NAD - Abschlussaufgabe

# Applikation 1: Informationssystem Mobilfunk

Es soll eine Applikation eigenständig, d.h. in Einzelarbeit, entwickelt werden, die das Kunden-Informations-System (CRM-System) eines Mobilfunk-Anbieters simuliert.

# Fachliche Anforderungen

Es gibt folgende Objekte:

* Tarif
  + Tarifname
  + Preis
  + Liste der verfügbaren Optionen
* Option
  + Optionsname
  + Preis
* Kunde
  + Nachname
  + Vorname
  + Geburtstag
  + Adresse
  + E-Mail-Adresse
  + Passwort
* Vertrag
  + Vertragsnummer
  + Tarif
  + Gebuchte Tarifoptionen
  + Kundenreferenz

Nach Verifikation des Passworts eines Kunden soll die Applikation folgende **Funktionalitäten** für Kundendaten bereitstellen:

* Den Vertrag / die Verträge im persönlichen Kundenkonto ansehen
* Zum aktuellen Tarif alle möglichen Tarif-Übergänge anzeigen
* Möglichkeit einen Tarifwechsel anzustoßen

Es gibt die Tarife S, M, L.

Ein Upgrade kann man in jeden „höheren“ Tarif machen. (S->M; S->L; M->L).

Ein Downgrade ist nur in einen „nächst-niedrigeren“ Tarif möglich (L->M; M->S), L->S ist nicht erlaubt.

* Zum aktuellen Tarif alle möglichen / buchbaren Optionen anzeigen, neue Optionen zum Tarif hinzufügen, vorhandene Optionen entfernen.

Buchbare Optionen können frei ausgedacht werden (z.B. MultiSIM, Hotspot Flat, Spotify, …).

Während der Bearbeitung von Verträgen auf einer Seite, d.h. vor Speicherung der Änderungen, soll ein Einkaufswagen angezeigt werden mit den vom Kunden ausgewählten Positionen.

Erst beim „Check-Out“, d.h. „kostenpflichtig bestellen“ des Einkaufswagens, werden die Positionen im Backend übernommen.

# Technische Anforderungen

Es soll eine multi-user client-server Applikation mit Netzwerk-Anbindung entwickelt werden.

Alle Daten sollen auf dem Server gespeichert werden. Jeder Client (Browser-Applikation) kann relevante Teile der Daten vom Server abfragen. Nach jeder Änderungs-Operation sollen die Änderungen mit dem Server entsprechend synchronisiert werden.

# Zu verwendende Technologien

Folgender Technologie-Stack soll verwendet werden:

* IDE: IntelliJ, Eclipse/STS, VSCode
* Server mit RESTful Web Services
  + Build Tool Maven
  + Spring Framework (Spring Boot, Spring Data JPA)
  + Database Backend – SQL
* Client
  + Browser based (Chrome, Firefox, Edge)
  + Build Tool npm
  + Single Page Application auf Basis von React oder Angular

# Erfolgskriterien

1. Die angeforderte Funktionalität ist implementiert, es gibt ein User Interface dazu
2. Das Maven-basierte Projekt ist in Module aufgeteilt (je ein eigenes Kommando für Build, und Deploy)
3. Es gibt eine fachliche Schnittstellen-Beschreibung.
4. Eine SQL Datenbank ist angebunden.
5. Die fachlichen Objekte sind implementiert, es gibt ein Mapping auf die jeweiligen Datenbank-Tabellen.
6. Das Deployment der Applikation erfolgt als Spring Boot Applikation innerhalb eines Docker Images.
7. Logging ist implementiert.
8. Eine technische Lösungsbeschreibung liegt vor.
9. Unit Tests für die Business Logik liegen vor.

# Applikation 2: Marketing / Werbung

# Fachliche Anforderungen:

Es gibt folgende Objekte:

* Kunde
  + Name
  + Vorname
  + Adresse
  + E-Mail-Adresse
  + Tarifname
* Campaign
  + Id
  + Schedule Date
  + Kunden
  + Inhalt

Nach dem Login soll ein Produkt Manager eine Marketing-Kampagne für ausgewählte Kunden erstellen können.

Die Kunden sollen anhand von Tarif oder Wohnort für eine Kampagne ausgewählt werden können.

Zu jeder Kampagne gehören, neben der Liste der ausgewählten Kunden, ein Startdatum, sowie ein Textfeld, das den Inhalt der Kampagne beschreibt.

# Technische Anforderungen

Die Objekte sollen in einer separaten Datenbank gespeichert werden.

Die Marketing-Applikation soll unabhängig von der ersten Applikation arbeiten.

Änderungen von Kundendaten oder Tarifdaten in der ersten Applikation sollen mit dieser Marketing-Applikation synchronisiert werden.