

Prüfungskandidatin: Nicole Stadelmann

Fachvorgesetzter: Benjamin Stettler

Berufsbildner: Sandra Walzer

Erstexperte: Andreas Auer

Zweitexperte: Kurt Flury

Prototyp: Webanwendung für GP Stammdaten

Individuelle Praktische Arbeit



Startdatum: 03.04.2018

Enddatum: 18.04.2018

Inhaltsverzeichnis

[Teil 1 5](file:///C:\Users\Nici\Desktop\IPA\IPA%20Bericht.docx#_Toc511569268)

[1 Aufgabestellung 6](#_Toc511569269)

[1.1 Ausgangslage 6](#_Toc511569270)

[1.2 Detaillierte Aufgabenstellung 6](#_Toc511569271)

[1.2.1 Ursprung des Prototyps 6](#_Toc511569272)

[1.2.2 Systemarchitektur 6](#_Toc511569273)

[1.2.3 Musskriterien 7](#_Toc511569274)

[1.2.4 Technische Vorgaben 7](#_Toc511569275)

[1.2.5 Abgrenzungskriterien 8](#_Toc511569276)

[1.2.6 Zielgruppen 8](#_Toc511569277)

[1.2.7 Betriebsbedingungen 8](#_Toc511569278)

[1.2.8 Testumfang 11](#_Toc511569279)

[1.2.9 Dokumentation 11](#_Toc511569280)

[1.3 Mittel und Methoden 12](#_Toc511569281)

[1.4 Vorkenntnisse 12](#_Toc511569282)

[1.5 Vorarbeiten 12](#_Toc511569283)

[1.6 Neue Lerninhalte 12](#_Toc511569284)

[1.7 Arbeiten in den letzten 6 Monaten 13](#_Toc511569285)

[2 Projektorganisation 14](#_Toc511569286)

[2.1 Projektrollen 14](#_Toc511569287)

[2.2 Projektmethode 15](#_Toc511569288)

[2.2.1 Umsetzung der Projektmethode 16](#_Toc511569289)

[3 Zeitplan 17](#_Toc511569290)

[4 Arbeitsjournal 18](#_Toc511569291)

[5 Gesprächsprotokolle 19](#_Toc511569292)

[5.1 Vorlage Gesprächsprotokoll (z.b. für Zwischengespräch) 19](#_Toc511569293)

[Teil 2 20](file:///C:\Users\Nici\Desktop\IPA\IPA%20Bericht.docx#_Toc511569294)

[6 Kurzfassung des IPA Berichts (Management Summary) 21](#_Toc511569295)

[7 Vorbereitung 22](#_Toc511569296)

[7.1 Infrastruktur und Organisatorisches 22](#_Toc511569297)

[7.2 Risikomanagement 22](#_Toc511569298)

[7.3 Analyse des Projektauftrags 22](#_Toc511569299)

[8 Entwurf Webservice 23](#_Toc511569300)

[8.1 Überlegungen 23](#_Toc511569301)

[8.2 Benötigte Funktionsbausteine und Tabellen 23](#_Toc511569302)

[8.2.1 Funktionsbauseine 23](#_Toc511569303)

[8.2.2 SAP Tabellen 24](#_Toc511569304)

[8.3 Entwurf Webservice 1: Kunden suche 24](#_Toc511569305)

[8.4 Entwurf Webservice 2: Kundendaten auslesen 25](#_Toc511569306)

[8.5 Entwurf Webservice 3: QMC Produkte auslesen 25](#_Toc511569307)

[8.6 Entwurf Webservice 4: Kundendaten ändern 26](#_Toc511569308)

[8.7 Entscheidungen 27](#_Toc511569309)

[9 Realisierung Webservice 27](#_Toc511569310)

[9.1 Enterprise Services Builder 27](#_Toc511569311)

[9.1.1 Namespace erstellen 27](#_Toc511569312)

[9.1.2 Data Type erstellen 28](#_Toc511569313)

[9.1.3 Messagetype erstellen 28](#_Toc511569314)

[9.1.4 Service Interface erstellen 30](#_Toc511569315)

[9.1.5 Message Mapping 30](#_Toc511569316)

[9.1.6 Operation Mapping 31](#_Toc511569317)

[9.2 Enterprise Services Builder 31](#_Toc511569318)

[9.2.1 Communication Channel erstellen!!!!!!! 31](#_Toc511569319)

[9.2.2 Integrated Configuration erstellen!!!!!!!!!!!!!!!!!!! 31](#_Toc511569320)

[9.3 SProxy 31](#_Toc511569321)

[9.4 31](#_Toc511569322)

[10 Glossar 32](#_Toc511569323)

[11 Quellenverzeichnis 33](#_Toc511569324)

[12 Abbildungsverzeichnis 34](#_Toc511569325)

[Anhang 35](file:///C:\Users\Nici\Desktop\IPA\IPA%20Bericht.docx#_Toc511569326)

[Anhang 35](file:///C:\Users\Nici\Desktop\IPA\IPA%20Bericht.docx#_Toc511569327)



Teil 1

Obligatorische Kapitel

# Aufgabestellung

Nachfolgend wird die Aufgabenstellung im Originaltext aus PkOrg aufgeführt.

## Ausgangslage

SAP ist in unserem Unternehmen das führende Stammdatensystem. Wir benötigen für die Zukunft eine Möglichkeit auf die Daten ohne "SAP GUI" oder auch mit Mobilen Geräten zuzugreifen. Aus diesem Grund soll ein Prototyp erstellt werden, welcher eine Webseite als UI hat und via Webservice aus dem SAP die Daten darstellen und mutieren kann. Es gibt in SAP die Möglichkeiten Daten auf Web Technologie auszugeben, aber wie immer sind die Technische Voraussetzungen und der Implementierungsaufwand im SAP Umfeld recht hoch. Es wäre unteranderem manchmal auch sehr hilfreich, Daten möglichst schnell und ohne grossen Aufwand aus dem SAP ziehen zu können und diese an den Benutzern zu Verfügungen stellen zu können. Denkbar wäre diese Lösung auch als Entwicklungswerkzeug in einem POC (Proof of Concept) zur Visualisierung späterer Spezifikationen in einem Projekt.

## Detaillierte Aufgabenstellung

### Ursprung des Prototyps

Die WWZ arbeitet mit dem Partnerunternehmen Quickline zusammen. Über Quickline werden Telekomprodukte verkauft. Es gibt eine Schnittstelle zu Quickline, welche die Workflows der Bestellungen, Umzüge und Stammdatenmutationen des Kunden zur WWZ übermitteln. Der Fokus des Prototyps ist der Prozess der Stammdatenmutationen. Für den Fachbereich gibt es im SAP ein einfaches Cockpit. In diesem Cockpit kann der Workflow für einen Kunden ausgewählt werden. Durch diese Selektion werden zwei Aktionen ausgelöst.

1. In einem GUI Container im SAP GUI wird eine Quickline Webseite aufgerufen, welche die Stammdaten des Kunden anzeigt.

2. In einem anderen GUI Container im SAP GUI, wird der SAP WebGUI gestartet und die Transaktion BP (ändern des Geschäftspartners) aufgerufen.

Der Facharbeiter vergleicht die Stammdaten der beiden Systeme. Nach einer Validierung der Daten passt er im SAP WebGUI die Daten des Kunden an. Danach bestätigt der Facharbeiter den Workflow und der Workflow wird bei Quickline abgeschlossen.

Da der SAP WebGUI von SAP veraltet ist und letztens ein Problem mit dem Memory auf unserem SAP System ausgelöst hat, kam die initiale Idee zustande den SAP WebGUI durch eine normale Webseite abzulösen.

Was ist der SAP WebGUI: https://blogs.sap.com/2015/07/04/what-where-when-why-webgui/

Was ist die Transaktion BP: In der Transaktion BP werden die Stammdaten zu einem Geschäftspartner (Kunde) gepflegt.

Zu den Stammdaten gehören die Üblichen Daten wie die Kontaktdaten der Person und deren Anschrift. Die Transaktion ist das Gengenstück zu XD01 bei dem Modul SD für den Kunden, aber speziell für die Versorgungsindustrie.

### Systemarchitektur

Die WWZ hat grundsätzlich im SAP Umfeld eine 3tier Architektur im Einsatz.

Es gibt 3 SAP ERP Systeme:

T01 - Customizing/Entwicklung

Q01 - Test/Stagesystem

P01 - Produktivsystem

Bei den SAP PI Systemen, haben wir nur eine 2tier Architektur:

SAP ERP Q01 -> SAP PI PT1

SAP ERP P01 -> SAP PI PP1

Aus diesem Grund müssen die Entwicklungen die im T01 gemacht werden und eine PI Schnittstelle benötigen, zwingend in das SAP Q01 mit dem SAP SolMan transportiert werden. Der SAP SolMan verwaltet die Änderungen und ist an das Transportmanagement angehängt ist.

Als Webserver für den Prototyp ist ein XAMPP lokal auf dem entwicklungs PC mit Windows 7 installiert. Bei einer eventueller Produktivsetzung nach der IPA wird die Webanwendung auf einem Server innerhalb der DMZ in der WWZ kopiert.

Die komplette Systemarchitektur welche an der IPA benötigt wird, ist bereits vollständig aufgebaut. An den Systemen ist keine Konfiguration mehr notwendig. Ein User für den Zugriff der Webanwendung auf das PI ist angelegt und geprüft.

### Musskriterien

Es ist eine Webanwendung zu realisieren welche Stammdaten vom Geschäftspartner, die im SAP ERP abgelegt sind, ausgibt. Neben der Anzeige sollen auch einige Daten geändert werden können, welche dann im SAP wieder verbucht werden. Die Schnittstelle zum Auslesen der Daten muss über das SAP PI erfolgen. Primäres Ziel dieses Prototypens ist es, die Machbarkeit und Schwierigkeiten einer Webanbindung an das SAP aufzuzeigen. Die Anwendung soll ohne spezielle Konfiguration auf dem Webserver und ohne zusätzlichem Modulen von XAMPP Webserver laufen.

Es soll eine Grundlage/Vorlage geschaffen werden, welche für schnelle Implementation z.b an einem POC hilft für einen Webservice ein GUI zu erstellen. Über dieses GUI kann der Webservice getestet werden und z.b dem Fachbereich oder externen Partner die Funktionen visualisiert werden. Denkbar sind auch kleinere/einfachere Anwendungen wie bei diesem Prototyp z.b Mutationen von Stammdaten.

### Technische Vorgaben

Webanwendung:

- Die Webanwendung soll auf einem Webserver: XAMPP auf WIN7 64Bit laufen.

- Die Daten sollen zwischen der Webanwendung und dem SAP PI über eine SOAP Schnittstelle kommunizieren. Innerhalb des SOAP Request werden die Daten im XML Format abgebildet (SAP PI Standard bei einem SOAP Adapter).

- Als Programmiersprache für die Webanwendung soll JavaScript und HTML verwendet werden.

- Zusätzliche benötigte Bibliotheken für die Webanwendung können frei gewählt werden.

SAP PI:

- Das Design der Schnittstellen (Webservice) muss im "Enterprice Service Builder" erstellt werden

- Die Konfiguration der Schnittstellen muss im "Integration Builder" gemacht werden

- Communications Channel soll vom Typ SOAP sein

- Die Kommunikation soll über HTTP erfolgen (HTTPS ist "out of scope")

SAP ERP:

- Das Coding für die Datenbeschaffung soll im generierten Proxy des Webservice erstellt werden

- Für die Datenbeschaffung sind SAP Funktionsbausteine oder Klassen zu verwenden, ausser es sind keine vorhanden oder deren Funktionalität recht nicht aus.

### Abgrenzungskriterien

Auf die Implementation der Sicherheit der Applikation wird bei dieser Umsetzung bewusst verzichtet, da dies der Rahmen der Umsetzungszeit sprengen würde. Die Angaben der Authentifizierung welche über Basic Authentication gemacht werden soll, darf fix im Code hinterlegt werden.

### Zielgruppen

Der Prototyp dieser Anwendung ist primär für Entwickler und IT Fachleute gedacht.

### Betriebsbedingungen

Die Web Anwendung soll im Internet Explorer Browser 11 auf einem Desktop oder Notebook laufen. Es sollen keine speziellen Voraussetzungen oder Installationen auf einem Web Server nötig sein.

Spezifikation der Webanwendung

Das GUI der Webanwendung soll die Zugehörigkeit zur WWZ Gruppe wiederspiegeln. Ein WWZ Logo und das WWZ Blau sollen Bestandteil des Designs sein. Als Vorlage dient die Webseite der WWZ. Ein spezifischer Style Guide wird nicht vorgegeben und auch nicht erwartet. Der Fokus des Prototyp liegt auf den Funktionen und dem Backendsystem. Die Bedienelemente und der Look and Feel sind der heutigen Zeit angemessen auszulegen. Bootstrap kann verwendet werden, ist aber kein Musskriterium. Rückmeldungen und Fehlerbehandlung nach dem Aufruf eines Service soll dem Benutzer als Feedback angezeigt werden.

Folgende Funktionen sollen in der Anwendung implementiert werden:

Suche von Geschäftspartner nach:

- GP Nummer

- Name

- Vorname

- Firmen-/Gruppenname

- Strasse

-Hausnummer

- Ort

- PLZ

**Funktion:**

Die Werte für die Suche sollen in einzelnen Inputfeldern eingegeben werden können.

Wenn die Suche ausgeführt wird, sollen maximal die ersten 50 Einträge zurückgegeben werden. Zusätzlich soll dem Benutzer eine Meldung angezeigt werden, dass es mehr Suchtreffer gibt als angezeigt werden. Die Suchergebnisse sollen in einer Liste angezeigt werden. Ein Treffer auf der Suchliste soll ausgewählt werden können, damit die Stammdaten des Kunden geladen und angezeigt werden.

**Validierung:**

Mindesten eines dieser Felder muss einen Wert enthalten bevor die Suche ausgeführt wird.

Wenn die GP Nummer ausgewählt wird, darf diese nur Numerische werte enthalten und 7 Zeichen länge haben.

Anzeigen von Geschäftspartner:

* Kundedaten anzeigen (Standartadresse)

- Anrede

- Name 1

- Name 2

- Vorname

-Nachname

- Ort

- Strasse

- PLZ

- Land

- Region

* Kommunikation

- Telefon

- Mobiletelefon

- E-Mail

• Quickline Produkte anzeigen

○ QMC-ID

- Internet Dienstleistungsprodukt

- Fix-Telefon

- DTV

- Verte!

- QLTV

- Mobile Dienstleistungsprodukt

- Kombi Dienstleistungsprodukt

**Funktion:**

Nachdem ein Treffer in der Suche angewählt wurde, werden die Stammdaten und die Produkte von Quickline des Kunden angezeigt. Die Daten sollen auf eine Übersichtliche weise angezeigt werden. Die Stammdaten und Produktdaten sollen Gruppiert sein.

Geschäftspartner ändern:

• Kundedaten ändern

○ Standartadresse

- Anrede

- Name 1

- Name 2

- Vorname

- Nachname

○ Kommunikation

- Telefon

- Mobiletelefon

- E-Mail

Bevor die Kundendaten geändert werden können, muss der Benutzer ein Änderungsmodus aktivieren. Der Modus soll Fehlmanipulationen an den Stammdaten verhindern. Vor dem Speichern soll eine einfache Validierung durchgeführt werden. Nach dem Speichern soll dem Benutzer eine Statusmeldung angezeigt werden. Wenn das Speichern erfolgreich war, soll der Änderungsmodus verlassen werden.

**Validierung Standartadresse bei:**

- Person: Anrede / Nachname / Vorname müssen gefüllt sein.

- Gruppe: Name 1 / Name 2 müssen gefüllt sein.

- Organisation: Name muss gefüllt sein.

**Validierung Kommunikation bei:**

- Telefon darf nur Nummern enthalten.

- Mobiletelefon darf nur Nummern enthalten.

- E-Mail muss ein @ enthalten.

### Testumfang

Für jeden erstellten Webservice im SAP PI soll ein Funktionstest mit dem SoapUI Tool gemacht werden. Mithilfe des WSDL vom SAP PI ist ein Projekt anzulegen. Mit Hilfe des Request soll die Funktionsweise des Webservice geprüft werden. Funktioniert der Aufruf über das PI bis ins ERP? Werden die Stammdaten korrekt ausgelesen oder geändert? Im Testprotokoll soll der Request und der Respons aufgeführt werden. Pro Service sollen mindestens 3 Testfälle durchgeführt werden.

Weiter soll ein Funktionstest der gesamten Anwendung gemacht werden. Besondere Aufmerksamkeit sollte der Mutation im SAP Backend geschenkt werden, es darf nicht zu Verbuchungsabbrüchen kommen! Der Funktionstest soll vom Umfang her pro Geschäftspartner Ausprägung (Person, Gruppe und Organisation) mindestens 2 Testfälle beinhalten. Der Funktionstest soll die Suche, Anzeige und Änderung beinhalten. Im Testprotokoll sollen die Angaben der Maske in der Webanwendung und die Stammdaten vom SAP ERP ersichtlich sein.

### Dokumentation

Da es sich hier primär um eine Umsetzung eines Prototypen handelt, ist der Fokus auf der technischen Dokumentation. Es wird eine kurze (max. 5 Seiten) technische Dokumentation erwartet, welche aufzeigt welche Schritte gemacht wurden um das Ziel zu erreichen. Zusätzlich sollte ein Diagramm für den Datenfluss und ein kurzer Abschnitt über das Fazit der Umsetzung enthalten sein. Auf alle anderen Dokumentationen wird verzichtet. Wenn Bedarf besteht, kann auf die IPA Dokumentation zurückgegriffen werden.

## Mittel und Methoden

Bei der Programmierung im SAP Umfeld ist die Namenskonvention einzuhalten. Im SAP ERP sollen Funktionsbausteine oder Klassen von SAP verwendet um Daten zu lesen und zu verbuchen. Wenn möglich soll der direkte Zugriff auf Tabellen vermieden werden. Die Webanwendung wird bis ins SAP Stage System Q01 transportiert und dort getestet und abgenommen.

Folgende Mittel stehen für die Umsetzung zu Verfügung:

- WWZ Windows 7 Standard PC mit Zugang auf das Internet.

- MS Office Packet

- SoapUI, Webservice Testtool

- Snagit, für Printscreens

- Notepad++, Texteditor mit XML Unterstützung

- Internet Explorer 11

- XAMPP, als Webserver

- atom.io, für die Erstellung der Webanwendung

- MockFlow, für das Erstellen der Mockup's

- HTML und JavaScript, für die Realisierung der Webanwendung

- SAP PI Weboberfläche, für das Anlegen der PI Schnittstellen

- SAP ERP ABAP, für das Programmieren in Proxyklassen

## Vorkenntnisse

Nicole hat Erfahrungen im Bereich der Programmierung im SAP sei ca. 1,5 Jahren. Jedoch hatte sie vom Letzten Herbst bis Februar dieses Jahrs ein Projekt geleitet ohne regelmässige Programmierungen. Im Bereich Webentwicklung bringt sie vor allem die Erfahrungen aus der Schule mit. In der WWZ hat sie im Bereich Webentwicklung nur 2-3 kleinere Aufgaben gemacht.

## Vorarbeiten

Es wurde ein kleiner POC mithilfe einer vorhandenen Webseite gemacht, um zu schauen ob für Nicole eine solche IPA umsetzbar ist.

## Neue Lerninhalte

Alle Systeme/Funktionen und Methoden zur Umsetzung der Anwendung sind bekannt. Funktionsbausteine für das Lesen und Mutieren der Daten sind evtl. noch nicht bekannt oder es müssen Alternativen gewählt werden.

SAP PI und das Anlegen von Webservice ist für Nicole recht neu. Die Behandlung der Datenstrukturen aus dem SAOP Request können evtl. für Schwierigkeiten sorgen.

## Arbeiten in den letzten 6 Monaten

Oktober bis Januar 2017 hat Nicole vor allem an dem Relaunch unserer Intranet Seite gearbeitet. Bei diesem Projekt hat sie die Teilprojektleitung und Koordination mit externen Partner gemacht, Programmierung von ihrer Seite war keine involviert. Februar bis März hat Nicole im SAP Umfeld Übungen, kleinere Programmierungen und Anpassungen an Programmen gemacht. Ebenfalls hat sie Erfahrungen mit Webservice im Zusammenhang mit dem SAP PI gesammelt.

# Projektorganisation

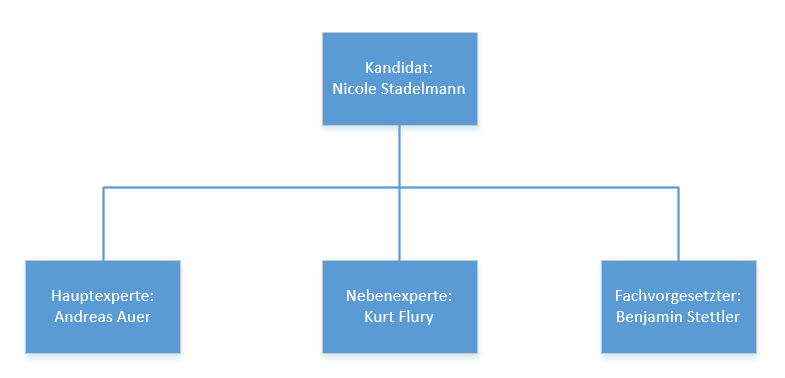


Abbildung 1: Projektorganisation

## Projektrollen

|  |  |
| --- | --- |
| Rolle | Tätigkeit |
| Kandidat | Führt die IPA anhand der Vorgaben der Detailbeschreibung durch. Wird nach der Ausführung anhand der Beurteilungskriterien bewertet. |
| Hauptexperte | Der Hauptexperte Validiert und kontrolliert die Kriterien der IPA. Desweitern besucht er den Kandidaten während der IPA in der Firma. Er bewertet die IPA anhand der Beurteilungskriterien und führt das Fachgespräch. |
| Nebenexperte | Der Nebenexperte unterstütz die Hauptexperte. Er nimmt ebenfalls an der Präsentation und dem Fachgespräch teil. |
| Fachvorgesetzter | Ist während der IPA Zeit anwesend und begleitet den Kandidaten in dieser Zeit. Es beurteilt das Arbeitsverhalten und bewertet am Ende ebenfalls die Arbeit des Kandidaten. |

## Projektmethode

Für die Umsetzung des Projektes wurde die Projektmanagementmethode IPERKA gewählt. IPERKA besteht aus 6 Phasen: Informieren, Planen, Entscheiden, Realisieren, Kontrollieren und Analysieren.

I Informieren

In der ersten Phase informieren sich die entsprechen Personen über den Projektauftrag. Anfällige Fragen können bereits zu Beginn auftauchen, und geklärt werden. Zudem wird dadurch klar, was als Ergebnis erwartet wird.

P Planen

In der zweiten Phase wird beschrieben wie man bei der Arbeit vorgehen möchte, und welche Optionen es gibt. Es wird ein Zeitplan mit den verschiedenen Teilschritten/ Aufgaben erstellt und deren Bearbeitungsdauer geschätzt. Des Weiteren werden verschiedene Lösungsmöglichkeiten geprüft.

E Entscheiden

In der dritten Phase muss das weitere Vorgehen bestimmt werden. Es wird beschrieben welche Lösungsvarianten gewählt wurden und warum.

R Realisieren

In dieser Phase werden die geplanten Arbeitsschritte umgesetzte. Der Zeitplan sollte hierbei so gut wie möglich eingehalten werden. Die effektive Arbeitszeit für die einzelnen Arbeitsschritte müssen festgehalten werden und im Arbeitsjournal notiert werden.

K Kontrollieren

In dieser Phase sollen die erhaltenen Ergebnisse getestet werden. Durch die Kontrolle wird überprüft, ob die umgesetzte Arbeit den Anforderungen entsprechen. Die Resultate der Tests sind schriftlich festzuhalten. Missstände müssen klar aufgezeigt werden.

A Auswerten

In der letzten Phase wird die gesamte Arbeit ausgewertet. Es wird festgehalten, ob alle Ziele und Anforderungen erfüllt wurden. Ebenfalls eine Selbsteinschätzung und Reflexion wird in dieser Phase erfasst.

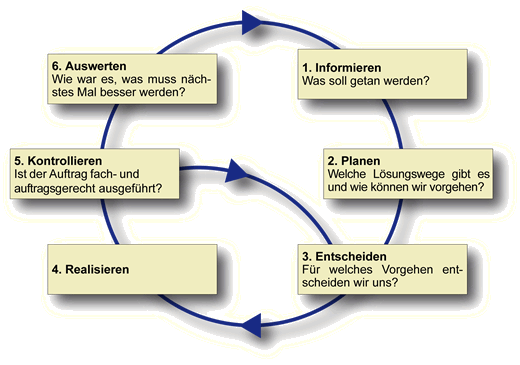


Abbildung 2: Das IPERKA Modell

### Umsetzung der Projektmethode

Für dieses Projekt wurde entschieden, die Projektmethode wie üblich auf das gesamte Projekt anzuwenden. Zusätzlich werden aber auch auf einzelne Arbeitspakete die Methode angewendet. So werden für einzelne Arbeitspakete Informationen gesammelt, geplant, Entscheidungen getroffen, realisiert und Kontrolliert.

Dadurch kann eine bessere Planung der einzelnen Arbeitspakete gewährleistet werden. Zusätzlich beeinflussen so Entscheidungen oder mögliche Änderungen in der Umsetzung die nachfolgenden Arbeitspakte nicht/weniger, da diese erst später geplant werden. Somit kann die Planung auf dem abgeschlossen vorgängigen Arbeitspaket aufgebaut werden.

Folgende Arbeitspakete beinhaltet das Projekt:

* Umsetzung des Webservice
* Umsetzung der Webanwendung

# Zeitplan

# Arbeitsjournal

**Legende:**

|  |  |
| --- | --- |
|  | An Arbeitspakt gearbeitet und beendet |
|  | Nicht an diesem Arbeitspaket gearbeitet |
|  | Arbeit an nicht geplantem Arbeitspaket |

|  |  |  |  |
| --- | --- | --- | --- |
| **Dienstag, 03. April 2018** | **Soll** | **Puffer** | **Ist** |
| 1. **Projektauftrag analysieren**   Mit der Analyse des Projektauftrags und der Beurteilungen konnte ich mir einen Überblick über die verlangten aufgaben machen und mir die ersten Gedanken zu der Umsetzung des Projektes machen. | **1** |  | **1** |
| 1. **Fragen & Abklärungen**   Kurze Beschreibung | **0.5** |  | **0.5** |
| 1. **Zeitplan erstellen**   Kurze Beschreibung   1. **Vorlage Dokumentation erstellen**   Ich habe heute die Dokumentationsvorlage erstellt. Ich habe dies bewusst heute eingeplant, da ich   1. **IPA Bericht teil 1 schrieben**   Später schrieb ich den 1. Teil des IPAs Bericht. Ich hatte dafür nicht so lange wie gedacht, da ich die Detail Beschreibung 1 zu 1 übernehmen konnte. Auch der Zeitplan war bereits erstellt und konnte nur noch eingefügt werden. Die meiste Zeit benötiget ich für dich Erstellung des Diagramms der Projektorganisation und die Rollenbeschreibungen.   1. **Zwischengespräch**   Kurze Beschreibung   1. **Arbeitsjournal** | **2**  1  2  0.5  0.5 | 0.5 | **3**  1.5  1  0.5  0.5 |
| **Total:** | 7.5 | 0.5 | 8 |

Der Start heute was leider etwas stressig. Am Morgen, bevor ich mit der Arbeit beginnen konnte, erfuhr ich dass es Probleme mit dem Intranet gab. Danach konnte ich endlich mit meiner IPA starten.

# Gesprächsprotokolle

## Vorlage Gesprächsprotokoll (z.b. für Zwischengespräch)

|  |  |  |  |
| --- | --- | --- | --- |
| Gesprächsprotokoll xyz | | | |
| Datum | xx.xx.xxxx | Teilnehmer: | * Xyz * Xyz * xyz |
| Zeit | xx.xx Uhr bis xx.xx Uhr |
| Themen: | * xyz * xyl |

Gesprächsnotizen:



Teil 2

Projektdokumentation

# Kurzfassung des IPA Berichts (Management Summary)

# Vorbereitung

## Infrastruktur und Organisatorisches

Für die IPA wurde ein Arbeitsplatz in einem separaten Raum eingerichtet, damit keine Störungen von Telefonaten, Gesprächen oder anderen Geräuschen verursacht wird.

Auf GitHub wurde ein Account eingerichtet, mit zwei repositories.

1. Repositorie :

Im ersten repositorie werden alle benötigten Dokumente, wie der IPA-Bricht, Testfälle, Zeitplan und notizen abgelegt. Über die GitHub Desktop Anwendung wird das Repositorie auf das persönliche Home-Verzeichnis geklont.

1. Repositorie :

Im zweiten repositorie wird der Projektordner mit je einem Ordner für JS und CSS Files, Bilder und Fonts erstellt. Dieses repositorie wird in den htdocs Ordner von Xampp geklont damit sie lokal über einen Webserver aufgerufen werden können.

Über die Gitib Desktopn Anwendung kann ich beide Repositories einfach auf die GitHub hochladen.

## Risikomanagement

Um einen Datenverlust zu verhindern werden folgende Vorkehrungen getroffen:

* Mindestens 2x Tag werden alle Dokumente und Files auf GitHub hochgeladen. Zusätzlich kann durch das Hochladen auf GitHub das Projekt Versioniert werden und alte Versionen von Files und Dokumenten wiederhergestellt werden.
* Den XAMPP Ordner sowie alle IPA Dokumente und Files werden zusätzlich mindesten 2x am Tag auf dem persönlichem Home Verzeichnis gesichert. Von diesem Verzeichnis erstellt die WWZ standartmässig jede Stunde ein Backup.
* SAP?

## Analyse des Projektauftrags

Phase: Informieren

Als erster Arbeitsschritt wurde der Projektauftrag und die Beurteilungskriterien analysiert. Anhand der erhalten Informationen wurden die einzelnen Arbeitsschritte erstellt, auf welchen der Zeitplan aufgebaut wurde.

Der Projektauftrag befindet sich in diesem Dokument unter Punkt 1 und der Zeitplan unter Punkt 3.

# Entwurf Webservice

## Überlegungen

Im SAP werden die Geschäftspartner in verschiedenen GP-Typen unterteilt. Zum erhalten der

Auf der Webseite soll schlussendlich nur die Daten von einem GP angezeigt werden. Um trotzdem eine Suchfunktion zu ermöglichen, welche mehrere Treffer zurückgeben kann, muss je ein Webservice für die GP suche und einen für das erhalten der GP Informationen erstellt werden.

Es sollen zwei verschiedene Arten von GP Informationen angezeigt werden. Die Persönliche Informationen und die QMC Produkte. Um dies zu verwirklichen könnte ein Web Service erstellt werden, welcher beide Informationen auf einmal ausliest oder zwei Web Service, welche die beiden Informationsarten getrennt beschafft.

Um die Informationen eines GPs zu ändern muss ein weiterer Web Service erstellt werden, welcher die neuen daten dem SAP ERP System übergibt und deren Speicherung auslöst.

Für die Verwirklichung aller Funktionen müssen daher 3- 4 Webservice erstellt werden.

Um beim erstellen der Webservice keine wichtigen Elemente bei den Request und Response Nachrichten zu vergessen, wird für jeden Webservice einen Entwurf erstellt. Die Entwürfe sollen Name und Funktion des Webservice beschriebt und die Struktur der Request und Response Nachrichten aufzeigen. Zusätzlich soll die Struktur die enthaltenen Elemente, ihre Kardinalität und Beziehung zu den anderen Elementen aufzeigen.

Des Weiteren soll jede Response Nachricht eine Message Element enthalten. Dies soll ermöglichen, Fehlermeldungen welche während dem ausführen des Web Service aufgetaucht sind, dem Anwender in der Webanwendung auszugeben.

## Benötigte Funktionsbausteine und Tabellen

Nachfolgend werden die für die Programmierung der Webservice benötigte Tabelle und Funktionsbausteine aufgeführt und ihre Funktionen und Informationen beschrieben.

### Funktionsbauseine

Folgende Funktionsbausteine werden für das Programmieren der Web Service im SAP ERP verwendet:

* BAPI\_BUPA\_SEARCH\_2
* BAPI\_BUPA\_CENTRAL\_GETDETAIL
* BAPI\_BUPA\_ADRESS\_GETDETAIL
* BAPI\_BUPA\_CENTRAL\_CHANGE

BAPI\_BUPA\_SEARCH\_2

Durch diesen Funktionsbaustein kann nach GPs gesucht werden. Als Suchkriterien können zentrale Geschäftspartnerdaten und Adressattribute angegeben werden.

BAPI\_BUPA\_CENTRAL\_GETDETAIL

Dieser Funktionsbaustein gibt zentrale Geschäftspartnerdaten, wie Personen-, Firmen- oder Gruppenname, GP-Typ, Anrede usw., zu einem GP zurück.

BAPI\_BUPA\_ADRESS\_GETDETAIL

Dieser Funktionsbaustein gibt die Adressdaten eines GPs zurück.

BAPI\_BUPA\_CENTRAL\_CHANGE

Mit diesem Funktionsbaustein können zentrale Geschäftspartnerdate, wie Personen-, Firmen- oder Gruppenname, Telefonnummern oder Mailadressen, geändert werden.

### SAP Tabellen

Folgende SAP Tabellen werden für das Programmieren der Web Service im SAP ERP verwendet:

* ZBC\_QMC\_CRM\_DATA
* BUT000

ZBC\_QMC\_CRM\_DATA

In dieser Tabelle befinden sich alle QMC IDs und Informationen zu deren QMC Produkte.

BUT000

In dieser Tabelle befinden sich alle GPs mit zentralen Informationen.

## Entwurf Webservice 1: Kunden suche

**Name:** BP\_Search

**Funktion:** Durch diesen Web Service soll der Anwender über verschiedene Angaben einen GP suchen können. Zurückgeben soll der Web Service eine Liste aller Treffer, auf welche die vom Anwendern angegeben Angaben zutreffen.

**Request Felder**

|  |  |  |  |
| --- | --- | --- | --- |
| Feld Name | Feld Wert | Kardinalität | Kind von: |
| GP\_No | GP Nummer | 0…1 |  |
| Lastname | Nachname | 0...1 |  |
| Firstname | Vorname | 0…1 |  |
| Org\_Name | Firma Name | 0…1 |  |
| Group\_Name | Gruppen Name | 0…1 |  |
| GP\_Type | Geschäftspartner Type | 0…1 |  |
| Street | Strasse | 0…1 |  |
| House\_No | Hausnummer | 0…1 |  |
| PLZ | PLZ | 0…1 |  |
| City | Stadt | 0…1 |  |

Response Felder

|  |  |  |  |
| --- | --- | --- | --- |
| Feld Name | Feld Wert | Kardinalität | Kind von: |
| GP\_No | GP Nummer | 0…1 |  |
| Name\_2 | Name 2 (Bei Privatpersonen Vorname) | 0…1 |  |
| Name\_1 | Name 1 | 1 |  |
| GP\_Type | Geschäftspartner Type | 1 |  |
| Message | Nachricht | 0…1 |  |
| Message\_Text | Nachrichten Text | 1 | Message |
| Message\_Type | Nachrichten\_Type | 1 | Message |

## Entwurf Webservice 2: Kundendaten auslesen

**Name:** BP\_Details

**Funktion:** Der Webservice soll ausgeführt werden, nachdem sich der Anwender einen GP, aus der vom Webservice 1 zurückgegebenen liste, ausgesucht hat.

Der Webservice soll alle relevanten GP daten des ausgewählten GPs suchen und zurückgeben.

**Request Felder**

|  |  |  |  |
| --- | --- | --- | --- |
| Feld Name | Feld Wert | Kardinalität | Kind von: |
| GP\_No | GP Nummer | 1 |  |

**Response Felder**

|  |  |  |  |
| --- | --- | --- | --- |
| Feld Name | Feld Wert | Kardinalität | Kind von: |
| GP\_Type | Geschäftspartner Type | 1 |  |
| Title | Anrede | 0…1 |  |
| Lastname | Name | 0…1 |  |
| Firstname | Vorname | 0…1 |  |
| Name\_1 | Name 1 | 0…1 |  |
| Name\_2 | Name 2 | 0…1 |  |
| City | Stadt | 0…1 |  |
| Street | Strasse | 0…1 |  |
| PLZ | PLZ | 0…1 |  |
| Country | Land | 0…1 |  |
| Region | Region | 0…1 |  |
| Telephone\_List | Telefon Nummern Liste | 0..1 |  |
| * Telephone\_Detail |  | 0…unbound | Telephone\_List |
| * Telephone\_NO | Telefon Nummern | 1 | Telephone |
| * Telephone\_ID | Telefon ID | 1 | Telephone |
| E\_Mail\_List | E-Mail Adressen Liste | 0..1 |  |
| * E\_Mail\_Detail |  | 0…unbound | E\_Mail\_List |
| * E\_Mail\_Adress | E-Mail Adresse | 1 | E\_Mail |
| * E\_Mail\_ID | E-Mail ID | 1 | E\_Mail |
| Message | Nachricht | 0…1 |  |
| Message\_Text | Nachrichten Text | 1 | Message |
| Message\_Type | Nachrichten\_Type | 1 | Message |

## Entwurf Webservice 3: QMC Produkte auslesen

**Name:** BP\_QMC\_Products

**Funktion:** Der Webservice soll ausgeführt werden, nachdem sich der Anwender einen GP, aus der vom Webservice 1 zurückgegebenen liste, ausgesucht hat.

Der Webservice soll alle relevanten QMC Produkte des GP zurückgeben.

**Request Felder**

|  |  |  |  |
| --- | --- | --- | --- |
| Feld Name | Feld Wert | Kardinalität | Kind von: |
| GP\_No | GP Nummer | 1 |  |

**Response Felder**

|  |  |  |  |
| --- | --- | --- | --- |
| Feld Name | Feld Wert | Kardinalität | Kind von: |
| QMC\_ID | QMC ID | 0…1 |  |
| Internet\_product | Internet Dienstleistungsprodukt | 0…1 |  |
| Telephone\_product | Telefon Dienstleistungsprodukt | 0…1 |  |
| DTV | DTV | 0…1 |  |
| Verte | Verte! | 0…1 |  |
| QL\_TV | QL TV | 0…1 |  |
| Mobile\_product | Mobile Dienstleistungsprodukt | 0…1 |  |
| Combi\_product | Kombi Dienstleistungsprodukt | 0…1 |  |
| Message | Nachricht | 0…1 |  |
| Message\_Text | Nachrichten Text | 1 | Message |
| Message\_Type | Nachrichten\_Type | 1 | Message |

## Entwurf Webservice 4: Kundendaten ändern

**Name:** BP\_Update

**Funktion:** Wen der Anwender im Änderungsmodus eine Änderung vornimmt und speichert soll der Web Service ausgeführt werden und die entsprechenden Daten richtig im SAP ERP ändern.

**Request Felder**

|  |  |  |  |
| --- | --- | --- | --- |
| Feld Name | Feld Wert | Kardinalität | Kind von: |
| Title | Anrede | 0…1 |  |
| Name\_1 | Name 1 | 0..1 |  |
| Name\_2 | Name 2 | 0..1 |  |
| Firstname | Vorname | 0..1 |  |
| Lastname | Nachname | 0..1 |  |
| Tel\_Change\_Llist | Telefon Nummer Liste | 0..1 |  |
| * Tel\_Change\_Detail |  | 0…unbound | Tel\_Change\_list |
| * Tel\_Number | Telefon Nummer | 1 | Tel\_Change\_Number |
| * Tel\_ID | Nummer ID | 1 | Tel\_Change\_Number |
| E\_Mail\_Change\_List | E\_Mail Liste | 0..1 |  |
| * Mail\_Change\_Detail |  | 0…unbound | E\_Mail\_Change\_List |
| * E\_Mail\_Adress | E-Mail Adresse | 1 | Mail\_Change\_Adress |
| * E-Mail\_ID | E-Mail ID | 1 | Mail\_Change\_Adress |

**Response Felder**

|  |  |  |  |
| --- | --- | --- | --- |
| Feld Name | Feld Wert | Kardinalität | Kind von: |
| Message | Nachricht | 1 |  |
| Message\_Type | Nachrichten\_Type | 1 |  |

## Entscheidungen

Wie unter Punkt 8.1 beschrieben, wäre es möglich die GP Informationen und QMC Informationen in einem Webservice auszulesen, oder auf zwei Webservice aufzuteilen.

Zwei getrennte Webservice bieten in der Zukunft bessere Weiterentwicklungsmöglichkeiten im Bereich der Webanwendung. Des Weiteren können bei getrennten Webservice immerhin eine Informationsgruppe angezeigt werden, sofern ein Webservice erfolgreich war.

# Realisierung Webservice

Die WWZ setzt das 3 Model ein. Das heisst, ein Service ist in 3 Teile aufgeteilt:

- Sendendes System (Web / Anfragesteller)

- SAP PI

- Empfangendes System (SAP / Antworter)

Aus diesem Grund müssen fast alle Komponenten 3mal erstellt werden.

## Enterprise Services Builder

### Namespace erstellen

Zu Beginn muss im Enterprise Services Builder in jedem ………….. ein Namespace erstellt werden. Folgende Namespaces wurden für das gesamte Projekt erstellt:

Web

* http://wwz.ch/S/99980/BP\_Search
* http://wwz.ch/S/99981/BP\_GetDetails
* http://wwz.ch/S/99982/BP\_QMCProducts
* http://wwz.ch/S/99983/BP\_Update

ERP

* http://wwz.ch/R/99980/BP\_Search
* http://wwz.ch/R/99981/BP\_GetDetails
* http://wwz.ch/R/99982/BP\_QMCProducts
* http://wwz.ch/R/99983/BP\_Update

PI

* http://wwz.ch/X/99980/BP\_Search
* http://wwz.ch/X/99981/BP\_GetDetails
* http://wwz.ch/X/99982/BP\_QMCProducts
* http://wwz.ch/X/99983/BP\_Update

### Data Type erstellen

Die Datentypen definieren die Struktur und Elemente für die Request oder Response Nachrichten inklusive deren Datentypen und Kardinalitäten. Als Datentyp wird in der WWZ standartmässig den xsd:string verwendet.

Die Datentypen werden anhand der Entwürfe erstellt und müssen jeweils im Web und im ERP …. Erstellt werden.

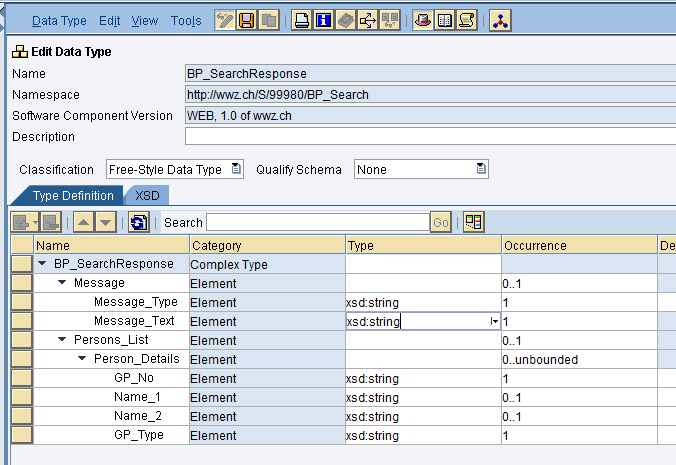


Abbildung 3: Erstellter Datentyp für BP\_Search Response

### Messagetype erstellen

Der Message Type definiert den Aufbau der XML Nachricht für die Request und Response Anfragen. Ihm wird ein Data Type zugewiesen, welcher die Struktur der XML Nachricht beschreibt. Die Struktu wird automatisch erstellt, sobald ein Data Type zugewiesen wurde.

Des Message Type muss ebenfalls immer für den Request und die Response angelegt werden. Dies jeweils im Web und ERP … .

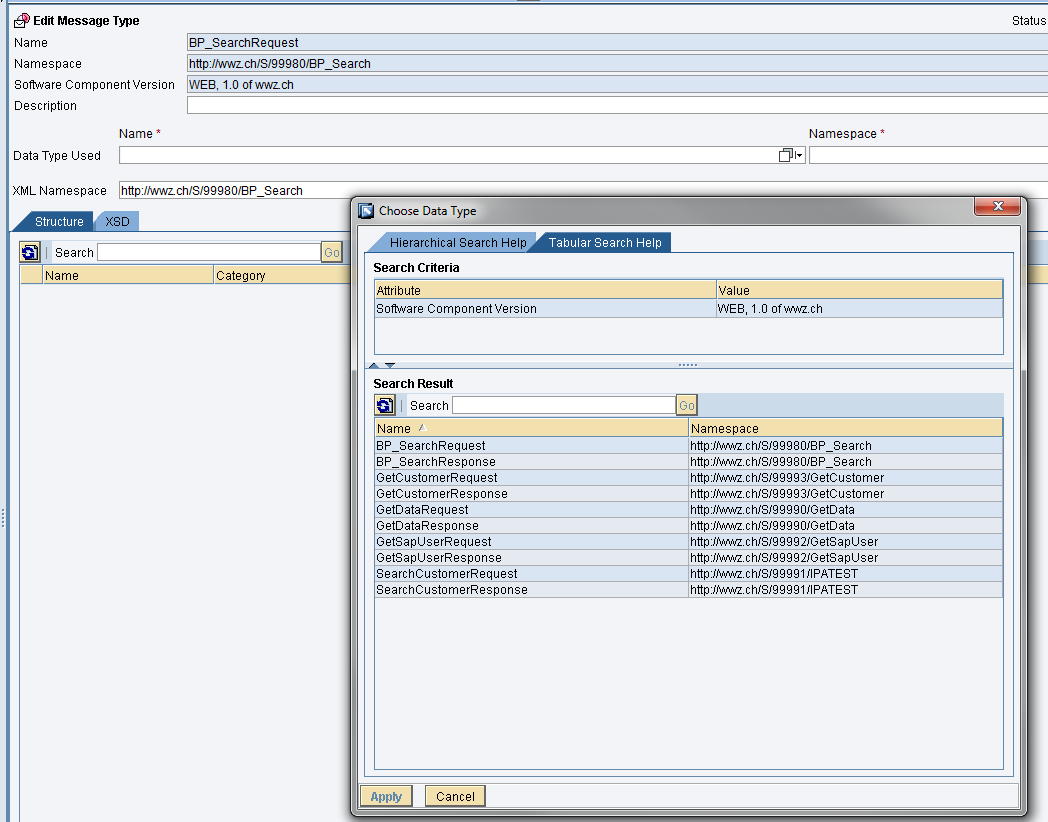


Abbildung 4: Auswählen des Data Types für den Request Message Type von BP\_Search

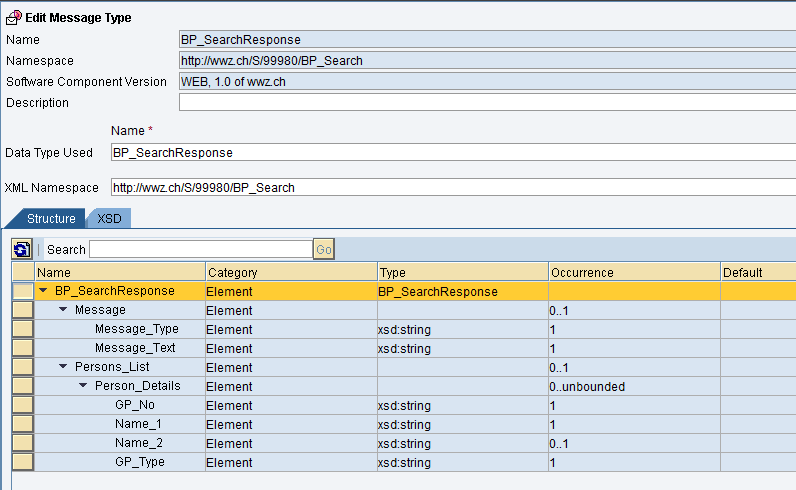


Abbildung 5: Der fertige Message Type mit der automatisch erstellen Struktur

### Service Interface erstellen

Dem Service muss eine Kategorie zugewiesen werden. Der Service Interface im ERP ….. bekommt die Kategorie Inbound. Damit der Service, mit der Inbound Kategorie, aufgerufen werden kann, benötig der Service einen passenden Outbound Service. Dieser wird im Web…. Erstellt.

Zusätzlich muss dem Service Interface jeweils die erstellten Message Types für die Request beziehungsweise für die Responsenachricht übergeben werden.

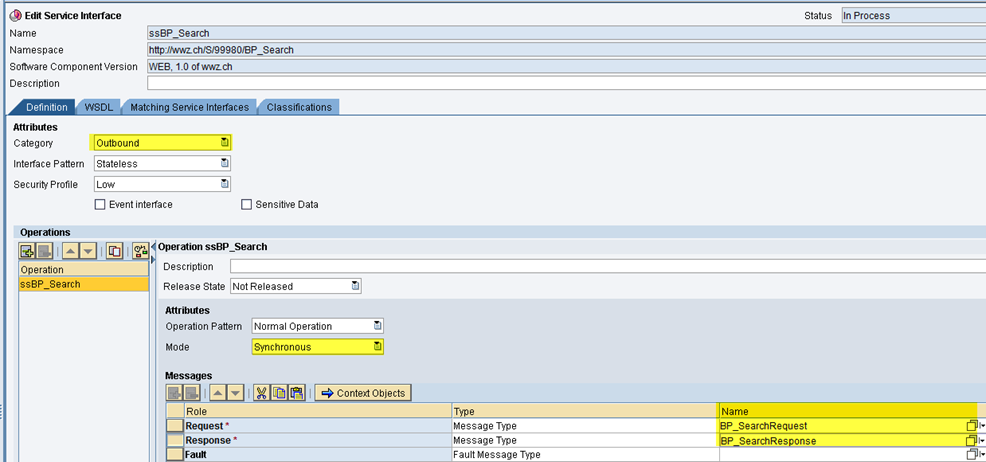


Abbildung 6:Das erstellte Service Interface mit der Outbound Kategorie und den zugewiesenen Message Types

### Message Mapping

Nachdem im WEB und ERP alle Komponenten erstellt wurden, muss im PI ….. die Messages Types miteinander verbunden werden.

Hierbei wurde der Message Type für den Request im Web mit dem Message Type für den Request aus dem Erp … miteinander verbunden werden. Zusätzlich müssen jeweils zusammengehörenden Elemente der Message Types verbunden werden. Wurden die Elemente in beiden Data Types genau gleich benannt, kann dieser schritt automatisch durchgeführt werden. Ansonsten muss die Verbindung die Elemente von Hand durchgenommen werden.

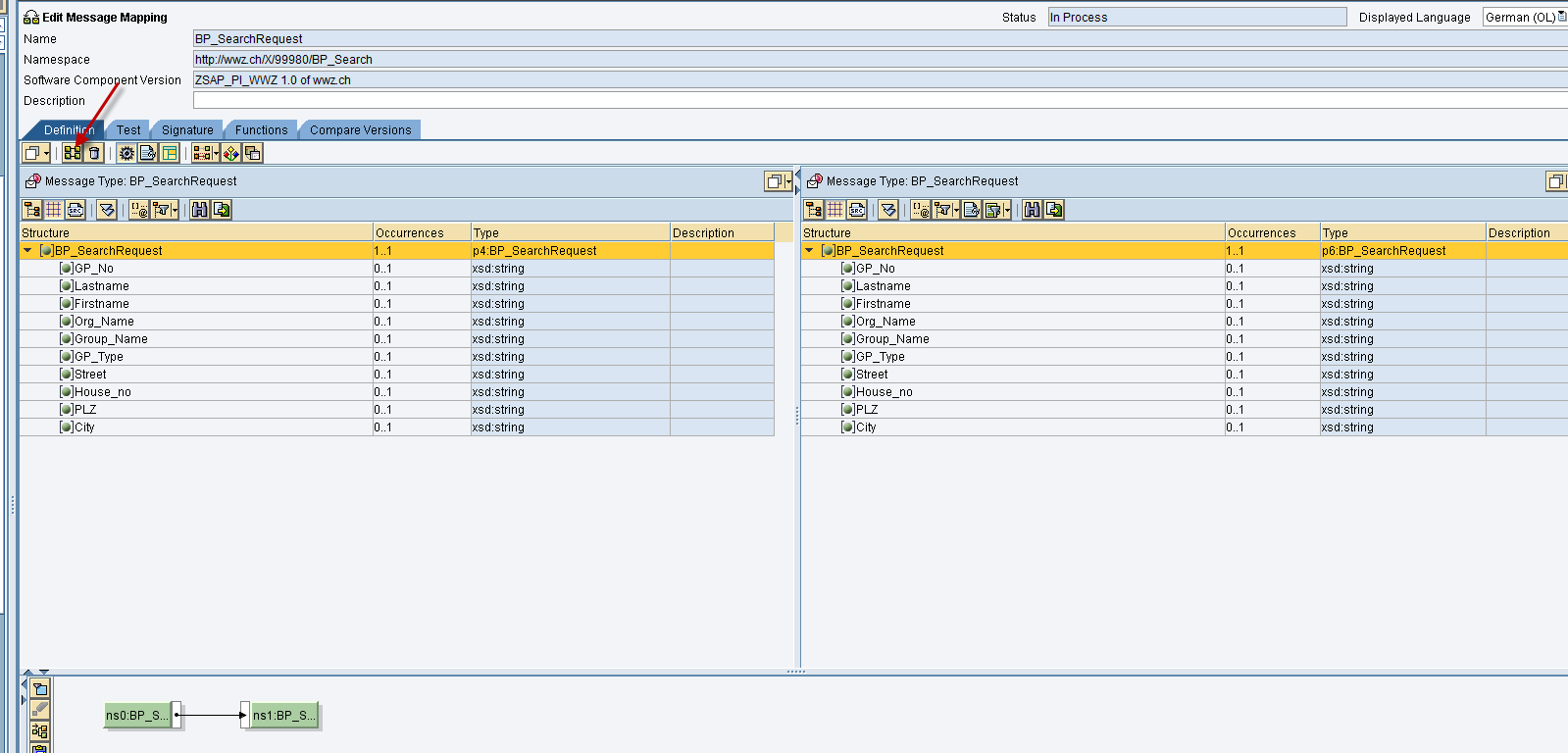


Abbildung 7: Mapping der BP\_Search Request Nachricht

### Operation Mapping

Über das Operation Mapping wird die Beziehung des Inbounds Service Interface mit dem Outbound Service Interface erstellt.

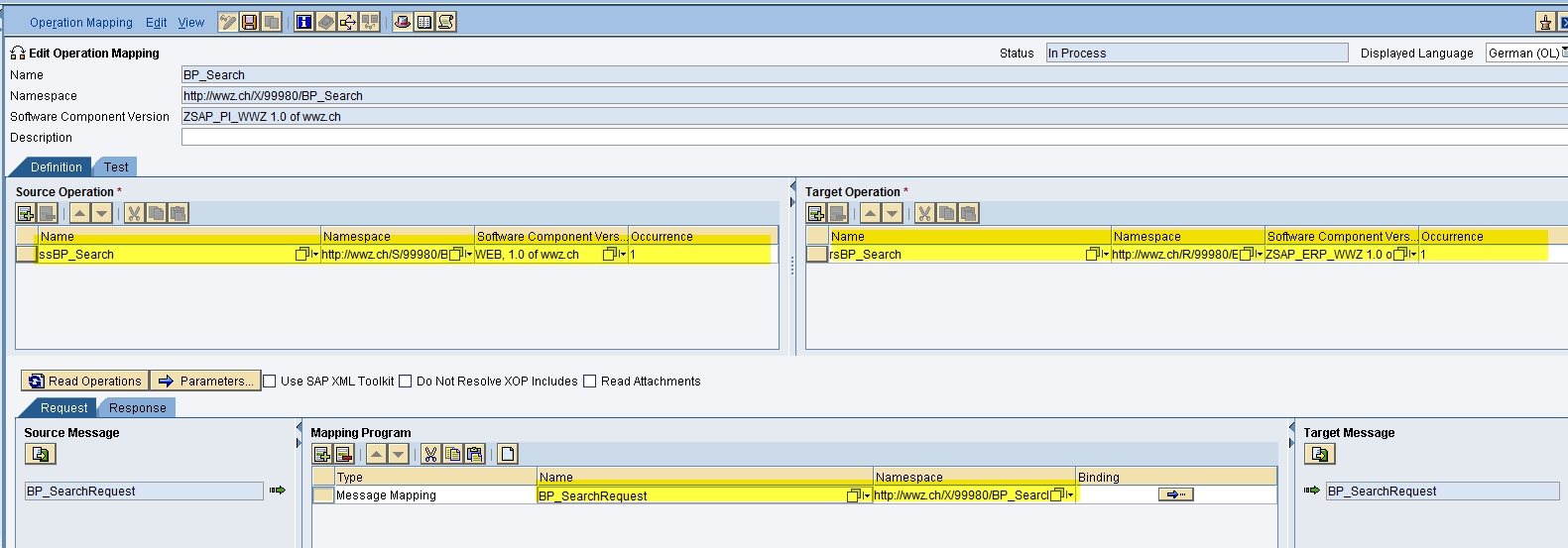


Abbildung 8: Das Operation Mapping für den Webservice BP\_Search

## Enterprise Services Builder

### Communication Channel erstellen!!!!!!!

### Integrated Configuration erstellen!!!!!!!!!!!!!!!!!!!

## SProxy

## 

# Planung Webanwendung

## Überlegungen

## Bibliotheken

## GUI Entwurf

Um bei der Erstellung der Webseite einen Anhaltspunkt zu haben, wie diese am aussehen soll, wird ein Entwurf für das GUI erstellt. Das GUI soll einfach und selbsterklärend sein, so dass selbst unerfahrene Benutzer die Anwendung leicht bedienen können.

### GUI Hauptseite

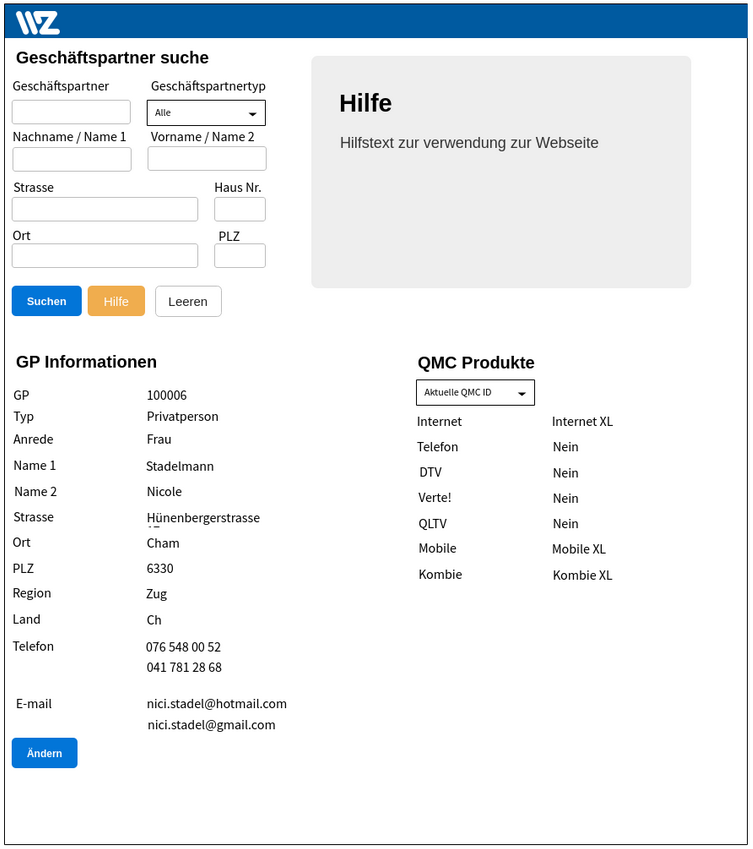


Abbildung 9: GUI für die Webseite

Überlegungen:

Über den Such-Button soll die GP Suche Ausgeführt werden. Werden mehrere Ergebnisse gefunden soll ein Fenster mit allen Gefunden GPs angezeigt werden, aus diesen kann ein GP ausgesucht werden. Sobald ein GP ausgesucht wurde, sollen die Abschnitte für die GP Informationen und die QMC Produkte angezeigt werden. Da ein GP mehrere QMC IDs besitzen kann, soll über eine Selection Box zwischen den QMC IDs auswählen. Bei den GP Informationen befindet sich ein änderungs-button. Wird in einem weiteren Fenster der Änderungsmodus aktiv. Der Hilfstext soll nur Angezeigt werden, wenn auf den Hilfe-Button geklickt wird.

### GUI für die GP Auswahl

Wen bei der Suche mehrere Treffer gefunden wurden, soll ein Fenster geöffnet werden, welches alle Treffer (maximal 50) anzeigt. Das Fenster wird über die Modal Funktion von Bootstrap umgesetzt. Die Gefunden GPs werden in einer Tabelle, angezeigt.

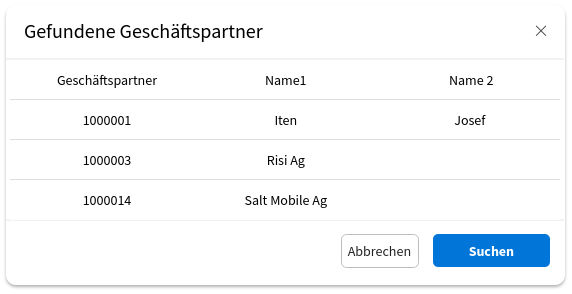


Abbildung 10: Die Auswahl eines GPs über die Modal Funktion von Bootstrap

### GUI für den Änderungsmodus

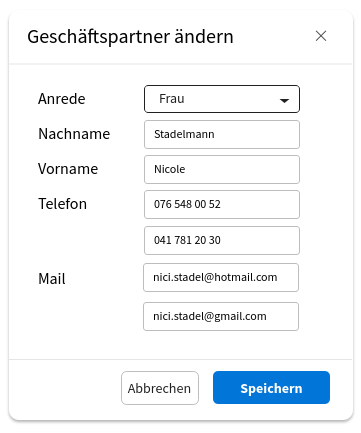
Der Änderungsmodus soll in einem Fenster, mit Hilfe der Bootstrap Modal Funktion, geöffnet werden.

Abbildung 11: Der Änderungsmodus

# Entwicklung Webanwendung

# Glossar

# Quellenverzeichnis

**Im aktuellen Dokument sind keine Quellen vorhanden.**

# Abbildungsverzeichnis

[Abbildung 1: Projektorganisation 13](#_Toc511578588)

[Abbildung 2: Das IPERKA Modell 14](file:///C:\Users\Nici\Desktop\IPA\IPA%20Bericht.docx#_Toc511578589)

[Abbildung 3: Erstellter Datentyp für BP\_Search Response 27](#_Toc511578590)

[Abbildung 4: Auswählen des Data Types für den Request Message Type von BP\_Search 28](#_Toc511578591)

[Abbildung 5: Der fertige Message Type mit der automatisch erstellen Struktur 28](#_Toc511578592)

[Abbildung 6:Das erstellte Service Interface mit der Outbound Kategorie und den zugewiesenen Message Types 29](#_Toc511578593)

[Abbildung 7: Mapping der BP\_Search Request Nachricht 30](#_Toc511578594)

[Abbildung 8: Das Operation Mapping für den Webservice BP\_Search 30](#_Toc511578595)

[Abbildung 9: GUI für die Webseite 32](#_Toc511578596)

[Abbildung 10: Die Auswahl eines GPs über die Modal Funktion von Bootstrap 33](#_Toc511578597)

[Abbildung 11: Der Änderungsmodus 33](file:///C:\Users\Nici\Desktop\IPA\IPA%20Bericht.docx#_Toc511578598)



Anhang

Anhang