**Author(s): Sandro Bürki**

**Team: OVWEB**

**Version No.: 1.0**

**Date: 2023.10.13**

**Table of Contents**

[Titel der IPA 3](#_heading=h.gjdgxs)

[OVWEB – Fertigstellung der Kundenerfassung 3](#_heading=h.30j0zll)

[Ausgangslage 3](#_heading=h.1fob9te)

[Aufgabenstellung 3](#_heading=h.3znysh7)

[1.1 Funktionale anforderungen & Vorgaben 3](#_heading=h.2et92p0)

[1.2 nichtfunktionale anforderungen & Vorgaben 3](#_heading=h.tyjcwt)

[1.2.1. Usability 3](#_heading=h.3dy6vkm)

[1.2.2. Security 3](#_heading=)

[1.2.3. Testing 4](#_heading=h.4d34og8)

[1.3 dokumentationsvorgaben 4](#_heading=h.2s8eyo1)

[1.3.1. Dokumentation des Ergebnisses 4](#_heading=h.17dp8vu)

[1.3.2. Interne Dokumentation 4](#_heading=h.3rdcrjn)

[1.3.3. API-Dokumentation 4](#_heading=h.26in1rg)

[1.3.4. VCS 4](#_heading=h.lnxbz9)

[1.3.5. Sonstiges 4](#_heading=h.35nkun2)

[1.4 Technische Vorgaben 4](#_heading=h.1ksv4uv)

[1.5 Abgrenzungen 4](#_heading=h.44sinio)

[Mittel und Methoden 4](#_heading=h.2jxsxqh)

[Vorkenntnisse 4](#_heading=h.z337ya)

[1.6 Weitere Vorkenntnisse 5](#_heading=h.3j2qqm3)

[Vorarbeiten 5](#_heading=h.1y810tw)

[Neue Lerninhalte 5](#_heading=h.4i7ojhp)

[Arbeiten in den letzten 6 Monaten 5](#_heading=h.2xcytpi)

# Titel der IPA

# OVWEB – Fertigstellung der Kundenerfassung

# Ausgangslage

Die Generali Schweiz AG ist eine Versicherungsgesellschaft mit ungefähr einer Million Kunden in der Schweiz. Die Generali bietet Produkte für verschiedene Lebenssituationen an. Darunter zählen beispielsweise Sach-, Rechtsschutz- und Lebensversicherungen sowie Vorsorgelösungen. Für die Beratung von Kunden verwenden die Versicherungsvertreter eine von der Generali entwickelte Software namens «ANGEBOTsoftware», welche unter anderem das Verwalten von Kundendaten und Berechnen von Prämien ermöglicht, sowie diverse Anleitungen und Erklärungen zu den verschiedenen Produkten und Verkaufsprozessen enthält. Die Software wird sowohl von internen als auch externen Versicherungsvertretern benützt.

Die ANGEBOTsoftware wird hauptsächlich mit der Programmiersprache Delphi als Monolith entwickelt. Der Berechnungskern der ANGEBOTsoftware steht als API zur Verfügung. Die Aufgabe des Lernenden-Teams ist es, Teile dieser Applikation Stück für Stück in neuere, webbasierte Technologien zu überführen. Diese Applikation namens «OVWEB» befindet sich aktuell noch in der Startphase. Bereits vorhandene Funktionalitäten sind die Authentisierung, die Autorisierung sowie ein erster Entwurf für das Erfassen von Basis-Kundeninformationen.

# Aufgabenstellung

Zwar existiert wie erwähnt in der aktuellen Version der Applikation eine Eingabemaske zum Erstellen von Kunden, jedoch fehlen für eine vollständige und qualitativ hochwertige Funktionalität noch einige Aspekte. Mit diesem Projekt soll das bestehende Formular fertiggestellt werden.

## Funktionale anforderungen & Vorgaben

Die Applikation soll für die folgenden Eigenschaften eines Kunden erfassen können:

* Persönliche Angaben
  + Anrede / Titel
  + Geschlecht (m / w)
  + Vorname
  + Nachname
  + Geburtsdatum
* Kontaktangaben
  + Telefonnummer
  + E-Mail-Adresse
* Adressangaben
  + ISO-Code des Landes des Wohnortes
  + Postleitzahl
  + Wohnort
  + Strasse
  + Hausnummer

## nichtfunktionale anforderungen & Vorgaben

### Usability

Alle Angaben sollen sowohl client- als auch serverseitig validiert werden (siehe Abgrenzungen). Bei Fehlermeldungen soll der Benutzer so darüber informiert werden, dass ihm klar ist, ob eine Lösung des Problems seinerseits möglich ist – und wenn ja, welche. Werden Daten geladen, sodass der Benutzer warten muss, so soll ihm dies angezeigt werden. Die Applikation muss auf mobilen Geräten bedienbar sein. Die Usability kann dabei jedoch vernachlässigt werden.

### Security

Nur authentifizierte Benutzer dürfen Kunden erstellen. Es muss ersichtlich sein, welcher Vertreter wann welchen Kunden erfasst hat.

### Testing

Die Applikation soll mit Unit- und/oder Integration-Tests versehen werden.

## dokumentationsvorgaben

### Dokumentation des Ergebnisses

Die Dokumentation der resultierenden Software soll eine technische Dokumentation sein, welche sich an Softwareentwickler oder -architekten richtet.

### Interne Dokumentation

Die für interne Entwickler relevanten Informationen sollen im REAMDE.md des jeweiligen Repository nachgeführt werden.

### API-Dokumentation

Jegliche Änderungen an der API sind in den entsprechenden Annotationen zu führen, sodass das generierte SwaggerUI ohne Codeeinsicht verständlich ist.

### VCS

Commits sollen strukturiert und durchdacht erstellt werden. Es wird mindestens einmal am Tag gepusht. Für die Benennung der Commits gilt es die [Conventional Commits Specification](https://www.conventionalcommits.org/en/v1.0.0/#summary) einzuhalten.

### Sonstiges

Auf Kommentare im Quellcode soll bewusst verzichtet werden, wenn die Verständlichkeit des Codes durch eine klarere Schreibweise oder Strukturierung des Codes selbst verbessert werden könnte.

## Technische Vorgaben

Änderungen an der Datenbank müssen als FlyWay Migrations definiert werden. Für die API-Endpoints werden genau definierte DTOs für Requests und Responses verwendet.

## Abgrenzungen

Die API der ANGEBOTsoftware bietet Endpoints zur Auflistung von Ländern, Wohnorten etc. Diese werden erst in einem späteren Stadium der Applikation verwendet und sollen nicht in der Applikation integriert werden. Die Validierung soll somit durch Überprüfung der Datentypen und einfachen Inhaltschecks (E-Mail-Adressformat etc.) implementiert werden.

# Mittel und Methoden

Die Applikation soll mit dem von der Generali zur Verfügung gestellten Laptop umgesetzt werden. Die zu verwendenden Technologien sind Docker, Next.js, npm, Spring Boot mit Java 17, Maven und PostgreSQL. Als Versionisierung soll git verwendet werden. Die Wahl der Entwicklungsumgebung ist der kandidierenden Person freigestellt. Empfohlen wird sowohl für Back- als auch Frontend IntelliJ IDEA Ultimate.

Die Firmenstandards der Generali sollen eingehalten werden. Dazu zählen der Design Guide und die in den Projekten konfigurierten Coding Styles. Werden Elemente implementiert, die nicht im Design Guide vorhanden sind, so gilt es sich an der offiziellen Webseite der Generali oder anderen bereits implementierten Tools zu orientieren.

Die Wahl der Projektmethodik ist der kandidierenden Person ebenfalls freigestellt. Empfohlen wird IPERKA.

# Vorkenntnisse

Fülle deine Vorkenntnisse mindestens über die untenstehenden Technologien ungefähr aus. Dabei zählen auch Kurse aus der Berufsschule, dem Basislehrjahr oder privaten Projekten, welche du umgesetzt hast. Halte dich dabei an das Schema X Monate/Jahre Y, bspw. 3 Jahre Spring Boot.

* Java
* Spring Boot
* JavaScript
* TypeScript
* React
* Next.js
* Docker
* GitHub Actions
* Git

## Weitere Vorkenntnisse

Gerne kannst du auch noch weitere Punkte hinzufügen. Jegliche Deklaration von Vorkenntnissen hilft den Expert\*innen dich besser einschätzen zu können.

# Vorarbeiten

Auf dem Arbeitsgerät wurde bereits die Applikation inkl. Entwicklungsumgebung eingerichtet. Für die Probe-IPA wurde eine Dokumentationsvorlage für den IPA-Bericht sowie den Zeitplan erstellt.

# Neue Lerninhalte

Womit hast du noch nie gearbeitet? Hast du gewisse Aspekte der Aufgabe noch nie in einem anderen Projekt entwickelt? Notiere diese hier.

# Arbeiten in den letzten 6 Monaten

Notiere hier deine Arbeiten in den letzten 6 Monaten. Eine Beispielformulierung kann entweder stichwortartig oder die folgende sein:

Die kandidierende Person hat seit sechs Monaten Software-Projekte im Devisenhandel, im FX-Entwicklungsteam der Musterbank-Abteilung ABCD, umgesetzt. Der erste Auftrag war das Beziehen von MMRates inklusive Konsolidierung und Weiterleitung an FIMAS (2 Monate). Der zweite Auftrag war die Tenor-unabhängige Erfassung von Margen und die dazugehörige Implementation der Business-Logik (2 Monate). Als Tools wurden Eclipse, Git, Maven und Java eingesetzt.