Álvaro Fernández

Ländliche Ostbahnen AG

Semesterarbeit TIN-16 2019

Inhaltsverzeichnis

[Ausgangslage 5](#_Toc3558432)

[Rahmenbedingungen 6](#_Toc3558433)

[Anforderungen 6](#_Toc3558434)

[Administratives 8](#_Toc3558435)

[Termine 8](#_Toc3558436)

[Projektrollen 9](#_Toc3558437)

[Lastenheft 10](#_Toc3558438)

[Abgrenzungen 11](#_Toc3558439)

[Zeitplan 12](#_Toc3558440)

[IST Zustand 14](#_Toc3558441)

[Problem 14](#_Toc3558442)

[Ziele 15](#_Toc3558443)

[Variantenentscheid 16](#_Toc3558444)

[Risikoanalyse 17](#_Toc3558445)

[Detailkonzept 18](#_Toc3558446)

[UseCases 18](#_Toc3558447)

[Sequenzdiagramme 19](#_Toc3558448)

[Klassendiagramm 20](#_Toc3558449)

[ERD 21](#_Toc3558450)

[Testkonzept 22](#_Toc3558451)

[Testfälle 23](#_Toc3558452)

[Test Ergebnisse 24](#_Toc3558453)

[Realisierung 25](#_Toc3558454)

[Linien: Datenstruktur & Verwaltung 26](#_Toc3558455)

[Berechnung: Fahrplan 27](#_Toc3558456)

[Berechnung: Verfügbares Material 28](#_Toc3558457)

[Glossar 29](#_Toc3558458)

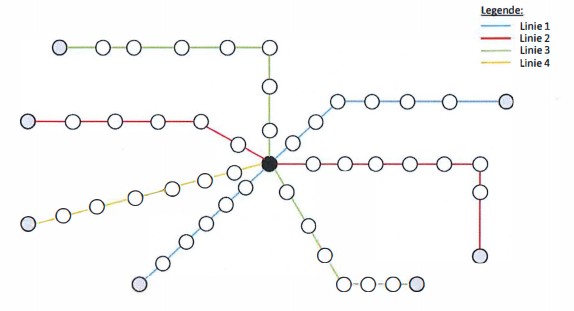
Abbildungsverzeichnis

Tabellenverzeichnis

Einleitung

# Ausgangslage

Die Ländliche Ostbahnen AG ist eine regionale Bahngesellschaft, welche von ihrem Hauptstandort mit insgesamt vier Linien die verschiedenen Ortschaften der Region verbindet (siehe Streckennetz mit den einzelnen Stationen in der nachfolgenden Abbildung). Das Streckennetz ist durchgängig zweispurig. Der Bahnbetrieb findet von 05:30 Uhr bis 24:00 Uhr statt. Die Züge verkehren auf allen Strecken generell im Halbstundentackt. An Werkstagen verkehren sie von 06:30 Uhr bis 08:30 Uhr sowie von 11 :30 Uhr bis 13:30 und von 16:00 Uhr bis 19:00 Uhr im Viertelstundentackt.



Die Linien 2 und 3 sind die meist frequentierten Verbindungen. Die Zugkompositionen, welche auf diesen beiden Linien verkehren setzen sich aus einer Lokomotive mit zwei Erstklassen- und vier Zweitklassenwagen zusammen. Bei den anderen Linien werden nur Kompositionen mit einem Erstklassenwagen und zwei Zweitklassenwagen eingesetzt. Es wird grundsätzlich den ganzen Tag mit denselben Kompositionen gefahren. Die Fahrpläne und die Einsatzplanung von Personal und Rollmaterial werden bei der Ländlichen Ostbahnen AG heute noch mit Excel geführt. Die Excel-Dateien sind kompliziert und fehleranfällig. Die auf dem Markt existierenden Software sind entweder zu umfangreich oder decken die Bedürfnisse der Ländlichen Ostbahnen AG nicht. Das Unternehmen hat sich deshalb entschieden, die Entwicklung einer auf ihre Bedürfnisse abgestimmten Applikation in Auftrag zu geben. Basierend auf den Angaben im vorliegenden Dokument muss zuerst ein Lastenheft erstellt werden. Dabei sind die vorliegenden Informationen auf ihre Vollständigkeit hin zu überprüfen und gegebenenfalls zu ergänzen. Anschliessend muss auf der Basis des erstellten Pflichtenhefts ein Prototyp der Applikation implementiert werden.

# Rahmenbedingungen

Die Semesterarbeit wird als Informatikprojekt durchgeführt. Es wird erwartet, dass die gelernten Methoden und Instrumente des Projektmanagements sowie des System Engineerings in die Praxis umgesetzt werden. Dazu gehören insbesondere (aber nicht nur) eine fundierte und detaillierte Planung der verschiedenen Aktivitäten sowie eine vollständige und nachvollziehbare Dokumentation des Projekts.

Die Wahl des Vorgehensmodells für die Durchführung des Projekts ist frei. Eine konsequente Umsetzung des gewählten Vorgehensmodells ist Pflicht. Das Projektmanagement umfasst nebst der bereits aufgeführten Planung noch ein entsprechendes Controlling, eine Risikoanalyse mit Massnahmen, ein Qualitätsmanagement sowie ein Konfigurationsmanagement in einer sinnvollen Granularität. Analyse und Design werden mittels UML vorgenommen.

Die Applikation muss in einer objektorientierten Programmiersprache implementiert werden, wobei die Wahl der konkreten Programmiersprache frei ist. Der Auftraggeber will eine echte objektorientierte Umsetzung mit Anwendung von Steuerungsklassen und sog. Fachklassen, was einen direkten Zugriff der GUI-Klassen auf die Datenbank ausschliesst. Er erwartet zudem eine mehrschichtige Architektur

# Anforderungen

Das Streckennetz der Ländlichen Ostbahnen AG muss in seiner heutigen Form (d.h. alle vier Linien mit ihren End- und Zwischenstationen) in der Applikation abgebildet werden können. Zudem müssen zukünftige Änderungen (und insbesondere eine Erweiterung des Streckennetzes) ohne Programmieraufwand möglich sein. Die Strecken müssen dynamisch definiert werden können mit Angabe der Distanz und der Fahrzeit je Teilstrecke sowie der Haltedauer in jeder Station. Letztere beträgt in der Regel zwei Minuten pro Station. Am Hauptstandort sind es fünf Minuten.

Der Rollmaterial der Ländlichen Ostbahnen AG muss in der Applikation verwaltet werden können. Der Einfachheit halber werden nur die Lokomotiven sowie die Erst- und Zweitklassenwagen berücksichtigt. Jedes Rollelement verfügt über eine eindeutige Identifikation und im Minimum über folgende Informationen: Typenkennzeichnung (Lokomotive, Wagen), Datum der Inbetriebnahme, Datum der letzten Revision, Datum der nächsten Revision, Klassenbezeichnung, Anzahl Plätze und Angabe, ob das Element verfügbar ist. Die Zugkompositionen setzen sich jeweils aus einer Lokomotive und der jeweiligen Anzahl an Erst- und Zweitklassenwagen zusammen. Es gibt keine Restriktion im Hinblick auf die Kombinierbarkeit des Rollmaterials.

Das rollende Personal der Ländlichen Ostbahnen AG (Lokomotivführer/innen und Kontrolleur/innen) muss ebenfalls in der Applikation registriert werden können. Nebst einer eindeutigen Identifikation (Personalnummer) müssen für jede Person im Minimum folgende Angaben erfasst werden können: Name, Vorname, Kategorie (Lokomotivführer/in, Kontrolleur/in), Abwesenheiten (Ferien/Krankheit). Jede Zugbesatzung besteht aus einem/einer Lokomotivführer/in und einem/einer Kontrolleur/in. Auch hier bestehen keine Restriktionen im Hinblick auf die Zusammensetzung der Zugbesatzungen.

Teil 1: Ablauf und Umfeld

# Administratives

## Termine

Die Semesterarbeit zwischen Dezember 2018 und März 2019 durchgeführt. Am 9. Januar 2019 und 13. Februar finden Reviews statt. An diesen Reviews soll ein A4 Blatt abgegeben werden, welches den aktuellen Stand des Projektes und die nächsten Schritte beschreibt. Diese sind im Anhang angefügt.

# Projektrollen

Austraggeber:  
Patrick Graber

Projektleitung:  
Alvaro Fernandez

Umsetzung:  
Alvaro Fernandez

# Lastenheft

Aus dem Situationsbeschrieb lassen sich folgende Anforderungen schliessen:

## Datenverwaltung Material

Es soll möglich sein, Rollmaterial zu erfassen, anpassen und löschen.

* Das Anpassen beschränkt sich auf die Verfügbarkeit des Materials.
* Folgende Daten sollen erfasst werden können:
  + Identifikationsnummer
  + Typenkennzeichnung (Lokomotive oder Wagen)
  + Datum der Inbetriebnahme
  + Datum der letzten Revision
  + Klassenbezeichnung, nur bei Wagen
  + Anzahl Plätze, nur bei Wagen
  + Verfügbarkeit

## Datenverwaltung Personal

Es soll möglich sein, Personal zu erfassen, anpassen und löschen.

* Das Anpassen beschränkt sich auf den Anwesenheitsstatus des Arbeiters.
* Folgende Daten sollen erfasst werden können:
  + Personalnummer
  + Name
  + Vorname
  + Kategorie (Lokomotivführer oder Kontrolleur)
  + Abwesenheit

## Datenverwaltung Station

Es soll möglich sein, Stationen zu erfassen und löschen.

* Stationen haben einen Namen und eine Haltedauer

## Datenverwaltung Linien

Es soll möglich sein, Linien zu erfassen, erweitern und löschen.

* Linien haben einen Namen und Zugkomposition (Anzahl Lokomotiven und Wagen)
* Beim Erstellen muss eine Start- und Endstation angegeben werden
* Es soll möglich sein, Stationen hinzuzufügen.
* Bei jeder Verbindung muss die Reisezeit zwischen den Stationen angegeben werden.

## Berechnung Fahrplan

Es soll ein Fahrplan generiert werden können. Die Züge fahren jeweils von beiden Endstationen los. Der Bahnbetrieb findet täglich zwischen 5:30 Uhr und 00:00 Uhr statt. Generell verkehren die Züge im Halbstundentackt. An Werktagen fahren die Züge an folgenden Zeiten im Viertelstundentackt:

* 06:30 bis 8:30
* 11:30 bis 13:30
* 16:00 bis 19:00

## Ressourcenprüfung

Es soll eine Prüfung möglich sein, zum Prüfen ob genug Material und Personal verfügbar ist, um den Fahrplan einzuhalten. Es soll eine Liste mit der Anzahl fehlenden Ressourcen erstellt werden.

Um die Prüfung zu vereinfachen wurden folgende Einschränkungen bestimmt:

* Das Material bleibt für den ganzen Tag auf der gleichen Linie
* Das Personal bleibt den ganzen Tag im gleichen Zug

# Zeitplan

Teil 2: Konzept & Realisation

# IST Zustand

Die Ländliche Ostbahnen AG ist eine regionale Bahngesellschaft, welche von ihrem Hauptstandort mit insgesamt vier Linien die verschiedenen Ortschaften der Region verbindet. Die ganze Datenverwaltung wird bis heute mit Excel gemacht.

# Problem

Die Organisation mit Excel ist sehr aufwändig, kompliziert und fehleranfällig.

# Ziele

Es soll eine Softwarelösung entwickelt werden welches die im Lastenheft definierten Anforderungen erfüllt. Die Software soll Objektorientiert umgesetzt werden mit einer mehrschichtigen Architektur.

# Variantenentscheid

Es haben sich folgende Varianten ergeben:

* Desktopanwendung mit JAVA
* Desktopanwendung mit C#
* Webanwendung mit MERN Stack
* Webanwendung mit WAMP Stack

## Variante 1: Desktopanwendung JAVA

Eine einfache Desktopanwendung mit Java.

**Vorteile**

* Läuft auf allen Plattformen
* Kann auf allen Plattformen entwickelt werden

**Nachteile**

* Sehr wenig Erfahrung

## Variante 2: Desktopanwendung C#

Eine einfache Desktopanwendung mit C#.

**Vorteile**

* Ein wenig Erfahrung (2 Semesterarbeiten in der TEKO)

**Nachteile**

* Windowsabhängig

## Variante 3: Webanwendung MERN Stack

Eine Webanwendung mit MERN Stack. Der MERN Stack besteht aus:

* **M**ongoDB als Datenbank
* **E**xpressJS als API
* **R**eactJS als Frontend Framework
* **N**odeJS als Backend Framework

Natürlich wird für den Grundaufbau und als Ergänzung HTML, CSS und JavaScript verwendet.

**Vorteile**

* Plattformunabhängig
* Erfahrung im Frontendbereich
* Einfach zu implementierten da Webbased

**Nachteile**

* Keine Erfahrung mit Backend und API
* Wenig Erfahrung mit NOSQL/MongoDB

## Variante 4: Webanwendung LAMP Stack

Eine Webanwendung mit LAMP Stack. Der LAMP Stack besteht aus:

* **L**inus als Serverbetriebssystem.
* **A**pache als http Server
* **M**ySQL/MariaDB als Datenbank
* **P**HP als Backend und teils Frontend

Natürlich wird für den Grundaufbau und als Ergänzung HTML, CSS und JavaScript verwendet.

**Vorteile**

* Plattformunabhängig
* Sehr viel Erfahrung (2 Jahre in einem Unternehmen)
* Einfach zu implementierten da Webbased

**Nachteile**

* Wenig Erfahrung mit objektorientiertem PHP

## Auswertung

Die Applikation wird nach der 4. Variante, LAMP Stack, umgesetzt. Bei allen anderen Varianten ist die fehlende Erfahrung ein zu grosses Risiko. Der Umgang mit PHP ist bekannt und obwohl ein neuer Aspekt dazu kommt (OOP), ist das kein grosses Problem, da die OOP Prinzipien gut bekannt sind.

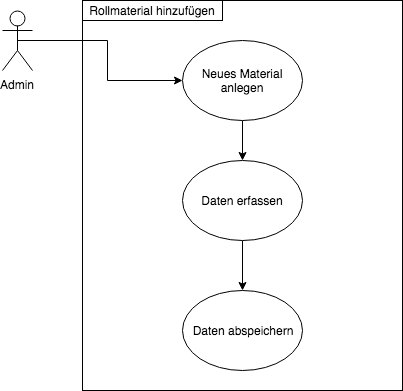
# Risikoanalyse

# Detailkonzept

## UseCases

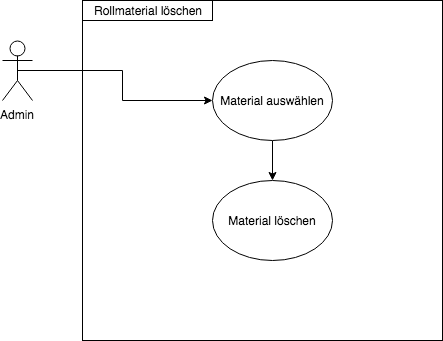
Aus den oben beschriebenen Anforderungen lassen sich folgende UseCases schliessen.

### Rollmaterial erfassen



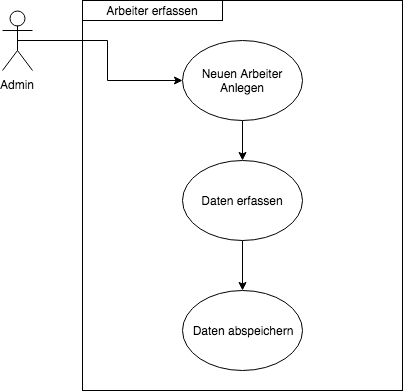
|  |  |
| --- | --- |
| Geschäftsanwendungsfall |  |
| Name | Rollmaterial erfassen |
| Kurzbeschrieb | Admin erfasst neues Rollmaterial im System |
| Akteure | Admin |
| Auslöser | Admin will neues Rollmaterial erfassen |
| Ergebnis | Rollmaterial erfasst |
| Eingehende Daten | Daten vom Rollmaterial |
| Vorbedingungen | Keine |
| Nachbedingungen | Rollmaterial existiert im System |
| Ablauf | * Admin erstellt neues Material * Admin erfasst Daten * Daten werden abgelegt |
| Priorität | Hoch |

### Rollmaterial löschen



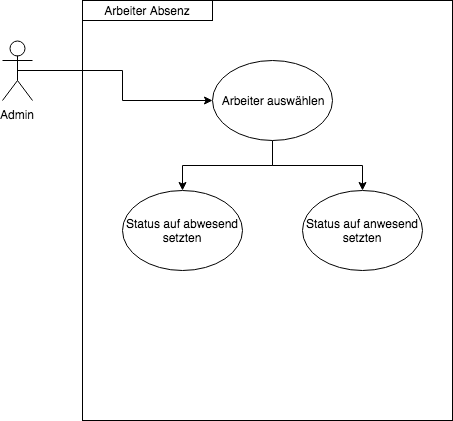
|  |  |
| --- | --- |
| Geschäftsanwendungsfall |  |
| Name | Rollmaterial löschen |
| Kurzbeschrieb | Admin löscht Rollmaterial aus dem System |
| Akteure | Admin |
| Auslöser | Admin will Rollmaterial löschen |
| Ergebnis | Rollmaterial gelöscht |
| Eingehende Daten | Keine |
| Vorbedingungen | Rollmaterial ist erfasst |
| Nachbedingungen | Rollmaterial existiert im System nicht |
| Ablauf | * Admin wählt Material aus * Admin löscht Material * Daten werden aus dem System gelöscht |
| Priorität | Tief |

### Arbeit erfassen



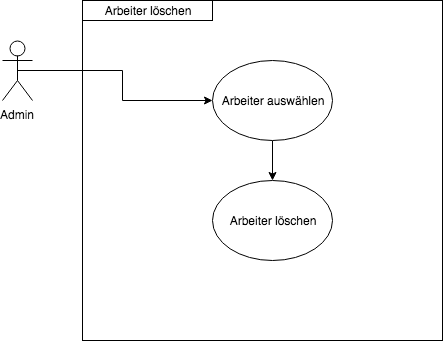
|  |  |
| --- | --- |
| Geschäftsanwendungsfall |  |
| Name | Arbeiter erfassen |
| Kurzbeschrieb | Admin erfasst einen neuen Arbeiter im System |
| Akteure | Admin |
| Auslöser | Admin will neuen Arbeiter erfassen |
| Ergebnis | Arbeiter erfasst |
| Eingehende Daten | Daten des Arbeiters |
| Vorbedingungen | Keine |
| Nachbedingungen | Arbeiter existiert im System |
| Ablauf | * Admin erstellt neuen Arbeiter * Admin erfasst Daten * Daten werden abgelegt |
| Priorität | Hoch |

### Arbeiter Anwesenheitsstatus ändern



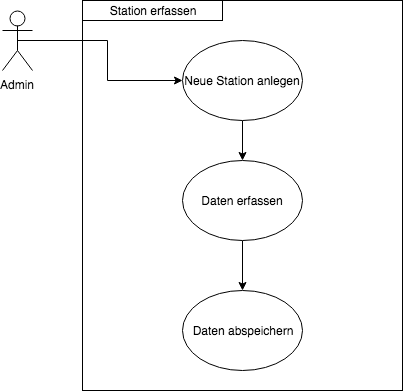
|  |  |
| --- | --- |
| Geschäftsanwendungsfall |  |
| Name | Arbeiter Anwesenheitsstatus ändern |
| Kurzbeschrieb | Admin ändert den Anwesenheitsstatus eines Arbeiters |
| Akteure | Admin |
| Auslöser | Admin will den Anwesenheitsstatus eines Arbeiters ändern |
| Ergebnis | Anwesenheitsstatus geändert |
| Eingehende Daten | Neuer Abwesenheitsstatus |
| Vorbedingungen | Arbeiter im System erfasst |
| Nachbedingungen | Abwesenheitsstatus geändert |
| Ablauf | * Admin wählt Arbeiter aus * Admin ändert den Status * Status wird im System aktualisiert |
| Priorität | Hoch |

### Arbeiter löschen



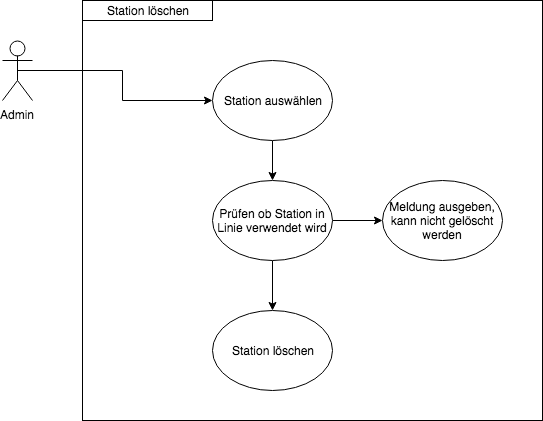
|  |  |
| --- | --- |
| Geschäftsanwendungsfall |  |
| Name | Arbeiter löschen |
| Kurzbeschrieb | Admin löscht Arbeiter aus dem System |
| Akteure | Admin |
| Auslöser | Admin will Arbeiter löschen |
| Ergebnis | Arbeiter gelöscht |
| Eingehende Daten | Keine |
| Vorbedingungen | Arbeiter ist erfasst |
| Nachbedingungen | Arbeiter existiert im System nicht |
| Ablauf | * Admin wählt Arbeiter aus * Admin löscht Arbeiter * Daten werden aus dem System gelöscht |
| Priorität | Tief |

### Station erfassen



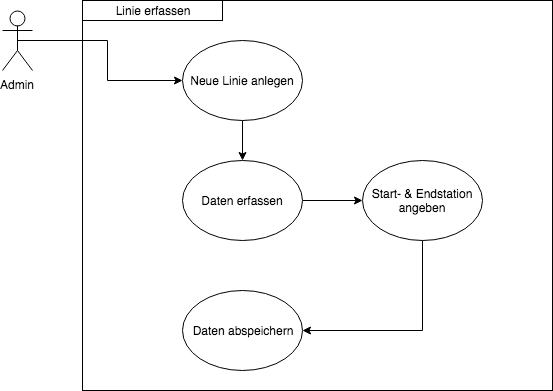
|  |  |
| --- | --- |
| Geschäftsanwendungsfall |  |
| Name | Station erfassen |
| Kurzbeschrieb | Admin erfasst eine neue Station im System |
| Akteure | Admin |
| Auslöser | Admin will neue Station erfassen |
| Ergebnis | Station erfasst |
| Eingehende Daten | Daten der Station |
| Vorbedingungen | Keine |
| Nachbedingungen | Station existiert im System |
| Ablauf | * Admin erstellt neue Station * Admin erfasst Daten * Daten werden abgelegt |
| Priorität | Hoch |

### Station löschen



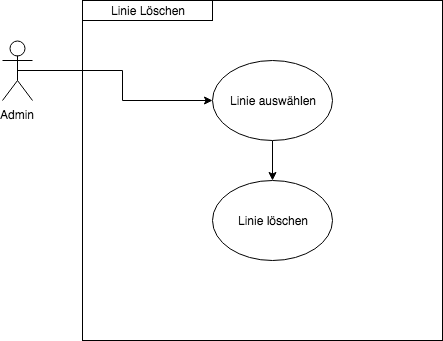
|  |  |
| --- | --- |
| Geschäftsanwendungsfall |  |
| Name | Station löschen |
| Kurzbeschrieb | Admin löscht Station aus dem System |
| Akteure | Admin |
| Auslöser | Admin will Station löschen |
| Ergebnis | Station gelöscht |
| Eingehende Daten | Keine |
| Vorbedingungen | Station ist erfasst |
| Nachbedingungen | Station existiert im System nicht |
| Ablauf | * Admin wählt Station aus * Admin löscht Station * Prüfen ob Station verwendet wird * Daten werden aus dem System gelöscht oder Meldung wird ausgegeben |
| Priorität | Tief |

### Linie hinzufügen



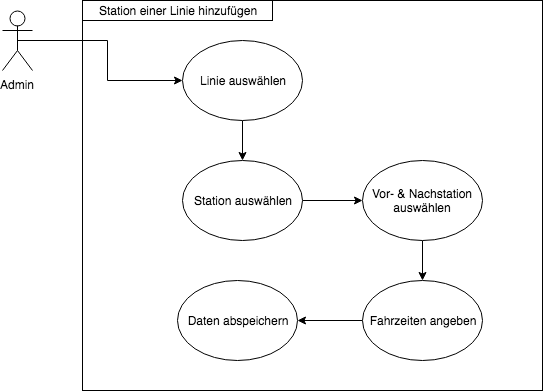
|  |  |
| --- | --- |
| Geschäftsanwendungsfall |  |
| Name | Linie erfassen |
| Kurzbeschrieb | Admin erfasst eine neue Linie im System |
| Akteure | Admin |
| Auslöser | Admin will neue Linie erfassen |
| Ergebnis | Linie erfasst |
| Eingehende Daten | Daten der Linie |
| Vorbedingungen | Keine |
| Nachbedingungen | Linie existiert im System |
| Ablauf | * Admin erstellt neue Station * Admin erfasst Daten * Admin gibt Start- & Endstation an * Daten werden abgelegt |
| Priorität | Hoch |

### Linie löschen



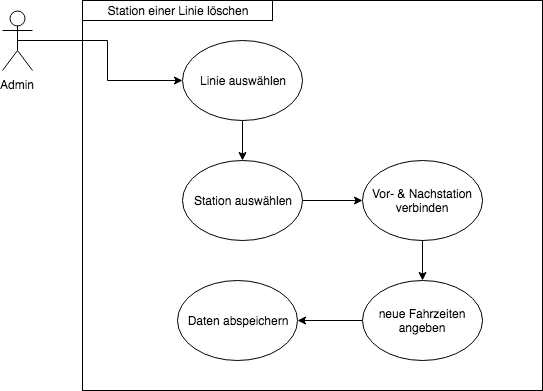
|  |  |
| --- | --- |
| Geschäftsanwendungsfall |  |
| Name | Linie löschen |
| Kurzbeschrieb | Admin löscht Linie aus dem System |
| Akteure | Admin |
| Auslöser | Admin will Linie löschen |
| Ergebnis | Linie gelöscht |
| Eingehende Daten | Keine |
| Vorbedingungen | Linie ist erfasst |
| Nachbedingungen | Linie existiert im System nicht |
| Ablauf | * Admin wählt Linie aus * Admin löscht Linie * Daten werden aus dem System gelöscht. |
| Priorität | Tief |

### Station einer Linie hinzufügen



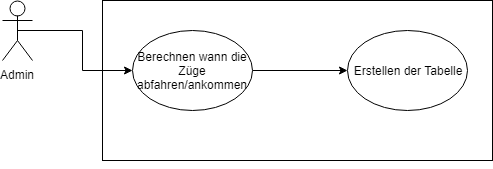
|  |  |
| --- | --- |
| Geschäftsanwendungsfall |  |
| Name | Station einer Linie hinzufügen |
| Kurzbeschrieb | Admin fügt eine Zwischenstation einer Linie hinzu |
| Akteure | Admin |
| Auslöser | Admin will eine Linie erweitern |
| Ergebnis | Linie wurde um eine Station erweitert |
| Eingehende Daten | Daten der neuen Haltestelle |
| Vorbedingungen | Linie ist erfasst  Station ist erfasst |
| Nachbedingungen | Die erweiterte Linie ist im System erfasst |
| Ablauf | * Admin wählt Linie aus * Admin wählt Position in der Linie aus * Admin gibt Fahrzeit an * Daten werden im System abgelegt |
| Priorität | Hoch |

### Station aus einer Linie löschen



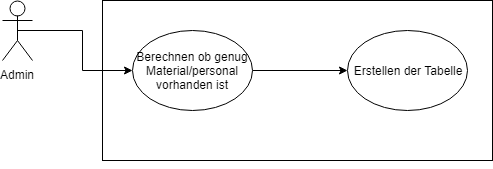
|  |  |
| --- | --- |
| Geschäftsanwendungsfall |  |
| Name | Station aus einer Linie löschen |
| Kurzbeschrieb | Admin löscht eine Zwischenstation in einer Linie |
| Akteure | Admin |
| Auslöser | Admin will eine Station in einer Linie löschen |
| Ergebnis | Linie wurde um eine Station gekürzt |
| Eingehende Daten | Daten der zu löschenden Haltestelle |
| Vorbedingungen | Linie ist erfasst |
| Nachbedingungen | Die verkürze Linie ist im System erfasst |
| Ablauf | * Admin wählt Linie aus * Admin wählt Station aus, die gelöscht werden soll * Admin gibt Fahrzeit an der neuen Verbindung an * Daten werden im System abgelegt |
| Priorität | Hoch |

### Fahrplan berechnen



|  |  |
| --- | --- |
| Geschäftsanwendungsfall |  |
| Name | Fahrplan berechnen |
| Kurzbeschrieb | Admin generiert den Fahrplan |
| Akteure | Admin |
| Auslöser | Admin will einen Fahrplan generieren |
| Ergebnis | Fahrplan wurde generiert |
| Eingehende Daten | keine |
| Vorbedingungen | Linien sind erfasst |
| Nachbedingungen | Der Fahrplan wird angezeigt |
| Ablauf | * Admin generiert Fahrplan * Fahrplan wird angezeigt |
| Priorität | Hoch |

### Material/Personal berechnen

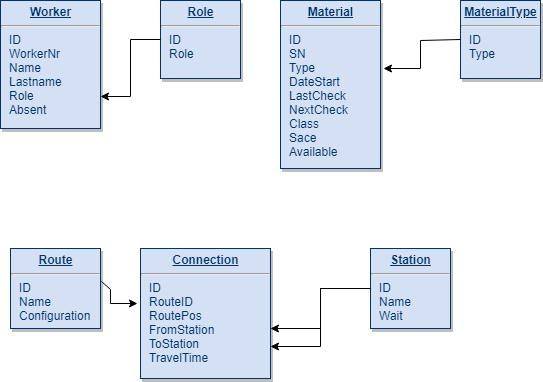


|  |  |
| --- | --- |
| Geschäftsanwendungsfall |  |
| Name | Material/Personal berechnen |
| Kurzbeschrieb | Admin generiert den Fahrplan |
| Akteure | Admin |
| Auslöser | Admin will einen Fahrplan generieren |
| Ergebnis | Fahrplan wurde generiert |
| Eingehende Daten | keine |
| Vorbedingungen | Linien sind erfasst |
| Nachbedingungen | Der Fahrplan wird angezeigt |
| Ablauf | * Admin generiert Fahrplan * Fahrplan wird angezeigt |
| Priorität | Hoch |

## Sequenzdiagramme

## Klassendiagramm

## ERD



# Testkonzept

Als Tests werden Manuelle Tests verwendet. Für jedes UseCase werden Testfälle geschrieben

## Testfälle

# Test Ergebnisse

# Realisierung

In diesem Abschnitt wird beschrieben wie 3 Anforderungen gelöst worden sind, die etwas anspruchsvoller sind. Auf die Erklärung wie die einfache Datenverwaltung aufgebaut ist, wird verzichtet.

## Linien: Datenstruktur & Verwaltung

Linien beschr.

## Berechnung: Fahrplan

## Berechnung: Verfügbares Material