

Konzeptionierung und prototypische Implementierung eines Buchungssystems für Individualreisen

Michael Peeß

Bachelor-Thesis
Studiengang Informatik

Fakultät für Informatik
Hochschule Mannheim

10.03.2017

Durchgeführt bei der Firma Tervado GmbH, Mannheim

Betreuer: Prof. Dr. Miriam Föllner-Nord, Hochschule Mannheim

Zweitkorrektor: Dipl.-Kfm. Michael Wurst, Tervado GmbH

Erklärung

Hiermit erkläre ich, dass ich die vorliegende Arbeit selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe.

Die vorliegende Arbeit enthält vertrauliche Daten des Unternehmens Tervado GmbH. Die Arbeit darf nur dem Erst- und Zweitgutachter sowie befugten Mitgliedern des Prüfungsausschusses zugänglich gemacht werden. Eine Veröffentlichung und Vervielfältigung der Arbeit ist – auch in Auszügen – nicht gestattet.

Eine Einsichtnahme der Arbeit durch Unbefugte bedarf einer ausdrücklichen Genehmigung des Verfassers und des Unternehmens Tervado GmbH.

Mannheim, 10.03.2017

Michael Peeß

Inhalt

1	Einleitung	1
1.1	Motivation und Problemstellung.....	1
1.2	Zielsetzung.....	1
1.3	Aufbau der Arbeit	2
2	Grundlagen	3
2.1	Rollen des Buchungssystems	3
2.1.1	Mister Trip	3
2.1.2	Agentur	3
2.1.3	Kunde.....	4
2.2	Buchungsablauf.....	4
2.3	Allgemeine Begriffserklärungen.....	7
2.3.1	Framework.....	7
2.3.2	Model View Controller Pattern	7
2.4	Eingesetzte Technologien	8
2.4.1	Spring MVC	8
2.4.2	AngularJS	8
2.4.3	PostgreSQL.....	9
2.4.4	Hibernate.....	9
2.4.5	Apache Maven	9
2.4.6	REST API.....	10
2.4.7	Bower.....	10
2.4.8	requireJS	10
2.4.9	Angular Module.....	11
3	Anforderungsanalyse.....	12
3.1	Funktionale Anforderungen	12
3.1.1	Kunde.....	12

3.1.2	Agentur	13
3.1.3	Mister Trip	14
3.2	Nichtfunktionale Anforderungen	18
4	Konzeptionierung	20
4.1	Sonderfälle	20
4.2	Datenbankkonzept.....	22
4.3	Visualisierungskonzept	24
5	Implementierung	39
5.1	Entwicklungsumgebung.....	39
5.2	Frontend	39
5.3	Backend.....	40
5.4	Aufbau des Frontend	41
5.5	Aufbau des Backend	43
6	Ausblick.....	44
	Abkürzungsverzeichnis.....	iv
	Tabellenverzeichnis	v
	Abbildungsverzeichnis	vi
	Literaturverzeichnis	vii

1 Einleitung

Diese Bachelorarbeit beschäftigt sich mit Konzeptionierung und prototypischen Implementierung eines Buchungssystems für Individualreisen. Im folgenden Kapitel werden Motivation und Zielsetzung der Arbeit beschrieben.

1.1 Motivation und Problemstellung

In der heutigen Welt ist der Begriff Lifestyle, welcher die Art und Weise einer Lebensführung beschreibt, von immer größerer Bedeutung. Der Mensch will sich nicht mehr nach einem Einheitsschema ausrichten und seine eigenen, für ihn richtigen, Wege gehen - seinen individuellen Weg.

Dies zeigt sich auch in der Reisebranche. Der Kunde von heute möchte keine gewöhnliche Pauschalreise, ein vordefiniertes Komplettpaket zum Einheitspreis das sich nicht an Wünsche und Bedürfnisse anpassen lässt, buchen. Deshalb wurde das Konzept der Individualreisen entwickelt.

Unsere Reiseexperten sind zertifizierte Reiseberater ausgewählter, lokaler Reiseagenturen, die über viel Erfahrung in Ihrem Reiseland verfügen und sich vor Ort bestens auskennen. Sie geben Ihnen wertvolle Insidertipps aus erster Hand, sodass Sie Ihr Reiseland authentisch und hautnah erleben. Ihr persönlicher Reiseexperte ist wichtigster Ansprechpartner und steht Ihnen vor, während und nach der Reise jederzeit zur Verfügung. Unsere Reiseexperten gehen auf jede Reiseanfrage vollkommen individuell ein und erstellen Ihnen ein 100% maßgeschneidertes Angebot, ganz auf Ihre Wünsche und Bedürfnisse zugeschnitten. (1)

Da Angebotserstellung, Buchungsverlauf und die Abrechnung mitunter sehr komplex und Zeitaufwendig werden können, ist ein Buchungssystem eine denkbare Alternative für die händische Abwicklung.

1.2 Zielsetzung

Ziel dieser Arbeit ist die Erarbeitung eines Anforderungskataloges, eines Konzepts und die anschließende prototypische Entwicklung eines Buchungssystems.

Die Hauptfunktion des Buchungssystems soll Kunden, Agenturen sowie der Firma Mister Trip (Tervado GmbH) einen Überblick über die erstellten Angebote, Buchungen und geleistete sowie noch ausstehende Transaktionen ermöglichen. Außerdem soll ersichtlich werden welche Angebote eine Buchung zur Folge hatten.

Zusätzlich soll der Firma Mister Trip eine Plattform geboten werden die es ermöglicht Angebote und Gutscheine zu erstellen, Sonderfälle für einzelne Agenturen einzutragen Kontenbewegungen einzusehen und eine Datenbank der Agenturen und Kunden anzulegen.

Daraus ergeben sich mehrere Rollen, die benötigt werden, um die verschiedenen Sichten auf die Plattform und damit einhergehend die unterschiedlichen Rechte zu verwalten. Aus den Anforderungen ist ersichtlich, dass mindestens drei Rollen vorhanden sein müssen: Die des Kunden, die der Agentur und die der Firma Mister Trip.

1.3 Aufbau der Arbeit

In Kapitel 2 *Grundlagen* werden die Grundlagen, welche für das Verständnis dieser Arbeit benötigt werden, vermittelt. Um dem Leser das Umfeld der Individualreisen näher zu bringen werden zu Beginn die Rollen Kunde, Agentur und Mister Trip erläutert. Daraufhin folgt ein Standard Buchungsaufbau der verdeutlichen soll welche Schritte für die Buchung einer Individualreise nötig sind. Danach folgt die Erläuterung der eingesetzten Software die für die Implementierung verwendet wurde.

Anschließend erfolgt in Kapitel 3 *Anforderungsanalyse* eine Aufstellung der funktionalen- und nichtfunktionalen Anforderungen die an die Plattform gestellt werden. Hierfür wurden die Anforderungen nach den einzelnen Komponenten geordnet und priorisiert. Der Anforderungskatalog wird dabei in tabellarischer Form dargestellt.

Kapitel 4 *Konzeptionierung* enthält das Konzept für den Umgang mit Sonderfällen, Visualisierungs- sowie Datenbankkonzept sind ebenfalls in diesem Kapitel vorzufinden.

In Kapitel 5 *Implementierung* wird auf die prototypisch entwickelte Software, die als Grundkonstrukt für die Weiterentwicklung dient, eingegangen.

Abschließend wird in Kapitel 6 *Ausblick* erörtert wo noch Verbesserungspotenzial ist und welche Funktionalitäten zwingend umgesetzt werden müssen.

2 Grundlagen

Dieses Kapitel beschreibt die Grundlagen welche zum Verständnis der Arbeit benötigt werden. Zuerst werden die Rollen Beschrieben die den Nutzern des Buchungssystems zugeordnet werden.

Zum besseren Verständnis wird vorab ein Standard Buchungsablauf vom Angebot bis zur Buchung erläutert. Daraufhin werden die Rollen des Buchungssystems beschrieben, nachfolgend kommen allgemeine Begriffserklärungen und abschließend werden die eingesetzten Technologien erläutert.

2.1 Rollen des Buchungssystems

In diesem Kapitel werden die einzelnen Rollen des Buchungssystems beschrieben, aus denen sich die Rechte und damit auch die einzelnen Sichten der entsprechenden Rollen ableiten lassen.

2.1.1 Mister Trip

Da es sich bei Mister Trip um den Betreiber der Online-Plattform handelt, handelt es sich hierbei um die Rolle mit den meisten Rechten. Den Rechteinhabern dieser Rolle stehen alle Funktionen der Plattform in vollem Umfang zur Verfügung.

2.1.2 Agentur

Bei der Agentur handelt es um die einzelnen Reiseagenturen mit denen die Firma Mister Trip kooperiert. Damit einhergehend repräsentiert die Rolle auch die Reiseexperten die mit den Agenturen kooperieren. Im Verlauf dieser Arbeit sind die Begriffe Reiseexperte und Agentur als gleichwertig zu betrachten. Ein Reiseexperte arbeitet für eine Agentur, Agenturen können mehrere Reiseexperten beschäftigen, jedoch bleiben die Sicht und somit auch die Rechte immer auf dem Level einer Agentur.

Den Agenturen steht nicht der komplette Funktionsumfang zur Verfügung. So ist es den Agenturen nicht möglich die Datenbank der Agenturen oder der Kunden einzusehen, ebenso wenig ist es den Agenturen möglich Kontenbewegungen einzusehen.

2.1.3 Kunde

Der Kunde repräsentiert die Rolle mit den geringsten Rechten. Er ist lediglich in der Lage das für ihn freigegebene Angebot der Agentur anzeigen zu lassen und dieses gegebenenfalls zu akzeptieren.

2.2 Buchungsablauf

In diesem Abschnitt wird ein regulärer Buchungsablauf einer Individualreise beschrieben, um zu verdeutlichen wie die Erstellung der Reise abläuft.

Ein Kunde interessiert sich für eine Reise, die er auf der Homepage der Firma Mister Trip vorfindet und entscheidet sich dieses Angebot in Anspruch zu nehmen. Jedes Angebot ist mit einem Reiseexperten verknüpft, der im entsprechenden Zielland wohnt. Durch ein Webformular wird der Kontakt, welcher über E-Mail abläuft, zwischen Kunde und Reiseexperte hergestellt. Der Kunde äußert seine Wünsche und der Reiseexperte stellt die Reise je nach Kundenwunsch zusammen und fertigt mehrere Angebote an. Sobald sich der Kunde für eines dieser Angebote entschieden hat kommt das Buchungssystem zum Einsatz.

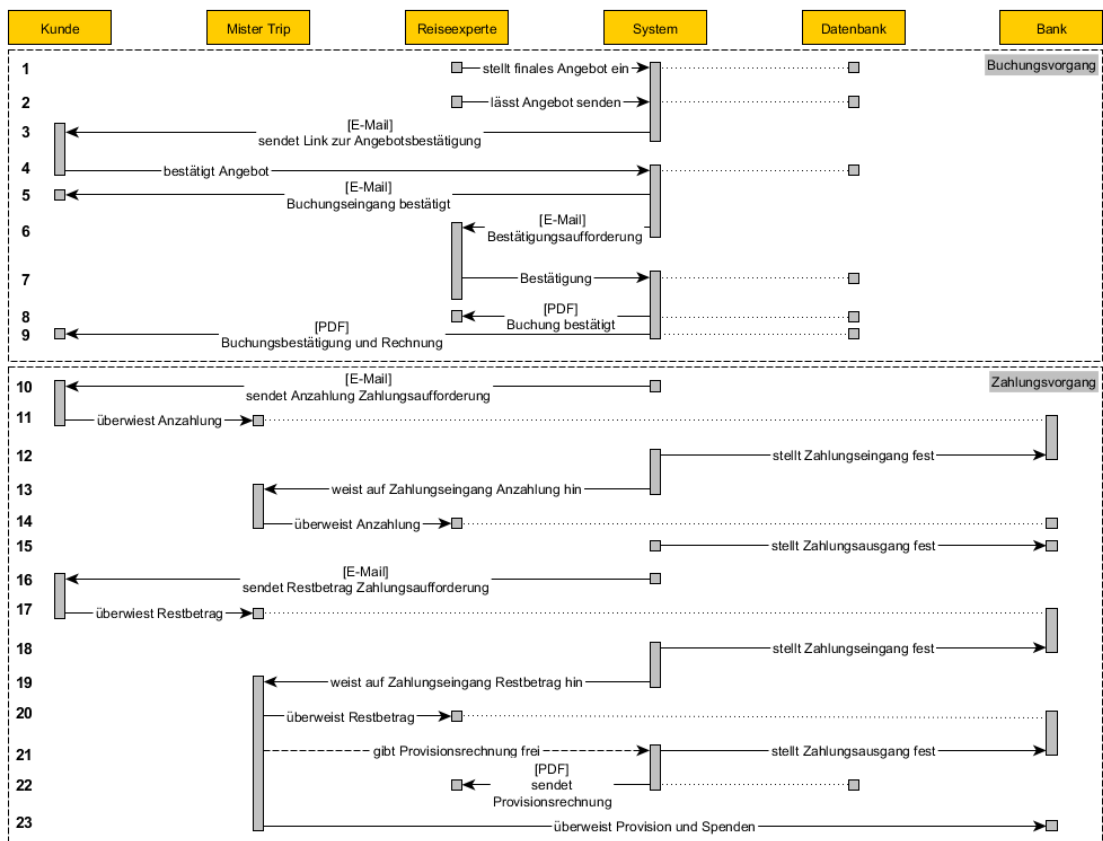


Abbildung 1: Sequenzdiagramm einer Standard Buchung

1. Der Reiseexperte legt ein neues Angebot im Buchungssystem an und lädt das Angebot als Portable Document Format (PDF) Datei hoch. Dieses wird vom System in der Datenbank gespeichert.
2. Nachdem der Upload erfolgt ist gibt der Reiseexperte das Angebot frei, so dass dieses an den Kunden gesendet werden kann. Das System speichert Datum, Uhrzeit, Username und IP von der die die Freigabe erfolgt ist.
3. Das System sendet dem Kunden einen Link zur Kundensicht des Buchungssystems.
4. In der Kundensicht des Buchungssystems überprüft der Kunde seine Kundendaten, das Angebot und muss die AGBs der Firma Mister Trip akzeptieren. Mittels eines Buttons kann er das Angebot verbindlich akzeptieren. Das System speichert Datum, Uhrzeit und IP in der Datenbank.
5. Das System sendet dem Kunden per E-Mail Bestätigung des Buchungseingangs.

6. Das System sendet zugleich eine Bestätigungsaufforderung an den Reiseexperten.
7. Der Reiseexperte prüft ob das gewünschte Angebot noch zur Verfügung steht. Ist dies der Fall bestätigt er die Buchung. Das System speichert Datum, Uhrzeit und IP in der Datenbank.
8. Das System generiert eine PDF Datei die alle Buchungsdaten und das Angebot enthält, speichert die PDF in der Datenbank und gibt es für den Reiseexperten frei.
9. Das System generiert zwei PDF Dateien. Die Buchungsbestätigung mit allen Informationen der Buchung und eine Rechnung die Fristen für An- und Restzahlung enthält. Die PDFs werden per E-Mail an den Kunden Versand und in der Datenbank gespeichert.

Ab diesem Zeitpunkt wird das Angebot im System als eine Buchung behandelt.

10. Eine Woche vor Fälligkeit der Anzahlung wird vom System ein Verwendungszweck für die Anzahlung generiert und per E-Mail an den Kunden Versand.
11. Der Kunde überweist, unter Angabe des Verwendungszwecks, die Anzahlung auf das Konto der Firma Mister Trip.
12. Das System stellt anhand des Verwendungszwecks den Zahlungseingang der Anzahlung fest.
13. In der Ansicht der Firma Mister Trip wird auf den Zahlungseingang der Anzahlung hingewiesen.
14. Ein Mitarbeiter der Firma Mister Trip überweist die Anzahlung an den Reiseexperten.
15. Der Zahlungsausgang wird vom System registriert und der Hinweis im Buchungssystem wird entfernt.
16. Eine Woche vor Fälligkeit der Restzahlung wird vom System ein Verwendungszweck für die Restzahlung generiert und per E-Mail an den Kunden Versand.
17. Der Kunde überweist, unter Angabe des Verwendungszwecks, die Restzahlung auf das Konto der Firma Mister Trip.

18. Das System stellt anhand des Verwendungszwecks den Zahlungseingang der Restzahlung fest.
19. In der Ansicht der Firma Mister Trip wird auf den Zahlungseingang der Restzahlung hingewiesen.
20. Ein Mitarbeiter der Firma Mister Trip überweist die Restzahlung an den Reiseexperten.
21. Sobald die Überweisung der Restzahlung getätigt wurde, wird über die Ansicht der Firma Mister Trip oder über den vom System festgestellten Zahlungsausgang die Generierung der Provisionsrechnung eingeleitet
22. Das System speichert die Provisionsrechnung in der Datenbank und sendet die Rechnung per E-Mail an den Reiseexperten.
23. Ein Mitarbeiter der Firma Mister Trip überweist die Provision auf ein separates Konto.

2.3 Allgemeine Begriffserklärungen

2.3.1 Framework

Ein Framework ist eine Art Rahmen, die dem Entwickler eine wiederverwendbare gemeinsame Struktur einer Software zur Verfügung stellt. Dabei wird das Framework in das eigene Projekt integriert und solange erweitert bis die gewünschte Funktionalität erreicht ist. Die einzelnen Teile des Frameworks können beliebig oft wiederverwendet werden und helfen dem Entwickler somit schneller eine einheitliche Software zu entwickeln.

2.3.2 Model View Controller Pattern

Das Model View Controller (MVC) Pattern¹ dient der Entwicklung klar strukturierter Software. Die strikte Trennung zwischen Geschäftslogik, der visuellen Darstellung und der Datenhaltung ermöglicht es, solange die Schnittstellen unverändert bleiben, die einzelnen Module zu verändern oder komplett auszutauschen ohne Änderungen an den anderen Modulen vornehmen zu müssen.

Dabei werden die Module wie nachfolgend definiert:

¹ Eine Patten ist ein Architekturmuster das Interaktion und Organisation von Komponenten definiert

- Model (Datenmodell)
 - Es speichert die Daten oder enthält die nötigen Datenbankabfragen.
- View (Präsentation)
 - Ist für die visuelle Darstellung der Daten zuständig.
- Controller (Programmsteuerung)
 - Beinhaltet die Logik und bearbeitet die anstehenden Ereignisse.

2.4 Eingesetzte Technologien

2.4.1 Spring MVC

Spring MVC ist ein quelloffenes Framework welches die Entwicklung von Webanwendungen im Java und JavaEE Umfeld erleichtern soll. Es ermöglicht die Entkopplung der Applikationskomponenten, fördert gute Programmierpraktiken. Hierzu werden einfache Java Objekte, oder auch Plain Old Java Objects (POJOs) als Spring Beans verwaltet, indem es einen Inversion of Control (IoC) Container zu Verfügung stellt, welcher mittels Dependency Injection (DI) von einander abhängige Beans verknüpft und verwaltet. (2)

Des Weiteren lassen sich mit dem Framework dynamische und REST-konforme Webanwendungen auf Basis der Servlet API realisieren. Dabei wird die stringente Implementierung des MVC Pattern vorausgesetzt. Eine Reihe von allgemeinen Web-Technologien wie HTTP, Servlet und WebSockets werden ebenfalls zur Verfügung gestellt.

2.4.2 AngularJS

Bei AngularJS handelt es sich um ein quelloffenes JavaScript Front-End Webframework das von Google Inc. entwickelt wurde. Mit dem Ziel die Erstellung von Architekturen sowie Komponenten die auf dem MV*-Pattern, welches das MVC und das Model-View-ViewModel (MVVM) Pattern enthält, basieren zu vereinfachen. (3) Dabei liegt der Fokus auf der Entwicklung von Single-Page-Web-Applications. Dies geschieht über die Einbettung zusätzlicher benutzerdefinierter Tag-Attribute im Hypertext Markup Language (HTML) Code, welche von Angular als Direktiven interpretiert werden um Input- oder Output-Teile der Seite an ein Model zu binden, das durch Standard-JavaScript-Variablen dargestellt wird. Dabei können die Werte der

Variablen innerhalb des Codes manuell festgelegt oder aus statischen oder dynamischen JSON-Ressourcen abgerufen werden. Die Generierung der HTML Seiten sowie das Interpretieren der Direktiven geschieht ausschließlich Clientseitig.

2.4.3 PostgreSQL

PostgreSQL ist ein objektrelationales Datenbankmanagementsystem, welches das Bindeglied zwischen klassischen relationalen Datenbanken und Objektdatenbanken darstellt, weitgehend konform zu SQL-Standard ist und die meisten dieser Funktionen anbietet. Es ist ein frei verfügbares Open-Source-Programm² und darf lizenzfrei eingesetzt werden. Als Datenbank-Server bestehen die Hauptaufgaben darin, Daten sicher zu speichern und diese auf Anfrage einer Softwareanwendung zurückzugeben. Somit basiert das Datenbanksystem auf Transaktionen und ist außerdem komplett ACID-konform. PostgreSQL verfügt über aktualisierbare Ansichten und materialisierte Ansichten, Trigger und Fremdschlüssel außerdem unterstützt PostgreSQL erweiterbare Datentypen, Operatoren, Funktionen und Aggregate.

2.4.4 Hibernate

Bei Hibernate handelt es sich um ein Object-Relational Mapping (ORM)- und Open-Source-Persistenz-Framework, dessen Hauptaufgabe es ist POJOs mit ihren Attributen und Funktionen umzuwandeln, um diese in einem relationalen Datenbankmodell zu speichern und bei Bedarf wieder als POJOs abrufen zu können. Dabei dürfen die Beziehungen der einzelnen Objekte zueinander nicht verloren gehen, sondern müssen ebenfalls als Relationen in der Datenbank gespeichert werden. So können die grundlegenden Unterschiede des relationalen Datenbankmodells zum objektorientierten Modell beseitigt werden und ein Object-relational Impedance Mismatch verhindert werden.

2.4.5 Apache Maven

Maven ist ein Build Management Tool von Apache. Das Tool erstellt und verwaltet Java-Programme standardisiert um den Entwickler beim Kompilervorgang zu unterstützen. Dies geschieht stets nach dem Softwaredesign-Paradigma „Konvention vor Konfiguration“ (4). Das bedeutet solange sich der Entwickler an vorgegebene Konventionen hält, ist ein großer Teil der Struktur bereits vorgegeben und dem Software-

² Open-Source ist eine Bezeichnung für eine Software dessen Quelltext frei einsehbar und modifizierbar ist.

Entwickler wird die Konfiguration der Strukturen und Abläufe, welche zum Kompilieren nötig sind, weitestgehend abgenommen. Die verborgene Verzeichnisstruktur blendet eher unwichtige Informationen aus und verbessert die Übersicht erheblich. Dabei werden sämtliche vom Projekt benötigten Abhängigkeiten im Extensible Markup Language (XML) Format in der Project Object Model (POM) Datei festgehalten. Wird Maven ausgeführt, prüft es die syntaktische Korrektheit der Datei und löst die Abhängigkeiten auf.

2.4.6 REST API

Representational State Transfer (REST) ist ein Programmierparadigma für verteilte Systeme. Die API schlüsselt eine Transaktion auf und erstellt daraus eine Reihe kleinerer Module. Jedes dieser Module ist für einen kleinen definierten Teil der Transaktion zuständig. Die daraus resultierende Modalität bietet in der Entwicklung eine hohe Flexibilität, muss jedoch von Entwicklungsbeginn an berücksichtigt werden.

Bei einem REST Application Programming Interface (API) handelt es sich um eine Programmschnittstelle, die HTTP-Anfragen verwendet, um per GET, PUT, POST und DELETE auf Daten zuzugreifen.

2.4.7 Bower

Bower ist ein Open-Source-Paketverwaltungstool welches mittels dem Node Package Manager (npm) installiert wird. Das Paketverwaltungstool ermöglicht, über ein in Node.js geschriebenes Kommandozeilentool, eine einfache Installation von Frameworks und Bibliotheken. Die Pakete werden direkt in das betreffende Projekt hinzugefügt und können bei Bedarf in einer JSON Datei gespeichert werden, der bower.json. Diese Datei enthält alle für das Projekt benötigten Abhängigkeiten und deren Versionsnummern.

2.4.8 requireJS

Die serverseitige asynchrone Open-Source-Moduldefinition requireJS ermöglicht es JavaScript-Dateien erst dann zu laden wenn diese benötigt werden. Dies entlastet den Server und verhindert unnötigen Datenverkehr.

2.4.9 Angular Module

Angular Route

Das Angular Modul Route bietet Routing-, Deeplinking-Services und -Direktiven für AngularJS-Apps an. Es wird für die Navigation von Single Page Applikations benötigt und ermöglicht das nachladen der View bei Routenänderungen.

Angular Material

Angular Material ein UI Component Framework welches von Google Inc. entwickelt wurde. Es zeichnet sich durch seinen einheitlichen und minimalistisch kantigen Gestaltungsstil aus. Das User Interface erinnert an Android und erzeugt durch leichte Schattierungen einen Tiefeneffekt der dem Nutzer dabei helfen soll interaktive Bereiche zu erkennen.

Angular Translate

Angular Translate ermöglicht die Internationalisierung mithilfe von Variablen im Code. Die Sprachdaten werden hierfür in Textdateien ausgelagert. Dies ermöglicht das asynchrone Laden anderer Sprachen und gewährleistet das hinzufügen neuer Sprachen ohne Quellcodeänderungen vornehmen zu müssen.

3 Anforderungsanalyse

In diesem Kapitel sind die wichtigsten funktionalen und nichtfunktionalen Anforderungen, die an das Buchungssystem gestellt werden, festgehalten. Wobei die Funktionalen Anforderungen den Funktionsumfang des Systems beschreiben und die nichtfunktionalen Anforderungen die Qualität in welcher die geforderte Funktionalität zu erbringen ist. Die Anforderungen sind nach Wichtigkeit priorisiert und zeigen die wesentlichsten Funktionsmerkmale auf.

3.1 Funktionale Anforderungen

Die Funktionalen Anforderungen sind in die einzelnen Rollen unterteilt, die im Buchungssystem vorhanden sind. Dies gewährleistet das den einzelnen Sichten die für sie zur Verfügung stehenden Funktionalitäten zugewiesen bekommen.

3.1.1 Kunde

Nr.	Kurzbeschreibung	Beschreibung	Priorität
FA-K01	Angebot akzeptieren	Das Angebot muss per Button akzeptiert oder abgelehnt werden können.	1
FA-K02	AGBs akzeptieren	Die AGBs müssen per Checkbox akzeptiert werden können.	1
FA-K03	Angebot herunterladen	Die Angebots PDF muss heruntergeladen werden können.	1
FA-K04	Kundendaten ansehen	Die Kundendaten müssen angezeigt werden.	1
FA-K05	Fehler melden	Fehler müssen über einen Button gemeldet werden können.	1
FA-K06	Gutscheinodes	Gutscheinodes müssen eingegeben werden können.	2

Tabelle 1: Funktionale Anforderungen aus Sicht des Kunden

3.1.2 Agentur

Angebote

Nr.	Kurzbeschreibung	Beschreibung	Priorität
FA-A01	Angebote	Angebote der Agentur müssen erstellt, angezeigt, bearbeitet und gelöscht werden können.	1
FA-A02	PDFs hochladen	Angebots-PDFs müssen hochgeladen werden können.	1
FA-A03	PDFs herunterladen	Angebots-, AGB- und Rechnungs-PDFs müssen heruntergeladen werden können.	1
FA-A04	Angebot freigeben	Angebote müssen für den Kunden freigegeben werden können.	1
FA-A05	Angebot akzeptieren	Das Angebot muss per Button akzeptiert oder abgelehnt werden können.	1
FA-A06	Details ansehen	Angebots-, Kunden- sowie Agenturdetails müssen angezeigt und bearbeitet werden können.	1
FA-A07	Angebots Status	Der derzeitige Status des Angebots sowie Statusleiste müssen angezeigt werden.	2
FA-A08	Angebote sortieren	Die Angebotsliste muss sortiert werden können.	3

Tabelle 2: Funktionale Anforderungen an die Komponente Angebote aus Sicht der Agentur

Buchungen

Nr.	Kurzbeschreibung	Beschreibung	Priorität
FA-A09	Buchungen	Buchungen müssen angezeigt werden können.	1
FA-A10	PDFs	Alle PDFs im speziellen die Angebots PDF dürfen nur noch heruntergeladen werden können.	1

FA-A11	Details ansehen	Buchungs-, Kunden- sowie Agenturdetails müssen angezeigt, dürfen aber nicht mehr bearbeitet werden können.	1
FA-A12	Buchungen stornieren	Buchungen müssen storniert werden können.	1
FA-A13	Zusatzpakete	Zusatzpakete müssen hinzugefügt werden können. Die Ermöglicht einen längeren Aufenthalt oder zusätzliche Exkursionen.	2

Tabelle 3: Funktionale Anforderungen an die Komponente Buchungen aus Sicht der Agentur

Einstellungen

Nr.	Kurzbeschreibung	Beschreibung	Priorität
FA-A14	Sonderfälle	Sonderfälle, bei abweichenden Steuerhöhen, müssen aus einer Liste gewählt werden können.	1
FA-A15	Experten	Neue Experten-Accounts müssen hinzugefügt werden können und automatisch der erstellenden Agentur zugeordnet werden.	2
FA-A16	Sprache	Eine Standardsprache muss gewählt werden können.	3

Tabelle 4: Funktionale Anforderungen an die Eigenschaften Angebote aus Sicht der Agentur

3.1.3 Mister Trip

Angebote

Nr.	Kurzbeschreibung	Beschreibung	Priorität
FA-M01	Angebote	Angebote der Agentur müssen erstellt, angezeigt, bearbeitet und gelöscht werden können.	1
FA-M02	PDFs hochladen	Angebots-, Rechnungs-, AGB- und Diverse-PDFs müssen hochgeladen werden können.	1

FA-M03	PDFs herunterladen	Angebots-, AGB- und Rechnungs-PDFs müssen heruntergeladen werden können.	1
FA-M04	PDFs bearbeiten	Automatisch generierte PDFs müssen bearbeitet werden können.	1
FA-M05	Angebot freigeben	Angebote müssen für den Kunden freigegeben werden können.	1
FA-M06	Details ansehen	Angebots-, Kunden- sowie Agenturdetails müssen angezeigt und bearbeitet werden können.	1
FA-M07	Angebots Status	Der derzeitige Status des Angebots sowie Statusleiste müssen angezeigt werden.	2
FA-M08	Angebote sortieren	Die Angebotsliste muss sortiert werden können.	3

Tabelle 5: Funktionale Anforderungen an die Komponente Angebote aus Sicht von Mister Trip

Buchungen

Nr.	Kurzbeschreibung	Beschreibung	Priorität
FA-M09	Buchungen	Buchungen müssen angezeigt werden können.	1
FA-M10	PDFs bearbeiten	Angebots-, AGB-, Rechnungs- und Diverse-PDFs können geändert werden, jedoch muss eine Kopie der alten Version gespeichert bleiben.	1
FA-M11	Details ansehen	Buchungs-, Kunden- sowie Agenturdetails müssen angezeigt und bearbeitet werden können.	1
FA-M12	Buchungen stornieren	Buchungen müssen storniert werden können.	1
FA-M13	Zusatzpakete	Zusatzpakete müssen hinzugefügt werden können. Die Ermöglicht einen längeren Aufenthalt oder zusätzliche Exkursionen.	2

FA-M14	Gutscheine	Gutscheine müssen auch nachträglich einer Buchung zugewiesen werden können.
--------	------------	---

Tabelle 6: Funktionale Anforderungen an die Komponente Buchungen aus Sicht von Mister Trip

Gutscheine

Nr.	Kurzbeschreibung	Beschreibung	Priorität
FA-M14	Gutscheine	Gutscheine müssen erstellt, angezeigt, bearbeitet und gelöscht werden können.	1
FA-M15	Gutscheincode	Der Gutscheincode muss automatisch generiert und nachträglich geändert werden können.	2
FA-M16	Gutscheine Gültigkeit	Gutscheine müssen eine Gültigkeitspanne besitzen.	2
FA-M17	Gutschein Status	Gutscheine müssen auf aktiv oder inaktiv geschaltet werden können.	2

Tabelle 7: Funktionale Anforderungen an die Komponente Gutscheine aus Sicht von Mister Trip

Sonderfälle

Nr.	Kurzbeschreibung	Beschreibung	Priorität
FA-M18	Sonderfälle	Sonderfälle müssen erstellt, angezeigt, bearbeitet und gelöscht werden können.	1
FA-M19	Sonderfall Standardwert	Sonderfälle müssen einen Standardwert besitzen, der nach dem Schema aus <i>Abbildung 2: Aktivitätsdiagramm eines Sonderfalls</i> zugewiesen wird.	1
FA-M20	Sonderfall Zuweisung	Sonderfälle müssen Agenturen zugewiesen und wieder entfernt werden können.	1
FA-M21	Zuweisungsanzeige	Es muss ersichtlich sein welchen Agenturen welche Sonderfälle zugewiesen sind.	1

Tabelle 8: Funktionale Anforderungen an die Komponente Sonderfälle aus Sicht von Mister Trip

Agenturen

Nr.	Kurzbeschreibung	Beschreibung	Priorität
FA-M22	Agenturen	Agenturen müssen erstellt, angezeigt, bearbeitet und gelöscht werden können.	1
FA-M23	Experten	Experten müssen erstellt, angezeigt, bearbeitet und gelöscht werden können.	1
FA-M24	Kontenbewegungen	Kontenbewegungen der Agenturen müssen ersichtlich sein und korrigiert werden können.	1
FA-M25	Zuweisungen	Experten sowie Agenturen müssen einander zugewiesen werden können.	1
FA-M26	Zuweisungen entfernen	Zuweisungen die getätigt wurden müssen wieder entfernt werden können	1

Tabelle 9: Funktionale Anforderungen an die Komponente Agenturen aus Sicht von Mister Trip

Kunden

Nr.	Kurzbeschreibung	Beschreibung	Priorität
FA-M27	Kunden	Kunden müssen erstellt, angezeigt, bearbeitet und gelöscht werden können.	1
FA-M28	Kontenbewegungen	Kontenbewegungen der Kunden müssen ersichtlich sein und korrigiert werden können.	1

Tabelle 10: Funktionale Anforderungen an die Komponente Kunden aus Sicht von Mister Trip

Kontenbewegungen

Nr.	Kurzbeschreibung	Beschreibung	Priorität
FA-M29	Kontenbewegungen	Kontenbewegungen müssen erstellt, angezeigt, bearbeitet und gelöscht werden können.	1

nen.

Tabelle 11: Funktionale Anforderungen an die Kontenbewegungen Angebote aus Sicht von Mister Trip

Einstellungen

Nr.	Kurzbeschreibung	Beschreibung	Priorität
FA-M30	Accounts	Neue Mister Trip-Accounts müssen hinzugefügt und gelöscht werden können.	1
FA-M31	Sprache	Eine Standardsprache muss gewählt werden können.	3

Tabelle 12: Funktionale Anforderungen an die Komponente Einstellungen aus Sicht von Mister Trip

3.2 Nichtfunktionale Anforderungen

Die nichtfunktionalen Anforderungen gelten für das gesamte Buchungssystem. Eine Einteilung in Rollen entfällt somit.

Nr.	Kurzbeschreibung	Beschreibung	Priorität
NFA-01	Modularität	Das Buchungssystem muss in die Module Model View und Controller aufgeteilt sein. Jedes der Module muss unabhängig von den anderen entwickelt werden können.	1
NFA-02	Maven	Das Build Management Tool Maven muss genutzt werden.	1
NFA-03	REST API	Für die Client Server Kommunikation muss die REST API verwendet werden.	1
NFA-04	Hibernate	Das Persistenz-Framework Hibernate muss für die Datenbankanbindung genutzt werden.	1
NFA-05	Sicherheit	Ein eingeloggter Benutzer darf nur die für ihn bestimmten Seiten und Daten	1

		sehen.	
NFA-06	Zugriffssicherheit	Alle Komponenten des Buchungssystems müssen vor unbefugten Zugriffen geschützt sein.	1
NFA-07	Robustheit	Das System muss robust sein.	1
NFA-08	Reaktionszeit	Die Reaktionszeiten auf Benutzereingaben sollen unter einer Sekunde betragen.	1
NFA-09	Zuverlässigkeit	Die Datenbank muss nach Ausfall wiederherstellbar sein.	1
NFA-10	Benutzbarkeit	Das System muss leicht erlernbar und verständlich sein. Kurze Erläuterungen müssen im System vorhanden sein.	1

Tabelle 13: Nichtfunktionale Anforderungen an das Buchungssystem

4 Konzeptionierung

Dieses Kapitel das Konzept des Buchungssystems erläutert. Der Umgang mit Sonderfällen wird beschrieben, das Datenbankkonzept und die Visualisierung werden beschrieben.

4.1 Sonderfälle

Ein Sonderfall ist ein Wert der variabel ist und vom üblichen Standard abweicht. Es gibt die verschiedensten Arten von Sonderfällen die bei einem Buchungsablauf auftreten können.

- Anfallende Steuersätze
- Abweichende Provisionshöhen
- Unterschiedliche An- und Restzahlungshöhen
- Höhe der Stornogebühren

Dies kann für den üblichen Geschäftsvorgang störend sein und soll, wenn es nach den Interessen der Firma Mister Trip geht, weitestgehend vermieden werden. Es ist nicht erwünscht Sonderfälle einzuführen und wird nur genehmigt wenn dies zwingend erforderlich ist. Um Anfragen diesbezüglich zu verringern werden alle Felder, die zwingend angezeigt werden müssen, in der Agentursicht als nicht bearbeitbare Textfelder angezeigt. Für die Agenturen nicht zwingend einzusehende Felder, werden komplett ausgeblendet. Sollte es dennoch zur Notwendigkeit der Einführung eines Sonderfalles kommen, wird betreffendes Feld als Dropdown Menu³ angezeigt. Daraus resultieren jedoch weitere Problemstellungen.

- Der Sonderfall kann ein Einzelfall sein und nur für eine oder wenige Reisen eine Gültigkeit besitzen.
- Existiert ein Sonderfall muss bei jeder Angebotserstellung und damit einhergehend bei jeder Buchung das betreffende Feld auswählbar sein.
- Bereits bestehende Buchungen oder Buchungen die keinen Sonderfall aufweisen, dürfen davon nicht beeinträchtigt werden.

³ Ein Dropdown Menu ist eine Liste mit Werten die durch Benutzeraktion nach unten ausklappt.

Dies führt zum Schluss das betreffende Felder einen Standardwert besitzen müssen, der gültig wird, bevor ein Sonderfall gewählt wurde oder keine Auswahl erforderlich ist, da kein Sonderfall vorhanden ist. Ebenso müssen alle Felder die infrage kommen einen Standardwert in der Sicht von Mister Trip besitzen. Dies dient als Rückfallversicherung, falls kein Standardwert der Agentur gefunden werden kann. Die Standardwerte von Mister Trip sind global gelten also für alle Agenturen, sofern kein abweichender Standardwert der Agentur gesetzt ist. Sonderfälle können für mehrere Agenturen verwendet werden da das Anlegen neuer Sonderfälle nur aus Sicht der Firma Mister Trip möglich ist.

Folgendes Diagramm zeigt die Vorgehensweise bei Bedarf eines Sonderfalls.

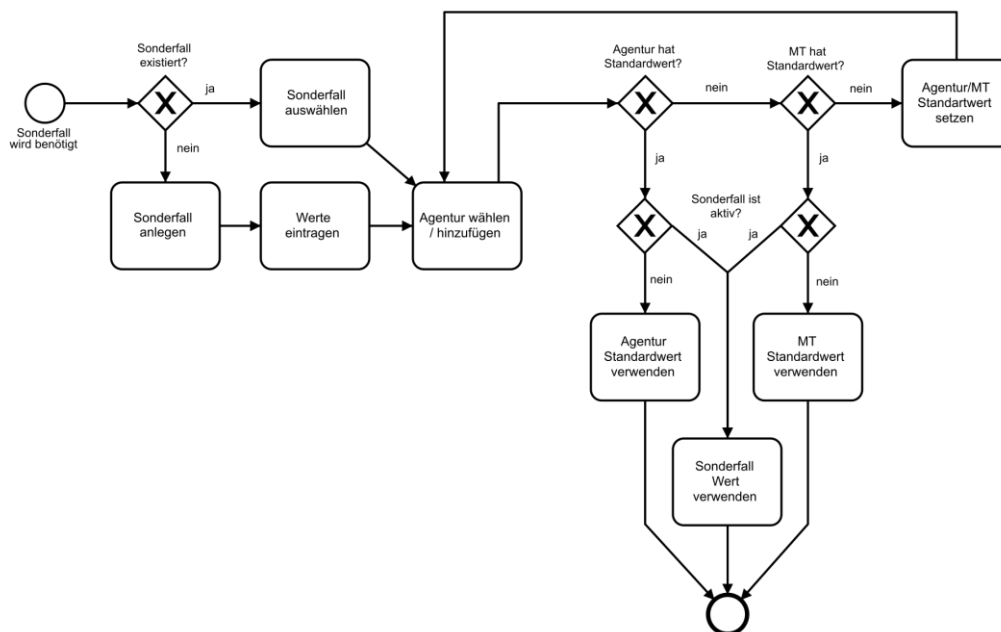


Abbildung 2: Aktivitätsdiagramm eines Sonderfalls

Sobald ein Sonderfall benötigt wird, ist zu prüfen ob ein derartiger Sonderfall bereits existiert und für die betreffende Agentur verwendet werden kann. Ist dies der Fall wird der Sonderfall gewählt und der Agentur zugewiesen. Ist dies nicht der Fall wird ein neuer Sonderfall erstellt, das Datenbankfeld wird gewählt, die Werte werden eingetragen und der Sonderfall wird der Agentur zugewiesen.

Da zu Anfang keine Standardwerte gesetzt sind müssen diese gesetzt werden. Die Standardwerte der Agentur, können aus Sicht der Agentur, in den Einstellungen aus einem Dropdown Menu gewählt werden. Aus Sicht von Mister Trip kann der Standardwert in den Agenturdetails über ein Dropdown Menu gewählt werden. Ebenso

muss beim Anlegen eines Sonderfalls der Mister Trip Standardwert gesetzt werden. Dies geschieht über die Einstellungen von Mister Trip. Existiert nur ein Wert wird dieser automatisch als Standardwert verwendet.

Existieren mehrere Werte und eine Agentur hat einen Sonderfall für eine Buchung verwendet, der im Verlauf der Buchung deaktiviert wird, wird der Standardwert der Agentur der die Buchung zugeordnet ist, verwendet. Besitzt die Agentur keinen Standardwert, weil dieser aus der Liste der Sonderfälle entfernt wurde wird der Standardwert von Mister Trip verwendet.

4.2 Datenbankkonzept

Das in dieser Arbeit beschriebene Datenbankkonzept dient als Grundlage für die Entwicklung der späteren Datenbank. Für die Entwicklung wird das objektrelationales Datenbankmanagementsystem PostgreSQL verwendet, da es für die geforderten Anforderungen am besten geeignet war. PostgreSQL ermöglicht komplexe Abfragen und lässt auch eine Schachtelung zu, bietet ein umfassendes Transaktionskonzept und bietet über Hibernate eine gute Schnittstellen zur Anbindung an Java und damit einhergehend auch an das Spring MVC Backend. Außerdem ist PostgreSQL durchweg ACID konform. Die Abkürzung ACID bildet sich aus den Anfangsbuchstaben der vier Eigenschaften Atomicity (Atomarität), Consistency (Konsistenz), Isolation (Isolation) und Durability (Dauerhaftigkeit). Wobei Atomicity für atomare Operationen steht, was bedeutet das alle Transaktionen als eine einzelne Operation angesehen werden die nicht unterbrochen werden kann. In der Praxis werden die Einzelaktionen zwar nacheinander ausgeführt, jedoch wird eine Transaktion erst dann für gültig erklärt wenn alle Einzelaktionen abgeschlossen sind. Consistency beschreibt die Konsistenzerhaltung der Datenbank. Werden Daten in eine Datenbank geschrieben die im Vorfeld konsistent war, so ist die Datenbank auch danach konsistent. Die Korrektheit der in der Datenbank befindlichen Daten ist extrem wichtig da durch inkorrekte Datensetze darüber liegende Abstraktionsschichten der Datenbank in ihrer Funktion beeinträchtigt werden können. Isolation wird durch Anwendung von Transaktionen gewährleistet, was verhindert das nebenläufig in Ausführung befindliche Daten-Operationen einander beeinflussen können. Unter Durability versteht man die dauerhafte Speicherung der Daten, die auch nach Systemausfällen sichergestellt sein muss. Dies kann über Transaktionslogs erreicht werden.

Das nachfolgende Entity-Relationship-Modell (ER-Modell) verdeutlicht den Aufbau der im Buchungssystem verwendeten Datenbankstruktur.

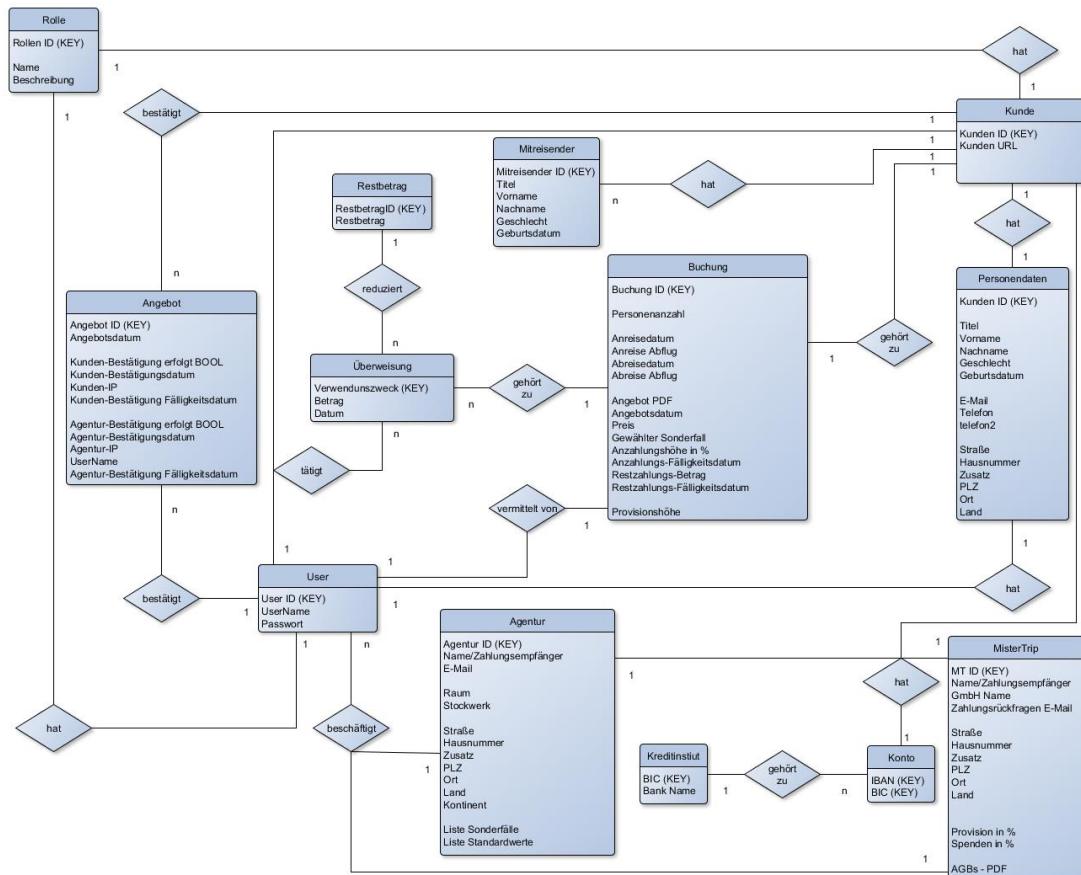


Abbildung 3: ER Modell der Datenbank

Aus diesem Modell geht hervor, dass die Agentur, MisterTrip und jeder Kunde jeweils ein Konto besitzen müssen. Dies ist für die ordnungsgemäße Funktion des Buchungssystems zwingend erforderlich. Die Konten wiederum stehen in Verbindung zu einem Kreditinstitut, wobei mehrere Konten zu einem Kreditinstitut gehören können.

Agentur und Mister Trip beschäftigen einen oder mehrere User, wobei jeder User eine Rolle innehaben muss. Der Kunde ist zwar ebenfalls Inhaber einer Rolle, wird aber nicht als User angesehen, da er kein Passwort benötigt. Der Kunde erhält lediglich eine E-Mail mit einer automatisch generierten Uniform Resource Locator (URL) und muss sich demnach nie einloggen, die Notwendigkeit eines Benutzernamens und eines Passworts entfällt somit. Kunden, User und demnach jeder Rolleninhaber besitzt Personendaten, welche die eindeutige Identifikation gewährleisten sollen.

Angebote müssen von Usern und Kunden bestätigt werden, wobei es sich in der Praxis nach dem Zeitpunkt der Bestätigung um eine Buchung handelt. Jedoch ist dies für das Datenbankschema irrelevant.

User vermitteln Reisen, welche wiederum mit Kunden und Überweisungen verknüpft sind. User und Kunden können beliebig viele Überweisungen tätigen, was den noch ausstehenden Restbetrag verringert.

Jeder Kunde kann beliebig viele Mitreisende haben, ob die Mitreisenden in der Datenbank gespeichert werden sollen wurde noch nicht abschließend geklärt.

4.3 Visualisierungskonzept

Im Folgenden Abschnitt wird das Visualisierungskonzept erläutert, welches im Rahmen dieser Arbeit mit der Software Axure RP erstellt wurde. Dabei handelt es sich um eine Software mit der schnell funktionsfähige Prototypen erstellt werden können die auch Animationen enthalten. Die Dynamic Panels die von dieser Software geboten werden sind perfekt dafür geeignet Single-Page-Web-Applications nachzuahmen, da die Funktionsweise nahezu die gleiche ist. Der Inhalt eines definierten Bereichs, dem Dynamic Panel, kann durch anderen Inhalt ausgetauscht werden und wird danach neu gezeichnet.

Der daraus resultierte Prototyp soll unabhängig von dieser Arbeit für Usability-Tests verwendet werden. Änderungen die daraus resultieren könnten, werden anschließend auch in der View des Buchungssystems durchgeführt. Der Modulare Aufbau des MVC Pattern wird dies noch begünstigen.

Der nachfolgende Teil zeigt ausschließlich die Sicht der Firma Mister Trip, da die Sicht alle Funktionalitäten in vollem Umfang enthält.

Übersicht

Abbildung 4: Konzept der Ansicht Übersicht

Die Übersicht fungiert als Startseite und ist die Ansicht zu der man gelangt nachdem man sich eingeloggt hat. Am linken Bildrand ist die Navigationsleiste vorzufinden, welche der Navigation zwischen den einzelnen Seiten dient. Die Menüpunkte die einen Pfeil enthalten lassen sich per Klick, ähnlich wie ein Dropdown Menu, nach unten ausklappen und bleiben bis zum nächsten Klick geöffnet.

Der obere Bildrand zeigt die Kopfleiste an, auf der linken Seite ist der Name des ausgewählten Menüpunkts abzulesen, was die Orientierung erleichtert. Mittels der Fahne, die rechts eingeblendet ist, lässt sich die Sprache ändern und über das Zahnrad gelangt man zu den Benutzereinstellungen. Im Falle eines Reiseexperten sind hier auch die Agenturbezogenen Einstellungen zu finden. Über den Logout Button gelangt der Nutzer zurück zum Anmeldemaske.

Oben in der Übersicht sind zwei Leisten eingeblendet. Diese fungieren als Filter. Darunter ist eine Liste von Panels dargestellt. Die Panels sind immer gleich aufgebaut und sind auch in den Angebots- und Buchungsansichten wiederzufinden. Sie enthalten alle für eine Reise erforderlichen Informationen. Die Angebots PDF ist hier ebenfalls eingefügt, da diese häufig benötigt wird. Buttons am rechten Rand des Panels ermöglichen die Einsicht der Details oder die Bearbeitung des betreffenden Angebots. Der unter Rand des Panels zeigt eine Art Ladebalken. Die Felder spiegeln wichtigen

Stadien, oder auch die Meilensteine, einer Reise wieder und werden grün eingefärbt wenn ein diese erreicht wurden. Liegt ein Problem vor wird das Feld rot eingefärbt. Der derzeitige Status wird blau angezeigt.

Angebote anzeigen

The screenshot shows the 'Angebote anzeigen' (View Offers) interface. It features a sidebar on the left with navigation options: Übersicht, Angebote, Anzeigen, Erstellen, Buchungen, Gutscheine, Sonderfälle, Agenturen, Kunden, and Kontobewegungen. The main content area is titled 'Angebote anzeigen' and includes a filter bar with buttons for 'Angebote ohne Buchungsfreigabe (17)', 'Angebote mit Buchungsfreigabe (9)', 'Angebote bei denen eine Buchung erfolgte (175)', and 'Angebote bei denen keine Buchung erfolgte (84)'. Below the filter bar are sorting and pagination options. Two travel offers are displayed, each with a progress bar and a status message. The first offer has a status of 'Ein Angebot wird benötigt' and the second offer has a status of 'Buchungsfreigabe erfolgt'. Both offers are for a trip to 'Das Beste von Südchina: Von Kunming bis Guilin'.

Angebot angelegt	Buchung freigegeben	Kunde bestätigt	Experte bestätigt	Kunde Anzahlung erfolgt	MT Anzahlung erfolgt	Kunde Restzahlung erfolgt	MT Restzahlung erfolgt	Kunde unterwegs	Kunde zurück
angelegt									
angelegt	freigegeben	bestätigt	bestätigt	erfolgt	erfolgt	erfolgt	erfolgt	unterwegs	

Abbildung 5: Konzept der Ansicht Angebote anzeigen

Die Ansicht Angebote anzeigen ist nahezu identisch zur Übersicht aufgebaut. Die Gestaltung der Seiten wurde so häufig wie möglich beibehalten, da Panels wiederverwendet werden können und die Orientierung deutlich erleichtert wird, da sich weniger häufig in wechselnde Oberflächen eingearbeitet werden muss. Lediglich die Filterarten wechseln je nach Anwendungsfall.

Angebote erstellen

MT Ansicht **Angebot erstellen** Logout

Reisedaten

Reisetitel *i*

Preis *i*

Anreisedatum *i*

Abreisedatum *i*

Personenanzahl *i*

Sonderfall *i*

Gutscheincode *i*

Kommentar *i*

Kosten

Anzahlung Höhe (berechnet) *i*

Anzahlung Fälligkeitsdatum (berechnet) *i*

Restzahlung Höhe (berechnet) *i*

Restzahlung Fälligkeitsdatum (berechnet) *i*

Provision Höhe (berechnet) *i*

Spenden Höhe (berechnet) *i*

Sorno Kosten derzeit in € (berechnet) *i*

Dokumente

Angebot *i*

Agentur Rechnung *i*

MT Kundenrechnung *i*

MT Provisionsrechnung *i*

MT AGBs *i*

Agentur AGBs *i*

Weitere Dokumente *i*

Buchungsbestätigung

Buchungsfreigabe erfolgt? Datum + Uhrzeit + User + IP

Kundenbestätigung erfolgt? Datum + Uhrzeit + IP

Link des Kunden *i*

Login *i*

Expertenbestätigung erfolgt? Datum + Uhrzeit + User + IP

Systeminformationen zur Reise

Reise ID *i*

Angebots ID *i*

Angebot-Erstellungsdatum User + IP *i*

Angebot-Änderungsdatum User + IP *i*

Buchungs ID *i*

Buchung-Erstellungsdatum User + IP *i*

Buchung-Änderungsdatum User + IP *i*

Kundeninformationen

Vor-/Nachname Telefon *i*

E-Mail Mobil *i*

[Kundendetails ansehen](#) [Kundendetails bearbeiten](#)

Agentur Informationen

Agentur Name *i*

Experte Telefon *i*

Vor-/Nachname Mobil *i*

E-Mail *i*

[Agenturdetails ansehen](#) [Agenturdetails bearbeiten](#)

[Angebot zur Buchung freigeben](#) [Angebot Preview](#) [Speichern](#) [Abbrechen](#)

Reise lösche

Abbildung 6: Konzept der Ansicht Angebot erstellen

Die Ansicht Angebot erstellen ist gleichzusetzen mit Angebot bearbeiten, da die gleichen Oberflächen dafür zur Benutzung kommen.

Oben links sind die Reisedaten angeordnet. Diese verändern sich im Normalfall nicht. Das Dropdown Menu der Sonderfälle ist hier ebenfalls angedeutet, wobei dies in der Praxis mehrfach vorkommen kann. Ein Sonderfall für jedes Datenbankfeld das variabel gehalten werden muss.

Darunter befinden dich die Kosten, welche nur zum Ablesen der Werte dienen. Alle Werte werden vom System anhand der Reisekosten und der hinterlegten Werte berechnet. Spenden, Provision, Stornokosten, An- und Restzahlung ergeben sich aus den vordefinierten Prozentwerten. Die Fristen können Zeitspannen eingestellt werden.

Unten auf der linken Seite sind die Dokumente dargestellt, die hoch sowie heruntergeladen werden können. Die Kundenrechnung und die Provisionsrechnung werden

vom System erstellt und können bei Bedarf auch manuell hochgeladen oder bearbeitet werden.

Die blauen i-Symbole neben den Textfeldern dienen als Anleitung. In wenigen Sätzen wird erläutert welche Informationen vorzufinden und einzutragen sind.

Rechts oben werden die Daten der Buchungsbestätigung angezeigt, darunter die Systeminformationen der Reise. Diese Werte können nur von Mister Trip eingesehen werden.

Rechts unten können die wichtigsten Kunden- und Agenturinformationen abgelesen werden. Über die Buttons gelangt man zur Detailansicht oder es können Änderungen vorgenommen werden.

Am unteren Ende der Seite kann das Angebot für den Kunden freigegeben werden. Dies wird vom Reiseexperten getätigt und der Kunde erhält einen Link per E-Mail, der auf eine Seite verweist auf der der Kunde das Angebot akzeptieren kann. Der Preview Button wurde nur zu Testzwecken eingebaut und führt zur Seite auf der der Kunde das Angebot annehmen oder ablehnen kann.

Kundensicht Angebot akzeptieren

Reisetitel z.B. (Das Beste von Südchina: Von Kunming bis Guilin)

Stammdaten

Titel

Anrede

Name

Vorname

Geburtsdatum

Anschrift

Straße

Hausnummer

Zusatz

PLZ

Ort

Land

Kontinent

Kontaktdaten

E-Mail

Telefon 1

Telefon 2

Fax

Skype



Reisezeitraum
14.11.17 - 24.11.17
(10 Tage)



Preis
1700,00 €



Personen
4



Angebot
ansehen



MT AGBS
ansehen

- Ich habe die AGBs gelesen und akzeptiert
 Ich möchte kostenpflichtig buchen

Angebot akzeptieren

Angebot ablehnen

Abbildung 7: Konzept der Kundenansicht Angebot akzeptieren

Eine Anpassung der Seite an das Design des Buchungssystems ist nicht erfolgt, da die Seite im Stil der Mister Trip Homepage gestaltet wird. Der Kunde hat die Möglichkeit seine Daten zu überprüfen und die für ihn freigegebenen PDF-Dateien einzusehen. Das Lesen der AGBs und der Wille zum verbindlichen Buchen müssen per Checkbox⁴ bestätigt werden, bevor das Angebot akzeptiert werden kann.

Angebots- und Buchungsdetailansichten

Die Detailansichten unterscheiden sich nur in wenigen Punkten von den Erstellungsansichten.

- Textfelder sind nicht bearbeitbar
- Info-Symbole werden nicht angezeigt
- Es gibt lediglich einen Zurück-Button, um zur vorherigen Ansicht zurück zu gelangen. Dies soll versehentliche Bearbeitungen vermindern.

Buchungen anzeigen und erstellen



Aufgrund der Tatsache, dass sich die Ansicht Buchungen anzeigen nicht von der Ansicht *Angebote anzeigen* unterscheidet, wird sie nicht separat erläutert.

Der Unterschied zwischen der Ansicht *Angebote erstellen* zur Ansicht zum Erstellen von Buchungen ist ebenfalls marginal. Lediglich ein Button zum Stornieren von Buchungen wurde hinzugefügt.

⁴ Eine Checkbox ist ein Auswahlkasten und dient zur Eingabe und Anzeige von Ja/Nein-Werten

Gutscheine anzeigen

The screenshot shows a web application interface for viewing vouchers. On the left is a dark sidebar menu with the following items: **MT Ansicht**, **Übersicht**, **Angebote**, **Buchungen**, **Gutscheine** (expanded), **Anzeigen**, **Erstellen**, **Sonderfälle**, **Agenturen**, **Kunden**, and **Kontobewegungen**. The main header is blue and contains the title **Gutscheine anzeigen**, a flag icon, a gear icon, and a **Logout** link. Below the header is a filter section with 'Sortieren nach:' (ID), 'Elemente pro Seite:' (50), 'Datum von:', 'Datum bis:', and a blue **Filter** button. The main content area features a table with the following data:

ID	Name	Code	Höhe	Datum von:	Datum bis:	Kommentar	Aktion
2133	Maimarkt	H5GZ12	100€	01.05.2017	31.07.2017	-	 

At the bottom of the page, there is a pagination control showing **<< < 1 2 ... 9 > >>** and a button labeled **Gutschein hinzufügen**.

Abbildung 8: Konzept der Ansicht Gutscheine anzeigen

Diese Ansicht zeigt das dritte und letzte Design, eine Anordnung in tabellarischer Form, das im Buchungssystem zum Einsatz kommt. Über die Buttons in der Aktionspalte der Tabelle können Gutscheine bearbeitet und gelöscht werden. Der Button rechts unten ermöglicht das Hinzufügen von Gutscheinen.

Gutscheine erstellen

Abbildung 9: Konzept der Ansicht Gutscheine erstellen

Das gewohnte Design der Erstellungsseiten wird auch in dieser Ansicht verwendet. Gutscheininformationen können bearbeitet und Systeminformationen abgelesen werden. Über die Datumsfelder wird der Zeitraum, über den der Gutschein aktiv ist, eingestellt. Die Farbe des Kreises unter dem Statuspunkt gibt an ob der Gutschein aktiv oder inaktiv ist. Die Checkbox links unten gewährleistet das Umschalten auf den manuellen Modus. Ist der Haken gesetzt sind die Datumsfelder außer Kraft gesetzt. Das Aktivieren und Deaktivieren ist dann nur noch über den Button darunter zu bewerkstelligen.

Sonderfälle anzeigen

Die Ansicht Sonderfälle anzeigen unterscheidet sich nicht von der *Gutscheine anzeigen* Ansicht. Die einzige Besonderheit dieser Ansicht ist eine mehrfachanzeige von Zeilen, wenn nach der Agenturliste sortiert wird. In der Liste sind alle Agenturen aufgeführt, die mit dem Sonderfall verknüpft sind.

Sonderfälle erstellen

Sonderfall erstellen

Sonderfalldaten

ID ⓘ

Name ⓘ

Datenbank-Feld ⓘ

Wert ⓘ

Standard-Wert ⓘ

Kommentar ⓘ

Status ● oder ● ⓘ

Systeminformationen zum Sonderfall

ID ⓘ

Erstellungsdatum ⓘ

Username ⓘ

Änderungsverlauf ⓘ

Agenturen

ID	Name	Hinzugefügt am	Wert	Standard Wert	Kommentar	Status	Aktion
0815	xyz Tours	07.12.15	0.3	0.2	-	aktiv	
4711	flying carpet	14.05.16	0.3	0.2	-	inaktiv	

Abbildung 10: Konzept der Ansicht Sonderfälle erstellen

Der obere Teil der Ansicht ist nach dem gewohnten Schema aufgebaut, mit der Besonderheit das der Standardwert eines jeden Sonderfalls in den jeweiligen Agenturen gewählt werden muss, denen der Sonderfall zugeteilt ist. Sonderfälle lassen sich ebenfalls aktivieren und deaktivieren. Im mittleren Teil der Ansicht wird eine Tabelle angezeigt die die referenzierten Agenturen enthält. Ein Klick auf das Icon in der ersten Spalte der untersten Tabellenzeile blendet ein modales Fenster⁵ ein, welches nachfolgend abgebildet ist.

⁵ Ein modales Fenster ist ein Dialogfenster das eingeblendet wird um Eingaben vom Benutzer einzuholen



Abbildung 11: Konzept des modalen Fensters beim Hinzufügen von Tabellenbelleneinträgen

Diese Fenster ermöglicht die Sonderfallauswahl aus einer Liste der bestehenden Sonderfälle oder das Erstellen eines neuen Sonderfalls.

Agenturen, Experten und Kunden anzeigen

Alle weiteren anzeigenden Ansichten sind im Tabellenstiel gestaltet der bereits aus der Ansicht *Gutscheine anzeigen* bekannt ist und werden daher nicht näher erläutert.

Agenturen erstellen

Logout

MT Ansicht

Agenturen erstellen

- Übersicht
- Angebote <
- Buchungen <
- Gutscheine <
- Sonderfälle <
- Agenturen >
- Agenturen anzeigen
- Agenturen erstellen
- Experten anzeigen
- Experten erstellen
- Kunden <
- Kontobewegungen <

Stammdaten Agentur

Name

Raum

Stockwerk

Straße

Hausnummer

Zusatz

Postleitzahl

Ort

Land

Kontinent

Kommentar

Standard AGBs

Kontaktdaten

E-Mail

Telefon 1

Telefon 2

Fax

Skype

Systeminformationen zur Agentur

ID

Erstellungsdatum

Username

Änderungsverlauf

Bankverbindung

Name

Vorname

Name des Geldinstituts

IBAN

BIC

Experten

ID	Name	E-Mail	Experte Hinzugefügt am	Laufende Angebote/Buchung	zu Agentur Hinzugefügt am	Kommentar	Aktion
10100	Linus Torvalds	Linus@MT.de	07.12.15	7 / 5	08.12.15	-	<input type="button" value="edit"/> <input type="button" value="delete"/>
111001	Klaus Knopper	Klaus@MT.de	14.05.16	12 / 3	20.05.16	-	<input type="button" value="edit"/> <input type="button" value="delete"/>

Sonderfälle

ID	Name	Datenbank-Feld	Wert	Standard Wert	Agentur	Kommentar	Aktion
1337	Anzahlung 30%	ANZAHLUNGSHOEHE	0.3	0.2	xyz Tours	-	<input type="button" value="edit"/> <input type="button" value="delete"/>
1739	Restbuchung Frist 10 Tage	REST_B_FRIST	10	25	xyz Tours	-	<input type="button" value="edit"/> <input type="button" value="delete"/>

Angebote / Buchungen

Angebote ohne
Buchungsfreigabe (17)

Angebote mit
Buchungsfreigabe (9)

Angebote bei denen
eine Buchung erfolgte
(175)

Angebote bei denen
keine Buchung erfolgte
(84)

Alle Angebote anzeigen

Offene Buchungen (17)

Abgeschlossene
Buchungen (9)

Alle Buchungen
anzeigen

Alle Reisen anzeigen

Kontobewegungen

Reise ID	Betrag	Typ	Zahlung von	Zahlung an	Frist	Überrweisung ausstehend	Kommentar	Aktion
10456	1624,00 €	Anzahlung	Michael (MT)	xyz Tours (Agentur)	1 Tage	ja	-	<input type="button" value="edit"/> <input type="button" value="delete"/>
20647	5463,00 €	Restzahlung	Herbert (Kunde)	Michael (MT)	-	nein	-	<input type="button" value="edit"/> <input type="button" value="delete"/>
34569	524,46 €	Provision	Michael (MT)	Michael (MT)	???	ja	-	<input type="button" value="edit"/> <input type="button" value="delete"/>
456789	371,85 €	Spende	Michael (MT)	WWF.D	33 Tage	ja	-	<input type="button" value="edit"/> <input type="button" value="delete"/>
56789	686,19 €	Storno	Karl (Kunde)	Michael (MT)	-3 Tage	ja	Fristüberschreitung!	<input type="button" value="edit"/> <input type="button" value="delete"/>
6123	867,89 €	Nachzahlung	Harry (Kunde)	Michael (MT)	-	nein	-	<input type="button" value="edit"/> <input type="button" value="delete"/>
777	978,62 €	Sonderposten	Michael (MT)	xyz Tours (Agentur)	-	nein	-	<input type="button" value="edit"/> <input type="button" value="delete"/>

Abbildung 12: Konzept der Ansicht Agenturen erstellen

Im oberen Teil der Ansicht werden die Daten der Agentur eingetragen. Darunter befinden sich die Textfelder für die Kontendaten. Diese finden sich nur in den Agenturansichten und in den Einstellungen von Mister Trip. Ob die Kontendaten der Kunden gespeichert werden, wurde noch nicht abschließend geklärt, da viele Kunden Bedenken deswegen haben könnten. Auf der rechten Seite lassen sich die Fristen bearbeiten, diese Option steht ausschließlich Nutzern zur Verfügung, die die Rolle Mister Trip innehaben.

Darunter befinden sich Listen der Experten und der Sonderfälle in tabellarischer Form. Die Filter unterhalb lassen die jeweiligen Anzeigeansichten einblenden. Der unterste Punkt zeigt die Kontenbewegungen, die mit der Agentur verknüpft sind, in einer Tabelle an.

Experten und Kunden erstellen

Die Ansicht Experten erstellen unterscheidet sich nur geringfügig von der vorherigen, weshalb auf eine Abbildung verzichtet wird. Sonderfälle und Kontobewegungen sind nicht enthalten. Der gravierendste Unterschied ist ein Button, der es ermöglicht das Passwort zurückzusetzen und als E-Mail an die in den Expertendaten eingetragene E-Mailadresse senden zu lassen.

Bei der Kunden erstellen Ansicht verhält es sich ähnlich, mit der Ausnahme das die Kontenbewegungen des Kunden als Tabelle aufgeführt werden.

Kontenbewegungen anzeigen

MT Ansicht
Kontobewegungen anzeigen
Logout

Offene Buchungen (12)

Abgeschlossene Buchungen (37)

Alle Buchungen anzeigen

Sortieren nach: ID

Elemente pro Seite: 50

Datum von:

Datum bis:

zeige meine Aufgaben (siehe ro)

Reise ID	Betrag	Typ	Zahlung von	Zahlung an	Frist	Überweisung ausstehend	Kommentar	Aktion
10456	1624,00 €	Anzahlung	Michael (MT)	xyz Tours (Agentur)	1 Tage	ja	-	
20647	5463,00 €	Restzahlung	Herbert (Kunde)	Michael (MT)	-	nein	-	
34569	524,46 €	Provision	Michael (MT)	Michael (MT)	???	ja	-	
456789	371,85 €	Spende	Michael (MT)	WWF-ID	33 Tage	ja	-	
56789	686,19 €	Storno	Karl (Kunde)	Michael (MT)	-3 Tage	ja	Fristüberschreitung	
6123	867,89 €	Nachzahlung	Harry (Kunde)	Michael (MT)	-	nein	-	
777	978,62 €	Sonderposten	Michael (MT)	flying carpet	-	nein	-	

« < 1 2 ... 9 > »

Manuelle Korrektur

Abbildung 13: Konzept der Ansicht Kontenbewegungen anzeigen

In dieser Ansicht werden alle Kontenbewegungen in tabellarischer Form angezeigt. Aufgaben, die vom Nutzer zu erledigen sind, werden im Buchungssystem hervorgehoben. Probleme wie Fristüberschreitungen werden stärker hervorgehoben. Die Symbole in der Aktionsspalte wechseln die Ansicht zur betreffenden Reise und ermöglichen die Bearbeitung über nachfolgend abgebildetes modales Fenster.

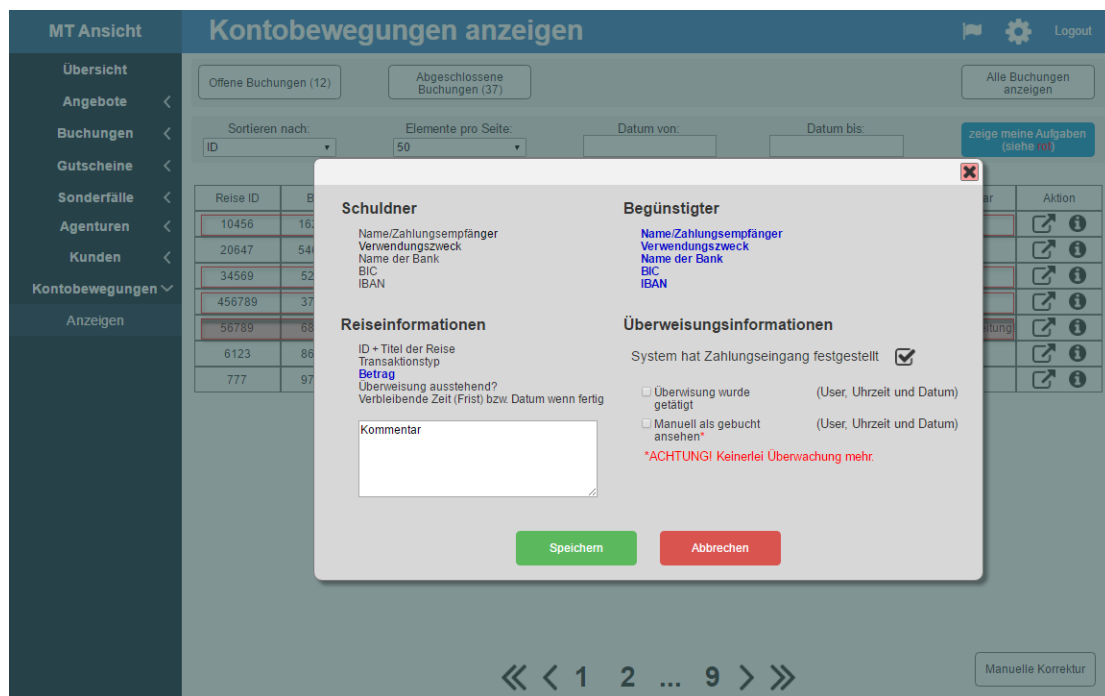


Abbildung 14: Konzept des modalen Fensters bei Kontenbewegungen

Das Fenster zeigt alle Informationen der betreffenden Buchung an. Überweisungen können per Checkbox als getätigt markiert werden, um die Markierung als Aufgabe zu entfernen. Ebenso bietet es eine Möglichkeit um Systemfehler, die durch nicht erkannte Zahlungseingänge entstehen können, manuell zu beheben.

Einstellungen

MT Ansicht
Einstellungen
Logout

- Übersicht
- Angebote <
- Buchungen <
- Gutscheine <
- Sonderfälle <
- Agenturen <
- Kunden <
- Kontobewegungen <

Stammdaten Mister Trip

Name:

Raum:

Stockwerk:

Straße:

Hausnummer:

Zusatz:

Postleitzahl:

Ort:

Land:

Kontinent:

Kommentar:

Kontaktdaten

E-Mail:

Telefon 1:

Telefon 2:

Fax:

Skype:

Systeminformationen zu Mister Trip

ID:

Erstellungsdatum:

Username:

Änderungsverlauf:

Bankverbindung

Name:

Vorname:

Name des Geldinstituts:

IBAN:

BIC:

Server URL:

Username:

Password:

Standard Dokumente

MT AGBS:

Weitere Dokumente:

User

ID	Name	E-Mail	Hinzugefügt am			Kommentar	Aktion
10100	Michael Wurst	Michael@MT.de	07.12.15			-	
111001	Christina	Christina@MT.de	14.05.16			-	

Speichern
Abbrechen

Abbildung 15: Konzept der Ansicht Einstellungen

Der obere Teil der Abbildung zeigt eine Auswahl an möglichen Feldern, die bearbeitet werden können. Sonderfälle sind in dieser Abbildung nicht vorhanden, werden aber dennoch hier mitaufgenommen. Auch einige weitere Login Daten, sowie eine Möglichkeit das Passwort des eingeloggtten Nutzers zu ändern, werden benötigt. In der unteren Tabelle sind alle Nutzer aufgeführt die der Eigenen Rolle zugewiesen sind. Ob dies bei den Agenturen, oder nur bei Mister Trip der Fall sein wird, wurde noch nicht abschließend geklärt.

5 Implementierung

In diesem Kapitel wird die Implementierung genauer beschrieben. Das Hauptaugenmerk dieser Arbeit liegt auf der Konzeptionierung einer soliden Basis die weiterentwickelt werden kann. Da es bei einem Buchungssystem um sensible Daten handelt, muss die Wahl der Technologie dem Anwendungsfall gerecht werden. Im Rahmen dieser Arbeit wurde das Grundkonstrukt mit funktionsfähigen Ebenen implementiert. Die Datenbankbindung per Hibernate sowie die Internationalisierung sind ebenfalls funktionsfähig. PDF Generierung, Versand von Mails und Bankanbindung nach Electronic Banking Internet Communication Standard (EBICS) sind derzeit noch nicht funktionsfähig.

5.1 Entwicklungsumgebung

Für die Implementierung wurde Eclipse, eine für Java entwickelte Integrated Development Environment (IDE)⁶, verwendet. Der Einsatz von anderen Entwicklungsumgebungen wie NetBeans ist ohne Probleme möglich, da alle verwendeten Erweiterungen auch hier existieren. Ein Tomcat 7 Server kann während dem Maven Kompiliervorgang, der aus Eclipse gestartet wird, heruntergeladen und gestartet werden.

5.2 Frontend

Für das Frontend wurde AngularJS in Verbindung mit dem Spring MVC Framework verwendet. Die Vorteile der hervorragend für die Erstellung von Single-Page-Web-Applications ausgerichteten Eigenschaften überwiegen den Bedenken der für ein Startup etwas überdimensionierten Unternehmensarchitektur, des Entwicklungskonzepts. Formularintensive Unternehmensanwendungen mit ihrer grundlegenden Idee den Server als eine Menge von zustandslosen wiederverwendbaren REST-Diensten zu entwickeln, mit dem Ziel den Controller des MVC Pattern aus dem backend in den Browser zu verschieben, schont die Ressourcen des Servers und erhöht die Performanz für den Anwender.

⁶ Eine IDE ist eine integrierte Entwicklungsumgebung und besteht aus einer Sammlung von Programmen die bei der Softwareentwicklung helfen

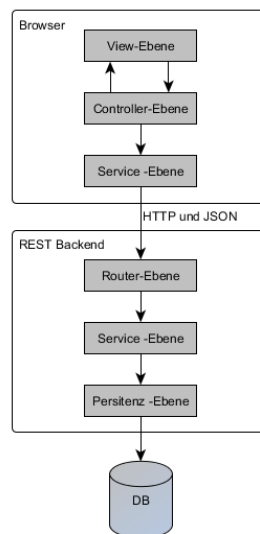


Abbildung 16: Front- und Backend-Ebenen

Aus der Abbildung wird ersichtlich, dass das MVC Pattern eingehalten wird, da Model View und Controller weiterhin getrennt bleiben, lediglich die Lokalisierung ändert sich. Die Controllerteile, die für die Präsentation zuständig sind, werden mit in den Browser verschoben. Die Präsentationslogik ist somit in drei Teile unterteilt: die View-Ebene, die Controller-Ebene und die Service-Ebene. Nach dem Anwendungsstart müssen nur noch HTTP und JavaScript Object Notation (JSON) Daten zwischen Client und Server ausgetauscht werden.

5.3 Backend

Das Backend kann mittels der REST API auf sehr webähnliche Weise konstruiert werden. Betrachtet man diesen Fall aus Sicht des Domain-driven Design (DDD) so verbleibt das Problemdomäne⁷ auf Service- und Persistenz-Ebene und somit auf dem Backend da ausschließlich Data Transfer Objects (DTOs)⁸ übertragen werden. Die Clientseite muss lediglich anzeigende Funktionalitäten übernehmen.

Das Backend wird in die üblichen Backendschichten aufgeteilt. Die Router-Ebene legt fest welche Service-Einstiegspunkte einer gegebenen HTTP-URL entsprechen und wie die Parameter aus der HTTP-Anfrage gelesen werden. Die komplette Ge-

⁷ Die Problemdomäne ist ein abgrenzbares Problemfeld mit speziellen Anforderungen an das technische System

⁸ DTOs dienen dem Datentransfer zwischen Services

geschäftslogik ist in der Service-Ebene enthalten und repräsentiert den Umfang der Geschäftsvorgänge. Die Persistenz-Ebene bringt die im Speicher befindlichen Objekte in die Datenbank und ermöglicht das erneute Abrufen dieser.

5.4 Aufbau des Frontend

Das Frontend befindet sich in der Projektstruktur im *webapp* Ordner. Alle Ressourcen, die für jeden der die Seite aufruft, unabhängig von vorhandenen Login Daten, zugänglich sind befinden sich im *public* Ordner. Der Ordner enthält die Angular Module die für das Anzeigen der Login Seite zuständig sind. Alle anderen Module befinden sich im *js* Ordner.

Nach erfolgreichem Login wird das Angular Modul *booking-system-app.js* ausgeführt. Diese enthält Konfiguration sowie alle Controller die für die Navigation erforderlich sind.

```
/**
 * Beispiel einer Konfiguration
 */
.config(function($routeProvider) {

    $routeProvider
        .when('/', {
            templateUrl : 'home.html',
            controller  : 'homeController'
        });

})

/**
 * Beispiel eines Controllers
 */
.controller('homeController', function($scope) {

    $scope.message = 'homeController';

})
```

In der *run-booking-system-app.js* befinden sich alle Module und ihre Abhängigkeiten die beim Start des Moduls geladen werden sollen. Diese Aufgabe wird von requireJS übernommen. Der Code repräsentiert eine Abhängigkeit, die von requireJS zur Laufzeit eingebunden wird, sollte sie benötigt werden.

```

/**
 * Beispiel einer requireJS Abhängigkeit
 */

require(['bookingSystemApp', 'translationApp'], function () {

    'use strict';

    angular.bootstrap(document.getElementById('bookingSystemApp'), ['book-
    ingSystemApp', 'translationApp']);

});

```

Im Hauptverzeichnis des Projekts befindet sich die *bower.json* Datei. Diese ermöglicht es, unter Zuhilfenahme des Paketverwaltungstools Bower, der Projektstruktur automatisch Programmbibliotheken und Frameworks hinzuzufügen zu lassen. Nachfolgend die derzeitige *bower.json* des Buchungssystems.

```

{
  "name": "mistertrip-booking-system",
  "version": "0.0.1",
  "authors": [
    "m.peess@gmx.de"
  ],
  "description": "Spring MVC | AngularJs project - Booking-System",
  "dependencies": {
    "angular": "~1.5.8",
    "angular-messages": "~1.5.8",
    "angular-cookies": "^1.5.8",
    "angular-translate": "~2.11.1",
    "angular-translate-loader-static-files": "^2.11.1",
    "angular-translate-storage-local": "^2.11.1",
    "angular-translate-handler-log": "^2.11.1",
    "angular-dynamic-locale": "^0.1.32",
    "angular-i18n": "^1.5.8",
    "lodash": "~4.15.0",
    "pure": "~0.6.0",
    "requirejs": "~2.2.0",
    "spring-security-csrf-token-interceptor": "0.1.5",
    "angular-material": "^1.1.1",
    "angular-route": "^1.5.8"
  },

```

```

"resolutions": {
  "angular": "~1.5.8",
  "angular-material": "^1.1.1",
  "angular-messages": "~1.5.8",
  "angular-translate": "~2.11.1",
  "angular-translate-loader-static-files": "^2.11.1",
  "lodash": "~4.15.0",
  "requirejs": "~2.2.0",
  "pure": "~0.6.0"
}
}

```

5.5 Aufbau des Backend

Die Verwaltung der Abhängigkeiten wird auch hier von einer Datei im Hauptverzeichnis des Projekts übernommen. Durch einen Eintrag innerhalb der *pom.xml* können auch hier Abhängigkeiten zu anderen Softwareprojekten oder Bibliotheken gesetzt werden. Maven legt diese in einem lokalen Verzeichnis, Repository genannt, ab und aktualisiert sie. Wird eine Abhängigkeit nicht mehr benötigt, wird der entsprechende Eintrag in der POM gelöscht. Auch transitive Abhängigkeiten werden automatisch entfernt.

```

<dependency>
  <groupId>org.springframework</groupId>
  <artifactId>spring-context</artifactId>
  <version>${org.springframework-version}</version>
  <exclusions>
    <exclusion>
      <groupId>commons-logging</groupId>
      <artifactId>commons-logging</artifactId>
    </exclusion>
  </exclusions>
</dependency>

```

Das Backend befindet sich in der Projektstruktur im üblichen Java Pfad. Derzeit verfügt das Backend lediglich über Controller und Services die für den Login-Vorgang zuständig sind. Eine Erweiterung dieser zu einer umfassenden Geschäftslogik war im Rahmen dieser Arbeit, aus Komplexitätsgründen und zu Gunsten der Frontendentwicklung, nicht möglich.

6 Ausblick

Das in dieser Arbeit entwickelte Buchungssystem muss in späteren Iterationen weiterentwickelt werden. Die Verwendung eines Buchungssystems bietet viele Vorteile, wenn geforderte Funktionalitäten vorhanden sind. Der durch die Software entfallende Mehraufwand wird sich im normalen Geschäftsbetrieb deutlich bemerkbar machen. Im speziellen die EBICS Bankanbindung wäre eine sinnvolle Erweiterung.

Auch könnten weitere Verbesserungen an der Architektur vorgenommen werden. So werden derzeit noch Front- und Backend im gleichen Vorgang kompiliert. Der Modulare Aufbau des MVC-Patterns würde auch eine komplett getrennte Entwicklung zulassen. Dies würde zu einer erheblichen Zeitersparnis führen, da der Kompilierungsvorgang derzeit bereits 50 Sekunden dauert, was bei der AngularJS Frontendentwicklung an der View zu erheblichen Verzögerungen führt. Zwar können die Backendtests übersprungen werden wenn nur an der View des Frontends gearbeitet wurde, doch die Trennung könnte, für die in AngularJS geschriebenen Teile, Änderungen im laufenden Betrieb zulassen.

Es sollten ebenfalls frühzeitig Usability Test mit dem entwickelten Prototyp gestartet werden, um den Änderungsaufwand so gering wie möglich zu halten.

Abkürzungsverzeichnis

API	Application Programming Interface
DTOs	Data Transfer Objects
DI	Dependency Injection
EBICS	Electronic Banking Internet Communication Standard
ER-Modell	Entity-Relationship-Modell
HTML	Hypertext Markup Language
IDE	Integrated Development Environment
JSON	JavaScript Object Notation
IoC	Inversion of Control
MVC	Model View Controller
MVVM	Model-View-ViewModel
npm	Node Package Manager
ORM	Object-Relational Mapping
PDF	Portable Document Format
POJOs	Plain Old Java Objects
POM	Project Object Model
REST	Representational State Transfer
URL	Uniform Resource Locator
XML	Extensible Markup Language

Tabellenverzeichnis

Tabelle 1: Funktionale Anforderungen aus Sicht des Kunden	12
Tabelle 2: Funktionale Anforderungen an die Komponente Angebote aus Sicht der Agentur	13
Tabelle 3: Funktionale Anforderungen an die Komponente Buchungen aus Sicht der Agentur	14
Tabelle 4: Funktionale Anforderungen an die Eigenschaften Angebote aus Sicht der Agentur	14
Tabelle 5: Funktionale Anforderungen an die Komponente Angebote aus Sicht von Mister Trip	15
Tabelle 6: Funktionale Anforderungen an die Komponente Buchungen aus Sicht von Mister Trip	16
Tabelle 7: Funktionale Anforderungen an die Komponente Gutscheine aus Sicht von Mister Trip	16
Tabelle 8: Funktionale Anforderungen an die Komponente Sonderfälle aus Sicht von Mister Trip	17
Tabelle 9: Funktionale Anforderungen an die Komponente Agenturen aus Sicht von Mister Trip	17
Tabelle 10: Funktionale Anforderungen an die Komponente Kunden aus Sicht von Mister Trip	17
Tabelle 11: Funktionale Anforderungen an die Kontenbewegungen Angebote aus Sicht von Mister Trip	18
Tabelle 12: Funktionale Anforderungen an die Komponente Einstellungen aus Sicht von Mister Trip	18
Tabelle 13: Nichtfunktionale Anforderungen an das Buchungssystem	19

Abbildungsverzeichnis

Abbildung 1: Sequenzdiagramm einer Standard Buchung.....	5
Abbildung 2: Aktivitätsdiagramm eines Sonderfalls.....	21
Abbildung 3: ER Modell der Datenbank	23
Abbildung 4: Konzept der Ansicht Übersicht	25
Abbildung 5: Konzept der Ansicht Angebote anzeigen	26
Abbildung 6: Konzept der Ansicht Angebot erstellen.....	27
Abbildung 7: Konzept der Kundenansicht Angebot akzeptieren	28
Abbildung 8: Konzept der Ansicht Gutscheine anzeigen.....	30
Abbildung 9: Konzept der Ansicht Gutscheine erstellen	31
Abbildung 10: Konzept der Ansicht Sonderfälle erstellen	32
Abbildung 11: Konzept des modalen Fensters beim Hinzufügen von Tabellenbelleneinträgen.....	33
Abbildung 12: Konzept der Ansicht Agenturen erstellen.....	34
Abbildung 13: Konzept der Ansicht Kontenbewegungen anzeigen.....	36
Abbildung 14: Konzept des modalen Fensters bei Kontenbewegungen	37
Abbildung 15: Konzept der Ansicht Einstellungen.....	38
Abbildung 16: Front- und Backend-Ebenen.....	40

Literaturverzeichnis

1. **Mister Trip.** Mister Trip: Individualreisen von Deinem Reiseexperten vor Ort. [Online] 01. 01 2016. [Zitat vom: 03. 03 2017.] <https://www.mistertrip.de/>.
2. **Johnsons, Rod.** *Expert One-On-One J2EE Design and Development*. s.l. : wrox Press, 2002. ISBN 0-7645-4385-7.
3. **Waikar, Manjo.** *Data-oriented Development with AngularJS*. s.l. : Packt publishing Ltd., 2015. ISBN 978-1-78439-805-7.
4. **John McBrewster, Frederic P. Miller, Agnes F. Vandome.** *Apache Maven*. s.l. : Alpha Press, 2010. ISBN 978-6-13-065219-7.