer Sicht die beste ist, um die vorher genannten Ziele zu erreichen. Formulieren Sie die Empfehlung als Vorschlag und nicht als Aufforderung oder gar Anweisung.

# Zweck des Dokumentes

# Programmierrichtlinien

## Naming

|  |  |  |
| --- | --- | --- |
| Stelle | REgel | Beispiel |
| Allgemein | Namen auf Englisch, aussagen welchen Zweck sie haben,Camel Case |  |
| Variablen | Anfang klein, lokale und Membervariablen werden nicht unterschieden | numberOfConnections |
| Methoden | Anfang gross | AreStationFieldsFilled() |
| Klassen | Gross | Station |
| GUI | Vor Name Kürzel des GUI-Elements (Angegeben in Modul 305 Dokument) | btnSearchConnections |

## Declaration

|  |  |
| --- | --- |
| REgel | Beispiel |
| Variablen: am Anfang deklarieren und im Konstruktor definieren |  |

## Comments

Code sollte selbsterklärend sein -> nur kommentieren falls sehr kompliziert  
If/Else kommentieren  
Kompliziertere Methoden kommentieren  
Möglichst kurz und verständlich  
Comments auf Englisch

## GUI-Standards

Schriftart:

# Funktionen des Programmes

## Vollständig umgesetzt

## Teilweise umgesetzt

## Nicht umgesetzt

## Bugs

# Mockups

# Use-Cases

# Systemtests

# Testprotokoll

# Installationsanleitung

Installation vom GIT auch?