|  |
| --- |
|  |
| Anforderungsspezifikation BLS Sitzplatzreservation  **Projektteam**  **Yann Roth**  **Aleksandar Andrejic**  **Version 1.4. 14.10.2019** |
| **Berner Fachhochschule**  Departement für Technik und Informatik |

Inhaltsverzeichnis

[1 Zweck des Dokuments 3](#_Toc22159647)

[2 Vision 3](#_Toc22159648)

[3 Projektzielsetzung 3](#_Toc22159649)

[3.1 Ausgangslage 3](#_Toc22159650)

[3.2 Stakeholder 3](#_Toc22159651)

[3.3 Projektziele 3](#_Toc22159652)

[3.4 Weiterführende Gedanken 4](#_Toc22159653)

[3.4.1 Offene Fragen 4](#_Toc22159654)

[3.4.2 Beantwortete Fragen 4](#_Toc22159655)

[4 Systemabgrenzung 4](#_Toc22159656)

[4.1 Prozessumfeld 4](#_Toc22159657)

[4.2 Systemumfeld 4](#_Toc22159658)

[4.3 Nicht unterstützte Projektziele 5](#_Toc22159659)

[5 Anforderungen 5](#_Toc22159660)

[5.1 Quellen und Vorgehen 5](#_Toc22159661)

[5.1.1 Quellen 5](#_Toc22159662)

[5.1.2 Vorgehen 5](#_Toc22159663)

[5.2 Funktionale Anforderungen 6](#_Toc22159664)

[5.2.1 Use Cases 6](#_Toc22159665)

[Admin Use Cases 9](#_Toc22159666)

[5.2.2 Detaillierte Anforderungen 12](#_Toc22159667)

[5.3 Qualitätsanforderungen 12](#_Toc22159668)

[5.4 Randbedingungen 12](#_Toc22159669)

[5.5 Datenmodell 13](#_Toc22159670)

[5.6 Business Objekte 13](#_Toc22159671)

[6 Glossar 14](#_Toc22159672)

[7 Literaturverzeichnis 14](#_Toc22159673)

[8 Anhang 14](#_Toc22159674)

[8.1 Abstimmung der Anforderungen 14](#_Toc22159675)

[8.2 Definition of Ready – Checklist 14](#_Toc22159676)

[9 Versionskontrolle 14](#_Toc22159677)

# Zweck des Dokuments

Dieses Dokument beschreibt die Ziele und Anforderungen für das Projekt «BLS Sitzplatzreservation».

Auftraggeber des Projekts ist die BLS; Ansprechperson und fachlicher Begleiter ist Peter Lange.

Auftragnehmer sind Aleksandar Andrejic und Yann Roth.

Das Projekt wird im Rahmen des Moduls «BTI7301 Projekt 1» im Herbstsemester 2019/2020 an der Berner Fachhochschule durchgeführt und dabei methodisch von Gerhard Schwab begleitet.

# Vision

Die BLS will mit einer Sitzplatzreservation ÖV-Reisenden ermöglichen, einfach einen Sitzplatz zu reservieren. Personen, die gerne sitzen, sollen dies im Voraus oder direkt im Zug tätigen können. Heutzutage bildet eine Sitzplatzreservation eher die Ausnahme. Evtl. würden mehr Personen die ÖV nehmen, wenn sie sicherstellen könnten, dass sie einen Sitzplatz haben: Schwangere, Familien mit Kindern, Personen mit Behinderungen, ältere Jahrgänge, ...

Es wird aller Wahrscheinlichkeit nach kein neues Geschäftsmodell geschaffen. Jedoch kann davon ausgegangen werden, dass der Komfort steigt und dies mehr Kunden anziehen könnte. Es besteht aber auch die Gefahr einer wer-zahlt-kann-sitzen-Mentalität, die das meist angenehme miteinander Reisen "vergiften" könnte.

# Projektzielsetzung

## Ausgangslage

Die BLS AG als Bahngesellschaft befördert jährlich eine grosse Anzahl von Passagieren. Viele Fahrgäste haben auch das Bedürfnis, einen Sitzplatz für Ihre Fahrt zu reservieren. Dies ist zurzeit nur am Schalter oder via SBB möglich. (stimmt das?) Die BLS will deshalb ihr Angebot erweitern, indem sie ihren Passagieren eine Möglichkeit anbietet, zusätzlich zur Fahrt einen Sitzplatz zu einem fixen Preis zu reservieren. Die Reservation soll einfach und schnell via Computer (Desktop oder Mobile) durchgeführt werden können. Eine Reservationsbestätigung soll nach erfolgter Bezahlung dem Kunden ausgehändigt werden.

## Stakeholder

* Entwickler: Yann Roth, Aleksander Andrejic
* öV Reisender / User: kann einfach eine Sitzplatzreservation vornehmen.
* Auftraggeber / Product Owner (BLS): repräsentiert durch P. Lange
  + weiss welche Art von Reisende an einer Sitzplatzreservation interessiert ist.
  + weiss wann für welche Reisen Reservationen getätigt werden.
  + kann bei Abo-Besitzer herausfinden, wer wohin fährt. -> User-Verhalten
  + bietet öV-Reisenden einen Mehrwert an.

## Projektziele

Ziel ist es, eine Applikation zu erstellen, die einfach, schnell, zuverlässig und benutzerfreundlich einem ÖV-Reisenden ermöglicht, eine Sitzplatzreservation vorzunehmen.

Die **öV-Reisende** soll in verschiedenen Situationen möglichst einfach zu einer Sitzplatzreservation gelangen.

Eine Reservation kann

* während dem Ticketkauf als Option gewählt werden.
* aufgrund von einer Position durch die Applikation vorgeschlagen werden.
* während der Fahrt manuell durchgeführt werden.

Siehe Use Cases für weitere Details.

Die **BLS** soll mit Hilfe der getätigten Reservationen herausfinden können:

* Wie Abobesitzer reisen (von ticketlösenden Reisenden weiss sie es schon)
* Wer zusätzlich zum Ticket noch eine Reservation benötigt.
* Welche Arten von Reservationen beliebt sind (z.B. Fensterplatz in Fahrtrichtung mit 220V Anschluss im oberen Stockwerk, …).
* Zu welchem Zeitpunkt eine Reservation gemacht wird (z.B. auf dem Perron, wenn viele Leute anstehen und der Zug erwartungsgemäss voll sein wird).

Die **Applikation** soll sicher, schnell, einfach zu bedienen, erweiterbar und ansprechend sein.

## Weiterführende Gedanken

Heute besteht meistens ein friedliches Miteinander beim Reisen in den Zügen. Man steht für Gebrechliche, Alte, Schwangere, Eltern mit Kleinkindern, ... auf und überlässt seinen Sitzplatz. Auch wenn dies nicht immer und überall problemlos passiert, so kann man doch sagen, dass das Zusammenreisen eher friedlich verläuft.

Mit der flächendeckenden Einführung eines Sitzplatzreservationssystem besteht die Gefahr, dass man vehementer auf seinem Recht beharrt, einen Sitzplatz "gekauft" zu haben.

Auch werden - je nach gewählter Applikationstechnologie - Leute diskriminiert, die keinen Zugang zum Reservationssystem haben oder dieses nicht zu benutzen wissen.

Es könnte aber auch sein, dass eine Sitzplatzreservation Konflikte vermeidet, da klar ist, wer den Sitzplatz reserviert hat.

(Siehe auch Punkt "offene Fragen" weiter unten...)

### Offene Fragen

### Beantwortete Fragen

Beantwortete Fragen sind ins File «Fragen-Antworten-Katalog» gezügelt worden. Siehe GIT: https://github.com/yaroth/bls/blob/master/Projekt1/Fragen-Antworten-Katalog.xlsx

# Systemabgrenzung

## Prozessumfeld

Die Sitzplatzreservation könnte in das System «Ticketkauf» eingebunden werden. Für das Projekt wird sie aber als standalone Applikation erarbeitet. Siehe auch Punkt 4.2 Systemumfeld.

Fazit: durch die Applikation wird der Geschäftsprozess «Sitzplatzreservation» verbessert und vereinfacht.

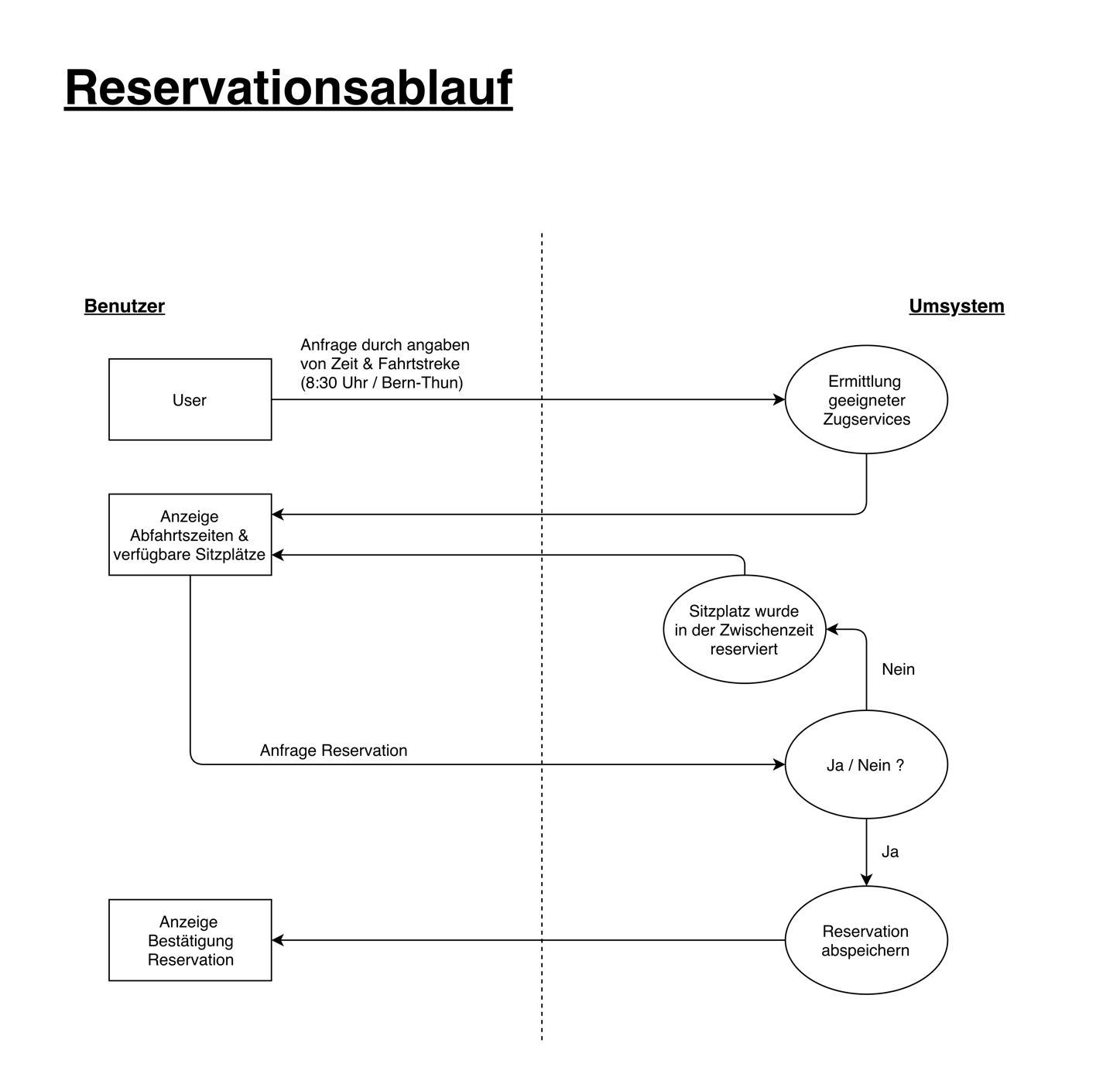
## Systemumfeld

Idealerweise würde die Applikation «Sitzplatzreservation» Anfragen zu einem Zugservice an eine zentrale Stelle (SBB? BLS?) schicken. Diese zentrale Stelle einzubinden würde aber den Rahmen dieses Projektes gemäss Aussagen von P. Lange sprengen. Darum ist entschieden worden, das Systemumfeld künstlich zu simulieren. Als Folge davon sind neue «Admin» Use Cases zu erarbeiten, die verschiedene Situationen der Zugservice-Anfragen simulieren können.

Das Umsystem «Zugservices»

* gibt auf Reiseanfragen die Zugservices heraus. Bsp.: auf die 8h30 Anfrage einer Verbindung von Bern nach Thun erfolgt die Antwort:
  + 8h35 Bern ab – 9h05 Thun an
  + 8h55 Bern ab – 8h25 Thun an
  + 9h20 Bern ab – 9h50 Thun an.
* Gibt auf Reservationsanfragen entweder eine Reservationsbestätigung oder eine «schon reserviert» Nachricht heraus.

Siehe auch folgende Grafik. Diese visualisiert auch grob den Reservationsablauf.



## Nicht unterstützte Projektziele

Nicht unterstützte Projektziele:

Die geobasierte Sitzplatzreservation wird nicht unterstützt. Siehe Use Cases 4 & 5. Aufgrund von Zeit und Geolokalisierung herauszufinden, in welchem Zug bzw. auf welchem Sitzplatz eine Person sich befindet, ist zurzeit technisch eine zu grosse Herausforderung. Diese Use Cases müssen adaptiert werden.

# Anforderungen

## Quellen und Vorgehen

### Quellen

Folgende Quellen können zwecks Ermittlung der Anforderungen konsultiert werden.

Interviews:

* Bahnbenutzer (Enduser)
* Auftraggeber (P. Lange): wöchentliche Meetings

Internet-Recherche:

* Wie funktionieren andere Zugreservationssysteme? SBB, öBB, DB, SNCR, TrenItalia,…

### Vorgehen

Folgende Fragen gilt es bei den Interviews und der Internet-Recherche zu beantworten.

* Interviews:
  + Was wünschen Enduser?
  + Welche Use Cases können neu entdeckt werden?
  + Was wünscht der Auftraggeber?
  + Was hat sich bewährt?
  + Was ist ein No-Go? Warum?
* Internet-Recherche:
  + wie haben andere ähnliche Gesellschaften das Problem der Sitzplatzreservation gelöst?
  + Durch welche Etappen wird die Benutzerin während einer Reservation geführt?

Mit dem Prototyping wird ein Vorgehen während der Entwicklung festgelegt.

* Prototyping: mit Hilfe von Use Cases & Erstellung erster Prototypen werden die Anforderungen mit Hr. P. Lange visualisiert, diskutiert und erweitert. Dies ermöglicht ein frühzeitiges Erkennen von Missverständnissen und eine Klärung der Anforderungen sowohl auf Seite des Auftraggebers wie auch auf Seite der Entwicklung.

## Funktionale Anforderungen

Als Vorabinformation: ein detaillierter Reservationsablauf sieht folgendermassen aus:



Es geht hierbei nur darum, herauszufiltern, für welche Fälle wie viele und wie komplexe Schritte zu einer Sitzplatzreservation führen. Die Anzahl Schritte sind vom Billettyp abhängig: ein Papierticket benötigt mehr Reservationsschritte als ein digitales Sparticket, das an einen Zugservice gebunden ist.

Am einfachsten ist der Use Case 5: die Reisende sitzt auf einem Platz und will diesen für die gelöste Strecke (Bsp: Bern - Thun) reservieren. 1 Klick in der mobilen App würde dies ermöglichen.

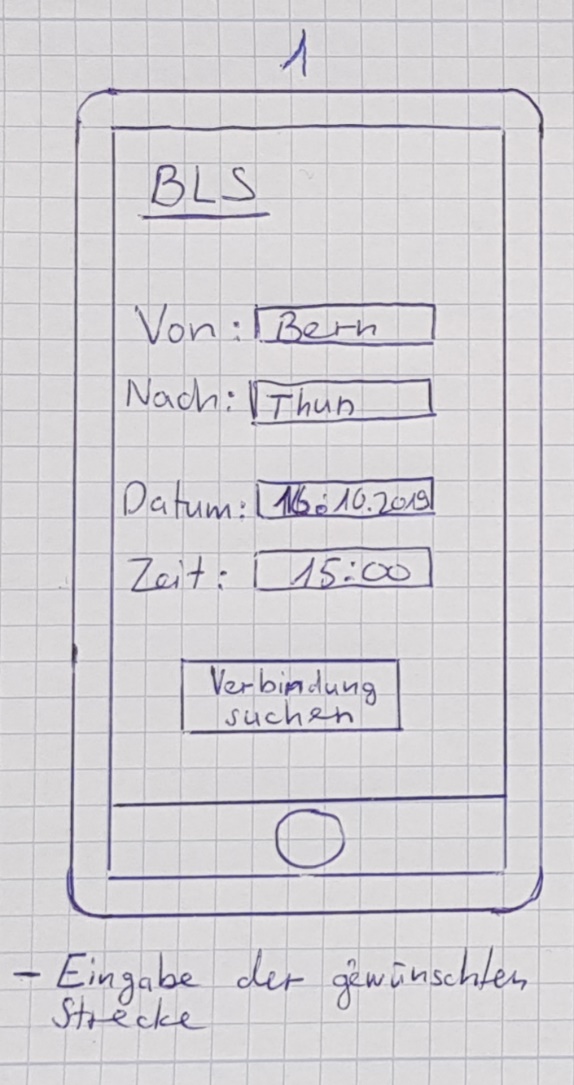
Der komplizierteste Ablauf ist derjenige mit einem Papierticket. Hier muss die Reisende zuerst den gewünschten Zugservice mit Hilfe von Abfahrts-, Ankunftsort und -zeiten ermitteln, dann einen Sitzplatz wählen und diesen reservieren. Es sind mindestens 6 Eingaben notwendig!

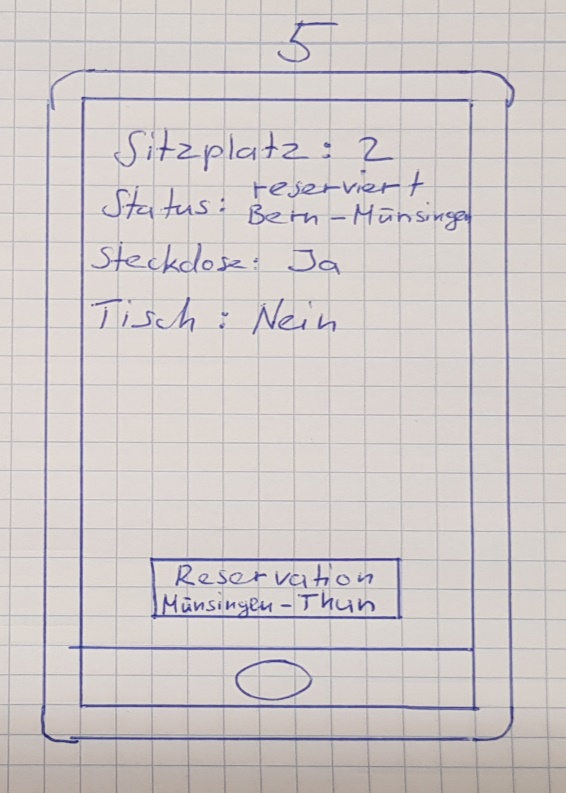
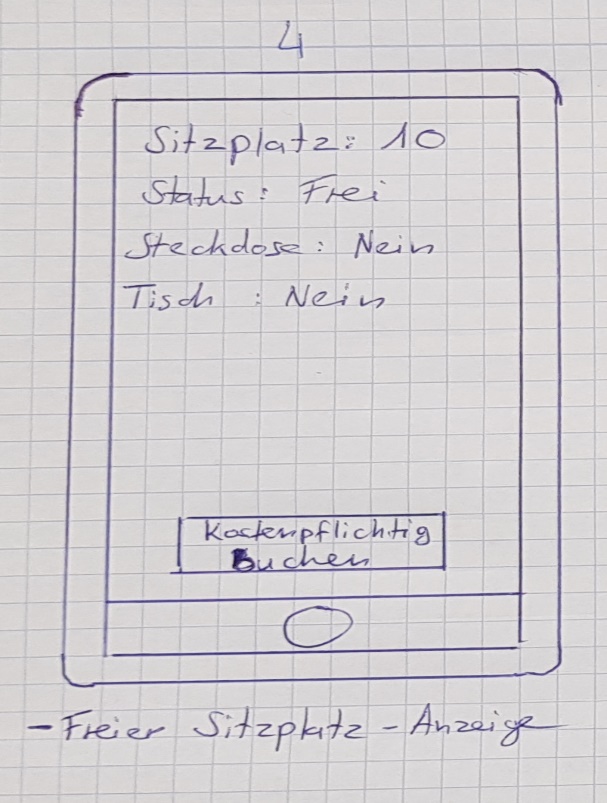
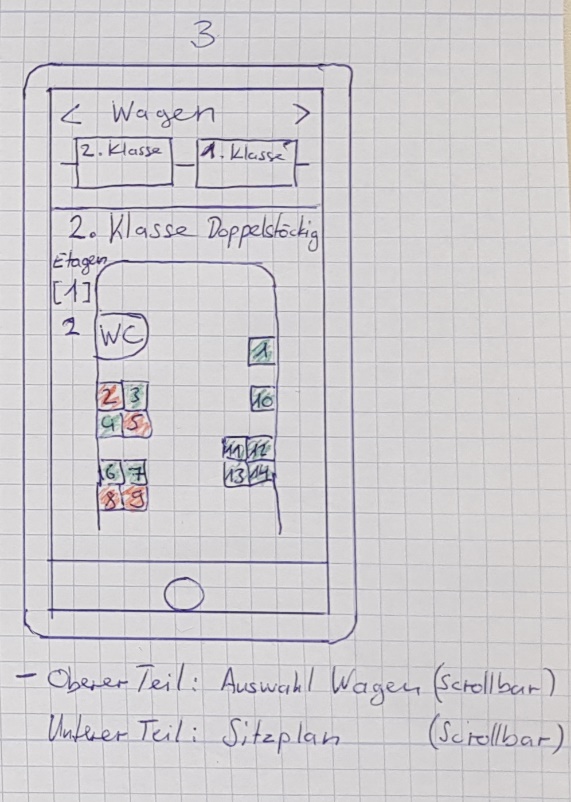
### Use Cases

Use Case 1: Ticket lange vorhanden - zusätzlich Sitzplatzreservation

*Ausgangslage*: Der Fahrgast hat im Voraus ein Fahrticket für eine längere Fahrt gekauft.

*Handlung*: Später entscheidet er sich, einen Sitzplatz zu reservieren. Er verwendet die Applikation, gibt seine Fahrstrecke und den Zeitpunkt an und wählt den Sitzplatz anhand der ihm angezeigten Möglichkeiten in den aktuellen Wagons. Er bezahlt für die Reservation und erhält eine Bestätigung.

Folgende Darstellungen zeigen den Ablauf einer möglichen Bestellung. Darstellung 4 & 5 gilt als gleiche Anzeige mit dem Unterschied, dass bei 5 ein reservierter Platz gewählt wurde.



Use Case 2: Ticket eben gekauft - viele Leute - Sitzplatz reservieren

*Ausgangslage*: Der Fahrgast hat vor einigen Minuten sein Fahrticket für die Strecke Bern-Interlaken gekauft.

*Handlung*: Später am Perron fällt ihm ein, dass der Zug für diese Strecke sehr voll sein wird. Er entscheidet sich kurzer Hand einen Sitzplatz zu reservieren. Mit der Applikation im Smartphone gibt er seine Fahrt an und sieht welche Plätze noch frei sind, reserviert einen Sitzplatz und bezahlt. Er erhält eine Bestätigung der Reservation.

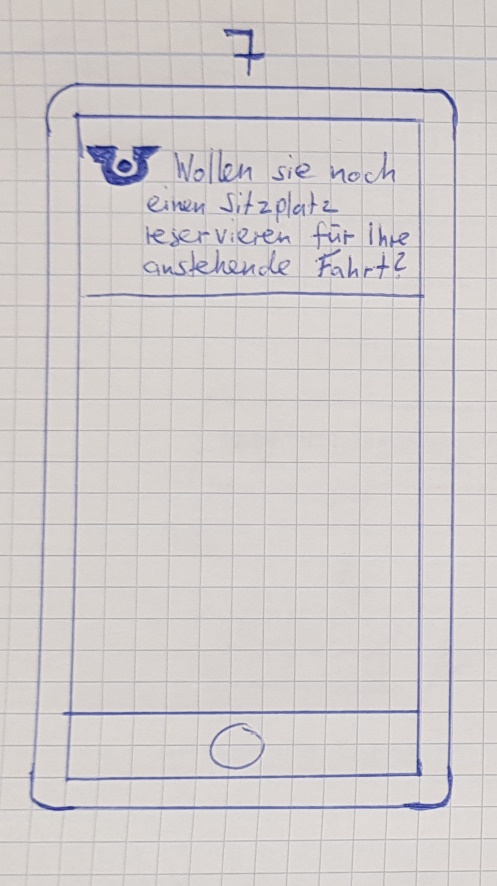
Gleicher Bestellablauf wie in Use Case 1. (Darstellungen 1-6)

Use Case 3: Abo - geolocation-basierter Reservation-Vorschlag durch Applikation

*Ausgangslage*: Der Fahrgast besitzt ein Strecken-Abo für seinen Arbeitsweg.

*Handlung*: Er möchte im Zug auf seinem Laptop noch arbeiten, weiss aber nicht, ob es noch freie (Arbeits-)Sitzplätze gibt. Auf dem Perron fragt die Applikation, ob er eine Sitzplatzreservation vornehmen will. Er öffnet die Applikation und reserviert einen Sitzplatz, bezahlt und erhält eine Bestätigung.

Anhand einer Notification erhält der User eine Benachrichtigung. Beim darauf Tippen wird der User zur Sitzplanansicht weitergeleitet (Siehe Use Case 1, Darstellung 3). Von hier an wird der übliche Bestellvorgang getätigt. (Use Case 1, Darstellungen 3-6)



Use Case 4: mit Ticket im Zug - nachträgliche Reservation

*Ausgangslage*: Reisende sitzt im Zug und sieht, dass keine Sitzplätze mehr frei sind.

*Handlung*: sie sucht mit der Applikation einen nicht reservierten Sitzplatz (jetzt oder ab einer späteren Haltestelle) und bucht diesen.

Gleicher Bestellablauf wie in Use Case 1. (Darstellungen 1-6)

Use Case 5: mit Ticket im Zug - Reservation des aktuellen Sitzplatzes

*Ausgangslage*: Reisende sitzt im Zug und möchte den Sitzplatz, auf dem sie sitzt, reservieren.

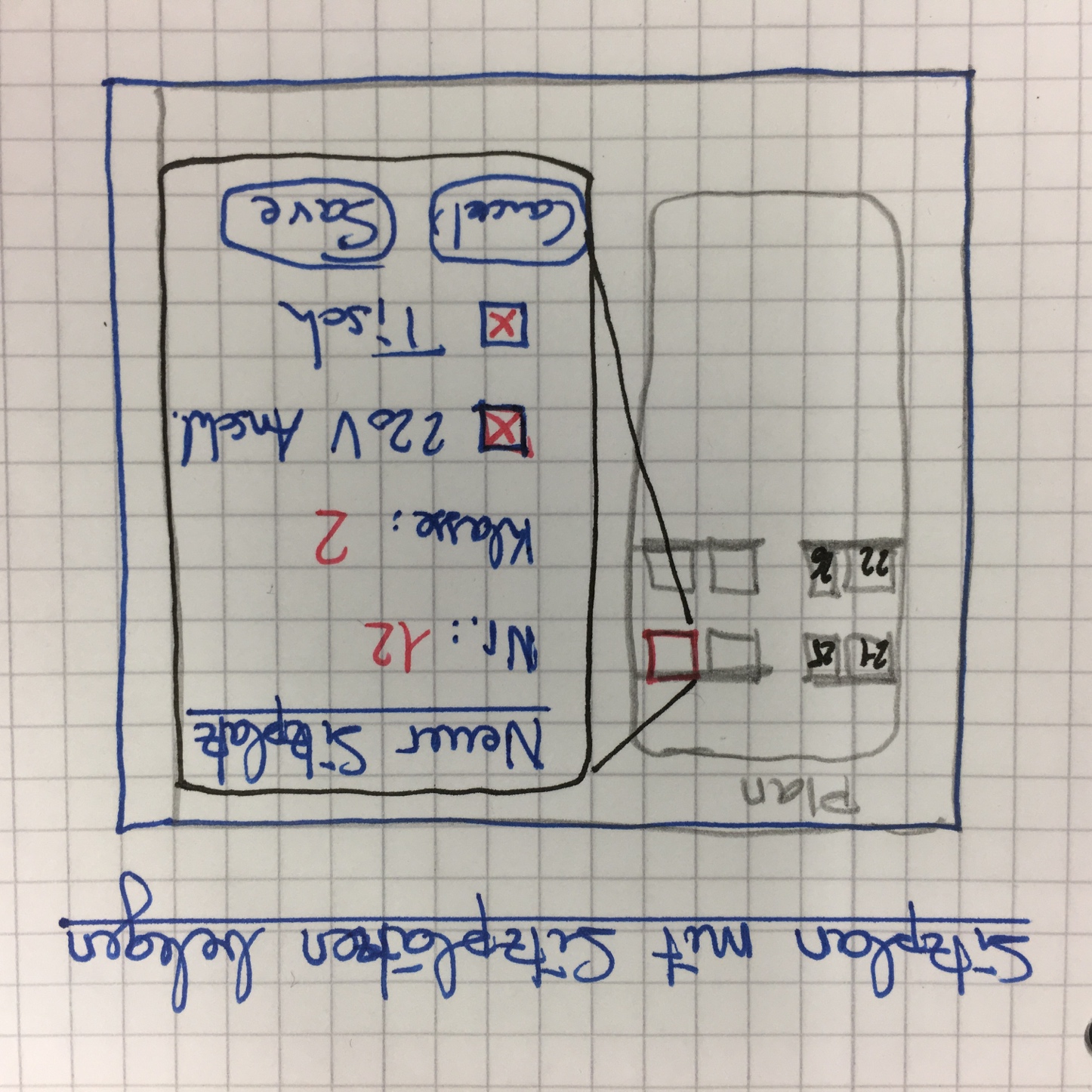
*Handlung*: in der Applikation klickt sie auf "aktuellen Sitzplatz" reservieren, bestätigt den Vorschlag, zahlt & erhält eine Bestätigung.

Gleicher Bestellablauf wie in Use Case 1. (Darstellungen 1-6)

### Admin Use Cases

Um das Systemumfeld simulieren zu können, sind folgende *Admin Use Cases* umzusetzen.

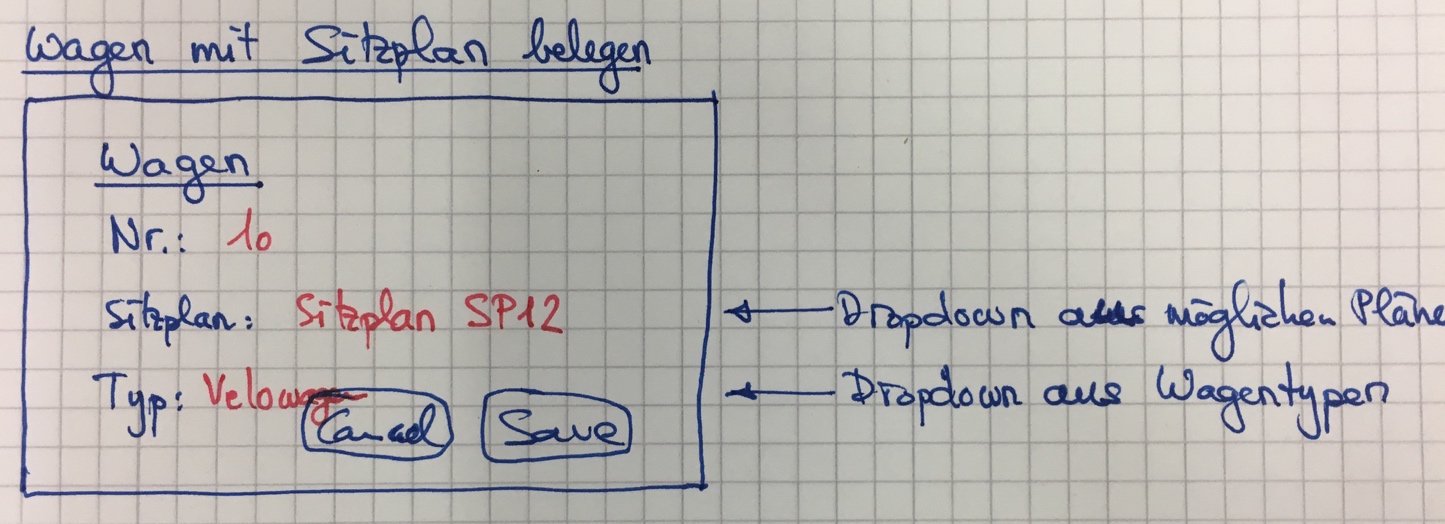
Admin Use Case 1: Sitzplatz erstellen (Nummer, 1./2. Klasse, 220V Anschluss, Tisch, …)



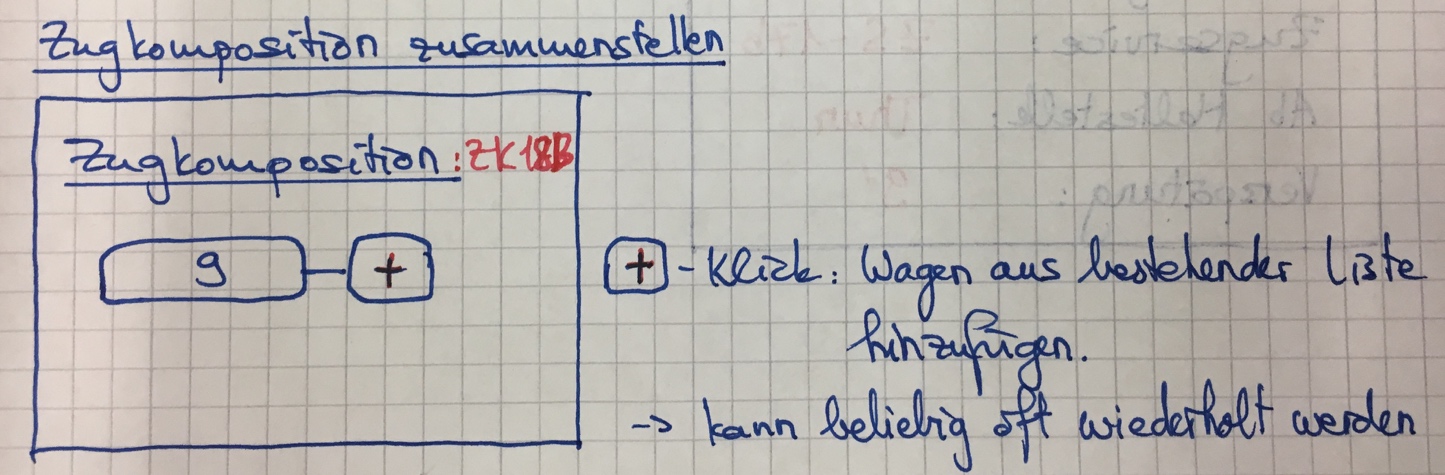
Admin Use Case 2: Sitzplan erstellen (einen kompletten Sitzplan erstellen, der einem Wagen zugeordnet werden kann)

GUI: Sitzpläne sind hardcoded.

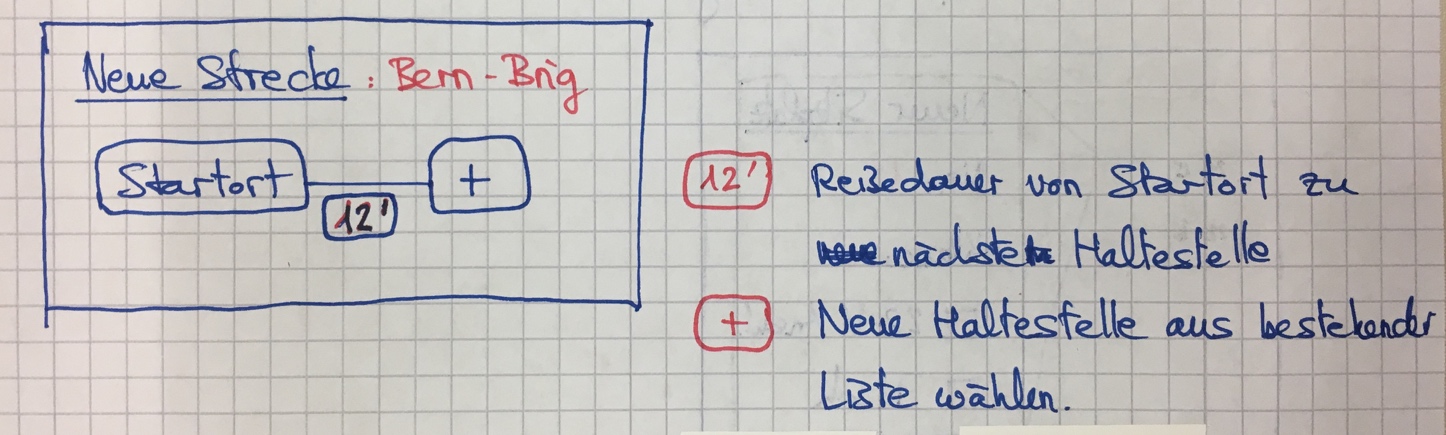
Admin Use Case 3: Sitzplan einem Wagen zuordnen



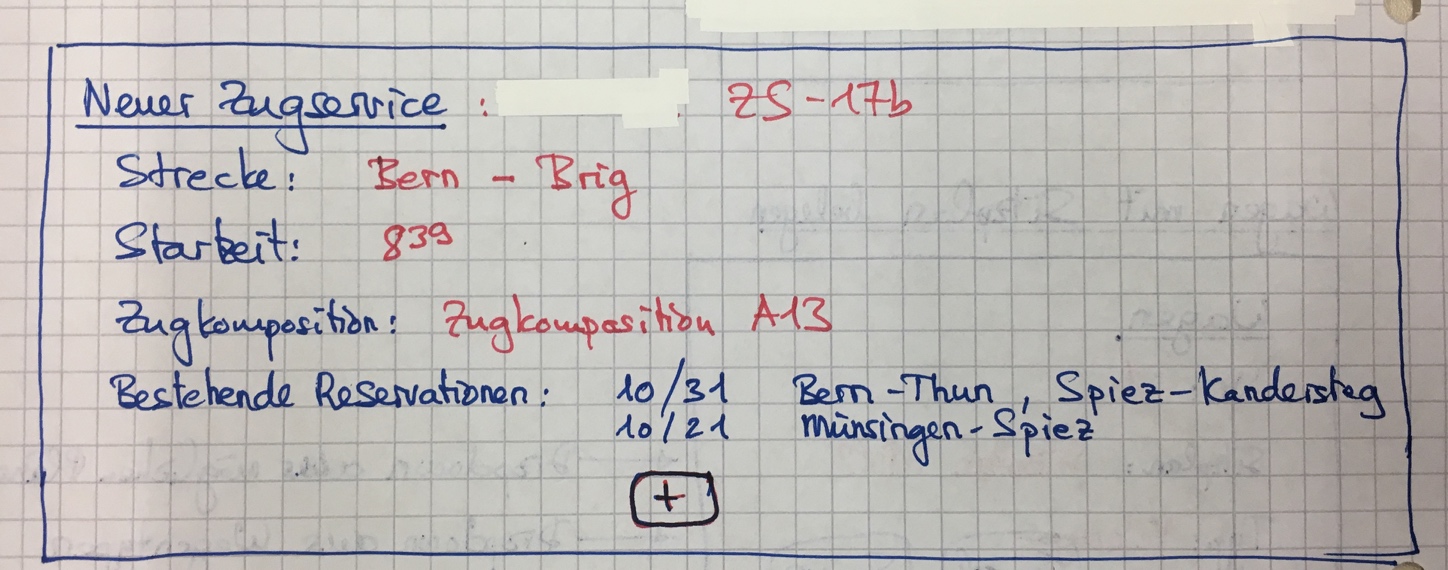
Admin Use Case 4: Zugkomposition zusammenstellen (verschiedene Wägen in eine Reihenfolge bringen, so dass eine Zugkomposition entsteht): Name/Code muss hinzugefügt werden.



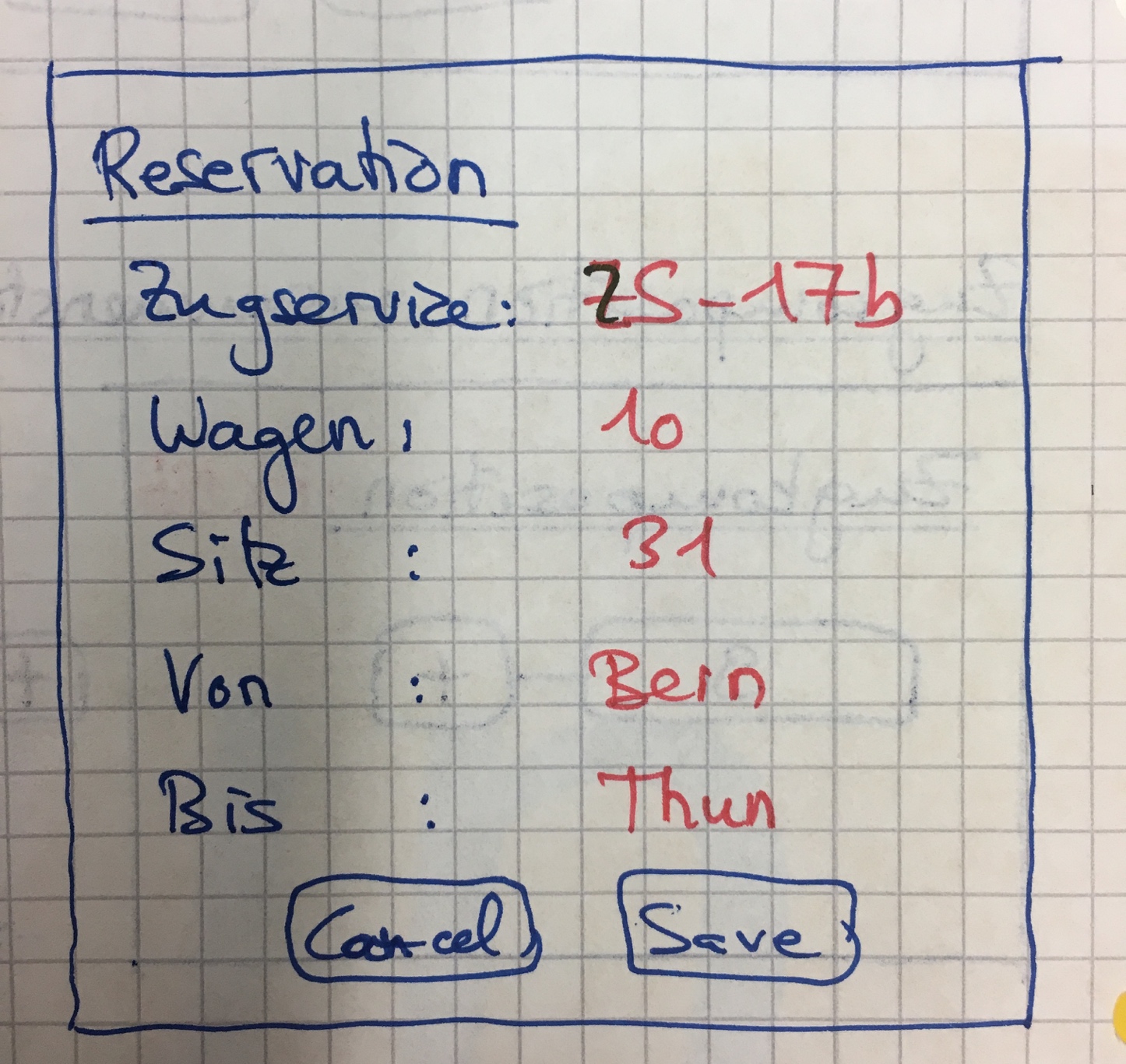
Admin Use Case 5: Strecke erstellen / ändern (Start – Zwischenhalte – Ende, Startzeit, Fahrtzeiten, Bahnhofsstehzeiten): Name/Code muss hinzugefügt werden.



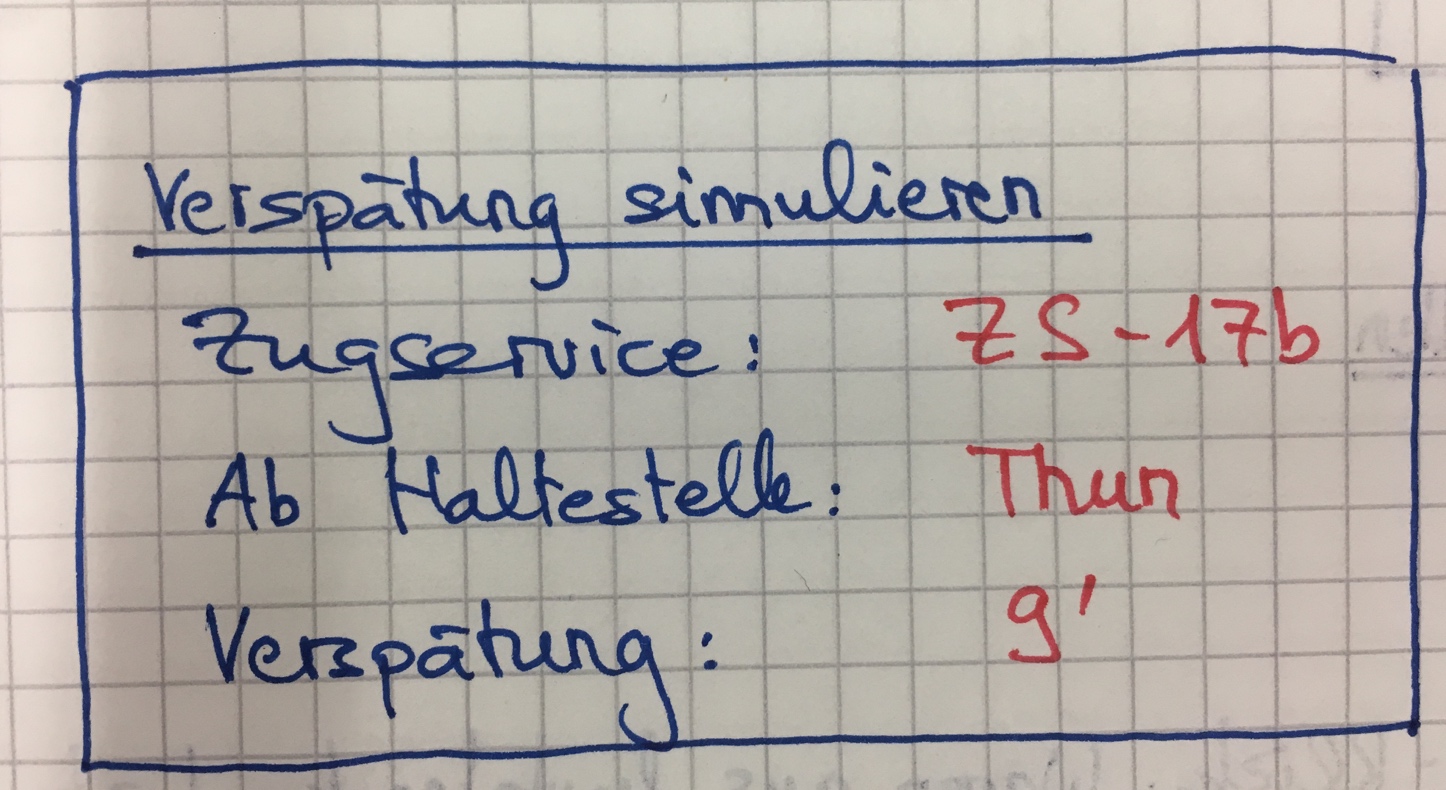
Admin Use Case 6: Zugservice erstellen /ändern (besteht aus Rollmaterial, Startzeit, Strecke)



Admin Use Case 7: in bestehendem Zugservice schon reservierte Sitzplätze simulieren.



Admin Use Case 8: Verspätung bei einem Zugservice simulieren.



## Qualitätsanforderungen

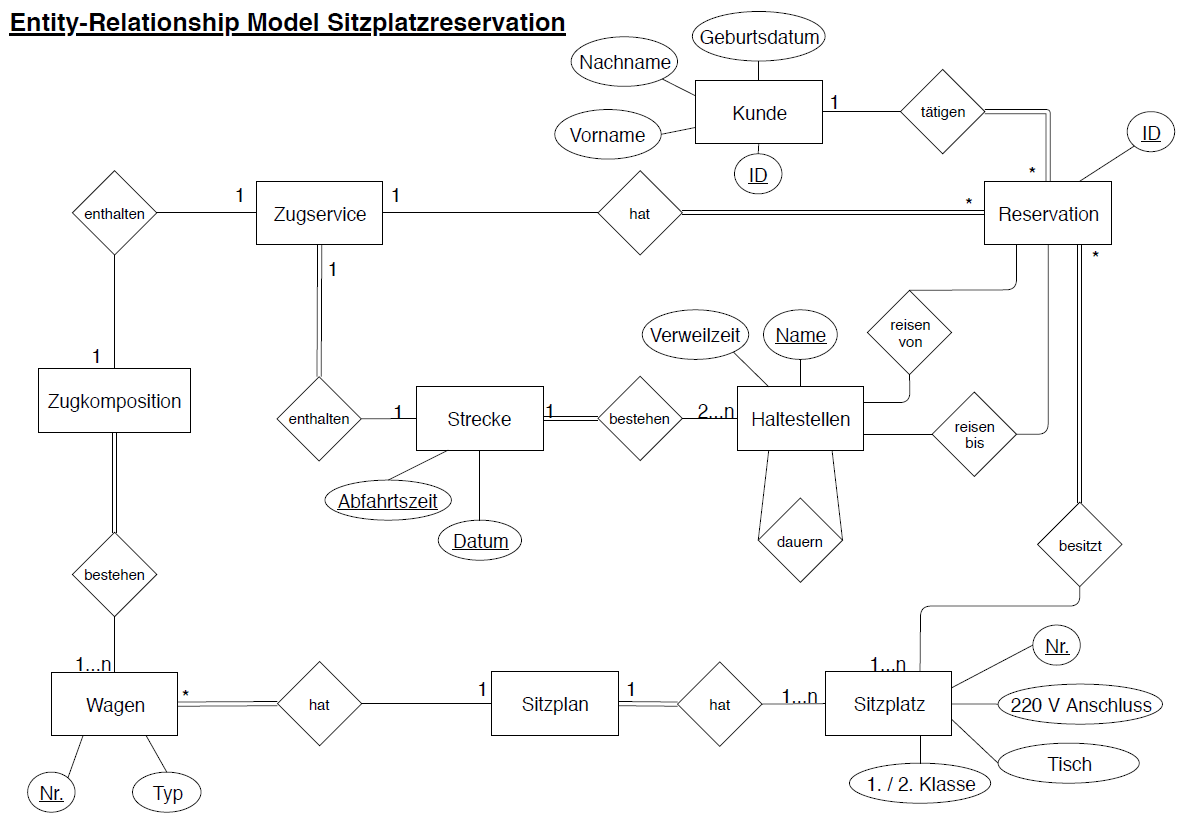
|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Status** | **Prio** | **Beschreibung** |
| **P.1.0** |  | M | **Performance: Antwortzeit**  Die Applikation soll innerhalb von 3 Sekunden starten. Ist die Applikation offen, folgt die Anzeige eines neuen Bildschirms auch innerhalb von 2 Sekunden.  Ausnahme ist der Zahlungsverkehr da dieser zwecks Sicherheitsüberprüfungen länger dauern kann. |
| **P.1.1** |  | H | **Performance: Parallelität**  Alle Benutzer können die Applikation gleichzeitig verwenden. Die maximale gleichzeitig abrufende Benutzer beträgt 50'000. |
| **Z.1.0** |  | H | **Zuverlässigkeit: Wiederherstellbarkeit**  Die Vollständigkeit der Daten hat oberste Priorität. Folgende 2 Fälle sollen abgedeckt werden:   1. Eine (Reservations-)Abfrage wird an den Server geschickt. Die Applikation stürzt ab. Beim Neustart der Applikation wird der letzte Status vom Server abgefragt. 2. Bevor eine (Reservations-)Abfrage abgesetzt wird, stürzt die Applikation ab. Beim Neustart der Applikation muss auf dem Gerät der letzte Status wiederhergestellt werden. |
| **Z.1.1** |  | M | **Zuverlässigkeit: Verfügbarkeit**  Die Applikation soll 99.99% der Zeit verfügbar sein.  Ausnahmen: bei Wartungen und Abschaltungen durch unerwartete Systemfehler kann es zu Ausfällen kommen; hierbei soll ein Backupsystem der Überbrückung dienen. Das Umschalten auf das Backupsystem kann zu kurzen Verzögerungen, maximal 5 Sekunden, führen.  Die erneute Verfügbarkeit der (Haupt-)Applikation soll nach 1 Std. erfolgen. |
| **S.1.0** |  | H | **Sicherheit: Zahlungsverkehr**  Der Zahlungsverkehr muss über abhörgeschützte, verschlüsselte Kanäle erfolgen. (ist nach Snowden heute gar nicht möglich!) |
| **S.1.1** |  | H | **Sicherheit: Vertraulichkeit**  Die Berechtigung für Manipulation und Einsicht der Daten soll unter den verschiedenen Benutzer-Rollen klar abgegrenzt sein. Das Sichten fremder Daten soll nur durch Administratoren erfolgen. Passwörter sind als Hash zu speichern. Genau gleiches gilt für Zahlungsmittel (Kreditkartennummer). |
| **G.1.0** |  | H | **GUI: Einfachheit**  Die Benutzung der Applikation soll einfach und simpel gehalten werden. Der Benutzer soll die Applikation intuitiv bedienen können.  Es sollen zu einem gegebenen Zeitpunkt nur die nutzbaren Informationen sichtbar sein. |
| **G.1.1** |  | H | **GUI: Design**  Die Applikation hat ein angenehmes, visuelles Design, das den Benutzer positiv anspricht. |
| **IT.1.0** |  | H | **IT: Wartbarkeit, Modifizierbarkeit, Erweiterbarkeit**  Es soll nach Clean Code und Best Practices gearbeitet werden damit auch im später die Applikation gut erweiterbar ist und gewartet werden kann. |

## Randbedingungen

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Status** | **Prio** | **Beschreibung** |
| R1.0 |  | H | Der Source Code der Applikation zwecks Code Review ist per 20.12.2019 der BFH abzugeben. Die Fertigstellung der Applikation muss nicht zu diesem Datum erfolgen. |
| R1.1 |  | T | Die zu verwendenden Entwicklungstechnologien für die Applikation sind…  (gilt es noch festzulegen….) |
| R1.2 |  | M | Fragen, die während der Erarbeitung der Spezifikationen entstehen, werden durch Peter Lange beantwortet. |
| R1.3 |  | H | Präsentation: 17.1.2019. |
| R1.4 |  | H | Abschluss: 22.1.2019 statt. Alle deliverables zu diesem Zeitpunkt. |

## Datenmodell

Im Folgenden Diagramm werden die Wichtigsten Entitäten dargestellt.



## Business Objekte

Folgende Business Objekte erachten wir als sinnvoll für eine effiziente Modellierung der Realität.

|  |  |
| --- | --- |
| **Objekt** | **Eigenschaften** |
| Sitzplatz | Nummer, 220V-Anschluss, 1./2. Klasse,… |
| Sitzplan | ID, «Blueprint» eines Wagens: wo sind welche Plätze situiert? |
| Wagen | Nummer, Wagentyp (Restaurant, Ruhe, …), Sitzplan |
| Zugkomposition | Welche Wagen werden in welcher Reihenfolge in eine Zugkomposition gestellt? |
| Strecke | Startort, Zielort, Zwischenstops, Fahrtzeiten, Stopzeiten |
| Zugservice | Strecke, Startzeit, Zugkomposition, Reservationen |
| Reservation | User: Name, Vorname, Geburtsdatum  Sitzplatz-Identifikation: Zugservice, Wagen, Sitzplatz, reservierte Teilstrecke, Abfahrtszeit, evtl. Ankunftszeit. |
| Haltestelle | Name (ID), Verweildauer (eines Zuges im Bahnhof) |
|  |  |

# Glossar

BLS Bern-Lötschberg-Simplon: Schweizerische Bahngesellschaft, mehrheitlich im Kanton Bern tätig.

# Literaturverzeichnis

**Literatureintrag**

*Autorname, Autorvorname, Buchtitel, Verlag, Ort, Ausgabe, Jahr* 7

**Literatureintrag**

*Autorname, Autorvorname, Buchtitel, Verlag, Ort, Ausgabe, Jahr* 9

**Literatureintrag**

*Autorname, Autorvorname, Buchtitel, Verlag, Ort, Ausgabe, Jahr* 11

# Anhang

## Abstimmung der Anforderungen

Eventuell aufgetretene Konflikte und die gewählte Lösung dafür werden kurz dokumentiert

## Definition of Ready – Checklist

Spezifische Kriterien, die die Anforderungen in diesem Projekt erfüllen müssen, um reif für die Umsetzung zu sein

# Versionskontrolle

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Datum** | **Beschreibung** | **Autor** |
| 1.4 | 14.10.2019 | Kapitel 5, Quellen identifizieren. | Y. Roth, A. Andrejic |
| 1.3 | 9.10.2019 | Kapitel 4 & business objects | Y. Roth, A. Andrejic |
| 1.2 | 2.10.2019 | Verbesserung Projektvision, Use Cases und Projektziele | Y. Roth |
| 1.1 | 30.09.2019 | Anforderungsspezifikationen | Y. Roth, A. Andrejic |