Konzeption und Prototyp eines Verwaltungstool für Buchungen von Wohnräumen

Analyse der verteilten Anwendung

Eine Arbeit von
Stefan Baumann, Florian Gümbel

entstanden im Sommersemester 2016 im Fach
„Entwicklung verteilter Anwendung“ bei
Prof. Dr. Karim Kremer
(16.09.2016)

Technische Hochschule Mittelhessen
Wilhelm-Leuschner-Straße 13
61169 Friedberg

Zusammenfassung

Eine Agentur, die Wohnräume vermietet, möchte ein Programm haben um die Buchungen zu Verwalten. Zu den Prämissen gehört, dass es Betriebssystem unabhängig nutzbar ist und Änderungen auf schnellstmöglichen weg an die Mitarbeiter kommuniziert wird um Doppebuchungen zu vermeiden. Es soll ebenfalls möglich sein zur gleichen Zeit Buchungen, Kunden und Wohnungen hinzufügen zu können.

Ein Dienstleister welcher in einem Feriengebiet seine Inserate verwalten will, bietet einen Buchungsservice per Telefon an. In seinen Call-Centern sitzen Mitarbeiter, welcher mithilfe der Software Buchungen aufnehmen und verwalten können. Dabei soll bei allen Anwendern die Synchronität der Daten sichergestellt werden, ohne, dass diese kontinuierlich manuell aktualisiern müssen. Eine Kalender-Ansiccht hilft bei der Verwaltung von Buchungsobjekten und den gebuchten Zeiträumen.

Inhaltsverzeichnis

1 Einleitung

Für die Vermietung seiner Ferienwohnungen und Häuser benötigt ein Unternehmen eine Software zur Verwaltung der jeweiligen Buchungen. Da das Unternehmen über mehrere Buchungsagenten verfügt, ist eine Anwendung nötig die das parallele arbeiten an Verschiedenen Geräten ermöglicht. Dazu gehören nicht nur Computer und Laptops sondern auch Tablets um schnell von unterwegs eine Buchung einzutragen. Besonders wichtig ist dem Unternehmen die schnelle Aktualisierung der Daten in der Software um eventuelle Doppelbuchungen zu vermeiden und das gleichzeitige Arbeiten zu ermöglichen.

2 Technologie

Dieses Kapitel stellt einen Überblick über die verwendeten Technologien dar.

2.1 AngularJS

AngularJS ist ein von Google entwickeltes JavaScript-Framework für clientseitige Webanwendungen. Es funktioniert nach dem Model-View-View-Model Prinzip. Es eignet sich besonders gut für Single-Page-Applications. Dabei werden die meisten Daten beim ersten Aufruf geladen. Das führt dazu, dass bei einer Änderung der URL nicht mehr die komplette Seite aktualisiert wird sondern lediglich die benötigten Daten per Ajax nachgeladen werden. Dadurch, dass bei AngularJS alles mittels JavaScript gerendert wird, stellt die Suchmaschinenoptimierung einen zusätzlichen Aufwand da, da die Suchmaschinen damit noch ihre Probleme haben. Abgesehen davon verfügt AngularJS über viele Stärken. Dazu gehören unter anderem Two-Way Binding, sehr gute Testbarkeit, Abstraktion von Low-Lovel-Operationen so wie die Lesbarkeit und Erweiterung von HTML-Code.

2.2 Firebase

Wie auch AngularJS stammt das Framework Firebase aus dem Hause Google. Allerdings wurde die Plattform nicht von Grund auf von Google entwickelt sondern erst im Jahr 2014 von Google übernommen. Es stellt eine universelle App-Plattform für Entwickler und Marketer zur Verfügung. Es dient zur Entwicklung von hoch-qualifizierten Anwendungen. Das Herzstück des Framework ist die Analyse von Apps und mobilen Anwendungen. Des Weiteren bietet es Cloud-Speicher, Cloud-Messaging, Remote Config und Test Lab. Ebenfalls stellt es die Möglichkeit zur Authentifizierung bereit so wie eine Echtzeit NoSQL Cloud Datenbank.

2.3 AngularJS Material Design

Passend zu Angular gibt das das User Interface Component Framework AngularJS Material Design. Es ist ein Zusammenschluss der Material Design Guidelines und dem AngularJs Framework. Dies soll dabei helfen, zeitgemäße

attraktive konsistente und funktionale Webseitendesigns zu Erstellen und auf allen Endgeräten zu gewährleisten.

2.4 Grunt

Grunt ist ein sogenannter JavaScript-Taskrunner der dazu da ist um wiederkehrende Aufgaben in Build-Prozessen in Frontend-Projekten zu automatisieren. Sobald er einmal konfiguriert ist, ist das Testen und Ausliefern selbst bei umfangreichen Projekten problemlos möglich. Es hilft viele Schritte wie zum Beispiel das minifizieren von JavaScript-Code oder das Umwandeln von SASS-Code zu CSS-Code von zentraler Stelle aus zu Steuern.

2.5 NPM

Node Package Manager ist ein Kommandozeilenprogramm für node.js. Es erleichtert JavaScript Entwicklern das Teilen von Code, welcher erstellt wurde um besondere Probleme zu lösen. Diese wiederverwendbaren Codeschnipsel werden Package oder manchmal auch Module genannt. Die Idee dahinter ist, dass ein Package ein Problem richtig löst. Das ermöglicht es, mit Hilfe von vielen kleinen Package zu einer Lösung zu gelangen.

2.6 Bower

Wie auch der node package manager ist Bower ein Kommandozeilen Paketverwaltungstool für die clientseitige Webentwicklung. Er ist sogar in Node.js geschrieben und wird über NPM installiert. Es dient zur Installation und Aktualisierung von Programmbibliotheken und Frameworks. Als Ergänzung zu NPM und in Zusammenarbeit mit Grunt kann der Workflow erheblich beschleunigt und verbessert werden.

2.7 SASS

SASS ist ein ausgereifter, etablierter und leistungsfähiger CSS-Präprozessor. Mit der Dateiendung .scss kann die Erweiterung genutzt werden. Da die Browser aktuell keine .scss-Dateien unterstützen, muss das SASS-Kommandozeilentool den Code in .css-Dateien übersetzen. Andersherum ist es ohne Probleme möglich CSS-Code in SASS-Dateien zu kopieren und zu nutzen. Dieser wird korrekt umgewandelt. Zu den Vorteilen von Sass gehört zum Beispiel Verschachtelungen, Variablen, Mixins, Vererbung und Importe.

2.8 Compass

Compass baut mit der Spracherweiterung Sass ein Framework. Es ist quasi die Standardbibliothek für Sass und bietet vieles was auch gängige CSS-Framewokrs enthalten. Compass erlaubt eine einheitliche Schreibweise die in verschiedene Eigenschaften übersetzt wird. Das hat zur Folge, dass sich die Entwickler keine Gedanken machen müssen,ob alle Browser abgedeckt werden. Ebenfalls ermöglicht es das automatisierte Erstellen von CSS-Sprites.

3 Projekt

Durch die in der Einleitung erwähnte Problemstellung ergeben sich folgende Zielsetzung und Umsetzung.

3.1 Zielsetzung

Prototypische Entwicklung einer Software zur Verwaltung von Buchungen für Ferienhäuser und Wohnungen. Eine der Anforderungen an die Software war die Nutzung auf verschiedenen Endgeräten. Das hat zur folge, dass auch die Betriebssysteme variieren können. Daraus ergibt sich, dass die Anwendung plattformunabhängig nutzbar sein muss.

Für diesen Fall eignet sich eine Webanwendung besonders gut. Sie ist von jedem Gerät aus nutzbar und benötigt in der Regel nicht mehr als einen Internetbrowser. Auch die Installation eines extra Programmes erübrigt sich. Allerdings ist eine stetige Internetverbindung Voraussetzung. In Anbetracht der weiteren Bedingungen an die Software ist diese aber auch bei einer nativ installieren Anwendung Voraussetzung. Da sie für den stetigen abgleich der Daten zwingend vorhanden sein muss. Das Ziel ist es also eine Webanwendung mit einer Benutzeroberfläche zu erstellen die auf jedem Gerät angezeigt werden kann sofern es über Internet und einen entsprechenden Browser verfügt. Des weiteren wird ein entsprechendes Backend benötigt, dass die Daten speichert.

Weitere Anforderungen an die Anwendungen sind die Verwaltung von Kunden, der zu vermietbaren Ferienhäuser und Wohnungen sowie die eigentliche Buchung. Das ganze soll übersichtlich dargestellt werden und einfach und schnell Benutzbar sein.

3.2 Umsetzung

Für die Umsetzung der eigentlichen Webanwendung kommen JavaScript, HTML und CSS zum Einsatz. Zu dem werden die in Kapitel 2 genannten Frameworks und Tools zur Unterstützung eingesetzt.

3.2.1 Backend

Für das speichern der Daten soll die in Firebase integrierte Objektbasierte Datenbank zum Einsatz kommen. Ebenfalls übernimmt ein weiterer Dienst von Firebase die Authentifizierung bei der Anmeldung.

3.2.2 Frontend

Neben den oben genannten Script- und Auszeichnungsprachen soll für die Logik und Strukturierung der Anwendung das Framework AngularJS genutzt werden. Dazu passend soll das Layout mittels der Angular Version von Materialize gestaltet werden.

3.2.3 Funktionen

Die Funktionen sollen sinnvoll in die Webanwendung integriert werden. Dabei sollen sie schnell und ohne Umwege erreichbar sein. Zu den Hauptfunktionen zählt das Hinzufügen und Bearbeiten von Kundendaten, Ferienwohnungen und der Buchung selbst. Folgende Informationen sollten für jeden Kunden zur eindeutigen Identifizierung gespeichert werden:

- Firma (Falls vorhanden)
- Vor- und Nachname
- Strasse und Hausnummer
- Postleitzahl und Stadt
- Telefonnummer
- Geburtstag
- Feld für Zusatzinformationen

Die zu vermietenden Ferienhäuser und Wohnungen sollten folgenden Attribute haben:

- Name des Objektes
- Anzahl der Personen die es maximal beherbergen kann

Um eine Buchung zu erstellen sollte die folgenden Informationen festgehalten werden:

- Kunde, der das Objekt bucht
- Objekt, das gebucht wird
- Anzahl der Personen
- Startdatum
- Enddatum

Um eine bessere Übersicht über die Buchungen zu bekommen, sollen diese optisch in einem Kalender dargestellt werden.

4 Aufbau Projekt

4.1 Frontend

4.2 Backend

5 Fazit

Abbildungsverzeichnis

Tabellenverzeichnis

Listings

Glossary

Cascading Style Sheet Gestaltungssprache für elektronische Dokumente und Programme wie HTML-Webseiten oder JavaFX-Anwendungen..

CSS *siehe* Cascading Style Sheet.

Database Management System System, welches die Datenbank aufbaut und verwaltet. Ziel ist ein möglichst effizientes und einfach zu bedienendes System..

DBMS *siehe* Database Management System.

Extensible Markup Language Kompaktes Datenformat, in etwa vergleichbar mit JSON..

Graphical User Interface Frontend-Ansicht, welche dem Benutzer eines Programms ausgeliefert wird..

GUI *siehe* Graphical User Interface.

HTML *siehe* Hypertext Markup Language.

HTTP *siehe* Hypertext Transfer Protocol.

Hypertext Markup Language Eine textbasierte Auszeichnungssprache zur Strukturierung digitaler Dokumente wie Texte mit Hyperlinks, Bildern und anderen Inhalten..

Hypertext Transfer Protocol Protokoll zur Übertragung von Daten über ein Rechnernetz..

JavaFX Framework zum Aufbau von einfachen bis komplexen Java-Frontendsystemen..

JavaScript Object Notation Kompaktes Datenformat in leicht verständlicher Textform, in etwa vergleichbar mit XML..

JSON *siehe* JavaScript Object Notation.

Plain Old Java Object Ein Java-Objekt im herkömmlichen Sinne..

POJO *siehe* Plain Old Java Object.

QR *siehe* Quick Response Code.

Quick Response Code Zweidimensionaler Code, der schnell durch Maschinen gelesen werden kann..

SQL *siehe* Structured Query Language.

Structured Query Language Datenbanksprache, um durch Datenbanken zu navigieren, Bearbeitungen, Löschvorgänge und Neueintragungen vorzunehmen..

Token Methode zur Autorisierung von Software-Diensten..

UI *siehe* User Interface.

Uniform Resource Locator Identifizierung einer Netzwerkressource, beispielsweise ein Server..

URL *siehe* Uniform Resource Locator.

User Interface Siehe Graphical User Interface.

XML *siehe* Extensible Markup Language.