M318 Dokumentation

|  |  |  |
| --- | --- | --- |
| Modulnummer | 318 | |
| Kompetenz | Eine Aufgabe analysieren mit einer Entwicklungsumgebung mit integriertem GUI-Designer implementieren, dokumentieren und testen. | |
| Handlungsziele | 1. | Problemstellung analysieren, Benutzerschnittstelle entwerfen und Abläufe darstellen. |
|  | 2. | Grafische Benutzerschnittstelle gestalten und mit GUI-Designer realisieren. |
|  | 3. | Programmentwurf mit einer auf Komponenten basierenden Programmierumgebung mit Einsatz von Ablaufstrukturelementen, eigenen Prozeduren und Funktionen implementieren. |
|  | 4. | Beim Programmieren vorgegebene Codekonventionen einhalten, den Programmcode kommentieren und dabei auf Wartbarkeit und Nachvollziehbarkeit achten. |
|  | 5. | Ausgehend von der Aufgabenstellung Testfälle erstellen, das Programm testen und Fehler beheben. |

Inhalt

[Management Summary 3](#_Toc38624311)

[Zweck des Dokuments 3](#_Toc38624312)

[Funktionalitäten 4](#_Toc38624313)

[Mockups (Vorführmodell) 4](#_Toc38624314)

[Use Case Diagramme / Use Case Beschreibung 4](#_Toc38624315)

[Systemtests Beschreibung 5](#_Toc38624316)

[Testprotokoll 5](#_Toc38624317)

[Programmierrichtlinien 5](#_Toc38624318)

# Management Summary

Diese Projektarbeit befasst sich mit dem Erstellen einer Fahrplan Applikation. Dies geschieht als benotete Arbeit als Abschluss des ÜK (Überbetrieblicher Kurs) Moduls 318. Der Grund dieser Arbeit ist das erlernen vom Analysieren und objektbasiert programmieren mit Komponenten. Diese Arbeit ist das Praxisbeispiel zur erlernten Theorie.

Das Ziel der Applikation ist es, einen Fahrplan zu haben, der einfach zu Handhaben ist und ein Benutzerfreundliches GUI zu enthalten. Der Kunde soll einfach und schnell Verbindungen von einem Startort zu einem Zielort haben.

Das Projekt befindet sich in der Endphase und wird am Freitag 24.April abgeschlossen.

# Zweck des Dokuments

Dieses Dokument sollte eine Zusammenfassung des Moduls 318 sein. Auch sollte es bei Veränderungen oder Überarbeitungen angeschaut werden, wenn das Abgebebene Projekt da es die Programmierrichtlinien des Projekts erhält und die Use Case und Aktivitätsdiagramme. Auch beinhaltet das Dokument Testfälle (Unit Tests) die im Verlauf der Projektentwicklung durchgeführt wurden.

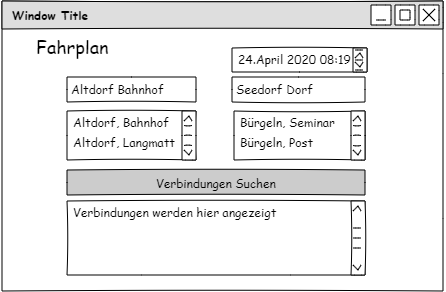
# Funktionalitäten

|  |  |
| --- | --- |
| **A-001: Der ÖV Benutzer kann via einfacher Textsuche den Start und den Zielort aussuchen** | |
| Funktionalität | 1. Bei Doppelklick auf die Textbox löst es die vorgegebene Eingabe Von das selbe tritt auch beim Doppelklick auf die Textbox Nach auf. |
| Bugs |  |
| Nicht umgesetzt | 1. Error wenn keine WLAN Verbindung vorhanden ist. |

# Mockups (Vorführmodell)

**Was ist ein GUI?**GUI heisst Grapical User Interface, dies ist die Grafische Benutzeroberfläche mit der man ein Programm bedienen kann. Beispielsweise ein Computerspiel oder ein Buchhaltungsprogramm etc.

Das GUI wurde mit Pencil erstellt. Ich habe dieses GUI gewählt da man effizient alle Verbindungen auswählen kann. Das GUI hat jedoch im Projekt eine fixe Grösse. Und beim Klick auf das X wird das beendet

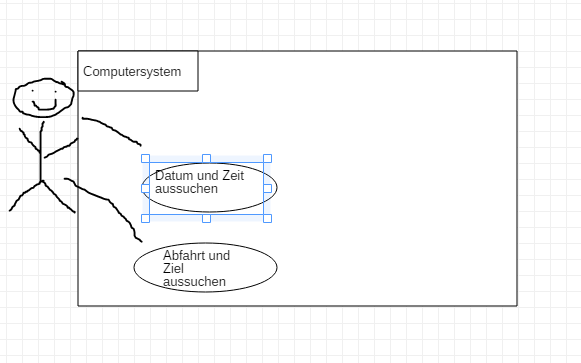


# 

# Use Case Diagramme / Use Case Beschreibung

**Was ist ein Use Case Diagramm?**Mit dem Use Case Diagramm kann man einem Kunden einfach erklären was das Programm welches der Entwickler schreibt macht und was es dafür braucht. Das Diagramm ist sehr einfach gehalten und hat einen

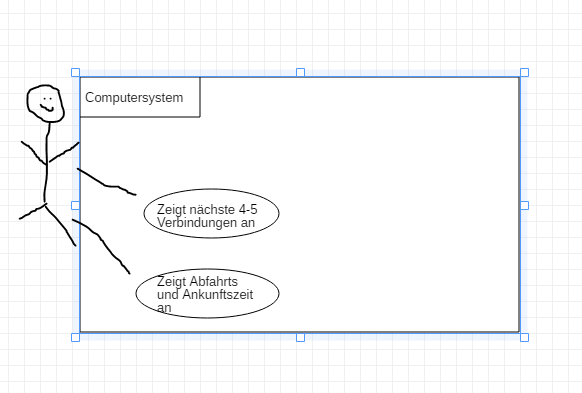
**Diagramm zu A001**Die erste Anforderung lautet, dass der Benutzer ein Abfahrts und Zielort aussuchen kann via einfache Textsuche. Und dass er Datum und Uhrzeit wählen kann.



**Diagramm Beschreibung A001**

|  |  |
| --- | --- |
| Diagramm Beschreibung A001 | |
| Beschreibung | Benutzer kann via 2 Eingabefelder eine Verbindung eines Zuges/Buses suchen |
| Akteure | Personen die eine ÖV Verbindung suchen. |
| Auslöser | Verbindung nach Zug wird gesucht |
| Vorbedingungen | Der Benutzer muss die App installiert haben |
| Ablauf | 1. Applikation öffnen 2. Von Eingeben 3. Nach eingeben |
| Alternativer Ablauf | Keiner |
| Ergebnis | Man kann den Button klicken für die Ausgabe der Zug Verbindungen |

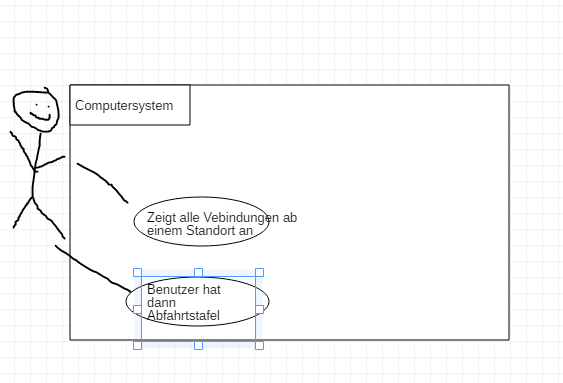
**Diagramm A002**Der Benutzer kann den Button wo Verbindungen Anzeigen steht klicken und es werden ihm die nächsten 5 Abfahrtszeiten um von A nach B zu gelangen.



**Beschreibung Diagramm A002**

|  |  |
| --- | --- |
| Diagramm Beschreibung A001 | |
| Beschreibung | Am Benutzer zeigt es durch den Button klick die nächsten 4-5 Verbindungen an |
| Akteure | Personen die eine ÖV Verbindung suchen. |
| Auslöser | Button klick auf Verbindungen anzeigen |
| Vorbedingungen | Der Benutzer Start und Zielort ausgewählt haben. |
| Ablauf | 1. Button «Verbindungen anzeigen» klicken |
| Alternativer Ablauf | Keiner |
| Ergebnis | Man sieht die nächsten Zug/Bus Verbindungen |

**Diagramm A003**Um einen eigenen Fahrplan zu haben kann man auswählen, dass man alle Vebindungen ab einer Station sehen kann.



|  |  |
| --- | --- |
| Diagramm Beschreibung A001 | |
| Beschreibung | Benutzer gibt ein Feld ein klickt Button und hat eigener Fahrplan |
| Akteure | Personen die an verschiedene Orte verreisen |
| Auslöser | Klick auf Button |
| Vorbedingungen | Benutzer muss Startort eingeben |
| Ablauf | 1. Startort eingeben 2. Button klicken |
| Alternativer Ablauf | Keiner |
| Ergebnis | Man hat einen Fahrplan |

# Aktivitätsdiagramme

**Was sind Aktivitätsdiagramme?**  
Das ist ein Flussdiagramm welches die Komplexen Abläufe der Software aufzeigt.

**Aktivitätsdiagramm A001**Auf diesem Diagramm sieht man dass man zuerst den Startort eingeben muss und danach den Zielort eingeben muss.

Start

Startort eingeben

Zielort eingeben

Ziel

**Aktivitätsdiagramm A002**Der Benutzer klickt den Button um dass es danach die nächsten 4-5 Verbindungen Anzeigt

Button klick

Ausgabe des Computers

Start

Start

Ziel

**Aktivitätsdiagramm A003**DerBenutzr klickt auf anderen Button damit es ihm den Fahrplan anzeigt

Start

Ausgabe des Computers

Button klick

Ziel

# Systemtests Beschreibung

|  |  |  |
| --- | --- | --- |
| Schritt | Aktivität | Erwartetes Resultat |
| 1 | Der User gibt Luzern Bahnhof ein | In Listbox sollte Luzern Bahnhof erscheinen |
| 2 | Bei doppelklick sollte das Resultat gelöst werden | Nichts mehr in Textbox |
| 3 | Nach sollte mit Bern ausgefüllt werden | In Listbox erscheint Bern |

# Testprotokoll

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Schritt | Aktivität | Erwartetes Resultat | Tatsächliches Resultat | Status |
| 1 | Der User gibt Luzern Bahnhof ein | Luzern Bahnhof | Luzern Bahnhof | gut |
| 2 | Bei doppelklick sollte das Resultat gelöst werden | Nichts mehr in Textbox | Nichts mehr in Textbox | gut |
| 3 | Nach sollte mit Bern ausgefüllt werden | Bern | Bern | gut |

# Programmierrichtlinien

**Was sind Programmierrichtlinien**

**Naming Conventions**Variablen werden klein geschrieben  
Elemente des GUI werden mit der Definiton zuerst geschrieben z.B. textboxeingabe

**Declaration**Lokale Variablen werden bevor sie gebraucht werden deklariert.   
Bei Variablen muss immer ein Wert zugewiesen werden.

**Comments**Kommentare werden dort gemacht wo man sie für sinnvoll hält.   
Lokale Variablen müssen auskommentiert werden.

**Statements**Die Klammer kommt immer auf eine seperate Zeile.