



Matthias Pojer, BSc

**Entwicklung und Integration eines smarten elektronischen  
Signaturservice zur Unterstützung von  
elektronischen Geschäftsprozessen**

**MASTERARBEIT**

zur Erlangung des akademischen Grades

Diplom-Ingenieur

Masterstudium Softwareentwicklung - Wirtschaft

eingereicht an der

**Technischen Universität Graz**

Betreuer

Brenner, Eugen, Ao.Univ.-Prof. Dipl.-Ing. Dr.techn.

Institut für Technische Informatik

Fakultät für Informatik und Biomedizinische Technik

Graz, Oktober 2016

## **EIDESSTATTLICHE ERKLÄRUNG**

Ich erkläre an Eides statt, dass ich die vorliegende Arbeit selbstständig verfasst, andere als die angegebenen Quellen/Hilfsmittel nicht benutzt, und die den benutzten Quellen wörtlich und inhaltlich entnommenen Stellen als solche kenntlich gemacht habe. Das in TUGRAZonline hochgeladene Textdokument ist mit der vorliegenden Masterarbeit identisch.

---

Datum

---

Unterschrift

## Kurzfassung

Ausgelöst durch das Phänomen der Tertiarisierung und den digitalen Wandel nehmen IKT-basierte Dienstleistungen eine immer wichtigere Rolle in der Wirtschaft ein. Die Lenken von Innovationen im Dienstleistungssektor ist längst als ingenieurstechnische Disziplin etabliert.

Diese Arbeit nutzt das Verfahren des Service Engineering zur Weiterentwicklung einer Dienstleistung im Bereich elektronischer Unterschriften. Ziel ist die Steuerung und Automatisierung von elektronischen Geschäftsprozessen. Dazu wird das Konzept der Unterschriftenplatzhalter überarbeitet und visuelle maschinenlesbare Datenrepräsentationen evaluiert. Mittels Customer Co-Creation werden die Bedürfnisse der Kunden ermittelt, sowie eine Barcode Syntax und Semantik entworfen.

Ein weiteres Ziel ist die Integration des Service in ein Word Add-in, um die Adaption im Office Bereich zu fördern. Hierzu wird bei der Konzeption das Evaluierungsrisiko der Technologieauswahl durch den Einsatz von experimentellem Prototyping minimiert. Der daraus resultierende Prototyp wird erfolgreich getestet und als Modul vollständig in die Dienstleistung aufgenommen.

Schlüsselwörter:

*Service Engineering, Elektronische Signatur, Office Add-in*

## Abstract

Driven by the phenomenon of tertiarisation and digitalization the importance of ICT-based services in the economy is ever-growing. The steering of service innovation is established as its own engineering discipline.

This paper utilizes the service engineering method to advance an existing electronic signature service. The goal is to control and automate electronic business processes. Therefore the concept of signature placeholders is revised and visual machine-readable data representations are evaluated. Customer co-creation is used to identify the latent customer needs, as well as a barcode syntax and semantic is designed.

To boost office adaption, an additional goal is to integrate the service into a Word Add-in. Accordingly experimental prototyping is used to minimize technological evaluation risks. The resulting prototype is tested successfully and entirely integrated into the service as a module.

Keywords:

*Service Engineering, Electronic Signatures, Office Add-in*

## Danksagung

Diese Diplomarbeit wurde im Jahr 2016 am Institut für Technische Informatik an der Technischen Universität Graz durchgeführt.

Zuallererst möchte ich mich bei meinem Betreuer Ao.Univ.-Prof. Dipl.-Ing. Dr.techn. Eugen Brenner bedanken, der mich beim Erstellen der Arbeit tatkräftig unterstützt hat und immer ein offenes Ohr für meine Fragen hatte.

Mein aufrichtiger Dank gilt dem ganzen Unternehmen XiTrust GmbH, welches mir diese Forschungsarbeit ermöglichte. Insbesondere möchte ich mich bei DI Dr. Helmut Aschbacher bedanken, ohne den die Organisation des Projektes nicht durchführbar und mit weniger Spaß verbunden gewesen wäre. Weiters möchte ich mich bei DI Dr. Sabine Hösch und DI Georg Lindsberger für die vielen wertvollen Diskussionen zum Thema Platzhalter und den Prozessen bedanken. Stellvertretend für das ganze Entwickler Team möchte ich mich bei Oliver Winter bedanken, dessen unermüdliche Mitarbeit bei technischen Problemen maßgeblich und motivierend war.

Besonders möchte ich mich auch bei DI Gerald Wagner bedanken, der dieses Projekt in die Wege leitete. Auch möchte ich mich bei allen Freunden bedanken, welche die Arbeit mit ihrem Feedback bereicherten.

Ein großes Dankeschön gilt auch meiner Familie, insbesondere meinen Eltern Edmund und Barbara, ohne die mein Studium nicht möglich gewesen wäre.

Großer Dank gebührt überdies meiner Freundin Juli, die mir stets mit Rat und Tat zur Seite stand und mir Kraft zur Umsetzung des Projektes gab.

Graz, im Oktober 2016

Matthias Pojer

# Inhaltsverzeichnis

<b>1</b>	<b>Einleitung</b>	<b>1</b>
1.1	Ausgangssituation . . . . .	2
1.2	Zielsetzung . . . . .	3
1.3	Gliederung . . . . .	4
<b>2</b>	<b>Situationsanalyse</b>	<b>6</b>
2.1	Digitale Signatur in Österreich . . . . .	6
2.1.1	Begriffsdefinitionen . . . . .	7
2.1.2	Funktionsweise . . . . .	9
2.2	XiTrust . . . . .	10
2.2.1	Produkte allgemein . . . . .	11
2.2.2	MOXIS . . . . .	12
2.2.3	Word2MOXIS . . . . .	15
2.3	Visuelle maschinenlesbare Datenrepräsentation . . . . .	16
2.3.1	QR Code . . . . .	18
2.3.2	Datamatrix . . . . .	21
2.3.3	PDF-417 . . . . .	22
2.4	IKT basierte Dienstleistungen . . . . .	23
2.5	Innovationsprozess für IKT basierte Dienstleistungen . . . . .	26
2.5.1	Service Engineering . . . . .	26
2.5.2	Customer Co-Creation . . . . .	28
2.5.3	Smart Services . . . . .	29
2.5.4	9 Fenster Methode . . . . .	30
2.6	Technologieframeworks . . . . .	30
2.6.1	OOXML . . . . .	30
2.6.2	JavaScript API für Office . . . . .	32
2.6.3	Vaadin . . . . .	34
2.6.4	BPMN & Activiti . . . . .	35

<b>3</b>	<b>Innovationsvorgang</b>	<b>36</b>
3.1	Strategische Analyse . . . . .	36
3.1.1	Analyse & Ideenfindung . . . . .	36
3.1.2	Use Cases Word2MOXIS . . . . .	38
3.1.3	Auswahl der Datenrepräsentation . . . . .	39
3.2	Ideensteckbriefe . . . . .	41
3.2.1	ISB 1: Platzhalter QR-Code Kodierung . . . . .	41
3.2.2	ISB 2: Platzhalter Visualisierung . . . . .	45
3.2.3	ISB 3: MOXIS Standard Prozess Editor . . . . .	47
3.2.4	ISB 4: Word Add-In . . . . .	49
3.3	Bewertung und Festlegung der Technologie . . . . .	50
<b>4</b>	<b>Praktische Umsetzung</b>	<b>53</b>
4.1	Prototyp 1 . . . . .	54
4.1.1	Evaluation . . . . .	55
4.2	Prototyp 2 . . . . .	57
4.2.1	Design der Anwendung . . . . .	58
4.2.2	Beschreibung . . . . .	60
4.3	Analyse . . . . .	63
<b>5</b>	<b>Conclusio</b>	<b>66</b>
5.1	Fazit . . . . .	66
5.2	Ausblick und weitere Forschungsfragen . . . . .	67
5.2.1	Entwicklung der Office.js Plattform . . . . .	68
<b>A</b>	<b>Begriffsbestimmung</b>	<b>70</b>
A.1	Abkürzungsverzeichnis . . . . .	70
	<b>Literaturverzeichnis</b>	<b>72</b>

# Abbildungsverzeichnis

1	Aufbau der Arbeit . . . . .	4
2	XiTrust Firmenlogo . . . . .	10
3	MOXIS Rollenplatzhalter, alte Version . . . . .	13
4	Abfolge von Iterationen . . . . .	14
5	Word2MOXIS, alte Version . . . . .	16
6	Aufbau eines QR Code . . . . .	19
7	Data Matrix Code, Enkodierungsschema . . . . .	22
8	PDF-417 Beispiel . . . . .	23
9	Service Engineering Vorgehen mit Evaluierungsrisiken . . . . .	28
10	Ribbon Benutzeroberfläche . . . . .	33
11	9 Fenster Operator für den MOXIS Platzhalter . . . . .	37
12	QR Code: Minimal Enkodiert . . . . .	44
13	QR Code: Standard Enkodiert . . . . .	44
14	Benutzer Icon . . . . .	46
15	Rollen Icon . . . . .	46
16	Job Icon . . . . .	46
17	Iteration Icon . . . . .	46
18	Großer Platzhalter, neue Version . . . . .	46
19	Kleiner Platzhalter, neue Version . . . . .	47
20	Prototyp 1, Kundenprozess A . . . . .	54
21	Prototyp 1, Standard Prozess . . . . .	55
22	Prototyp 1, Strukturen einfügen (Gesamtansicht) . . . . .	56
23	Prototyp 2, Benutzerführung Kundenprozess . . . . .	57
24	Prototyp 2, Komponentendiagramm (schematisch) . . . . .	59
25	Prototyp 2, Anlegeseite . . . . .	60
26	Prototyp 2, Unterschreiber hinzufügen . . . . .	61
27	Prototyp 2, Platzhalter einfügen . . . . .	61
28	Prototyp 2, Auftrag abschließen (Gesamtansicht) . . . . .	63



# Tabellenverzeichnis

1	Gegenüberstellung: QR Code, Data Matrix & PDF-417 . . . . .	40
2	QR Code Versionsübersicht . . . . .	43
3	QR Enkodierung Minimal . . . . .	44

# Kapitel 1

## Einleitung

Dienstleistungen stellen mittlerweile den Großteil der wirtschaftlichen Aktivitäten in vielen entwickelten Ländern dar. Getrieben von der zunehmenden Digitalisierung werden viele existierende Unternehmensprozesse aus Kosten und Effizienzgründen durch IT-basierte Lösungen ersetzt. Um im globalen Wettbewerb attraktiv zu bleiben, wird dieser Trend von der EU mittels Agenden [SZS12] und rechtlichen Rahmenbedingungen [PotC14] unterstützt. Dabei sind sowohl im Design, als auch im Beziehen von Dienstleistungen moderne Management Methoden notwendig.

Im Rahmen dieser Arbeit wird eine Dienstleistungsinnovation im Bereich elektronischer Unterschriften durchgeführt. Kern des Produktes sind sogenannte Unterschriftenplatzhalter, die Positionen zum Signieren festlegen. Mittels maschinenlesbarer Parameter sollen diese mit Informationen versehen werden, um ganze Unterschriftenprozesse der Kunden zu automatisieren.

Dienstleistungen existieren nie in einem Vakuum, sondern Bestehen aus einer Kombination von sozialem und technologischem Potential. Wertsteigerungen des Dienstes sind oft durch Koppelungen mit ihrem Marktumfeld zu realisieren. Im vorliegenden Fall wird das Produkt mittels einer neuen Schnittstelle in die Microsoft Word Office Suite eingebettet.

Neben der Sammlung und Aufbereitung von Ideen ist die Auswahl der passenden Technologie zentral für das Gelingen des Projektes. Die daraus entstehenden Risiken werden durch sorgfältige Planung mittels des Service Engineering Verfahrens reduziert.

Ziel der Arbeit ist es, fundierte Handlungsempfehlungen abzugeben, sowie einen technischen Prototypen zu implementieren.

## 1.1 Ausgangssituation

Diese Diplomarbeit wird mit der XiTrust GmbH in Zusammenarbeit mit dem Institut für Technische Informatik durchgeführt. Das Unternehmen bietet Beratung, Dienstleistungen und Produkte in den Bereichen digitaler Signatur und Verschlüsselung an.

Zurzeit wird eine bereits am Markt erfolgreiche Dienstleistung zum elektronischen Unterschreiben und Verwalten von Dokumenten modernisiert und von Grund auf neu entwickelt. Dabei wird sowohl die verwendete Technologie erneuert, als auch neue Funktionalität hinzugefügt. Die Kernkomponente der Anwendung bleibt das rechtsgültige Signieren von Dokumenten, wobei an dafür freigehaltenen und spezifizierten Stellen eine Signaturvisualisierung angebracht wird. Diese Platzhalter sollen mit einer Prozess-Engine, welche sich gerade in Entwicklung befindet, kombiniert werden und diese sinnvoll erweitern.

Dazu wird eine breite Analyse des sozialen, rechtlichen und technologischen Umfelds durchgeführt. Untersuchungen zeigen, dass für die Innovation von IKT-basierten Dienstleistungen eine klassische straffe Projektstruktur nicht optimal ist. Eine offene Organisationsstruktur, die alle beteiligten Stakeholder, vor allem die Kunden selbst involviert, erhöht die Chance auf einen Erfolg. [BL09]

Das Angebot der Firma ist stark modularisiert, um Kunden eine maßgeschneiderte Auswahl an Dienstleistungspaketen anbieten zu können und die Wertschöpfung für alle beteiligten Akteure transparent zu gestalten [Asc09]. Ein Teilmodul soll das Service in die Microsoft Office Word Applikation eingliedern. Dies erfüllt identifizierte Kundenbedürfnisse, da die Anwendung in der Praxis sehr weit verbreitet ist und einen Marketingvorteil für die XiTrust darstellt.

Da diese Integration einen teilweisen Paradigmenwechsel im Vergleich zu einer reinen Webanwendung erfordert, ist eine strukturierte Ideengenerierung und Planung vonnöten. Hierzu werden verschiedene etablierte Werkzeuge aus dem Innovations-Management eingesetzt und beurteilt. Die in der Firma vorhandene Wissensbasis in der Entwicklung von sicherheitsbasierten Web-Anwendungen soll optimal genutzt und erweitert werden.

## 1.2 Zielsetzung

In diesem Abschnitt werden zwei im Rahmen der Arbeit gestellten Fragestellungen dargestellt und daraus abgeleitete Ziele und Nicht-Ziele formuliert.

### **Forschungsfrage 1:**

*Platzhalter mit QR Code, wie kann man das Konzept der Unterschriftenplatzhalter im XiTrust MOXIS innovieren?*

Das bestehende Konzept der Unterschriftenplatzhalter soll um einen smarten Bestandteil mit Informationsspeicherung erweitert werden und so mit der Process-Engine zusammenarbeiten. Dazu muss die geeignete Datenrepräsentation und ihr Format gefunden werden. Dabei wird ein Gleichgewicht zwischen sinnvoller Spezifikation und Anpassungsfähigkeit an Kundenwünsche angestrebt.

### **Forschungsfrage 2:**

*Wie kann man das Modul Word2MOXIS mit Hilfe des neuen Unterschriftenplatzhalterkonzeptes verbessern?*

Die an die neuen Anforderungen angepassten Platzhalter sollen auch in das Word2MOXIS Modul eingebaut werden. Da dies mit der zurzeit verwendeten Architektur nicht zufriedenstellend möglich ist, soll ein Konzept erstellt werden, mit dem das Word Add-in angepasst oder neu entwickelt wird. Hierbei soll das Problem als Ganzes betrachtet werden und Anforderungen, die das Modul an die Platzhalter stellt, in deren Konzeption rück-einfließen.

Als Ergebnis dieser Forschung erwartet sich die Firma XiTrust detaillierte Vorschläge an die Geschäftsführung, welche als Grundlage für Management Entscheidungen dienen. Ein fertiges Konzept für die Syntax und Semantik der Platzhalter soll erstellt werden und deren Nutzen an realen Use-Cases erprobt werden. Am Ende des Dienstleistungsinnovationsprozesses soll ein marktorientierter Prototyp entstehen. Zur Umsetzung soll das Potential bereits in der Firma verankerter Philosophien wie der Customer Co-Creation genutzt, und um sinnvolle Werkzeuge erweitert werden.

Nicht-Ziele sind eine komplette Neuausrichtung der Entwicklung und der sie steuernden Management Prozesse. Darüber hinaus wird auch keine vollständige Integration der erarbeiteten Vorschläge erwartet.

## 1.3 Gliederung



Abbildung 1: Aufbau der Arbeit

Die Struktur der vorliegenden Arbeit mit ihren drei Hauptteilen (Situationsanalyse, Innovationsvorgang und Praktische Umsetzung) ist in Abbildung 1 zu sehen.

Zuerst wird in der Situationsanalyse der Status von digitalen Signaturen mit ihrer rechtlichen Lage und Begriffsdefinition erläutert. Danach wird die Firma XiTrust mit ihrer Produktpalette vorgestellt. Darauf folgt eine Präsentation aktueller maschinenlesbarer Datenrepräsentationen, mit speziellen Fokus auf die drei in Erwägung gezogenen Barcodes, sowie eine Erklärung ihrer Funktionsweise.

Hiernach werden IKT-basierte Dienstleistungen mit ihren Merkmalen und Herausforderungen diskutiert. Gleich darauf werden das Service Engineering Verfahren, Customer Co-Creation und der 9 Fenster Operator als Innovationswerkzeuge präsentiert. Schlussendlich werden für die Prototypen relevante Technologien ausgewählt und dargestellt.

Den zweiten großen Abschnitt stellt der Innovationsvorgang dar. Dazu wird der 9 Fenster Operator angewandt und die Dienstleistung auf ihre Zeit- und Systemdimensionen untersucht. Anhand dieser Aufbereitungen werden Ideen generiert und Use Cases vorgestellt. Dies schließt die Anforderungsanalyse ab. Eine Entscheidung über den verwendeten Barcode wird getroffen und begründet. Darauf werden mit Ideensteckbriefen die vier Hauptkonzepte zur Verbesserung des Service vorgestellt. Das Kapitel endet mit dem Entschluss der Geschäftsführung und der Planung des weiteren Vorgehens.

Der dritte Abschnitt behandelt die praktische Umsetzung der Prototypen. Nach der Präsentation der Methode und den befolgten Prinzipien der Benutzerführung wird der erste Prototyp präsentiert. Die aus ihm gewonnenen Erkenntnisse werden im Detail behandelt und dienen als Grundlage der Entwicklung des zweiten Prototypen. Dessen Aufbau wird genau beschrieben und die Vor- und Nachteile der gewählten Technologie diskutiert. Zentrale Komponenten des Add-ins werden vorgestellt und die Anwendung mit Screenshots dokumentiert.

Die Endresultate werden mit den gesetzten Zielen verglichen und auf Erfolg geprüft.

# Kapitel 2

## Situationsanalyse

Zu Beginn dieses Kapitels werden die rechtlichen Rahmenbedingungen der digitalen Signatur in Österreich erörtert. Dabei werden die für die Diplomarbeit notwendigen Begriffe definiert und erklärt. Anschließend wird auf die technische Funktionsweise und die praktische Implementierung des Systems in Österreich eingegangen.

Danach wird die Firma XiTrust Secure Technologies GmbH (XiTrust) näher betrachtet, welche sich auf diesen neu geschaffenen Markt spezialisiert hat. Dazu wird ihre strategische Ausrichtung sowie ihre Produktpalette vorgestellt. Der Fokus wird auf das für die Diplomarbeit zentrale Produkt MOXIS gelegt, wofür die Terminologie und Use-Cases spezifiziert werden. Darüber hinaus wird auch der theoretische Hintergrund für den umgesetzten Innovationsprozess zur Produkterweiterung definiert. Zuletzt werden alle im Lösungsprozess erwogenen und verwendeten Technologien beschrieben.

### 2.1 Digitale Signatur in Österreich

Mit der zunehmenden Digitalisierung unseres Alltags wird die herkömmliche eigenhändige Unterschrift zunehmend obsolet. Oft sind gesamte Geschäftsvorgänge rein elektronisch und der Medienbruch durch den Ausdruck - Unterschreiben - Einscannen - Vorgang ist zeitaufwändig, teuer und fehleranfällig. Außerdem ist die naive Zwischenlösung, eine eingescannte Unterschrift wiederholt als Stempel in Dokumente einzufügen, aufgrund der Manipulationsgefahr rechtlich einer eigenhändigen Unterschrift nicht gleichgesetzt.

Um diese neu geschaffenen Anforderungen im europäischen Binnenmarkt einheitlich umzusetzen, definierte die EU mit der Signaturrechtlinie [PotC99] Vorgaben bezüglich elektronischer Signaturen. Die Richtlinie ist von den einzelnen Mitgliedsstaaten in die nationalen Gesetze zu übernehmen. Österreich ist im Umsetzen der legislativen Regelung Vorreiter und verabschiedete noch im selben Jahr das Signaturgesetz (SigG), welches zwischen einer

einfachen elektronischen Signatur und einer sicheren elektronischen Signatur unterscheidet [RB99]. 2008 wurden die Bestimmungen durch die Signaturverordnung (SigV) angepasst und insbesondere zwischen fortgeschrittenen und qualifizierten Signaturen unterschieden. Darüber hinaus sind einige Bestimmungen gelockert worden, um die Verbreitung von elektronischen Signaturen zu fördern.

Im Rahmen der Digitalen Agenda wurde 2014 die Verordnung über elektronische Identifizierung und Vertrauensdienste für elektronische Transaktionen im Binnenmarkt und zur Aufhebung der Richtlinie 1999/93/EG (eIDAS) veröffentlicht. Durch den Status einer Verordnung wird sie mit dem 1. Juli 2016 in allen EU-Staaten rechtswirksam. Die Unterscheidung zwischen einfachen und fortgeschrittenen Signaturen wird beibehalten, es gleicht hauptsächlich die rechtliche Verbindlichkeit in allen EU-Staaten an. Während eine fortgeschrittene elektronische Signatur der eigenhändigen Unterschrift gleichgestellt ist, ist die qualifizierte elektronische Signatur nicht abstreitbar und damit rechtlich stärker. Hinzu kommt auch die Regelung von elektronischen Siegeln. [PotC14]

Anstelle des aufgehobenen SigG regelt nun das Signatur- und Vertrauensgesetz (SGV) die notwendigen Anpassungen anderer österreichischer Gesetze aufgrund der eIDAS.

### 2.1.1 Begriffsdefinitionen

Im Folgenden werden die für die Diplomarbeit wichtigen Elemente der Verordnung vorgestellt. Sofern möglich, wird direkt im Originaltext zitiert. Bei sehr ausführlichen Querverweisen wird zusammengefasst und an die genaue Stelle in [PotC14] verwiesen.

**Elektronische Identifizierung** ist der Prozess der Verwendung von Personenidentifizierungsdaten in elektronischer Form, die eine natürliche oder juristische Person oder eine natürliche Person, die eine juristische Person vertritt, eindeutig repräsentieren [§1 Art. 3 eIDAS].

**Elektronisches Identifizierungssystem** ist ein System für die elektronische Identifizierung, in dessen Rahmen natürlichen oder juristischen Personen oder natürlichen Personen, die juristische Personen vertreten, elektronische Identifizierungsmittel ausgestellt werden [§4 Art. 3 eIDAS].

**Unterzeichner** (oder auch Signator, Anm. d. Verf.) ist eine natürliche Person, die eine elektronische Signatur erstellt [§9 Art. 3 eIDAS].

**Elektronische Signatur** sind Daten in elektronischer Form, die anderen elektronischen Daten beigelegt oder logisch mit ihnen verbunden werden und die der Unterzeichner zum Unterzeichnen verwendet [§10 Art. 3 eIDAS].



**Fortgeschrittene elektronische Signatur** ist eine elektronische Signatur, welche gemäß [Art. 26 eIDAS] folgende Anforderungen erfüllt:

- Sie ist eindeutig dem Unterzeichner zugeordnet.
- Sie ermöglicht die Identifizierung des Unterzeichners.
- Sie wird unter Verwendung elektronischer Signaturerstellungsdaten erstellt, die der Unterzeichner mit einem hohen Maß an Vertrauen unter seiner alleinigen Kontrolle verwenden kann.
- Sie ist so mit den auf diese Weise unterzeichneten Daten verbunden, dass eine nachträgliche Veränderung der Daten erkannt werden kann.

**Qualifizierte elektronische Signatur** ist eine fortgeschrittene elektronische Signatur, die von einer qualifizierten elektronischen Signaturerstellungseinheit erstellt wurde und auf einem qualifizierten Zertifikat für elektronische Signaturen beruht [§12 Art. 3 eIDAS].

**Elektronische Signaturerstellungsdaten** sind eindeutige Daten, die vom Unterzeichner zum Erstellen einer elektronischen Signatur verwendet werden [§16 Art. 3 eIDAS].

**Vertrauensdienst** ist ein meist gegen Entgelt erbrachter elektronischer Dienst, der für die Erstellung, Überprüfung, Validierung und Bewahrung der Zertifikate, welche für die oben genannten Signaturen notwendig sind, zuständig ist [§16 Art. 3 eIDAS].

**Qualifizierter Vertrauensdiensteanbieter** ist ein Vertrauensdiensteanbieter, der einen oder mehrere qualifizierte Vertrauensdienste erbringt und dem von der Aufsichtsstelle der Status eines qualifizierten Anbieters verliehen wurde [§20 Art. 3 eIDAS].

**Zertifikat für elektronische Signaturen** ist eine elektronische Bescheinigung, die elektronische Signaturvalidierungsdaten mit einer natürlichen Person verknüpft und die mindestens den Namen oder das Pseudonym dieser Person bestätigt [§14 Art. 3 eIDAS].

**Qualifiziertes Zertifikat für elektronische Signaturen** ist ein Zertifikat für elektronische Signaturen, bei dem ein Vertrauensdienst für den Inhalt des Zertifikats zum Zeitpunkt der Ausstellung haftet. Die exakten Bedingungen sind in Anhang I beschrieben.

**Qualifizierte elektronische Signaturerstellungseinheit** müssen die Verfahren zur Signaturerstellung vertraulich, kryptographisch einwandfrei und sicher gestalten (genauer siehe Anhang II). Sie dürfen die zu unterzeichnenden Daten nicht verändern sowie ihre Anzeige verhindern. Außerdem haben sie das alleinige Recht, elektronische Signaturerstellungsdaten im Namen eines Unterzeichners zu erstellen [Anh. II eIDAS].

In Österreich nimmt die A-Trust als staatlich akkreditierter qualifizierter Vertrauensdienst diese Rolle wahr.

**Elektronisches Siegel** sind Daten in elektronischer Form, die anderen Daten in elektronischer Form beigefügt oder logisch mit ihnen verbunden werden, um deren Ursprung und Unversehrtheit sicherzustellen [§25 Art. 3 eIDAS].

Fortgeschrittene und qualifizierte Siegel sind Verbesserungen, ähnlich denen der elektronischen Signatur, genauer siehe [§25, §26 Art. 3, sowie Art. 36 eIDAS] .

**Elektronischer Zeitstempel** bezeichnet Daten in elektronischer Form, die andere Daten in elektronischer Form mit einem bestimmten Zeitpunkt verknüpfen und dadurch den Nachweis erbringen, dass diese anderen Daten zu diesem Zeitpunkt vorhanden waren [§33 Art. 3 eIDAS].

### 2.1.2 Funktionsweise

Während der Begriff der elektronischen Signatur den rechtlichen Rahmen regelt, bezieht sich der Begriff der digitalen Signatur auf die technische Implementierung des Signatursystems. Alle im Rahmen dieser Arbeit verwendeten qualifizierten elektronischen Signaturen bauen auf digitalen Signaturen auf. In der Praxis wird hier asymmetrische Kryptographie verwendet, deren Schlüssel mittels Public Key Infrastruktur (PKI) verteilt werden. Die A-Trust stellt eine solche PKI bereit <sup>1</sup>.

Der zentrale Aspekt der Public Key Kryptographie ist die Trennung von privatem und öffentlichem Schlüssel. Die Geheimhaltung des privaten Schlüssels ist wichtig, da dieser zum Signieren der Nachricht verwendet wird. Der öffentliche Schlüssel dient dem Verifizieren der Nachricht und der Authentifizierung des Senders. Eine Nachricht kann jegliche digitale Information sein. Ein Digitales Signaturverfahren besteht aus den folgenden drei Schritten:

1. Im ersten Schritt wird das Schlüsselpaar generiert. Dabei ist es notwendig, dass der private Schlüssel diskret gleichverteilt ist und der Algorithmus und seine Implementierung keine probabilistische Lücke aufweisen. Diese Dienstleistung wird von einem Vertrauensdienstleister erbracht und der private Schlüssel verlässt diesen nie.
2. Der Unterschriften-Algorithmus nutzt die Nachricht und den privaten Schlüssel, um eine digitale Signatur zu erzeugen. Dabei wird meist nicht die Nachricht als Ganzes verwendet, sondern ein Hash dieser Nachricht berechnet. Die verwendete Hashfunktion muss kollisionsresistent sein, wodurch es unmöglich sein soll, zwei Nachrichten

---

<sup>1</sup><https://www.a-trust.at/Business/Anwendungen.aspx>, zuletzt zugegriffen am 05.10.2016

zu finden, deren Hash Wert identisch ist. Die Verwendung des Hash Wertes hat folgende Vorteile:

- Die Effizienz des Systems wird erhöht, da die Signatur viel kürzer ist.
- Die Generierung der Signatur ist unabhängig von der Nachrichtenkodierung.
- Die Länge der Nachricht ist egal und muss nicht in Blöcke aufgeteilt und wieder zusammengesetzt werden.

Hierauf wird das Dokument vom Vertrauensdienstleister gehasht und dieser Hash Wert signiert. Der Anwender autorisiert das Unterschreiben mittels seiner digitalen Identität, indem er auf seine Handy Nummer einen TAN Code bekommt, mit dem er den Vorgang bestätigt.

3. Der Verifikationsalgorithmus verwendet den öffentlichen Schlüssel aus einem öffentlichen Verzeichnis des Vertrauensdienstleisters gemeinsam mit der Nachricht und ihrer digitalen Signatur, um die Authentizität des Absenders und die Integrität der Nachricht zu prüfen. Dabei wird der Unterschreiber unbestreitbar als Signator der Nachricht identifiziert.

## 2.2 XiTrust

Die in Graz ansässige und inzwischen auf drei Standorte in Österreich und Deutschland expandierte Firma XiTrust begann ihre Entwicklung bereits im Jahr 2000, damals noch unter dem Namen Xicrypt GmbH. Von Beginn an war das Bestreben, Produkte rund um das Thema Digitale Signatur zu entwickeln [Asc09]. Nach der

Umbenennung in XiTrust im Jahr 2002 begann ein kontinuierlicher Expansionsprozess, der bis heute anhält. Das Unternehmen beschreibt seine Kompetenzen als: <sup>2</sup>



Abbildung 2: XiTrust Firmenlogo

*Innovation sowie beste Qualität für Kunden sind nur einige Faktoren, die XiTrust seither zur ersten Anlaufstelle für Beratung, Dienstleistungen und Produkte rund um elektronische Signatur und medienbruchfreie Geschäftsprozesse macht. [...] verfolgt XiTrust gemeinsam mit seinen Kunden die Vision einer papierlosen Zukunft. Die Umsetzung dieser Vision wird ermöglicht durch Technologien wie digitale Signatur und Verschlüsselung, Handy-Signatur, Time-Stamping sowie Archivierung elektronischer Dokumente.*

---

<sup>2</sup><http://www.xitrust.com/xitrust/ueber-uns/>, zuletzt zugegriffen am 30.07.2016

### 2.2.1 Produkte allgemein

Das Produktportfolio spiegelt die Kernkompetenzen des Unternehmens wider und bietet folgende Lösungen an:

**XiTrust MOXIS** Die digitale Unterschriftenmappe. Wird in Kapitel 2.2.2 genauer beschrieben.

**XiTrust Business Server** Ist ein modularer zentraler Dienst, der bestehende IT Infrastruktur vernetzt und um Funktionalität erweitert. Die folgende Liste stellt einen Auszug der zurzeit bereit gestellten Features dar: <sup>3</sup>

- Dokumente signieren, prüfen und archivieren
- E-Mail Verschlüsselung und Signatur
- Erstellung von Workflows, um Geschäftsprozesse zu automatisieren
- E-Government Integration

**XiTrust Timestamp Server** Dieser Dienst wird oft mit dem Business Server zusammen gebündelt, erweitert diesen um sichere elektronische Zeitstempel und umfasst folgende Features: <sup>4</sup>

- Inhalt und Zeitpunkt von Dokumenten mit Aufbewahrungspflicht nachweisen
- Sichere Versionierung von geschäftskritischen Dokumenten
- Nachweis der rechtzeitigen Übermittlung zeitkritischer E-Mails
- Nachweis der Urheberschaft an einem Patentantrag oder Gebrauchsmuster

**XiDentity** Ermöglicht europaweit einen einfachen Zugang zu einer Digitalen Identität, welche normalerweise an die Meldedaten eines Staates gebunden sind. Diese ist notwendig, um gemäß der europäischen Signatur-Richtlinie von 1999 elektronisch rechtsgültig qualifiziert unterschreiben zu können. Die rechtlichen Rahmenbedingungen wurden mit der eIDAS Verordnung weiter entwickelt und ersetzt. Im Kapitel 2.1 werden die einzuhaltenen rechtlichen Rahmenbedingungen erörtert.<sup>5</sup>

Bei vielen Produkten wird auf die A-Trust als staatlich anerkannte Zertifizierungsstelle zurückgegriffen, um Benutzern ihre digitale Identität zuzuweisen.

---

<sup>3</sup><http://www.xitrust.com/produkte/xitrust-business-server/>, zuletzt zugegriffen am 30.07.2016

<sup>4</sup><http://www.xitrust.com/produkte/xitrust-timestamp-server/>, zuletzt zugegriffen am 30.07.2016

<sup>5</sup><https://www.xidentity.eu/>, zuletzt zugegriffen am 30.07.2016

### 2.2.2 MOXIS

Die Mobile XiTrust Signaturlösung (MOXIS) ist ein web-basiertes System, das alle Funktionen einer herkömmlichen papiergebundenen Unterschriftenmappe übernimmt. Es dient dem Hochladen und Verwalten von Dokumenten, welche mit einer elektronischen Signatur versehen werden. Dokumente können beliebig oft fortgeschritten oder qualifiziert unterschrieben werden. Die Software wird in die bestehende IT Infrastruktur des Kunden integriert und von diesem betrieben, sodass Dokumente ihre Unternehmen nie verlassen. Für das Anbringen von qualifizierten elektronischen Signaturen wird die Handy-Signatur der A-Trust verwendet, welche diesen Vertrauensdienst erbringt. Die Vorteile des Produktes gegenüber der klassischen eigenhändigen Unterschrift und Mitbewerbern sind:

- Zeit & Kostenersparnis durch das Wegfallen von Papierstapeln
- Erlaubt paralleles unterschreiben
- Weniger fehleranfällig durch medienbruchfreie Prozesse
- Örtlich und zeitlich ungebunden rechtsgültig unterschreiben
- Übersichtliche Dokumentenverwaltung
- Datenschutz, da unternehmensinterne Lösung
- Dokumente können nur von den adressierten Benutzern gelesen werden
- Stapelsignatur & Freigaben möglich

**Begriffsdefinitionen** Zusätzlich zu den in Abschnitt 2.1 angeführten Definitionen sind für das Verständnis von MOXIS folgende Begriffe notwendig:

**Auftrag** Ein spezifischer Signaturvorgang, dem Dokumente und Unterschreiber angehören.

**Auftragsmappe** Beinhaltet alle aktiven Aufträge mit den untergeordneten Dokumenten, welche der Benutzer erstellt und hochgeladen hat.

**Unterschriftenmappe** Beinhaltet alle aktiven Aufträge, denen ein Benutzer zugeordnet ist. Hier werden alle Dokumente angezeigt, die seine Signatur oder Freigabe erfordern.

**Freigeben** Dient oft dem öffentlich machen oder der Weitergabe von signierten und un-signierten Dokumenten. Dabei wird auf eine fortgeschrittene elektronische Signatur zurückgegriffen, welche den Inhalt zum Zeitpunkt der Freigabe garantiert.

Im Unterschied zur qualifizierten Signatur kann diese auch auf das Unternehmen, oder eine Gruppe von Personen ausgestellt werden. Dadurch sind die unterschreibenden Personen nicht mehr namentlich erkenntlich, mit der Ausnahme von bereits angebrachten qualifizierten Signaturen.

**Rollen** Damit wird eine Gruppe von Personen bezeichnet, die eine bestimmte Aufgabe wahrnehmen. Hier wird ein Subset aller Benutzer gebildet und meist in Verbindung mit Freigaben benutzt.

**Platzhalter** Der Platzhalter ist ein Bereich auf einem Dokument, der für eine Unterschriften- oder Freigabevisualisierung freigehalten wird. Um diese Funktion zu erfüllen, besteht er aus einem Logo, Beschreibungstext und QR Code. Das Logo kennzeichnet ihn sichtbar als MOXIS Platzhalter und hebt ihn vom restlichen Inhalt ab.

Der Beschreibungstext schildert die auf diesem Platzhalter erwartete Unterschrift. Wenn eine qualifizierte elektronische Signatur erwartet wird, kann hier „Max Mustermann“ stehen. Wird jedoch eine fortgeschrittene Signatur einer Gruppe von Personen erwartet, wird dies mittels „Vorstand“ gekennzeichnet. Diese Vorgaben stellen lediglich eine Hilfestellung für den Betrachter des Dokuments dar und werden von MOXIS selbst nicht weiter verwendet.

Diese Aufgabe fällt dem QR Code zu, der den gewünschten Unterschreiber maschinenlesbar abspeichert. MOXIS scannt das Dokument und listet alle gefundenen Platzhalter auf. Der Benutzer kann entscheiden, wo und wie er unterschreiben will. MOXIS bringt abschließend bei erfolgreicher Unterschrift eine Signaturvisualisierung über dem Platzhalter an. Abbildung 3 zeigt einen solchen Platzhalter.



Abbildung 3: MOXIS Rollenplatzhalter, alte Version

**Signaturvisualisierung** Ist ein vom Benutzer frei wählbarer Stempel, der nach erfolgreichem Unterschreiben über dem Platzhalter angebracht wird. Oft wird hier eine eingescannte eigenhändige Unterschrift verwendet. Die Visualisierung hat keine rechtliche Wirkung, sondern ist lediglich ein Indiz, dass eine rechtsgültige elektronische Signatur am Dokument angebracht worden ist. Sie erfüllt ästhetische Erwartungen, die von Benutzern an eigenhändig unterschriebene Schriftstücke gestellt werden.

**Unterschriftsprozess** Ein solcher Prozess definiert und steuert den Signatur-Vorgang. Benutzer legen Aufträge an, welche den Prozess instantiiieren. Er lädt die beteiligten Personen ein, benachrichtigt diese, bringt die Unterschriften an und schließt diesen Auftrag nach erfolgreichem Ablauf.

**Geschlossene Aufträge** Hier werden alle vollendeten Aufträge gesammelt. Aufträge können durch das Anbringen aller erforderlichen Unterschriften, sowie nach Zeita-

blauf oder Beendigung des Auftraggebers vollendet werden. Sowohl vom Benutzer erstellte, als auch von ihm signierte Aufträge werden gezeigt.

**Iterationen** Damit werden serielle und parallele Abläufe in MOXIS modelliert. Auf Platzhaltern derselben Iterations-ebene kann zeitgleich unterschrieben werden. Zugleich wird aber erst mit der nächsten Iteration fortgefahren, wenn alle spezifizierten Unterschriften der vorherigen Iteration erfolgreich am Dokument angebracht sind. Zum Projektstart ist es erst rudimentär umgesetzt worden und noch gar nicht in Word2MOXIS vorhanden.

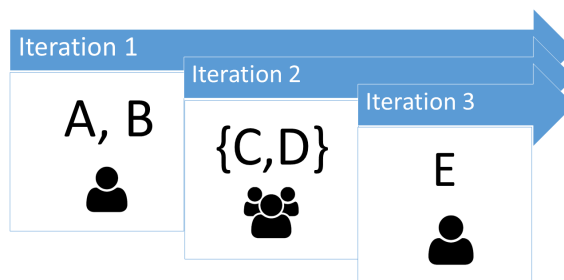


Abbildung 4: Abfolge von Iterationen

Abbildung 4 zeigt einen Prozess mit drei Iterationsebenen. Die zweite Ebene wird erst gestartet, nachdem sowohl A als auch B unterschrieben haben. Da C und D eine Gruppe bilden, reicht es aus, wenn einer von beiden unterschreibt, um die Iterationsebene abzuschließen.

**MOXIS Alt vs. MOXIS Prozess-Engine** Zum Start dieser Diplomarbeit wird MOXIS von Grund auf erneuert. Dabei wird die technologische Basis aktualisiert, sowie die Lösung mittels Spring Komponenten stärker modularisiert. Für die Kunden sichtbar ist vor allem die Neugestaltung der Benutzeroberfläche.

Hauptmotivation für die Neuentwicklung ist die Integration der sogenannten Prozess-Engine, weshalb die neue MOXIS Version firmenintern mit MOXIS-PE abgekürzt wird. Im Zentrum steht die Integration eines Business Process Model and Notation (BPMN) Systems, mit dem Geschäftsprozesse modelliert werden können. Dies soll die Erstellung und Wartung von Arbeitsprozessen (Workflows) erleichtern. Ein weiterer Vorteil ist es, dass stets eine visuelle Repräsentation des jeweiligen Prozesses zur Verfügung steht. Dadurch können Kunden den Ablauf verstehen und selbst konfigurieren.

In Abschnitt 2.6 wird näher auf die dafür verwendeten Technologien eingegangen. Da MOXIS-PE parallel zu dieser Arbeit entwickelt wird, beeinflussen sich beide Projekte gegenseitig stark.

**Ablauf** Ein authentifizierter Benutzer mit Auftraggeber-Berechtigung kann einen Auftrag starten. Dazu wählt er in einer Eingabemaske das zu unterschreibende Dokument aus. Zurzeit werden ausschließlich PDF Dokumente akzeptiert. Danach können ein oder mehrere Platzhalter am Dokument angebracht werden.

In weiterer Folge werden alle Signatoren für das Dokument bestimmt, die Anzahl der notwendigen Signaturen festgelegt und das Ablaufdatum des Auftrages bestimmt. Darüber hinaus kann der Auftrag einer Kategorie zugeordnet werden. Nach einer Bestätigung wird der Auftrag erstellt.

Alle angegebenen Signatoren werden über eine E-Mail Benachrichtigung zum Unterschreiben eingeladen. Zusätzlich wird dieser Auftrag nun in ihrer Unterschriftenmappe angezeigt, und sie können ihn jederzeit signieren. Bei jeder erfolgreichen Signatur wird der Auftraggeber verständigt. Nachdem die erforderliche Anzahl an Unterschriften erreicht wird, gilt der Auftrag als abgeschlossen und wandert in die Kategorie der geschlossenen Aufträge aller beteiligten Personen.

### 2.2.3 Word2MOXIS

Word2MOXIS ist ein Visual Studio Tools for Office (VSTO) Add-in für Microsoft Word, das MOXIS in dieses integriert. Mittels des Ribbon Interface können Platzhalter über eine Liste ausgewählt und in das Word Dokument eingefügt werden. Zum Abschluss wird eine automatische Konvertierung in das PDF Format vorgenommen und das Resultat in die MOXIS Installation hochgeladen.

Word wird in vielen Unternehmen eingesetzt. Da viele Dokumente ohnehin mit diesem Programm erstellt werden, ist diese Integration in der Praxis sehr nützlich. Durch das automatische Konvertieren wird ein weiterer Arbeitsschritt eingespart. Darüber hinaus ermöglicht diese Konfiguration das Erstellen von Templates, bei denen die Platzhalter schon im Word Format angebracht sind und immer nur aktuelle Details und Inhalte angepasst werden müssen.

Aufgrund der Natur von VSTO Add-ins stellt sich das Ausliefern und Updaten als Achillesferse heraus. Ein Installationsskript muss an jeden Kunden angepasst werden. Dieses liefert alle Platzhalter mit und installiert sie lokal. Daneben wird über ein mit Java realisiertes Interface die Add-in zu MOXIS Kommunikation abgewickelt. Die Verbesserung dieser Pipeline ist ein Kernaspekt der Arbeit.



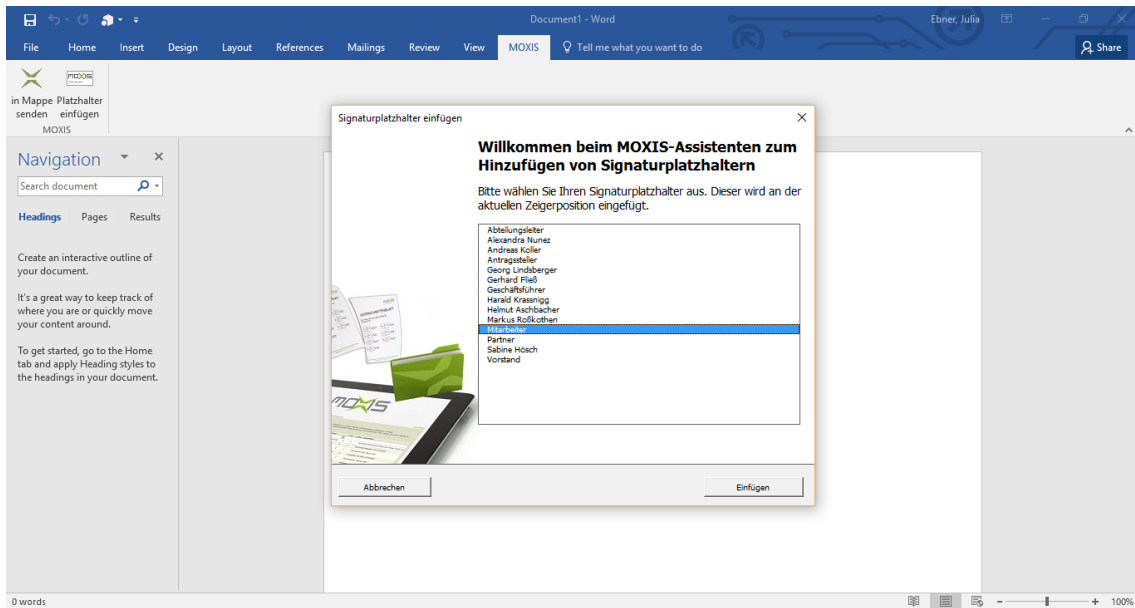


Abbildung 5: Word2MOXIS, alte Version

## 2.3 Visuelle maschinenlesbare Datenrepräsentation

Bei MOXIS werden Unterschriftenplatzhalter (Unterschriften-PH) in Dokumente eingefügt, um Prozessinformation abzuspeichern. Zugleich dient dieser Platzhalter (PH) auch dazu, später die Unterschriften Visualisierung am Dokument anzubringen. Da dies sowohl digital als auch physisch an ausgedruckten Materialien möglich sein soll, benötigen wir eine maschinenlesbare Datenrepräsentation, welche die angebrachten Informationen auslesen kann.

Dank jahrzehntelanger Forschung im Bereich der automatischen Identifizierung (Auto-ID) kann für diesen Einsatzzweck auf eine große Basis verfügbarer und erprobter Technologien zugegriffen werden. Für die optische Übertragung von Information bieten sich Methoden der optischen Zeichenerkennung und Barcodes an. Aufgrund ästhetischer Merkmale, Informationsdichte und Scangeschwindigkeit sind Barcodes Klartext vorzuziehen und werden daher nachfolgend erörtert.

**Ausgangssituation** Durch die zunehmende Automatisierung von Produktions- und Verarbeitungsprozessen, die zunehmend von Computern gesteuert wurden, kam schon früh der Wunsch auf, Daten maschinenlesbar an Objekten anzubringen. Den Grundstein für die heute eingesetzten Barcodes legten N.J Woodland und B. Silver mit einer Patentanmeldung im Jahr 1949 [Sei11].

Ende der 1960er Jahre experimentierten amerikanische Eisenbahngesellschaften mit Automatic Car Identification (ACI), welches aber aufgrund mangelnder Zuverlässigkeit wieder fallen gelassen wurde. Hauptgrund war die Anfälligkeit der blau-roten Streifen für Verschmutzung oder leichte Beschädigungen. [Cra11]

Um Kosten zu sparen und die Effizienz der Wareninventur zu erhöhen, wurde weiter an den Ideen von Woodland und Silver geforscht. 1974 begann mit dem Scan eines Kaugummi durch Alan Haberman der Erfolgslauf des nun allgegenwärtigen Universal Product Code (UPC), welcher als einer der erfolgreichsten Standards, die je entwickelt wurden, gilt [Fox11]. In Europa setzte sich der Code minimal verändert als European Article Number (EAN) durch, welcher seit 2009 gemeinsam mit dem UPC weltweit einheitlich als Global Trade Item Number (GTIN) verwendet wird [Bro01].

Gemeinsam ist allen Barcode Arten, dass sie auf Linien, Quadrate und andere geometrische Muster setzen, um binäre Informationen zu kodieren. Darüber hinaus werden verschiedene Fehlerkorrektur-Algorithmen angewandt, um bei Beschädigung des Codes oder optischen Effekten trotzdem einen fehlerfreien Scan zu gewährleisten. Auf dieser Grundtechnologie aufbauend unterscheidet man mehrere Arten von Barcodes, welche im Folgenden beschrieben werden.

**Eindimensionale Barcodes** Codes dieser Art werden auch Strichcode genannt und bestehen aus hintereinander gestaffelten parallelen Linien. Eindimensionale Barcodes sind am weitesten verbreitet, die oben erwähnten GTIN Codes finden breite Anwendung in der Logistik, Buchhandel, Flugzeugtickets und Gesundheitswesen. Durch die geringe Speicherkapazität ist zur Nutzung immer eine Datenbank notwendig, welche dem Code in der Datenbank gespeicherter Information zuordnet.

Speziell im Gesundheitsbereich treten die Limitierungen dieser Technologie zu Tage. Die Fehlerkorrektur funktioniert gemäß einer Studie von M. L. Snyder nicht zuverlässig genug, da in der Praxis öfter Substitutions-Fehler auftreten als erwartet. Durch die geringe Speicherkapazität bei relativ großem Platzbedarf stehen auch nur wenige Ziffern für die Prüfsumme der Fehlerkorrektur zur Verfügung. Im untersuchten Fall wird eine 2-stellige Prüfziffer verwendet, die maximal 102 verschiedene Zahlen erlaubt. [SCJF10]

**Zweidimensionale Barcodes** Diese Codes sind die logische Evolution der eindimensionalen Barcodes und erweitern die Speicherkapazität dieser enorm. Dadurch wird die direkte Nutzung der Codes als mobiler Datenspeicher ermöglicht und sie sind nicht mehr nur der Schlüssel zu einem Datenbankeintrag. Darüber hinaus werden enorm verbesserte Fehlerkorrekturen verwendet. Der Begriff der Barcodes wird in der Literatur auch für 2D-Codes verwendet, obwohl sie nicht mehr Striche (engl. Bars) verwenden.

Man unterscheidet zwischen **Stapelcodes**, welche aus übereinander gelegten 1D Codes bestehen, und echten **Matrixbarcodes**. Kato und Tan [KT05] attestieren den Matrix Codes vor allem in der mobilen Anwendung Vorteile, da sie omni-direktional lesbar sind. Es gibt viele verschiedene Implementierungen von 2D Barcodes, welche sich im Einsatzgebiet, der Verbreitung und dem Lizenzmodell unterscheiden. Offene Systeme, die eine kostenlose Erstellung und Lesen der Codes ermöglichen, sind geschlossenen Systemen vorzuziehen [UH12].

2D Barcodes erfüllen die in dieser Diplomarbeit gestellten Anforderungen am Besten, weshalb die zur Erwägung gezogenen Vertreter dieser Kategorie - *Portable Data File 417 (PDF417)*, *Quick Response Code (QR Code)* und *Data Matrix Code (DMC)* - in diesem Kapitel genauer betrachtet werden.

**Dreidimensionale Barcodes** Es existieren 3D Barcodes, bei denen die Matrix um Farben erweitert wird, wodurch eine weitere Steigerung der Kapazität erreicht wird. Als Vertreter ist hier der von Microsoft entwickelte High Capacity Color Barcode (HCCB) zu nennen. Allerdings schränken hier Lichtverhältnisse, Ausbleichen und Abbildungsfehler den Einsatz ein, und stellen hohe Anforderungen an die Computer Vision [PJ08].

Darüber hinaus wird auch an 4D Barcodes geforscht, welche die Komponente Zeit hinzufügen und farbige Codes über einen Zeitraum animieren [LB07].

### 2.3.1 QR Code

Der QR Code wurde 1994 von dem japanischen Automobilzulieferer DENSO WAVE entwickelt, um Bauteile an Fertigungslinien zu verfolgen [UH12]. Durch dessen Herkunft kann ein QR Code neben den gängigen numerischen, alphanumerischen und binären Daten auch Kanji Zeichen effizient speichern. Ein Code besteht aus einer Matrix von schwarzen und weißen Quadraten, die von einem Bildsensor aufgenommen und von einer Software ausgewertet werden.

Ein solches Quadrat wird als Modul bezeichnet. Die Größe der Codes wird durch die Anzahl der Module bestimmt, wobei die kleinste Version über 21x21 Module verfügt. Deren Anzahl lässt sich in Vierer Schritten bis zu einer maximalen Anzahl von 177 Modulen in der Version 40 erhöhen. In einem solchen Code lassen sich maximal 7089 numerische, 40296 alphanumerische, 2953 binäre, oder 1817 Zeichen in Kanji Einkodierung abspeichern. Es ist möglich, verschiedene Kodierungsmodi zu vermischen. [fS15b]

**Aufbau** Beim Erstellen eines QR Codes wird zunächst der Modus festgelegt und danach die Fehlerkorrektur-Stufe gewählt. Es gibt vier Stufen, welche den Reed-Solomon Algorithmus verwenden. Im kleinsten Modus können 7% der Daten und im größten 30% der

Daten wiedergewonnen werden. Danach wird die kleinstmögliche Version für die zu speichernde Datenmenge gewählt und die Anzahl der Zeichen abgespeichert. Darauf folgt das Enkodieren der Daten, sowie das Berechnen der Fehlerkorrekturwörter und das Einbinden dieser mittels verschiedener Filter. Zuletzt wird das Ergebnis binär abgespeichert, bevor mit dem Zeichen des Codes begonnen wird.

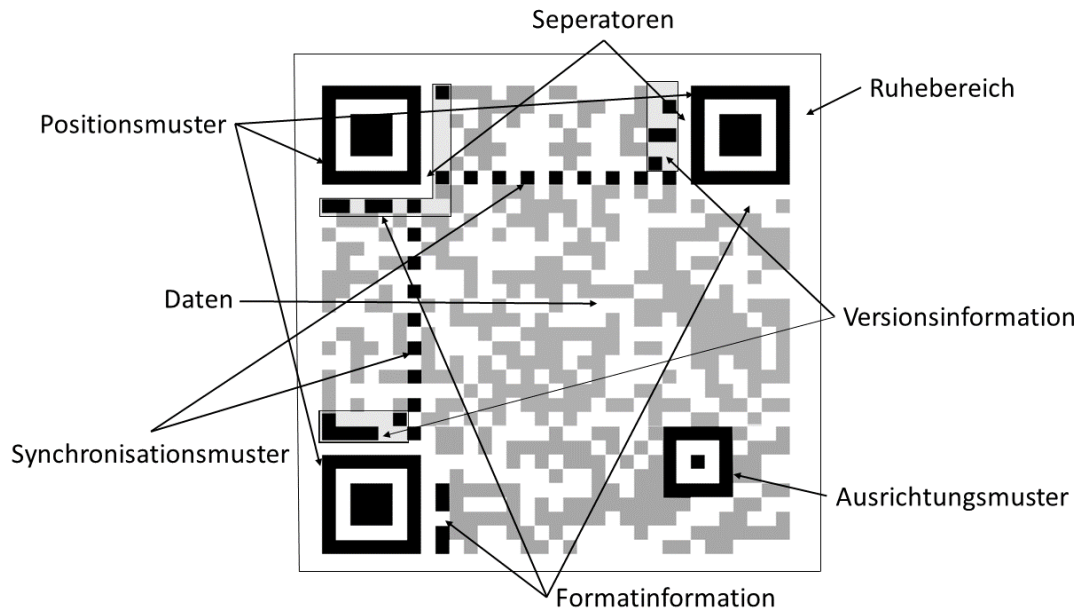


Abbildung 6: Aufbau eines QR Code

Abbildung 6 zeigt den in der Folge im Detail beschriebenen Aufbau eines QR Codes. Um ihn muss sich ein Ruhebereich befinden, damit das Lesegerät diesen sicher von der Umgebung unterscheiden kann. Dieser Bereich soll mindestens einen Abstand von vier Modulen in alle Richtungen gewährleisten. Der eigentliche Code enthält zwei verschiedene Areale. Die Function Patterns ermöglichen das fehlerfreie Auslesen der eigentlichen Encoding Region.

Die visuell auffälligen Positions-Muster sind an den beiden oberen, sowie der linken unteren Ecke des Codes angebracht. Sie dienen sowohl der Größe und Positionsbestimmung, als auch der Winkelkorrektur. Diese Muster werden aufgrund ihrer Form auch als Augen bezeichnet. Sie bestehen aus einem äußeren schwarzen Quadrat mit 7x7 Modulen, einem inneren weißen Quadrat mit 5x5 Modulen und einem durchgängig schwarzen Quadrat mit 3x3 Modulen. Scanner identifizieren QR Codes, indem sie nach dem Kontrastverhältnis 1:1:3:1:1 suchen, welches sich aus dieser Spezifikation ergibt. An den innen liegenden Kanten der Augen werden Separatoren eingesetzt, um die Erkennung zu erleichtern. [fS15b]

Im inneren des Codes werden Ausrichtungsmuster verwendet, um durch Kameralin-

sen entstandene Verzerrungen auszugleichen. Sie bestehen aus einem schwarzen Quadrat mit 5x5 Modulen, einem inneren weißen Quadrat mit 3x3 Modulen, und einem einzelnen schwarzen Modul in der Mitte. Dabei ist zu beachten, dass sie die zuvor platzierten Positions-Muster nicht überlagern dürfen. Deshalb ist ihre Anzahl variabel. Codes der Version 1 besitzen überhaupt kein solches Muster, aber größere Versionen weisen bis zu 46 auf. Dies funktioniert in der Praxis gut, da das Problem der Verzerrung zunehmend bei größeren Flächen auftritt. [fS15b]

In einem weiteren Schritt werden Synchronisationslinien angebracht, welche die Augen horizontal und vertikal verbinden. Die Linien bestehen aus einer abwechselnden Abfolge von weißen und schwarzen Modulen. Eine Überlappung mit den Ausrichtungsmustern ist unproblematisch, da sie genau dieselbe Abfolge aufweisen. Mit ihrer Hilfe werden dem Scanner die exakten Koordinaten jeder Zeile und Spalte signalisiert. [fS15b]

Nun werden die vorbereiteten Datenbits von rechts unten startend in den Enkodierungsbereich eingetragen. Dabei werden sequentiell immer zwei Module in jede Zeile geschrieben. Wird die Kante der Matrix erreicht, startet man in der entgegengesetzten Richtung links davon, die nächsten beiden Spalten zu befüllen.

Abschließend wird eine der acht verfügbaren Masken auf den Datenbereich angewandt, um das Lesen des Codes zu erleichtern. Dabei wird die Farbe von Modulen auf ihr Gegenüber invertiert, um die fertigen QR Codes so unregelmäßig wie möglich zu machen. Ausgewählt wird die Maske, welche die geringsten Störfaktoren produziert. Zuletzt wird Information zur verwendeten Maske und Fehlerkorrektur an die vorgegebene Stelle geschrieben. Bei Versionen der Größe 7 oder größer wird auch die Version in den Code geschrieben. [fS15b]

**Micro QR Code** Dies sind sehr kleine QR Codes, die auf minimalen Platzverbrauch optimiert sind. Aufgrund dessen wird nur ein Positions-Muster verwendet, und der Ruhebereich kann auf zwei Module verringert werden. Das Zeitraster wird ähnlich der Data-matrix an den Rand des Codes verschoben. Auf ein Ausrichtungsmuster wird verzichtet. Es gibt vier Versionen von 11x11 bis 17x17 Modulen, wobei die größte Variante maximal 21 numerische Zeichen bei einer Fehlerkorrektur von bis zu 25% speichert. <sup>6</sup>

**Verwendung** Unter den Matrix Codes sind die QR Codes am häufigsten im Alltag anzufinden, insbesondere aufgrund ihrer Verwendung im Marketing. Durch die Verbreitung von Smartphones und die in ihnen eingebauten Kameras besitzt nahezu jeder Mensch einen Scanner. Meist wird mit ihnen auf Webseiten verlinkt und erweiterte Produktinformation angezeigt. Außerdem ermöglichen sie das Verbinden zu Drahtlosnetzwerken, sowie das

---

<sup>6</sup><http://www.qrcode.com/en/codes/microqr.html>, zuletzt zugegriffen am 18.09.2016

Abwickeln von Zahlungen. Durch die weite Verbreitung wird die Bezeichnung des QR Codes inzwischen als Synonym für alle anderen Matrixcodes verwendet, obwohl andere Vertreter nicht die typische Form mit den drei Augen aufweisen.

Sowohl die Barcode-Erzeugung als auch das Einlesen wird durch eine Vielzahl an frei zugänglicher Software ermöglicht.

### 2.3.2 Datamatrix

Diese Technologie wurde Ende der 80er Jahre vorangetrieben, um Millionen von Teile, die zum Bau von Weltraumraketen verwendet werden, während des Herstellungsprozesses zu identifizieren und verfolgen. Der DMC Code ist mittlerweile in vielen Branchen verbreitet und wird oft mittels Lasergravur auf Werkstücke angebracht.

Wie der QR Code besteht ein Datamatrix Barcode aus schwarz-weißen Modulen, die eine Größe von 10x10 bis 144x144 haben können, wobei sie eine maximale Kapazität von 2335 alphanumerischen Zeichen speichern. Es gibt vier verschiedene Kodierungsmodi, aber keine Kanji Unterstützung.

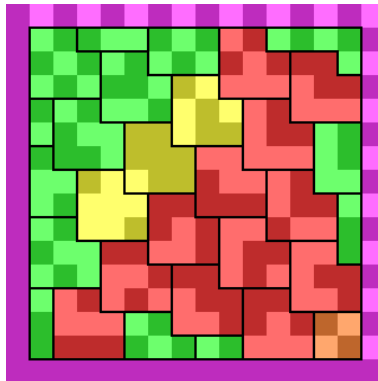
**Aufbau** Fehlerkorrektur wird im aktuellen ECC 200 Standard mittels des Reed-Solomon Verfahrens verwirklicht und ermöglicht das Wiederherstellen eines bis zu 30% beschädigten Codes. Früher wurden weniger robuste Verfahren verwendet. [fS06]

Um die DMC muss sich ein Ruherbereich von mindestens einem Modul befinden. Als Positions-Muster werden zwei aneinander liegende Kanten verwendet, deren Module durchgehend schwarz gezeichnet sind. Die anderen beiden Kanten fungieren als Synchronisationslinien und bestehen aus abwechselnd weißen und schwarzen Modulen. Alle davon eingeschlossenen Module dienen der Speicherung des Inhaltes. Ab einer Modulanzahl von 32 wird ein Ausrichtungsmuster eingesetzt, das den Datenbereich in 4, 16 oder 36 gleich große Teilbereiche trennt. Wie beim Positionsmuster wird eine paarweise Verbindung von durchgehenden und unterbrochenen Linien verwendet. [fS06]

Die eigentlichen Datenbits werden durch einen Algorithmus in einem wabenartigen Muster in die Datenfläche geschrieben, nachdem diese gemäß der gewählten Enkodierung und der angewandten Fehlerkorrektur im Binärformat vorliegen. Dieses Schema ist in Abbildung 7 zu sehen.

---

<sup>7</sup>Jonathan Laventhol, Creative Commons Zero, <https://upload.wikimedia.org/wikipedia/commons/f/fb/Datamatrix-modulecolours.png>, zuletzt zugegriffen am 20.09.2016

Abbildung 7: Data Matrix Code, Enkodierungsschema<sup>7</sup>

**Verwendung** Weit verbreitet ist der Barcode im pharmazeutischen Bereich, und in der Elektronikbranche. Dort werden Seriennummern und weiterführende Informationen an sehr kleinen Produkten angebracht. Da in solchen Fertigungsanlagen die Lesegeräte sehr genau mit der Codebeschaffenheit abgestimmt werden, sind außerordentlich kleine Codegrößen möglich.

Zunehmend findet er auch ausgedruckt in verschiedenen Ticket-Systemen Verwendung und viele Smartphone Barcode Scanner können auch DMC lesen.

### 2.3.3 PDF-417

Als Vertreter der gestapelten Barcodes besteht dieser aus aufeinander gelegten eindimensionalen Barcodes. Er weist immer mindestens drei, und maximal 90 einzelne Strichcode Zeilen auf. Aufgrund dessen hat er im Vergleich zu den Matrix Barcodes einen höheren Flächenbedarf.

Um ein korrektes Auslesen zu gewährleisten, wird eine Ruhezone benötigt. Jede Zeile beginnt mit einem Startmuster, die den Barcode als PDF417 definiert. Ein rechter Zeilenindikator speichert Information über die Zeile, wie ihr Fehlerkorrektur-Level, Spaltenanzahl und Zeilennummer. Danach folgen 1-30 Codewörter, welche 929 verschiedene Werte annehmen können. Ein Subset von 29 Codewörtern ist für spezielle Funktionen reserviert, die restlichen 900 repräsentieren je nach Enkodierungsart die Daten. Folgende Schemen stehen zur Verfügung: Text, Byte und Numerisch. [fS15a]

Die Codewörter bestehen aus Mustern mit jeweils vier weißen und schwarzen Strichen, wodurch der Barcode die 4 in seinem Namen erhält. Die 17 bezieht sich auf die Breite des Codeworts, welche 17 mal der minimal erlaubten vertikalen Zeile entspricht. [fS15a]

Nach den Codewörtern gibt es einen linken Zeilenindikator, gefolgt von einem Stop-

muster. Ein PDF417 Code wird linear gescannt und erfordert eine spezielle Vorgehensweise beim Scannen. Dadurch können auch 1D Laserscanner die Codes lesen. Er kann sehr einfach an den zur Verfügung stehenden Platz maßgeschneidert werden, da die Breite und Höhe variabel anpassbar sind.



Abbildung 8: PDF-417 Beispiel, 'XiTrust MOXIS Platzhalter'

Auch hier basiert die Fehlerkorrektur auf dem Solomon-Reed Algorithmus, dessen Fehlerkorrektur-Level stark regulierbar ist. In der höchsten Stufe können bis zu 50% beschädigte Codes gelesen werden. [fS15a]

Der Micro-PDF-417 Code ist als kompakte Variante entwickelt worden, bei dem vor allem Start- und Endmuster reduziert wurden. Der Barcode ist in der Luftfahrtindustrie weit verbreitet.

## 2.4 IKT basierte Dienstleistungen

Der Begriff der Informations- und Kommunikationstechnologien (IKT) wurde Ende der 80er Jahre geprägt, als sich eine Konvergenz der Informationstechnik, welche damals meist in riesigen Rechenzentren heimisch war, und der Kommunikationstechnik abzeichnete. Letztere war zu diesem Zeitpunkt hauptsächlich mit der Telefonie und damit verbundenen Diensten beschäftigt. Das Zusammenwachsen der beiden Felder war aufgrund der Digitalisierung unausweichlich. Ein großer Faktor war der Wunsch, mit den neuen Technologien Kosten einzusparen.

Heutzutage fasst IKT als breite Kategorie alle Produkte und Dienstleistungen zusammen, welche sich mit dem Speichern, Berechnen, Übermitteln und dem Empfangen von Daten in digitaler Form beschäftigen. Die zunehmende wirtschaftliche Bedeutung lässt sich am enormen Anstieg von drei Schlüsselindikatoren im Zeitraum von 1986 zu 2007 messen. Die folgenden Größenangaben beziehen sich auf optimale Kompression der Daten.

- Die gesamte Kapazität der Menschheit, um Informationen in digitaler Form bidirektional zu übertragen, ist von 281 Petabyte auf 65000 Petabyte gestiegen. Dies bedeutet eine Steigerungsrate von 28% im Jahr. [HL11]
- Die Kapazität der Speicherung von Informationen ist von 2,6 Exabyte auf 295 Exabyte angestiegen, was einer jährlichen Steigerung von 23% pro Jahr entspricht. [HL11]



- Die Fähigkeit, mit Allzweck Computern mittels Algorithmen Information zu berechnen, stieg jährlich um 58% auf  $6.4 * 10^{18}$  im Jahr 2007 an. [HL11]

Die Wichtigkeit der IKT lässt sich auch an ihrem Anteil an der gesamten Wertschöpfung ablesen. In einer Studie der OECD [OEC15] lag der Mittelwert aller OECD Staaten bei 5.5%, mit Südkorea an der Spitze 10,7%. Österreich liegt hier mit 3,56% unter dem Durchschnitt und weist Nachholbedarf auf.

Während die IKT in den hoch entwickelten Ländern hohe Wachstumsraten haben, gibt es in den Entwicklungsländern noch großes Wachstumspotential. Insbesondere, da viele Menschen und Organisationen noch gar keinen Zugang zu modernen Informationstechnologien haben. [Uni14]

Durch die Globalisierung wurde ein weiterer Strukturwandel angetreten, der auch die IKT erfasst. Besonders in den entwickelten Staaten tritt die Tertiarisierung auf, damit bezeichnet man den Übergang von einer Industriegesellschaft in eine Dienstleistungsgesellschaft [KMS<sup>+</sup>97]. Getragen von diesem Trend wird das Entwickeln und Anbieten von Software und Serviceleistungen stark nachgefragt.

**KIBS** Die Anbieter dieser Lösungen (im englischen 'KIBS' für 'knowledge intensive business services') sind hochspezialisierte Unternehmen, welche wissens-intensive Waren und Dienstleistungen für andere Unternehmen durchführen. In der Vergangenheit wurden sie lediglich als Umsetzer von Innovationen des Produktionssektors gesehen. Gestützt von der Forschung von Muller und Doloreux [MD<sup>+</sup>07] gelten KIBS zunehmend als eigenständige Innovationsträger und spielen eine wichtige Rolle als Innovationsauslöser in ihren Kunden. Im weiteren Verlauf der Arbeit wird der englische Begriff des Service sinngemäß mit Dienstleistungen verwendet, da sich die Bezeichnung auch im deutschsprachigen Raum etabliert hat.

Bei wissens-intensiven Dienstleistungen ist Wissen der Hauptproduktionsfaktor, und zu einem großen Teil das eigentliche Produkt, welches angeboten wird. KIBS sind durch folgende Merkmale gekennzeichnet: [SZS12]

1. Sie lernen kumulativ durch anhaltende Interaktion zwischen dem Anbieter und Benutzer
2. Probleme werden gelöst, indem die Anbieter ihr Wissen und ihre Fachkompetenz den Bedürfnissen ihrer Klienten anpassen.

Herausfordernd beim Erbringen dieser Dienste ist, sich die organisatorische Anpassungsfähigkeit an soziale, technologische und regulatorische Veränderungen zu erhalten. Auch werden hohe Anforderungen an die Qualifikation der Mitarbeiter gestellt.

Ein Grund für das Wachstum im KIBS Sektor ist neben dem technologischen Wandel der Trend zum **Outsourcing**. Dabei werden Services, welche ursprünglich intern in den jeweiligen Organisationen durchgeführt wurden, an externe Dienstleister abgegeben. Basierend auf der Philosophie, die eigenen Kernkompetenzen zu forcieren, werden externe Spezialisten bevorzugt. Laut Miles [Mil05] soll die Qualität und Effizienz der Services durch die Erfahrung mit anderen Kunden höher sein, und durch den Marktdruck die Gesamtkosten der Leistung geringer liegen als bei einer internen Lösung. Auch soll ein Wechsel von Anbietern den Unternehmen Flexibilität geben.

**MOXIS als KIBS** Eine Hürde bei jeder Marktplatzierung einer Dienstleistung zu überwinden ist es, bei den Kunden das Bedürfnis nach dieser zu wecken. Obwohl in allen Unternehmen Dokumente unterschrieben werden, ist der Wunsch nach einer Verbesserung dieses Prozesses nicht unmittelbar gegeben. Im Marketing wird daher viel auf Netzwerkbildung durch zufriedene Kunden gesetzt.

Sobald das Interesse besteht, geht es darum, den Kundenmehrwert zu quantifizieren. Das Finden von passenden Dienstleistern und deren Kosten stellen laut Miles [Mil05] vor allem bei kleinen Unternehmen eine Barriere dar. Überzeugungsarbeit ist auch zu leisten, da für den Kunden geschäftskritische und strategische Dokumente signiert werden und eine Abneigung vorhanden ist, in solchen Bereichen externe Services einzusetzen.

MOXIS wird in der IT Infrastruktur des Kunden betrieben, wodurch Sicherheit und Vertraulichkeit der Dokumente gewährleistet werden. Zugleich verursacht diese Integration Kosten und Personalaufwand, da die in den Kundenunternehmen vorhandenen IT Infrastrukturen sehr heterogen sind. Daher muss die XiTrust als Dienstleister eine breite Wissensbasis in seinen Mitarbeitern aufbauen. Außerdem muss MOXIS fortlaufend gewartet und an Veränderungen des Kunden selbst angepasst werden.

Bei Problemen, die in Kundeninstallationen auftreten, gestaltet sich der Fernzugriff auf die Host Maschine teils als schwierig, da die Computer aus externen Netzwerken nicht erreichbar sind. Das Erhalten der Zugriffsrechte geht dann aus Sicherheitsgründen mit bürokratischen Hindernissen einher.

Obwohl bei der MOXIS Entwicklung der Fokus auf eine einfache Benutzeroberfläche gelegt wird, muss auf die unterschiedliche technische Kompetenz bei den Endbenutzern Rücksicht genommen werden. Schulungen im Rahmen der Dienstleistung sind immer wichtig. Insbesondere da MOXIS eine kritische Funktion in den Prozessen des Kunden einnimmt.

Beim Entwickeln der Dienstleistung ist es schwierig, den gemeinsamen Nenner in den Bedürfnissen der potentiellen Kunden zu finden. Da nahezu jeder Kunde eigene Abläufe und Hierarchien hat, ist es notwendig, einen gelungenen Mittelweg an Kundenkonfigura-

tion und Basis Features zu schaffen. Gleichzeitig muss auf die Wirtschaftlichkeit geachtet werden, wobei jeder Aufwand ein günstiges Kosten-Nutzen Verhältnis aufweisen soll.

Um diese Risiken zu mitigieren, werden die im nächsten Abschnitt vorgestellten Innovationstechniken verwendet.

## 2.5 Innovationsprozess für IKT basierte Dienstleistungen

Durch die zunehmende Sensibilität für Dienstleistungen als Auslöser und Träger von Innovationsprozessen werden neue Anforderungen an das Management gestellt. Grundsätzlich stellt sich die Frage, was Innovation im Servicebereich eigentlich bedeutet. Außerdem soll der Einfluss von Unternehmensstrukturen auf die Fähigkeit einer Organisation, Neuerungen zu entwickeln, untersucht werden.

### 2.5.1 Service Engineering

Service Engineering wurde in Deutschland entworfen, um Unternehmen ein Instrument zu geben Dienstleistungen strategisch und operativ zu planen. Wissenschaftlich lange Zeit ignoriert, nahm das Fraunhofer Institut für Arbeitswirtschaft und Organisation im deutschsprachigen Raum eine führende Rolle ein. [BS06]

Bullinger et al. definieren den Begriff wie folgt:

*Service Engineering can be understood as a technical discipline concerned with the systematic development and design of services using suitable procedures, methods and tools.*[BS06]

Damit betrifft Service Engineering nicht nur die technische Herangehensweise an die Dienstleistungsentwicklung. Vielmehr geht es um den Interaktionsprozess zwischen Unternehmen und Kunden, der breite Anforderungen an den Service Engineer stellt. Neben ingenieurstechnischen Fertigkeiten wie strukturiertem und methodischem Vorgehen, sind laut Aschbacher je nach Entwicklungsschritt verschiedene soziale Kompetenzen gefragt. [Asc14]

Im Folgenden wird das von Bullinger et al. entworfene iterative Vorgehensmodell vorgestellt. Dabei ist das sequentielle Abarbeiten der Punkte eher zweitrangig, und es wird ein Fokus auf die Arbeitsschritte innerhalb der Phasen empfohlen. [Asc14]

**Ideensammlung und Ideenbewertung** Hier wird mit der Sammlung und der Bewertung der Ideen gestartet. Auch wird die Erstellung einer Machbarkeitsstudie vorbereitet.

**Anforderungsanalyse** In dieser Phase werden die Anforderungen der Dienstleistung an das Unternehmen selbst und den Markt ermittelt. Dabei ist der Ressourcenverbrauch entscheidend, um eine Preisvorstellung zu erhalten. Diese Erkenntnisse fließen in die Machbarkeitsstudie ein.

**Dienstleistungskonzeption** Dieser Abschnitt beginnt mit der Gestaltung des Leistungsmodells, auf dessen Basis man den Prozess plant und die Ressourcen zuweist. Außerdem wird hier ein Marketingkonzept erstellt, da die Dienstleistung von Beginn an für die Kunden umgesetzt wird.

**Dienstleistungsimplementierung** Unter der Implementierung versteht man das Setzen von organisatorischen und personellen Maßnahmen, um die Dienstleistung umzusetzen. Hierbei wird immer wieder die Zielsetzung aus der Konzeption mit dem aktuellen Status verglichen, und falls notwendig angepasst.

**Markteinführung** Zum Abschluss werden die Marketingmaßnahmen gestartet, und mit der Auslieferung begonnen. Von der ersten Minute an wird das Sammeln von Feedback forciert, um Rückmeldungen in das Produkt einpassen zu können. Bei gravierenden Änderungen muss entschieden werden, ob das bestehende Produkt angepasst, oder eine komplett neue Dienstleistung entworfen wird.

Dieses Modell wird in der XiTrust aktiv umgesetzt und durch erfolgreiche Projekte evaluiert. Aschbacher beschreibt die Flexibilität der Werkzeuge als Auswahlgrund für das Service Engineering [Asc09].

Fallen hinsichtlich Management-Entscheidungen Evaluierungsrisiken an, können diese durch die Ergänzung des iterativen Verfahrens um einen Prototyping-Schritt reduziert werden. Grob und Seufert [GS96] beschreiben den Prototypen als vorläufige und unvollständige Testversionen der Anwendung, mit der bereits frühzeitig wesentliche Merkmale des untersuchten Systems festgestellt werden können.

Abbildung 9 zeigt einen solchen Prozesszyklus, bei dem eine Testversion als Resultat der ersten drei Methoden des Service Engineering konzipiert wird. Grob und Seufert unterscheiden zwischen folgenden Arten des Prototyping: [GS96]

*Das experimentelle Prototyping versucht unbekannte Eigenschaften der Systemarchitektur zu erproben und dient zur Festlegung des Systementwurfs.*

*Beim evolutionären Prototyping entsteht das Endsystem durch die kontinuierliche Weiterentwicklung und Verbesserung der anfänglichen Testversion.*

Nach dem Abschließen der Evaluierung des Prototypen auf die verfolgte Zielsetzung wird iterativ erneut der Service Engineering Prozess verfolgt.

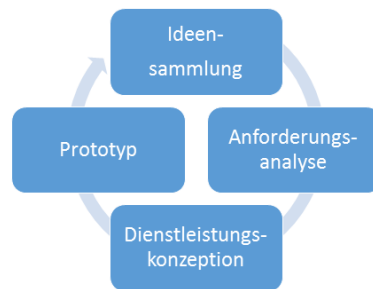


Abbildung 9: Service Engineering Vorgehen mit Evaluierungsrisiken

In Hinsicht auf die Verbesserung der Dienstleistungsqualität, sowie der Effektivität des Prozesses wird in der Literatur ein Kontinuierlicher Verbesserungsprozess (KVP) empfohlen. Insbesondere in Kombination mit dem Einsatz von Wissensmanagement im Service Engineering soll der Kundennutzen optimiert werden. [KF06]

Das Produktfeedback und Marktreaktionen sollen in den ganzen Engineering Prozess einfließen und diesen stetig verbessern. Dazu sind flexible Organisationsstrukturen nötig. Die Art des zu erfassenden Wissens unterscheidet sich bei Dienstleistungsentwicklungen erheblich von normaler Produktentwicklung [KF06]. Zum Überwinden dieser Hürde eignet sich das im Folgenden vorgestellte Konzept der Customer Co-Creation, bei der im dienstleistenden Unternehmen inhärent vorhandene Wissenslücken durch verstärkte Kommunikation mit den Kunden gefüllt werden.

### 2.5.2 Customer Co-Creation

Eine weitere Strategie, um Services zu entwickeln, stellt die Ko-Kreation mit Kunden dar. Hierbei ist der Kunde nicht nur passiver Käufer und Konsument, dessen Bedürfnisse befriedigt werden, sondern er wird aktiv in den Entwicklungsprozess miteinbezogen.

Gustafsson et. al. beschreiben Innovation als Veränderung in den Kompetenzen der Organisation selbst, in den Kompetenzen der Kunden, sowie auch den Voraussetzungen des Angebotes und dem, was der Kunde selbst ko-kreiert [VGKW12]. Als besonders wichtig werden 'value-creational processes' [VGKW12] beschrieben, die den Fokus nicht auf das eigentliche Angebot, sondern auf den Wert, den der Kunde aus diesem Produkt zieht, lenkt. Dieser Wert ist kontextspezifisch und kann nur vom Kunden selbst bewertet werden.

Gemäß dieser Definition kann alles, das einen Mehrwert für Kunden darstellt, als Service gesehen werden. Als besonders wichtig wird hier eine proaktive Marktorientierung gesehen, um den inhärenten Wert für einen Kunden zu identifizieren. Hierzu ist Kommunikation der Grundstein für eine erfolgreiche Interaktion,, bei der es um Ko-Kreation für Andere geht. Eine Rückbesinnung auf vermehrte Face-to-Face Treffen stellt sich als

effektiv heraus, da bei bi-direktionaler Kommunikation mehr über latente Bedürfnisse des Kunden herausgefunden werden kann. Direktes Feedback ist wertvoll, um in den jeweiligen Kontext einzutauchen. Je demokratischer der Austausch, desto besser sind die Ergebnisse. Übermäßig dominante Akteure wirken sich negativ auf die Ergebnisse des Entwicklungsprozesses aus. Die Kommunikationsbereitschaft soll nicht nur die Führungsebene einer Firma betreffen, sondern alle Ebenen der beteiligten Mitarbeiter. [VGKW12]

Gleichzeitig empfehlen Gustafsson et al., sich nicht ausschließlich auf die Vorschläge von Kunden zu stürzen, da diese oft zu sehr mit ihrem eigenen Kontext verwurzelt sind. Auf diesem Weg sind keine radikalen Innovationen möglich, da die Nachfrage für etwas absolut Neues unvorhersehbar ist. In diesem Sinne wird die Ko-Kreation dazu verwendet, alle Bedürfnisse des Kunden zu erfassen, um bisher verborgene Potentiale auszuschöpfen. [VGKW12]

### 2.5.3 Smart Services

Durch kundenorientierte Erhebungsmethoden wie der Customer Co-Creation erhält man eine große Menge an Daten. Man spricht von Smart Services, wenn man versucht, diese Informationen proaktiv für eine Dienstleistungserbringung zu nutzen. Im Gegensatz zum herkömmlichen reaktiven Verhalten arbeiten proaktive Dienstleister schon vor der ersten Servicenachfrage an einer Dienstleistung. [Asc14]

Präemptive Smart Services erweitern diese Definition, indem sie versuchen, sich vorbeugend auf sich abzeichnende Entwicklungen zu konzentrieren. Hier steht eine individualisierte langfristige Partnerschaft im Fokus, sowie das Ziel, den Kunden kontinuierlich bei seinem Geschäftsmodell zu unterstützen. [Asc14]

Davis et al. nähern sich dem Thema von einer anderen Richtung. Für sie sind Services „*smart*“, wenn sie ihre Informationen intelligent filtern und ihren Kunden dadurch eine effektivere Nutzung ermöglichen [DB94]. Darüber hinaus sehen sie Smart Services als interaktiv und Kunden als Lernende, wenn sie solche Dienstleistungen nutzen. Diese andauernde Lernbeziehung soll durch Kundenanpassungen Services noch „*smarter*“ machen [DB94].

Gemäß dieser Auffassung ist auch MOXIS ein Smart Service, da proaktiv Kundenbedürfnisse erhoben werden. Auch die Phase der Prozessdefinition wird entscheidend von dieser Einstellung geprägt. Die Kombination mit dem Ansatz des experimentellen Prototypings ist vielversprechend, denn der Prototyp kann dazu genutzt werden, eine proaktive Dienstleistungsidee mit geringerem Kosten- und Zeitaufwand bei potentiellen Kunden zu platzieren.

### 2.5.4 9 Fenster Methode

Diese Methode wurde erstmals in TRIZ Journal <sup>8</sup> verwendet, (im englischen 9-Window-Operator) und dient der vollständigen Analyse einer Problemsituation. Sie ist sowohl auf physische Produkte als auch auf Dienstleistungen anwendbar. Ziel ist es, die Sichtweise auf das zu untersuchende Projekt zu erweitern, und damit die Chancen für erfolgreiche Innovation zu erhöhen. Um den psychologischen Fokus auf die Gegenwart, sowie einer zu starken internen Fokussierung entgegen zu wirken, wird ein Problem strukturell und zeitlich extrahiert. Die Ergebnisse werden übersichtlich in 9 Fenstern dargestellt und dienen der Entscheidungsfindung. [Ste10]

Die Untersuchung der Dienstleistung auf Systemebene unterteilt dieses in Subsysteme, System und Supersysteme. Der aktuelle Status der Dienstleistung wird als System bezeichnet. Das Betrachten des Subsystems wird auch als Zoom-In bezeichnet. Dabei werden die Einzelteile der Dienstleistung analysiert, sowie dessen Ressourcen und deren Herkunft betrachtet. Beim Zoom-Out wird auf das Ökosystem, in dem der Service eingebettet ist, eingegangen. Im Fokus stehen die Marktsituation und die Kundschaft, für welche das System in Frage kommt. Auch der sozioökonomische Kontext spielt eine Rolle. [Ste10]

Entlang der Zeitachse wird die Dienstleistung auf ihre Vergangenheit, Gegenwart und Zukunft untersucht. Dabei sind auch Abläufe und kausale Zusammenhänge interessant. Der Betrachtungszeitraum kann individuell an die Gegebenheiten angepasst werden .

Abschließend wird aufgrund der gewonnenen Betrachtungsweisen versucht, neue Ideen zu generieren.

## 2.6 Technologieframeworks

Im Folgenden werden einige Technologien und Konzepte vorgestellt, welche für die Umsetzung des Prototypen maßgeblich sind.

### 2.6.1 OOXML

OOXML steht für Office Open XML und ist ein gezipptes XML basiertes Format für Dokumente, Kalkulationstabellen und Präsentationen. Es wurde von Microsoft entwickelt und ursprünglich nur in deren Office Produktpalette verwendet. Das Format wurde von der ISO Gruppe 2008 mit dem Standard ISO/IEC 29500 akzeptiert und 2012 weiter entwickelt [fS12]. Der Standardisierungsprozess wurde von vielen Seiten kritisiert, insbesondere,

---

<sup>8</sup><https://triz-journal.com/system-operator-tutorial-1-9-windows-world/>, zuletzt zugegriffen am 25.09.2016

da mit ODF bereits ein anderes Dokumenten-Format zur Verfügung stand<sup>9</sup>. Lizenzrechtlichen Bedenken versuchte Microsoft mit dem Open Specification Promise<sup>10</sup> entgegenzuwirken. Dies stellt aber lediglich ein Versprechen dar, Patentverletzungen nicht juristisch zu verfolgen und gibt keine Rechte ab.

Im Rahmen der Diplomarbeit wird nur das Office Open XML Document Format verwendet. Dies umfasst eine WordProcessingML Datei mit der Endung docx. Ein Dokument besteht aus einem hierarchischen Aufbau von Ordnern und Relationships. Verschiedene Metainformationen wie die Font oder verwendete Einstellungen sind in eigenen Dateien getrennt vom eigentlichen Hauptinhalt gespeichert. [fS12]

Für MOXIS relevant ist das Einfügen von Bildern. Dies wird über DrawingML Objekte realisiert, welche verschiedene Positionen, Ausrichtungen und Ausmaße aufweisen können. Nachfolgend wird ein Auszug aus der XML Struktur gezeigt, mit welcher ein Platzhalter relativ zum Text an der Cursor Position eingefügt wird. Das eigentliche Bild kann entweder als getrennte Ressource im Dateipfad der WordProcessingML Datei liegen, oder Base64-encodiert in das XML selbst eingefügt werden.

Die OOXML Spezifikation ist hierbei umfassender als die Implementation von Microsoft. So wird zum Beispiel der Parameter 'noResize', welcher das Verändern der Größe von Bildern verhindern soll, ignoriert.

Folgend wird beispielhaft ein OOXML Ausschnitt gezeigt, der einem auf einem Dokument angebrachten Platzhalter entspricht. Redundante oder nicht aufschlussreiche Elemente wurden entfernt:

```
<?xml version="1.0" encoding="utf-8"?>
<w:r>
  <w:drawing>
    <wp:anchor allowOverlap="1" behindDoc="0" distB="0" distL="114300"
      distR="114300" distT="0" layoutInCell="1" locked="0"
      relativeHeight="251659264" simplePos="0"
      wp14:anchorId="63830E2E" wp14:editId="35C3ED34">
      <wp:simplePos x="4286250" y="923925"/>
      ...
      <wp:positionH relativeFrom="character">
        <wp:posOffset>0</wp:posOffset>
      </wp:positionH>
      <wp:positionV relativeFrom="line">
```

<sup>9</sup>Google's Position on OOXML as a Proposed ISO Standard, [https://www.csun.edu/~hcmth008/odf/google\\_ooxml.pdf](https://www.csun.edu/~hcmth008/odf/google_ooxml.pdf), zuletzt zugegriffen am 02.10.2016

<sup>10</sup><https://msdn.microsoft.com/en-us/openspecifications/dn646765>, zuletzt zugegriffen am 26.09.2016



```

    <wp:posOffset>0</wp:posOffset>
  </wp:positionV>
  ...
  <a:picLocks noChangeArrowheads="1" noChangeAspect="1" noResize="1"/>
  <a:ext cx="{ImgWidthInEMU}" cy="{ImgHeightInEMU}"/>
  ...
</wp:anchor>
</w:drawing>
</w:r>

```

### 2.6.2 JavaScript API für Office

Die erstmals in den 2013er Versionen von Office eingebaute API erlaubt es, Webanwendungen zu schreiben, die mit den Office Inhalten über JavaScript interagieren <sup>11</sup>. In der aktuellen Produktpalette wurde das generelle Feature Set dieser Office.js Schnittstelle um anwendungsspezifische Funktionen, wie der Word JavaScript API und der Excel JavaScript API, erweitert.

Das Add-in wird in einer gekapselten Laufzeitumgebung betrieben, welche je nach Betriebssystem einen unterschiedlichen Webbrowser verwendet. Der Zugriff auf die Host-Anwendung wird nur indirekt gestattet. [Mic16]

Der Inhalt des Dokumentes wird über ein Modell abgebildet, welches direkt manipuliert werden kann. Dies erhöht auch die Sicherheit, da kein Teil der Office Applikation selbst exponiert werden muss [Mic16]. Zum Nutzen der Bibliothek muss das API Skript im Header der Webanwendung referenziert werden. Nach dem Laden aller Komponenten wird über eine Initialisierungsfunktion die Verbindung zum Model hergestellt. Dies passiert über einen 'Reason' Parameter, der zusätzliche Informationen über die Art der Anbindung liefert.

Die Anwendung muss nicht lokal installiert werden, da lediglich ein Manifest XML File vorliegen muss, welches das Add-in deklarativ beschreibt. In diesem wird mit dem Element „SourceLocation“ die URL des Add-ins deklariert <sup>12</sup>. Zusätzlich werden darin die von der Anwendung implementierten Add-in Commands definiert, sowie Icons und Rechte festgelegt.

Dadurch sind sie sehr einfach zu aktualisieren, da die eigentliche Anwendung sepa-

<sup>11</sup><http://dev.office.com/reference/add-ins/javascript-api-for-office>, zuletzt zugegriffen am 05.10.2016

<sup>12</sup><https://msdn.microsoft.com/de-de/library/office/fp161044.aspx>, zuletzt zugegriffen am 05.10.2016

rat wartbar und losgelöst ist. Zusätzlich können Add-ins plattformunabhängig konzipiert werden und laufen prinzipiell auf Windows, Mac oder auch in der Office 365 Cloud Anwendung. Allerdings wird zum Zeitpunkt dieser Arbeit noch nicht jede Funktionalität in jeder Anwendung unterstützt. Durch die geteilten Basisschnittstellen ist es möglich, Excel, Word und Outlook Anwendungen mit gemeinsamer Codebasis zu implementieren. Für das Verständnis der Benutzeroberfläche sind die im Folgenden beschriebenen zwei Begriffe vonnöten:

**Ribbon** Dies ist ein von Microsoft mit Office 2007 erstmals eingeführtes grafisches Bedienkonzept. Es besteht aus einem Band von kontextuell sortierten Registerkarten mit Unterbefehlen. Das Konzept soll die Zugänglichkeit gegenüber herkömmlichen hierarchischen Menüs erleichtern. [Dos10]

**Add-in Commands** Damit wird die Integration von Office Add-ins in die Ribbon Leiste bezeichnet. Es ist möglich, eine eigene Kontextgruppe mit den Add-in Befehlen zu erstellen. Abbildung 10 zeigt die MOXIS Gruppe des alten Add-ins mit ihren zwei Befehlen.

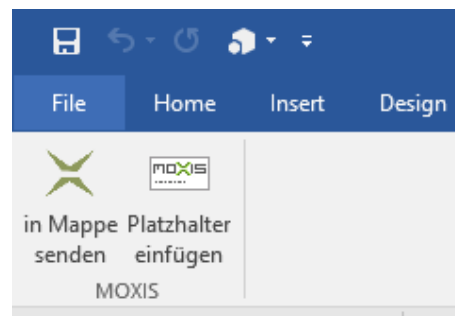


Abbildung 10: Ribbon Benutzeroberfläche

Das Word Add-in wird über Add-in Kommandos in der Ribbon Leiste gestartet, und Seite an Seite mit dem Dokument in einem Task Panel dargestellt. Mehrere Add-ins können parallel benutzt und in ihrer Größe und Anordnung verändert werden.

Das Kontext Objekt der API regelt den Zugriff auf das Dokument. Es ist möglich, Teile, sowie auch das gesamte Dokument auszulesen und zu manipulieren. Durch das asynchrone Event-Modell bleibt die Anwendung stets ansprechbar. Dasselbe Objekt stellt auch einen Mechanismus bereit, um zu erfahren, welche API Version von der Hostanwendung unterstützt wird. Auf diesem Weg ist ein Fallback-Mechanismus auf das Basisfunktionsset durch den Anwendungsentwickler durchführbar.

## Vorteile

- Alle Webtechnologien sind verwendbar
- Plattformübergreifend, ohne herkömmliche Installation
- Performant durch asynchrone Operationen

## Nachteile

- In Bezug auf Kompatibilität und zukünftige Erweiterungen vertraut man auf Versprechen von Microsoft (genauer in Unterabschnitt 5.2.1 Anmerkung Updatepolitik)
- Offline-Modus muss bei Webanwendungen in Betracht gezogen werden
- Entwicklung und Debugging nur auf Microsoft Windows Betriebssystemen komfortabel möglich

### 2.6.3 Vaadin

Vaadin ist ein Web-Framework, welches das Entwickeln von Rich Internet Applications ermöglicht<sup>13</sup>. Es ist ein Vertreter der Thin-Client Philosophie und basiert auf dem Google Web Toolkit (GWT). Dadurch können Web Anwendungen grundsätzlich ohne direkten HTML und JavaScript Code, sondern rein in Java realisiert werden. Dies ist ein Vorteil für Firmen mit vielen erfahrenen Java Entwicklern. Das Framework abstrahiert die Browser zu Server Kommunikation und die Entwicklung ist mehr an klassische GUI Frameworks wie Swing angelehnt.

Eine Vaadin Anwendung setzt sich aus den folgenden Komponenten zusammen. Die Client-Side Engine ist für das Rendern der Benutzeroberfläche zuständig und enthält keine Anwendungslogik. Sie erfasst lediglich Benutzer Inputs und andere Ereignisse und leitet diese über AJAX an den Server weiter [Grö14].

Das Server-Side Framework ist für die gesamte Anwendungslogik zuständig. Die vom Client gesendeten Events werden aufbereitet und sind über Widgets in Java Klassen repräsentiert. Alle Änderungen dieser Elemente werden von Vaadin automatisch zum Client gesendet, sodass dieser stets den aktuellen Status der Anwendung zeigt [Grö14].

## Vorteile

- Schnellere Entwicklung durch Fokus auf eine Programmiersprache
- Browser Unterstützung durch GWT gewährleistet
- Einfache Datenverknüpfung und fertige Templates
- Falls notwendig, ist es client-seitig erweiterbar

---

<sup>13</sup><https://vaadin.com/framework>, zuletzt zugegriffen am 25.09.2016

## Nachteile

- Skalierbarkeit durch Speicherbedarf pro aktiver User Session
- Komplexe moderne GUIs mit den neuesten Front-End Technologien sind aufwändig zu integrieren

### 2.6.4 BPMN & Activiti

**BPMN** Ist eine Notation für Prozesslogik, deren Ziel es ist, von allen Akteuren in Unternehmen verstanden und genutzt zu werden. Damit können dieselben Diagramme für das Design, Implementieren und Überwachen von Geschäftsprozessen verwendet werden. [Whi04]

Zum Erstellen der Diagramme stehen vier Kategorien zur Verfügung [Whi04]:

**Flow Objects** Enthält die Events, Activities und Gateways. Damit werden Ereignisse abgebildet, generelle Aktivitäten beschrieben und Zugangspunkte definiert.

**Connecting Objects** Die Flow Objects können mit drei zur Verfügung stehenden Werkzeugen verbunden werden. Zur Auswahl stehen Sequence Flow, Message Flow und Association Verbindungen.

**Swimlanes** Pools und Swimlanes dienen dazu, die am Prozess beteiligten Objekte zu gruppieren und Workflows(Arbeitsläufe) zu definieren.

**Artifacts** Diese können Daten, Gruppen oder zusätzliche textuelle Beschreibungen in Form von Kommentaren enthalten

**Activit** Ist ein in Java geschriebenes Workflow Management System. Es kann Geschäftsprozesse in der BPMN Notation erstellen und ausführen. [Rad12]

MOXIS-PE verwendet dieses System als Spring Komponente, um seine Geschäftsprozesse zu modellieren. Durch seine Ausdrucksfähigkeit kann man damit verschiedenste Arten von Abläufen darstellen, ausführen und überwachen. Aufgrund seiner relativen Einfachheit und Verbreitung wird diese Notation auch von vielen Kunden verstanden, wodurch sie ihre Prozessbeschreibung sogar selbst gestalten können.

# Kapitel 3

## Innovationsvorgang

### 3.1 Strategische Analyse

Für die Verbesserung des in Abschnitt 2.2.2 beschriebenen Platzhalters wird zunächst eine geeignete Auswahl der visuellen maschinenlesbaren Datenrepräsentation getroffen. Dazu werden die in Abschnitt 2.3 vorgestellten Barcodes hinsichtlich ihrer Eignung für das Projekt geprüft und eine Wahl getroffen.

#### 3.1.1 Analyse & Ideenfindung

Abbildung 11 zeigt die Anwendung der 9 Fenster Methode auf den MOXIS Platzhalter. Mit dieser Sichtweise lassen sich die Produktdimensionen aufspalten, und neue Ideen werden generiert.

Auffallend ist die Umgebung, in welcher die Platzhalter verwendet werden. Sie stellen einen Übergang von einzelnen getrennten Dokumenten in eine geordnete Prozessautomatisierung dar. Dabei ist es wichtig, eine Vielzahl an möglichen Subsystemen zu unterstützen, welche das Dokument vor der MOXIS Einbettung erzeugen. Hierbei ist neben dem geplanten Word2MOXIS auch ein Fokus auf mobile Systeme wie Tablets sinnvoll. Nichtsdestotrotz sollen auch papiergebundene Dokumente mittels Scan verwendbar sein.

Darüber hinaus besteht die Möglichkeit, eine vollautomatisierte Pipeline mit Dokumentenverwaltungssystemen zu erschaffen und so den Kundenmehrwert zu steigern.

In jedem Fall muss die Menge an gespeicherter und verwendeter Metainformation in den Platzhaltern erhöht werden, um sinnvolle Prozessabwicklungen zu ermöglichen. Ein weiteres Anwendungsszenario ist es, Platzhalter nicht nur für Signaturen und Freigaben zu verwenden, sondern auch als Prozesskennzeichnung. So können Dokumente unabhängig von Unterschriften erkannt, zugewiesen und verarbeitet werden.



Abbildung 11: 9 Fenster Operator für den MOXIS Platzhalter

Es soll zwei Grundarten von Platzhaltern geben. Diese sollen im Gegensatz zur Lösung im alten MOXIS (siehe Abschnitt 2.2.2) verbindlich sein und nicht länger nur einen Vorschlag darstellen.

**Personenplatzhalter** Dieser Platzhalter ist eindeutig einer elektronisch identifizierten Person zugeordnet. Nur diese soll darauf unterschreiben dürfen. Zur Unterschrift wird eine qualifizierte elektronische Signatur verwendet.

**Rollenplatzhalter** Zum einen kann dieser Platzhalter als Gruppenplatzhalter verwendet werden. Hierbei kann aus einem Set von elektronisch identifizierten Personen jeder Benutzer qualifiziert elektronisch signieren. Zum anderen kann er eine Rolle definieren. Dabei kann jeder MOXIS Benutzer, der diese Rolle inne hat, am Dokument mit einer fortgeschrittenen elektronischen Signatur der Rolle unterschreiben.

Das Anbringen der fortgeschrittenen elektronischen Signatur wird nicht von der A-Trust, sondern vom XiTrust Business Server durchgeführt. Daher ist diese Funktion nur in Kombination mit diesem als Modul erwerbbar.

### 3.1.2 Use Cases Word2MOXIS

In einem offenen Gesprächsformat wird mit allen XiTrust Mitarbeitern, insbesondere jenen mit direktem Kundenkontakt und den Product Owners, mit der Ideensammlung begonnen. Dabei werden in der Vergangenheit geäußerte Kundenwünsche und aktuelles Feedback eingearbeitet. Zugleich wird auch die bestehende Vision der Firma, von MOXIS als zentralem Ökosystem für ein Prozessverwaltungstool mit elektronischen Signaturen, eingebaut.

Als Resultat dieser Informationssammlungen werden Use-Cases definiert, von denen zwei repräsentative Beispiele im Folgenden angeführt werden. Diese Phase entspricht der Anforderungsanalyse des Service Engineering Modells zur Dienstleistungsinnovation.

Um Rücksicht auf die offen gelegten Geschäftsprozesse der beteiligten Kunden zu nehmen, werden sie im Rahmen der Diplomarbeit nicht namentlich erwähnt, sondern als Firma A, B, usw. bezeichnet.

**Standard Use Case** Dieser Fall stellt den kundenunabhängigen Standard Prozess dar, der einfache Werkzeuge für simple Workflow Modellierung zur Verfügung stellt. Es sollen sowohl leere Dokumente von Grund auf mit Platzhaltern versehen werden, als auch schon bereits vorhandene Templates, die Platzhalter enthalten, verwendet werden.

1. Öffne Word2MOXIS über Ribbon und Login
2. Wähle Standard Prozess
  - (a) Suche nach Benutzern oder Rollen
  - (b) Auswahl der Iterationsebene
  - (c) Optional: Auswahl ob Freigabe oder Signatur
  - (d) Platzhalter auf Dokument anbringen
  - (e) Starte mit neuer Suche (a), oder Abschluss der Bearbeitung (3)
3. Konvertiere Dokument zu PDF und Upload zu MOXIS, starte Standard Auftrag
4. Optional: Öffne MOXIS-Web in einem Browser Fenster

**Firma A Use Case** Dieser Fall ist auf einen Kunden zugeschnitten, der einen Prozess mit vielen benötigten Unterschriften definiert. Das zu signierende Dokument ist nach einer fixen Vorlage aufgebaut, und fällt im täglichen Betrieb des Kunden an. Es zeigt eine Übersicht von verschiedenen Produktionsparametern.

Dabei sollen die Platzhalter in einer übersichtlichen Tabellenstruktur an einer definierten Stelle angebracht werden. Um den Kunden die Positionierung zu erleichtern und

repetitive Arbeit zu ersparen, wird die gesamte Tabelle mit den Platzhaltern als OOXML Inhalt in das Dokument eingefügt.

Zusätzlich ist jeder Platzhalter mit einer vom Prozess definierten Maske versehen, die automatisch auf die Auswahl angewandt wird. Das bedeutet, dass etwa an der Position 2 nur der Prüfer signieren kann und bei Position 4 ein bestimmtes Subset der Benutzer freigeben kann.

1. Öffne Word2MOXIS über Ribbon und Login
2. Wähle kundendefinierte Prozesse
3. Suche und Auswahl von Prozess A
4. Anzeige aller geforderten Platzhalter Positionen
  - Suche und Auswahl jedes benötigten Platzhalters, mit den prozessspezifischen Subsets und Masken jeder Position
  - Anbringen der Platzhalter am Dokument in der vom Prozess definierten Tabellenstruktur
  - Abschluss der Bearbeitung
5. Konvertiere Dokument zu PDF und Upload zu MOXIS
6. Optional: Einschränkung der gefundenen Platzhalter, vor Prozess Start
7. Starte Auftrag Prozess A
8. Optional: Öffne MOXIS-Web in einem Browser Fenster

Auch hier sollen Dokumente mit bereits vorhandenen Rollenplatzhaltern erkannt und nach erfolgter Auswahl der korrekte Prozess gestartet werden.

### 3.1.3 Auswahl der Datenrepräsentation

Für die neuen Prozessplatzhalter muss mehr Information in die Platzhalter kodiert werden. Die Repräsentation soll zuverlässig und schnell gelesen werden können. Darüber hinaus ist der Platzbedarf, die Erweiterbarkeit und Ästhetik von großer Bedeutung.

Da hinsichtlich des Erzeugens und Auslesens auf etablierte Bibliotheken zurückgegriffen werden soll, ist lizenzfreie Software von großer Bedeutung. Qualität und Auswahl der zur Verfügung stehenden Programme sind stark von der Verbreitung der Barcodes abhängig und werden mitberücksichtigt. Zugleich sind bereits QR Codes in Verwendung, sodass



Tabelle 1: Gegenüberstellung: QR Code, Data Matrix &amp; PDF-417

QR Code	Data Matrix Code	PDF-417 Code
Bis zu 4296 Zeichen	Bis zu 2335 Zeichen	i.d.R. maximal 800 Zeichen
Kleiner Flächenbedarf	Kleiner Flächenbedarf	Größter Flächenbedarf
Auswählbare Fehlerkorrektur	Fixe Fehlerkorrektur	Hohe Fehlerkorrektur
Matrix Code	Matrix Code	Stapel Code
Lizenzfrei	Lizenzfrei	Lizenzfrei
Kamerascan	Kamerascan	Scan mit Laserscannern

hier bestehende Implementierungen erweitert werden können. Dies bedeutet, dass Alternativen einen besonderen Vorteil gegenüber dem QR Code aufweisen müssen, um diesen Arbeitsaufwand zu rechtfertigen.

**Gegenüberstellung** Alle drei Codes erfüllen die Voraussetzungen bezüglich des Lizenzmodelles und der Speicherkapazität. In Bezug auf den Flächenbedarf fällt der PDF-417 jedoch aus der Reihe, da er bis zu vier mal mehr Platz als ein QR Code einnimmt. Die ausgewiesenen Vorteile des gestapelten Strichcodes, wie die hohe Fehlerkorrektur und das Lesen mit verbreiteten Laserscannern sind für den MOXIS Use-Case nicht von Bedeutung. Deshalb werden sie nicht mehr zur Erwägung gezogen. Tabelle 1 listet die Merkmale der untersuchten Barcodes auf.

Im Vergleich von DMC und QR Code fällt auf, dass DMC effizienter hinsichtlich ihres Platzverbrauches sind. Vor allem in Größen mit weniger als 22 Modulen ist bei der Data Matrix der Bereich, welcher zur Datenenkodierung bereit steht, größer, da die Funktionsmuster bei QR Codes einen höheren fixen Anteil haben.

Im Folgenden bezeichnet 'M' die Kantenlänge der Codes in Modulen. So brauchen allein die drei Platzierungsmuster 147 Module ( $7 \times 7 \times 3$ ). Jedes Ausrichtungsmuster verbraucht 25 Module und das Synchronisationsmuster benötigt  $2 \times (M - 14)$  Module.

Im Gegensatz dazu startet die Datamatrix mit einem Overhead von  $4 \times M - 4$  und fügt dieselbe Anzahl bei jedem Ausrichtungsmuster ein, dass in den Code integriert wird. Der Unterschied ist vor allem bei kleinen Versionen beträchtlich und der QR Code wird erst bei mittleren Versionsgrößen effizienter als ein DMC. In der Praxis wird dies aber durch den Einfluss der Fehlerkorrektur auf die Codegröße mitigiert. Im Gegensatz zu der fixen Fehlerkorrekturrate von DMC (30%), können QR-Codes mit einer Fehlerkorrekturrate von 7% im L Modus kreierte werden. Wird die zusätzliche Redundanz nicht benötigt, muss ein DMC inklusive der Korrektur-Codewörter 23% mehr Daten speichern.

Diese Redundanz Level werden bei MOXIS nicht benötigt, da im vollständig digitalen Umfeld keine große Beschädigung zu erwarten ist. Falls ein physisch eingescanntes Dokument fehlerhaft eingescannt wird, und ein Barcode dermaßen unlesbar ist, gibt es andere

beschädigte Bereiche des Dokumentes, weshalb dieses nicht unterschrieben werden sollte.

Bezüglich der Informationsdichte ist der QR-Code weiter im Vorteil, da er im alphanumerischen Modus 5.5 Bits pro Zeichen benötigt, der DMC hingegen 6 im EDIFACT Modus.

Durch seine markanten drei Positionsmuster ist der QR Code beim Lokalisieren der Barcodes leicht im Vorteil. Gleichzeitig bieten diese auch einen Wiedererkennungswert für die Endbenutzer, da QR-Codes weiter verbreitet sind. Dies ist ein wichtiger Punkt, da der ihn enthaltende Platzhalter in externe und interne Dokumente des Kunden eingefügt wird.

Mit dem Vorteil einer bereits vorhandenen und integrierten Softwarepipeline in MOXIS, und den hier angeführten Vorteilen und Erklärungen wird der QR Code als visuelle maschinenlesbare Datenrepräsentation für die zukünftigen MOXIS Platzhalter ausgewählt.

## 3.2 Ideensteckbriefe

Im Folgenden werden die der Geschäftsführung vorgelegten Ideensteckbriefe aufgelistet. Sie sind maßgeblich an der Entscheidung zur Produktentwicklung beteiligt und stellen gleichzeitig eine Spezifikation dar.

### 3.2.1 ISB 1: Platzhalter QR-Code Kodierung

**Problembeschreibung** Es existiert bisher keine einheitliche und kundenübergreifende Informationskodierung in den Platzhaltern. Systeme wurden verschieden angepasst ausgeliefert, was die Anbindung untereinander, sowie das Aktualisieren bestehender Installationen erschwert. Des Weiteren müssen im neuen MOXIS auch die notwendigen Parameter für die Process-Engine in den Platzhaltern enthalten sein.

**Ideenbeschreibung** Eine Anpassung und Vereinheitlichung der bestehenden Kodierung und eine Erweiterung der gespeicherten Informationen im Hinblick auf die MOXIS Process Engine wird vorgeschlagen. Dazu werden zu den bisher verwendeten Feldern, der XiTrust Identifikation und dem User (oder der Rolle), weitere Felder hinzugefügt. Um das Funktionsspektrum des Standardprozesses abzudecken, werden die MOXIS ID (identifiziert eine konkrete Installation bei einem Kunden), eine Versions Nummer der Platzhalter Visualisierung und Kodierung, die Iteration (für parallele und serielle Abläufe), die Signaturqualität (Freigabe oder Signieren) sowie die Dimensionen des Platzhalters, der den QR Code enthält.

Es wird zwei Arten von Informations-Strings geben. Zum einen die **Standard Enko-**

**dierung**, bei welcher die E-Mail Adresse des Unterschreibers zur Identifikation genutzt wird, zum anderen die **Minimale Enkodierung**, bei welcher eine Zahl eindeutig einem Unterschreiber zugeordnet wird und pro MOXIS Instanz identifiziert.

Alle prozessbezogenen Daten werden Base36 enkodiert abgespeichert, da dies im alphanumerischen Modus der QR Codes mit 11 bits pro 2 Zeichen die größte Informationsdichte bringt [fS15b]. Die in diesem Modus ungenutzten erlaubten Zeichen [Space, \$, %, \*, +, -, ., /, :] werden als mit Semantik versehene Trennzeichen (Delimiter) genutzt. Dies erleichtert das Parsen der Strings, da keine positionellen regulären Ausdrücke notwendig sind und beim Erweitern des Wertebereiches eines Feldes keine Änderungen am Parser notwendig sind.

Die genaue Zuordnung wird in den Details beschrieben und resultiert darin, dass die kompletten prozessrelevanten Informationen, ohne die Identifikation des Benutzers nur 21 Zeichen benötigen. Zur Identifikation des Benutzers wird im Standardfall die E-Mail-Adresse verwendet, wodurch man aufgrund deren variabler Länge einen QR Code der Version 3 oder 4 erhält. Dieser Teil bleibt gut menschenlesbar, obwohl die vollständige Prozessinformation gegeben ist.

Für den Anwendungsfall mit geringem Platzangebot wird anstatt der E-Mail-Adresse eine Base36 enkodierte Mapping ID per User eingeführt, wodurch die Länge des gesamten Strings 25 Zeichen beträgt. Dies führt immer zu einem QR Code der Version 1 und bildet trotzdem das gesamte Prozessspektrum ab. Rollen und Workflows können entweder im Klartext oder mittels ID Mappings zugewiesen werden, wobei der Klartext wenig Vorteile bringt, da jede Rolle MOXIS spezifisch ist und intern immer mit einem Mapping umgesetzt wird.

### Chancen & Vorteile

- Automatisierung von Prozessdokumenten durch die standardisierte Enkodierung
- Kundenspezifische Form und Größenanpassungen sind einfacher umzusetzen (Standard und minimale Platzhalter verwenden dieselbe Prozessinformation)
- Vereinheitlichung der Platzhaltersyntax, um an alle Kunden dasselbe System auszuliefern
- Unterschiedliche Versionen der Syntax erlauben trotzdem Anpassung mit Rückwärtskompatibilität. Eine Erweiterung des Wertebereiches ist ebenso möglich
- Die Signaturvisualisierung ist immer auch ohne genaue Versionsinfo durchführbar, da die Größe im QR Code gespeichert ist
- Man hat bei gleichem Funktionsumfang die Auswahl zwischen besserer Menschenlesbarkeit und Kompaktheit
- MOXIS externe Personen können in der Standard Enkodierung direkt eingeladen werden, da die E-Mail-Adresse bekannt ist

### Risiken & Nachteile

- Jedes MOXIS sollte seine Versionshistorie speichern
- Spezialfälle, wie Kunden, die sich ihre QR Codes selbst generieren, müssen noch immer eingepflegt werden
- Prozessinformationen sind nur teilweise für uneingeweihte Menschen lesbar, aber die notwendigen Informationen wie z.B. für Iterationen sollen ohnehin in der Platzhalter Visualisierung (siehe 3.2.2) und nicht im QR Code ersichtlich sein

**Details** Zuerst werden die QR Code Versionen präsentiert und mit Vorschlägen zur besten Verwendungsart versehen. Nachfolgend wird die vorgeschlagene Syntax genauer definiert und Verwendungsbeispiele gezeigt.

**QR Code Versionen & Empfehlungen** Tabelle 2 listet die QR Code Versionen anhand ihrer Modulanzahl auf und zeigt, wie sich die Speicherkapazität bei verschiedenen Fehlerkorrektur Levels verändert. Die Spalte #Alph gibt die maximale Anzahl an Zeichen in alphanumerischer Kodierung an. Die Spalte #Binär gibt die maximale Anzahl an Zeichen in binärer Kodierung an.

Tabelle 2: QR Code Versionsübersicht<sup>14</sup>

Version	Module	Fehlerkorrektur	#Alph	#Binary	Geeignet für
1	21x21	L (7%)	25	17	Minimale Größe
		H (30%)	10	7	
2	25x25	L	47	32	
		H	20	14	
3	29x29	Q	77	53	Standard Größe: Normale E-Mail-Adresse Minimale Größe, Embedded Logo
		H	35	24	
4	33x33	Q	114	78	Standard Größe: Lange E-Mail-Adresse
		H	50	34	
5	37x37	Q	154	106	
		H	64	44	
6	41x41	Q	195	134	Standard Größe, Embedded Logo
		H	84	58	

**Minimale Enkodierung Beispiel** Der Informationsstring einer Beispielfirma wird, wie in Tabelle 3 spezifiziert, gebildet und ein QR-Code erzeugt. Dieser hat die Version 1 und ist damit sehr klein.

Das Resultat von "XI\$VOE.1:2/S UUXXYZZ+02A" ist in Abbildung 12 zu sehen.

<sup>14</sup>Quelle: <http://www.qrcode.com/en/about/version.html>, zuletzt zugegriffen am 02.10.2016

Tabelle 3: QR Enkodierung Minimal

Teil	Trennzeichen	Syntax	#Zeichen	Beschreibung
1		„XI“	2	Identifikation
2	\$	„\$ZZZ“	4	MOXIS ID
3	.	„.Z“	2	Version NR
4	:	„:Z“	2	Iteration
5	/	„/Z“	2	Qualität
6	Space	„ UUVVXXYY“	9	Dimensionen
7 min	+	„+ZZZ“	4	Benutzer
	-	„-ZZZ“	4	Rolle
	*	„*ZZZ“	4	Job
Gesamt			25	Zeichen im alphanum. Enkodierung



Abbildung 12: QR Code: Minimal Enkodiert

**Standard Enkodierung Beispiel** Hier wird die E-Mail Adresse als Erkennungsmerkmal verwendet, wodurch neben variabler Länge der Enkodierungsmodus gewechselt wird. Es werden dieselben Trennzeichen wie in Tabelle 3 verwendet.

Das Resultat von *„XI\$VOE.2:2/S UUXXYZZ+example.moxisuser@xitrust.com“* ist in Abbildung 13 zu sehen.



Abbildung 13: QR Code: Standard Enkodiert

**Vergrößerung der Fehlerkorrektur** Desweiteren besteht die Möglichkeit, die Fehlerkorrektur auf das Level H anzuheben, wodurch man ein Firmenlogo in den QR Code einbringen kann.

**Enkodierung der Dimension** Um die Signaturvisualisierung nach dem Unterschreiben korrekt über den Platzhalter legen zu können, ist es notwendig, dessen Ab-

messungen in den QR Code zu speichern. Die QR-Code Erkennungssoftware kann dessen Position auf dem Dokument bestimmen und anhand der Modulgröße die Distanz zu den Platzhalterkanten ermitteln. Da 8 Zeichen im Enkodierungsschema für die Dimension zur Verfügung stehen, kann man in Base36 Enkodierung 1295 Modulschritte in jede Richtung gehen. Dies ist für die meisten Anwendungszwecke ausreichend, außer der Platzhalter ist um ein Vielfaches größer als der Barcode selbst.

z.B. "2X0F0K0F" definiert den Abstand zur linken Kante des Platzhalters, welche 105 Module von der linken Kante des QR Codes entfernt ist. Analog dazu beträgt der Abstand unten 15, rechts 20 und oben 15 Module.

Eine andere Möglichkeit ist es, den Mittelpunkt des Codes relativ im Koordinatensystem des Platzhalters festzuhalten. Ein QR-Code, der ähnlich wie im oberen Beispiel vertikal mittig und horizontal weit rechts im Platzhalter angebracht ist, kann mittels 0.5 & 0.85 im zwischen 0 und 1 aufgespannten Koordinatensystems des Platzhalters beschrieben werden. Diesen Bereich kann man wieder in Base36 aufteilen, sodass man HZ & RT an die ersten vier Stellen des Strings schreiben und somit dem Mittelpunkt des QR Codes festhalten würde. Die Breite und Höhe des QR Codes kann ebenfalls im selben Wertebereich angegeben werden.

### 3.2.2 ISB 2: Platzhalter Visualisierung

**Problembeschreibung** Mit der Einführung der MOXIS Process Engine reicht das bisherige Design der Platzhalter nicht mehr aus, um dem Benutzer ausreichend über den Typ und die Bedeutung des Platzhalters zu informieren. Um den gegebenen Platz ideal auszunutzen, sowie auf Kundenwünsche in Bezug auf Größe und Branding reagieren zu können, sollte zudem nicht alles als Volltext in die Platzhalter geschrieben werden.

**Ideenbeschreibung** Das bisherige Standarddesign, bestehend aus einem Firmenlogo, dem Beschreibungstext und dem QR Code soll um weitere Merkmale ergänzt werden. Dies sollte mit aussagekräftigen Icons realisiert werden, um die Funktion klar zu kommunizieren. Die Spezifikation verwendet Icons von Font Awesome.<sup>15</sup>

Bis auf den QR Code, der gemäß Unterabschnitt 3.2.1 enkodiert ist und prozessrelevante Infos enthält, sollen alle Bestandteile, je nach Kundenanforderung und gewählter Vorlage, entfernbar sein.

**Platzhalter Spezifikation** Diese Icons beschreiben auf den ersten Blick, ob auf diesem Platzhalter eine Person (siehe Abbildung 14) oder eine Rolle (siehe Abbildung 15)

---

<sup>15</sup><http://fontawesome.io/>, zuletzt zugegriffen am 02.10.2016

vorgesehen ist. Auch Prozessplatzhalter können so visuell unterschieden werden (siehe Abbildung 16).



Abbildung 14: Benutzer Icon

Abbildung 15: Rollen Icon

Abbildung 16: Job Icon

**Signatur Qualität** Eine Freigabe wird durch ein „F“, eine qualifizierte Unterschrift durch ein „U“ oder „Q“ vor dem Beschreibungstext gekennzeichnet. Alternativ kann auch ein passendes Icon eingesetzt werden.

**Iterationsebene** Hier sollte ein Icon in Kombination mit einer Zahl benutzt werden, welche die geforderte Iterationsebene angibt (siehe Abbildung 17).



Abbildung 17: Iteration Icon

Zudem kann die bisherige Standardgröße und Kodierungsart um weitere Vorlagen erweitert werden, um oft anfallende Kundenwünsche abzudecken. Die folgenden drei Beispiele stellen Vorschläge für öfter verwendete Templates dar:

**Großer Platzhalter** Enthält alle oben angeführten Komponenten, inklusive dem Branding des Kunden oder der XiTrust.



Abbildung 18: Großer Platzhalter, neue Version



Abbildung 19: Kleiner Platzhalter, neue Version

**Kleiner Platzhalter** Dies ist ein um den Beschreibungstext und Firmenlogo reduzierter großer Platzhalter

**Winziger Platzhalter** Dieser ist auf den den kleinsten Platzverbrauch optimiert und besteht nur aus dem QR Code. Siehe Abbildung 12 oder Abbildung 13. Dieser erfordert auch eine Anpassung der Signaturvisualisierung auf diesen Formfaktor.

#### Chancen & Vorteile

- MOXIS Benutzer bekommen die benötigte Information über den Inhalt der Platzhalter
- Entwurf eines einheitlichen Designs & Vorlagen passend für die meisten Kunden
- Weitere Kundenanpassung ist trotzdem möglich, da sie in Versionen archiviert werden
- Branding Möglichkeiten sowohl für die Kunden als auch die Firma selbst

#### Risiken & Nachteile

- Dem Kunden muss die Bedeutung der Icons in der Anwendung selbst erklärt werden

### 3.2.3 ISB 3: MOXIS Standard Prozess Editor

**Problembeschreibung** Bisher existiert kein genereller Standard Prozess, sondern nur kundenspezifische Prozesse. Ausgehend vom zurzeit umfassendsten Prozess in der MOXIS Process-Engine, dem Kundenprozess A, ist es sinnvoll, einen kundenunabhängigen Prozess zu erstellen, der dann als Demonstrator genutzt werden kann.

Darüber hinaus sollte ein Standard Prozess angestrebt werden, der für Kundenwünsche nur konfiguriert werden muss und nicht mehr Stück für Stück erweitert werden muss. Auch der bisherige Kundenprozess A ist dann eine Ableitung des MOXIS Standard Prozesses und nutzt dessen Logik und Platzhalter. Dadurch kann eine MOXIS Instanz grundsätzlich ein Dokument einer anderen Installation funktional verarbeiten, ohne den genauen Prozess und seine Definition zu kennen.



Stößt man beim Scannen des Dokumentes auf Personen oder Gruppen, die nicht bekannt sind, soll der Auftraggeber diese Information eingeben.

**Ideenbeschreibung** Im Folgenden sind Vorschläge für die Erweiterung des Featuresets einzeln aufgelistet:

**Iterationen** Aufheben der bisher festen Verknüpfung der Freigeber oder Unterschreiber mit einer Iterationsebene. Beide Signaturqualitäten sollen in jeglicher Iterationsebene, auch in Kombination mit Gruppenplatzhaltern möglich sein. Dies erfordert eine umgesetzte Anpassung der QR Code Encodierung (siehe Unterabschnitt 3.2.1), sowie der Visualisierung (siehe Unterabschnitt 3.2.2).

**Gruppen & Rollen** Zurzeit werden bei jedem Prozessesstart ad-hoc Rollen vom Auftraggeber zugewiesen. Diese Möglichkeit sollte um permanente Gruppen erweitert werden, die dann von Kunden (mit Administrator Rechten) selbst erstellt und verwaltet werden können. Das bringt den Kunden auf lange Sicht Zeitvorteile. Zum Beispiel müssen nicht jedes Mal vier Manager selbst gesucht und ausgewählt werden, um sie einer Rolle zuzuweisen. Gleichzeitig sollten ad-hoc Rollen weiterhin verfügbar bleiben, denn sie ermöglichen das Bereitstellen von generellen Templates, die dann beim Auftragsstart vervollständigt werden.

**Zeit** Es soll möglich sein, Zeitfenster sowohl für den gesamten Auftrag, als auch für einzelne Iterationsebenen einzustellen. Damit verbunden ist das Erstellen von Fail-Zuständen, die Abbruchbedingungen von Iterationsebenen regeln. Das Einstellen dieser Optionen soll primär im Backend geschehen, kann aber falls gewünscht auch beim Anlegen eines Auftrags konfigurierbar sein. Diese Änderung erfordert keine Anpassung der Platzhalter.

**Unsichtbar Signieren und Freigeben** Es soll die Möglichkeit geschaffen werden, ganz ohne Platzhalter und Signaturvisualisierung Prozesse zu starten, welche die korrekten Unterschriften- und Freigabevorgänge auslösen.

### Chancen & Vorteile

- Zukünftige Anpassungen sind weniger zeitintensiv
- Klarere Platzierung eines Kernproduktes am Markt
- Die Features sind teilweise schon jetzt von Kunden gewünscht, oder in alten Projekten inkludiert
- Stellen ersten Schritt zu MOXIS instanzübergreifenden Features dar

### Risiken & Nachteile

- Initialer Entwicklungsaufwand ist hoch
- Es besteht die Gefahr, am Markt „vorbei zu entwickeln“

#### 3.2.4 ISB 4: Word Add-In

Dieser Ideensteckbrief ist bereits durch das aus dem ersten Prototypen gewonnene Feedback verbessert worden (siehe Unterabschnitt 4.1.1). Um unnötige Informationsduplizierung im Rahmen dieser Arbeit zu vermeiden und die Übersichtlichkeit zu wahren, wird hier die finale Version vorgestellt.

Das Vorgehen entspricht dem iterativen experimentellem Prototyping, da hier nach der Analyse des Prototypen wieder mit einer Konzeption und erneuter Dienstleistungsimplimentation begonnen wird.

**Problembeschreibung** Das zurzeit verwendete VSTO Word2MOXIS Add-in benötigt ein Installationspaket, welches die Platzhalter mitliefert und auf dem System bereit stellt. Diese Abhängigkeit limitiert Anpassungen, da immer dafür gesorgt werden muss, dass alle Benutzer die aktualisierten Platzhalter installiert bekommen.

Darüber hinaus ist die Benutzeroberfläche nicht mehr zeitgemäß, da sie aus bis zu zwei Pop-Up Fenstern übereinander besteht und kein zeitgleiches Bearbeiten des Dokuments erlaubt. Auch ist diese UI nicht für moderne breite Office Monitore optimiert und lässt viel Platz ungenutzt.

Daher ist das Add-In mit den bisher in Unterabschnitt 3.2.1, Unterabschnitt 3.2.2 und Unterabschnitt 3.2.3 vorgeschlagenen Änderungen inkompatibel, da diese den Benutzern dynamische Inhalte liefern und adaptiert werden.

**Ideenbeschreibung** Word2MOXIS soll von Grund auf neu entwickelt werden. Dabei werden die von Microsoft entwickelten Office.js und Word.js Schnittstellen verwendet und soweit möglich in die MOXIS Dienstleistung eingebaut.

Ziel ist es, die Benutzerführung zu vereinheitlichen, sowie auch das Design und die Logik der Webseite wiederzuverwenden, um bei neuen Prozesseinführungen in Zukunft den Arbeitsaufwand zu minimieren. Da das Add-in ein normales Servlet ist, bleibt es immer aktuell und bildet Veränderungen der Benutzer, Rollen und Prozesse, sowie deren zugeordneten Platzhaltern in Echtzeit ab. Ein weiteres Ausliefern der Updates wird hiermit obsolet.

Verschiedene Prozesse sollen außerdem über eigene Ribbon-Kommandos gestartet werden. Diese können zukünftig für eine Generierung von Templates im Rahmen der Prozessparameter genutzt werden. Anstatt nur die Platzhalter für einen Urlaubsantrag zu laden, könnte so das gesamte Dokument inklusive Platzhalter als Ganzes generiert werden.

### Chancen & Vorteile

- Die Installation beschränkt sich auf das Veröffentlichen eines XML Deskriptors
- Das Updaten und Erweitern des Systems wird extrem erleichtert
- Eine Veröffentlichung im Office Store ist möglich, und erhöht eventuell die Sichtbarkeit der Dienstleistung
- Side-by-Side UI anstatt von Pop-Up Fenstern
- Wiederverwendung und Einbindung des bestehenden Systems
- Theoretisch kann MOXIS vollständig von Word aus bedient werden
- Mittels Login sind benutzerspezifische Anpassungen der sichtbaren Prozesse und Platzhalter umsetzbar
- Das Einfügen von beliebigen Word Strukturen ist möglich, z.B. Unterschriften Platzhalter in Tabellenform einfügen, Anhängen eines formatierten Unterschriften-Blattes
- Ganze Formatvorlagen reduzieren den Verwaltungsaufwand

### Risiken & Nachteile

- Die Unterstützung beschränkt sich auf Office 2013/ 2016
- Teilweise vertraut man beim Funktionsumfang auf Versprechen von Microsoft
- Vorerst ist kein Offline-Modus des Add-ins möglich, wobei man MOXIS sowieso nicht offline benutzen kann
- Sicherheitsaspekt, da die Add-in Laufzeitumgebung auf Windows noch immer den Internet Explorer einsetzt
- Ein umfassendes Testen ist schwierig, da die Kombination von eingesetzter Word- und Betriebssystemversion eine Rolle spielen kann

## 3.3 Bewertung und Festlegung der Technologie

Im Rahmen der Dienstleistungskonzeption wurden alle vier Ideensteckbriefe als relevant eingestuft, um die Dienstleistung zu innovieren. Die ersten drei Ideen stellen allgemeine Verbesserungen von MOXIS dar, welche alle Bereiche des Produktes betreffen. Besonders die vierte Idee der kompletten Neuentwicklung des Word Add-ins fand unternehmensweit Zustimmung. Dieses erleichtert das Ausliefern und spart Arbeitszeit der Entwickler ein, da nicht erst ein Installationspaket erstellt und ausgerollt werden muss.

Zugleich ist der Service günstig zu vermarkten, da das Add-in nun keine statische Anwendung mehr darstellt. Im Gegensatz zu früher stehen Veränderungen, wie zum Beispiel das Anlegen von neuen Benutzern, sofort im Add-in zur Verfügung.

Bei der Zuweisung der Ressourcen des XiTrust Entwicklungs-Teams wird beschlossen, die Platzhalter Kodierung sowie ihre Visualisierung an das Kernteam zu übergeben. Die notwendigen Änderungen und der Einbau in das neu entwickelte MOXIS mit der Process-Engine legen den Grundstein für die anderen beiden Ideen. Zuerst werden die Platzhaltererzeugung und der Scan geändert. Die Übergabe an das XiTrust Team stellt sich als effizient heraus, da der Aufwand einer vollständigen Integration den für diese Diplomarbeit zur Verfügung stehenden Zeitrahmen überschreitet.

Da der in Unterabschnitt 3.2.3 vorgeschlagene Standardprozess noch nicht fertig umgesetzt ist, wird die Arbeit am Prototypen im Hinblick auf den aktuell verfügbaren Kundenprozess fortgesetzt.

Um die Umsetzbarkeit und Praktikabilität der Word JavaScript API abzuschätzen, wird ein in der Funktionalität experimenteller statischer Prototyp entwickelt. Dieser hat die folgende Ziele:

- Testen der für Word2MOXIS relevanten Funktionen der Office.js Umgebung
- Design Mockup der Benutzeroberfläche
- Die Reaktionszeiten der Anwendung sollen überprüft werden
- Endgültige Technologieauswahl

Die Add-in Laufzeitumgebung lehnt aus Sicherheitsgründen das Laden von ungesicherten HTTP Ressourcen ab [Mic16]. Deshalb muss der prototypische Webservice über HTTPS Verschlüsselung verfügen. Es werden neben HTML, JavaScript und einer QR Code Bibliothek die folgenden Technologien eingesetzt:

**REST** Wird zum Übertragen der in OOXML gespeicherten Platzhalter über AJAX Aufrufe verwendet

**jQuery** Für schnellen Zugriff auf Elemente, sowie kürzeren JavaScript Code. Darüber hinaus sind die AJAX Call Wrapper sehr praktisch. Außerdem werden weitere Libraries wie **Masonry** (Responsive Grid Layout von Bildern) und **Image Picker** (Für die Platzhalter Selektion) verwendet

**Django** Wird als Webframework ausgewählt, um die Entwicklungszeit kurz zu halten, da der Autor mit Python erfahren ist. Gute Unterstützung von REST Services, daher besonders geeignet

**SCSS** Um schnell und übersichtlichen CSS Code zu produzieren. Weitere Bibliotheken für das visuelle Design sind **Pure CSS** und **Font Awesome**

Mittels eines experimentellen ersten Prototypen soll das Add-in Konzept evaluiert und präsentiert werden. Daraus gewonnene Erkenntnisse, insbesondere hinsichtlich technologischer Risiken, sollen die weitere Entscheidungsfindung leiten. Bei Erfolg soll dieser evolutionär zur fertigen Dienstleistungsimplementierung weiterentwickelt werden. Bei gravierenden Problemen soll eine Veränderung der Konzeption erfolgen.

## Kapitel 4

# Praktische Umsetzung

Die bereits erfolgten Arbeiten über das Aussehen und die Funktion der Platzhalter (siehe Unterabschnitt 3.2.2 und Unterabschnitt 3.2.2) im neuen MOXIS, sowie die Integration von Prozessmodellen mit der Prozess-Engine, bieten die Grundlage für die folgende praktische Umsetzung. Prototyp 1 erfüllt den Zweck einer Machbarkeitsstudie und entspricht daher im Service Engineering Modell einer Kombination aus Anforderungsanalyse und Dienstleistungskonzeption. Der Prototyp hat den Zweck, ein Feedback über die neue Add-in Umgebung sowohl von der Geschäftsführung als auch von interessierten Kunden zu bekommen.

Aufbauend auf diesen Rückmeldungen soll der zweite Prototyp eng mit MOXIS verbunden sein und ein nahezu fertiges Produkt darstellen, welches nur mehr an die Anforderungen des jeweiligen Kunden angepasst werden muss.

Die Umsetzung der Prototypen wird von folgender in der Einleitung gestellten Forschungsfrage angestoßen:

*Wie kann man das Modul Word2MOXIS mit Hilfe des neuen Unterschriftenplatzhalterkonzeptes verbessern?*

Dabei ist das Erlebnis des Benutzers von zentraler Bedeutung. Um das Add-in homogen in die Word Benutzeroberfläche zu integrieren, muss es an die vorgegebenen Anforderungen hinsichtlich des Platzangebotes angepasst werden. Ein nützliches Werkzeug, um dies sicherzustellen, sind die von Microsoft veröffentlichten "Design guidelines for Office Add-ins"<sup>16</sup>. Daraus abgeleitet werden folgende Design Prinzipien umgesetzt:

---

<sup>16</sup><http://dev.office.com/docs/add-ins/design/add-in-design>, zuletzt zugegriffen am 30.09.2016

**Benutzerkontrolle** Benutzer sollen Entscheidungen verstehen, und rückgängig machen können. Hierzu wird eine Navigationsleiste unter dem Add-in Titel eingefügt, welche die Hauptnavigation enthält. Wird in aktuellen Office Versionen von Add-in Commands ersetzt

**Explizit für Office Entwickelt** Nutzen der Ribbon Integration, zur Office UI passender Stil

**Steuerungsmethoden** Hauptsächlich für Desktopbereich entwickelt, aber trotzdem werden die Elemente groß genug gehalten, um auch Touch Bedienung zu ermöglichen

**Inhaltsfokussierung** Effiziente Nutzung des begrenzten Platzangebots durch ein Responsive Design, welches den vom Benutzer gegebenen Platz ausfüllt. Kein überflüssiger Inhalt

## 4.1 Prototyp 1

Nachfolgend wird der erste Prototyp, der in Abbildung 20 im Detail präsentiert wird, vorgestellt. Die Benutzeroberfläche und der Bedienfluss wird anhand der zuvor ermittelten Kriterien erörtert. Aus den gezeigten Screenshots wird ein eingebettetes Kunden-Firmenlogo entfernt und mit dem XiTrust Logo ersetzt. Zum Zeitpunkt der Implementierung werden auch verschiedene Platzhalter Visualisierungen evaluiert.

Das Firmenlogo dient als Home Button, um zum Startfenster zurück zu kommen. Über den Reiter „Simple Placeholder“ wird der Standardprozess gestartet, wie in Abbildung 21 zu sehen ist. Das Suchfenster ermöglicht es Personen und Rollen zu suchen. Darunter werden die passenden Platzhalter angezeigt, wobei über Drop-Down Menüs die Größe (Standard, Minimal, QR Code Only - siehe Unterabschnitt 3.2.2) und die Signaturqualität (Freigabe oder Unterschrift) bestimmt werden kann. Zusätzlich kann auch mittels der Ebenenanzeige eine Iterationsebene zugewiesen werden.

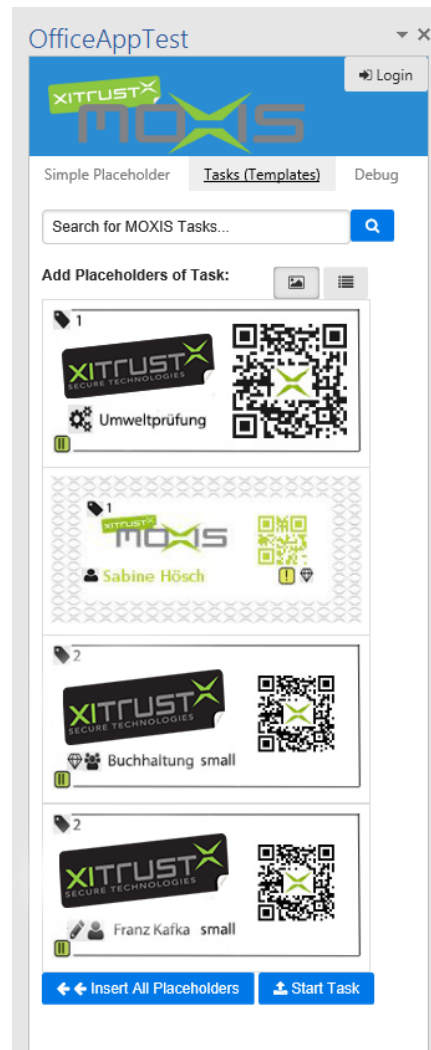


Abbildung 20: Prototyp 1, Kundenprozess A

Unterhalb befindet sich ein Button, mit dem der ausgewählte Platzhalter im Dokument an der aktuellen Cursorposition eingefügt wird. Neben dem Upload Button konvertiert das Dokument in das PDF Format, und lädt dies zum prototypischen Webservice hoch. Danach kann der Benutzer entscheiden, ob er einen neuen Auftrag starten oder MOXIS-Web in einem Browserfenster öffnen möchte.

Abbildung 20 zeigt den Prozess Reiter, der dem in Abschnitt 3.1.2 skizzierten Kundenprozess A nachempfunden ist. Hier kann nach Platzhaltern im Rahmen der Prozessdefinition gesucht werden, welche dann eine Liste aller benötigten Platzhalter liefert. Die Suchfunktion ist im Rahmen des ersten Prototypen nicht implementiert. Danach werden alle Platzhalter in der vom Prozess definierten Form eingefügt. Abbildung 22 zeigt das Resultat der Operation mit der gewünschten Tabellen-Platzhalter Struktur.

#### 4.1.1 Evaluation

Die Rezeption des experimentellen Prototypen bei einer Präsentation ist durchwegs positiv, weswegen die Entwicklung einer voll integrierten Dienstleistung begonnen wird. Er zeigt deutlich, dass die neue Task Panel UI (siehe Abbildung 5) der alten Pop-up UI von der Benutzerfreundlichkeit deutlich überlegen und die geforderte Funktionalität durchführbar ist. Zugleich erwiesen sich die nachfolgend angeführten Teile des ersten Prototypen als nicht optimal, weswegen ein Technologiewechsel forciert wird:

**REST Integration** Die umgesetzte Lösung setzt auf ein REST Service als Datenquelle. In Zukunft setzt MOXIS sehr stark auf Konfigurierbarkeit mit vielen kundenangepassten Prozessen. In diesem Fall muss jede Add-in Prozessseite einzeln erstellt werden, oder serverseitig eine generische Lösung implementiert werden, welche aus Prozessdefinitionen dynamisch eine visuelle Repräsentation erstellt. Zugleich muss auch der Platzhalter Filter und die OOXML Spezifikation geladen werden.

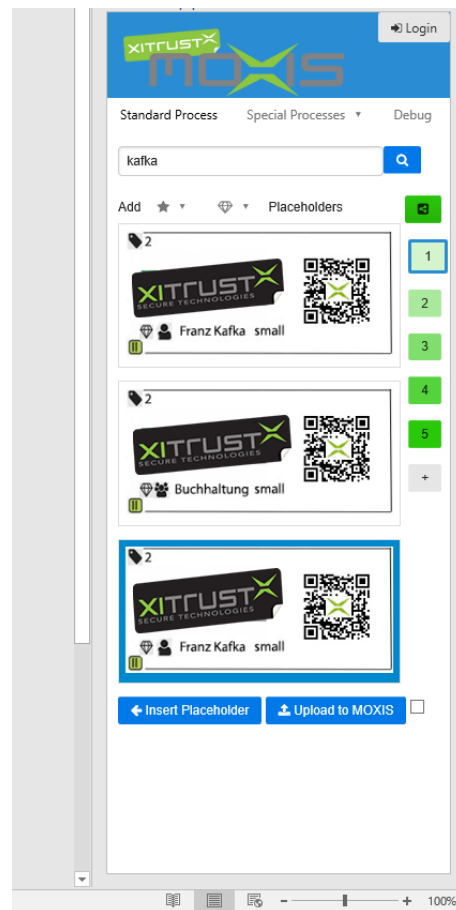


Abbildung 21: Prototyp 1, Standard Prozess



**Kompetenzen der Firma** Der vorherige Punkt resultiert entweder in sofortig hohem Zeitaufwand, oder in immer wiederkehrendem Aufwand bei jeder Auslieferung. Zugleich ist zurzeit kein Front-End Web Technologiefokus bei der XiTrust vorhanden, um vollständig auf zielgerichtete Frameworks wie React oder Angular umzusteigen.

Die finale Implementierung wird aufgrund dieser Risiken erneut als zweiter Prototyp bezeichnet, da bei Nicht-Kompatibilität der Parameter ein neuer Entwicklungsprozess gestartet wird. Die folgende Auflistung zeigt erneut die positiven Aspekte, welche bei der erfolgten Evaluierung des ersten Prototypen ausschlaggebend für die positive Entscheidung bezüglich des Office Add-ins waren.

**Installation & Einrichtung** Das Installieren und Verwenden des Add-in ist wie erwartet simpel. Die Einfachheit des zukünftigen Update Vorgangs ist erstrebenswert.

**JavaScript Wiederverwendung** Die für den Prototypen entwickelten Funktionen zum Einfügen der Platzhalter und Generieren des PDFs können recycelt werden.

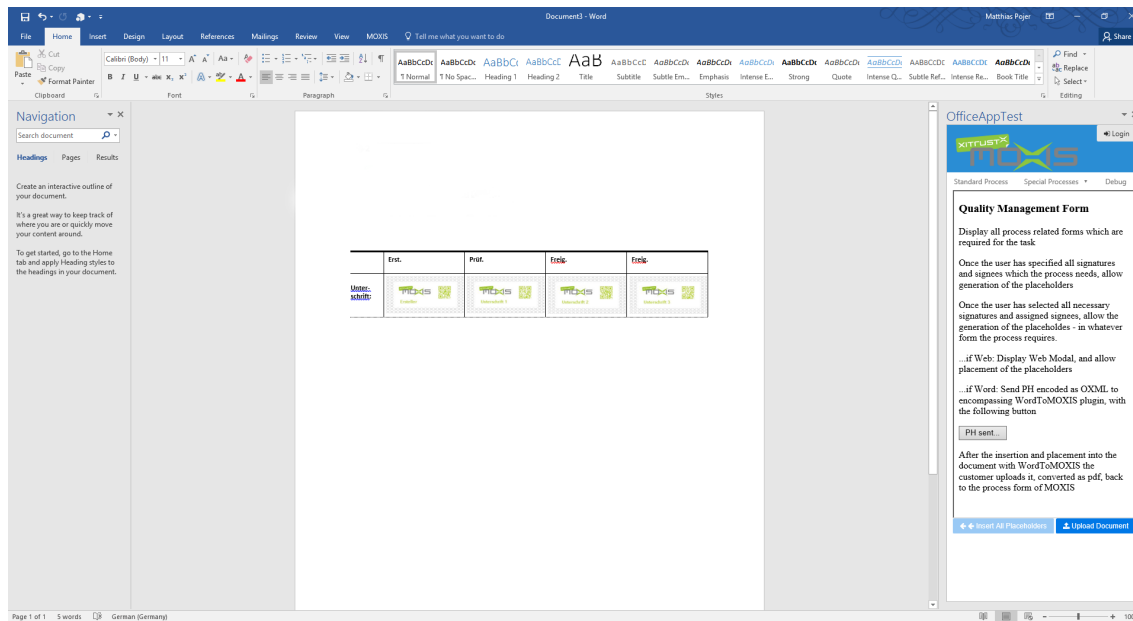


Abbildung 22: Prototyp 1, Kundenprozess A mit in einem Schritt eingefügter Tabellen-Platzhalter Struktur

## 4.2 Prototyp 2

Dieser finale Prototyp hat eine vollständige Integration in die MOXIS PE zum Ziel. Um die beim ersten Prototypen dokumentierten technologischen Nachteile zu umgehen wird ein neuer Ansatz gewählt. Das Bestreben ist es, soviel wie möglich der von MOXIS PE verwendeten Auftragsanlegeseiten für das Word Add-in wiederzuverwenden, um den Implementationsaufwand gering zu halten. Als positiver Nebeneffekt vereinheitlicht sich das Design und die Benutzerführung und macht es für den Kunden einfacher, die Anwendung zu verwenden.

Prinzipiell ist es möglich, das gesamte MOXIS aus Word zu bedienen, da ein Add-in jegliche Webseite sein kann. Schlussendlich ist dies aber nicht zielführend, da das Platzangebot begrenzt ist, und viele von der Weboberfläche angebotenen Funktionen im Texteditor nicht sinnvoll sind. Dies widerspricht den in Kapitel 4 beschriebenen Design Prinzipien.

Getragen durch das von Vaadin strikt forcierte komponentenbasierte Anwendungsdesign ist es aber möglich, einzelne Komponenten, die bereits für MOXIS Web entwickelt wurden, in eine Webseite für das Add-in einzufügen. Zum Beispiel kann das Suchfenster (siehe Abbildung 25), mit dem Unterschreiber eingeladen werden, samt der dazugehörigen Filter-, Anzeige- und Platzhalter Erzeugungslogik eingesetzt werden. Im Falle nötiger kundenspezifischer Anpassungen können die bestehenden Komponenten der Benutzeroberfläche kombiniert und abgeleitet werden.

Das im Rahmen dieser Arbeit entwickelte Add-in wird ein generelles Gerüst darstellen, und im vorliegenden Fall einen Kundenprozess implementieren.

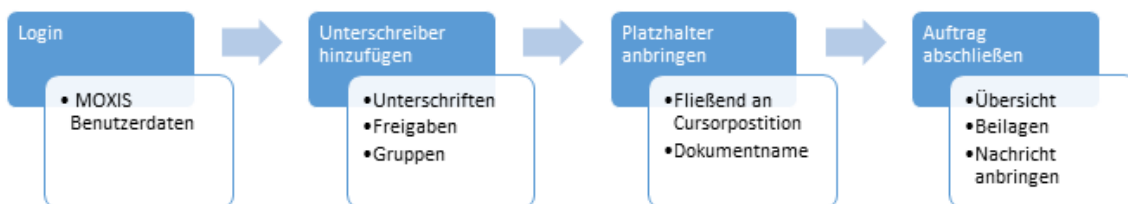


Abbildung 23: Prototyp 2, Benutzerführung Kundenprozess, genauer in Unterabschnitt 4.2.2

### 4.2.1 Design der Anwendung

Folgend wird die Zusammensetzung des neuen Add-in Prototypen gezeigt. Dabei wird jede Entscheidung begründet, und auf die Vor- und Nachteile eingegangen.

**Vaadin als Add-in** Durch die Entscheidung, Word2MOXIS mit Vaadin Komponenten zu gestalten, stellt das Add-in lediglich ein weiteres Servlet im Produkt dar. Damit muss kein separates Webservice gehostet werden, was die Verwaltung und das Updaten der Anwendung erleichtert. Potentiell kann jedes MOXIS Feature in das Word Add-in integriert werden. Überdies steht die gesamte in Vaadin integrierte Funktionalität der Benutzerautorisierung zur Verfügung.

Dabei werden die bereits implementierten Vaadin Components und Views abgeleitet, und nicht benötigte Teile entfernt und anders angeordnet. Zusätzlich werden das CSS Styling und die Layouts verändert und so an die Word Umgebung angepasst. In einem letzten Schritt wird die neu benötigte Funktionalität über die Office.js Schnittstelle eingebaut (wird in Abschnitt 4.2.1 genauer beschrieben).

#### Vorteile

- Nutzung der breiten Java & Vaadin Wissensbasis im Unternehmen
- Monolithisches System erleichtert Ausrollen, Wartung und Updaten
- Zeitersparnis in Entwicklung durch Code Re-Use
- Verwendung von getesteten Komponenten & JavaScript des ersten Prototypen
- Einheitliche Benutzeroberfläche & Responsiveness durch VaadinUI realisierbar

#### Nachteile

- JavaScript nutzbar, aber mit Aufwand verbunden
- Vereinheitlichung des Designs für MOXIS Web und Word schwierig
- Beim Design der Komponenten muss auf Add-in Rücksicht genommen werden

**Vaadin - JavaScript Interaktion** Vaadin erlaubt ein direktes Aufrufen von JavaScript Funktionen. Darüber hinaus werden sogenannte „**Connectoren**“ angeboten, welche es erlaubt JavaScript Komponenten in Vaadin Komponenten zu integrieren. Der benötigte

Verbindungscode wird ad hoc vom Framework veröffentlicht und auf dem Browser initialisiert. Dadurch kann die Client-Seite mit dem Server kommunizieren, indem sie einen Remote Procedure Call (RPC) auslöst.

Diese bidirektionale Verbindung ist notwendig, da die asynchronen Office.js Operationen einige Zeit dauern können. Insbesondere das Generieren des PDF Dokumentes ist aufwändig. Dabei muss die Vaadin UI jedoch Rückmeldung über den aktuellen Status der Operation geben, sowie bei Fehlern Information über deren Ursache anzeigen. Die für den Prototypen entwickelte Lösung besteht aus den folgenden vier Kernkomponenten welche in Abbildung 24 gezeigt werden:

**WordComponent.java** Hauptakteur der Verbindung und eine Vaadin Komponente. Stellt das Laden der benötigten JavaScript Dateien über Annotationen sicher. Durch öffentliche Setter und Getter, sowie Value-Change Listener wird der Status der Anwendung synchronisiert. Ansonsten ist es eine ganz gewöhnliche Komponente und kann beliebig in Layouts eingefügt werden.

**WordComponentState.java** Hält Status Variablen, über welche die Server-Seite mit der Client-Seite kommuniziert.

**addin.js** Beinhaltet den eigentlichen Code, mit dem die Office.js Funktionalität umgesetzt ist. Zum Großteil wird der für den ersten Prototypen geschriebene Code eingesetzt.

**addin-connector.js** Dieser initialisiert die JavaScript Komponente und ist für die Kommunikation der beiden Seiten zuständig. Wird an das Window-Objekt angefügt und ruft sowohl die Funktionen in addin.js, als auch die RPC des Servers auf.

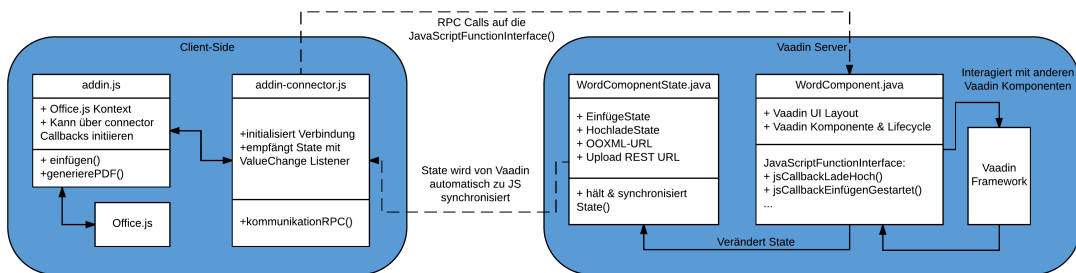


Abbildung 24: Prototyp 2, Komponentendiagramm (schematisch)

**Vorteile**

- Ermöglicht die Nutzung von vielen JavaScript Bibliotheken in Verbindung mit Vaadin

- Geschäftslogik bleibt am Server, Anwendung für andere Bereiche von MOXIS zukunftsträchtig

### Nachteile

- Langsamer wegen Kommunikations-Aufwand
- Daher sollte sich der Datenaustausch auf Steuerungslogik beschränken

### 4.2.2 Beschreibung

Nachfolgend wird die Dienstleistungsimplementierung gezeigt. Die Unterschiede zu den in Unterabschnitt 3.1.2 skizzierten Use-Cases ist eine Folge der Fokussierung auf den konkreten Kundenprozess.

Word2MOXIS wird durch das Hinzufügen der in Unterabschnitt 2.6.2 beschriebenen Manifest XML Datei zum Trust Center installiert. Dieses kann in einem geteilten Ordner oder einem SharePoint Ordner platziert sein.

Der Benutzer startet das Add-in in der 2016er Office Version über die im Ribbon integrierten Add-in Commands. Nutzer der älteren Word 2013 Version starten es über den Einfügedialog. Danach wird das Standard Loginfenster von MOXIS in einem Task Panel angezeigt.

Nach erfolgter Authentifizierung wird direkt die Auftrags-Anlegeseite angezeigt, welche in Abbildung 25 zu sehen ist. Die Navigationsleiste passt ihre angezeigten Elemente dem Platz an und ermöglicht dem Benutzer, Einstellungen zu ändern und sich aus dem Add-in abzumelden. In zukünftigen Versionen mit mehreren Kundenprozessen wird eine Übersicht aller auswählbaren Prozesse angezeigt.

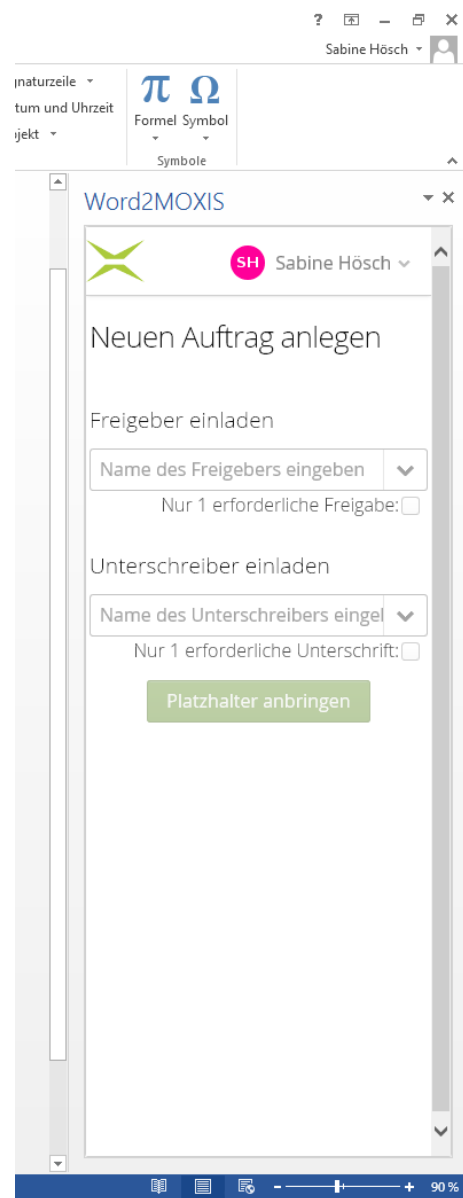


Abbildung 25: Prototyp 2, Anlegeseite

Wie Abbildung 26 zeigt, sind Platzhalter in Freigeber und Unterschreiber unterteilt. Die beiden Suchleisten verwenden Autovervollständigung, um dem Benutzer eine schnellere Auswahl zu erlauben. Dabei liegen jeder Leiste eigene vom Prozess definierte Suchfilter zugrunde. Mittels einer Checkbox kann gesteuert werden, dass die jeweilige Liste von Personen als Gruppe unterschreibt, sodass die Unterschrift einer Person ausreicht, um den Iterationsschritt abzuschließen. Dies stellt eine abgespeckte Version der in Unterabschnitt 3.2.3 vorgeschlagenen und in Abbildung 4 skizzierten Rollen-Funktionalität dar, welche sukzessive vom XiTrust Entwicklerteam implementiert werden soll.

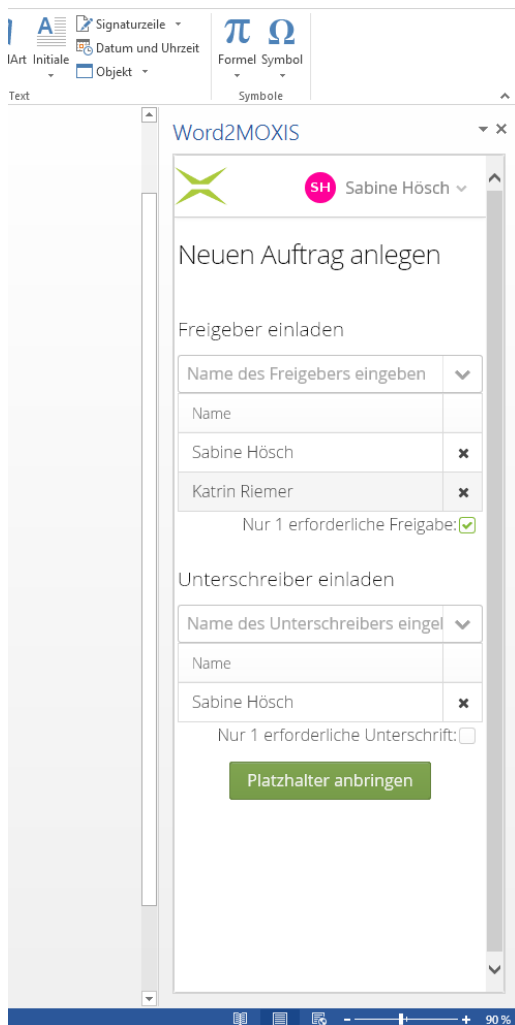


Abbildung 26: Prototyp 2, Unterschreiber hinzufügen

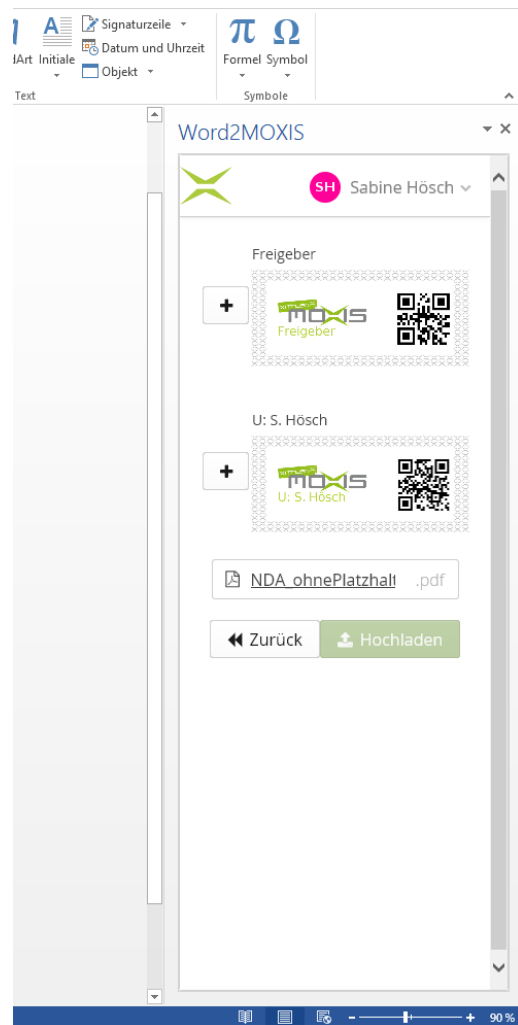


Abbildung 27: Prototyp 2, Platzhalter einfügen

Sind alle gewünschten Platzhalter gewählt, wird mit dem Einfügefenster fortgefahren. Dort werden die Platzhalter nach der im Prozess hinterlegten Konfiguration generiert und wie in Abbildung 27 zu sehen, angezeigt. Es kann auch fortgefahren werden, ohne Perso-

nen einzuladen und neue Platzhalter am Dokument anzubringen. In diesem Fall wird angenommen, dass es sich um Vorlagen handelt, an denen bereits Platzhalter vorangebracht sind. Allerdings kann man einen Auftrag nur starten, wenn im Scanschritt mindestens ein Freigabe- und Signaturplatzhalter vorhanden ist. Jeder Platzhalter wird einzeln an der aktuellen Cursor Position im Dokument eingefügt. Wurden Einladungen vergessen, kann jederzeit zurück gegangen werden.

Vor dem nächsten Schritt kann der Benutzer den Dokumentnamen ändern, oder den voreingestellten Namen der .docx Datei übernehmen. Hiernach wird das Dokument als PDF exportiert und zum MOXIS Server hochgeladen. Nach erfolgreicher Übertragung wird das Dokument auf Platzhalter gescannt. Hierzu wird es auf die QR Codes durchsucht, welche in Unterabschnitt 3.2.1 definiert wurden.

Die gefundenen Platzhalter werden in einer Auftragsübersicht angezeigt, welche in Abbildung 28 sichtbar ist. Falls das Resultat nicht der Prozessspezifikation entspricht, wird eine Fehlermeldung angezeigt und das Dokument abgelehnt. Zugleich hat der Benutzer die Möglichkeit, eine Nachricht an alle Empfänger anzugeben und auch Beilagen an den Signaturauftrag anzuhängen.

Nach dem Senden wird eine Erfolgsmeldung über den soeben erstellten Auftrag ausgegeben und angeboten, die Auftragsmappe des Benutzers in einem Browser Fenster, mit voller MOXIS Web Funktionalität, zu öffnen oder zum Start des Add-ins zurückzukehren.

Nun übernimmt MOXIS-PE mittels der Activiti Workflow-Engine die korrekte Ausführung des Prozesses und lädt per E-Mail die angegebenen Signatoren ein.

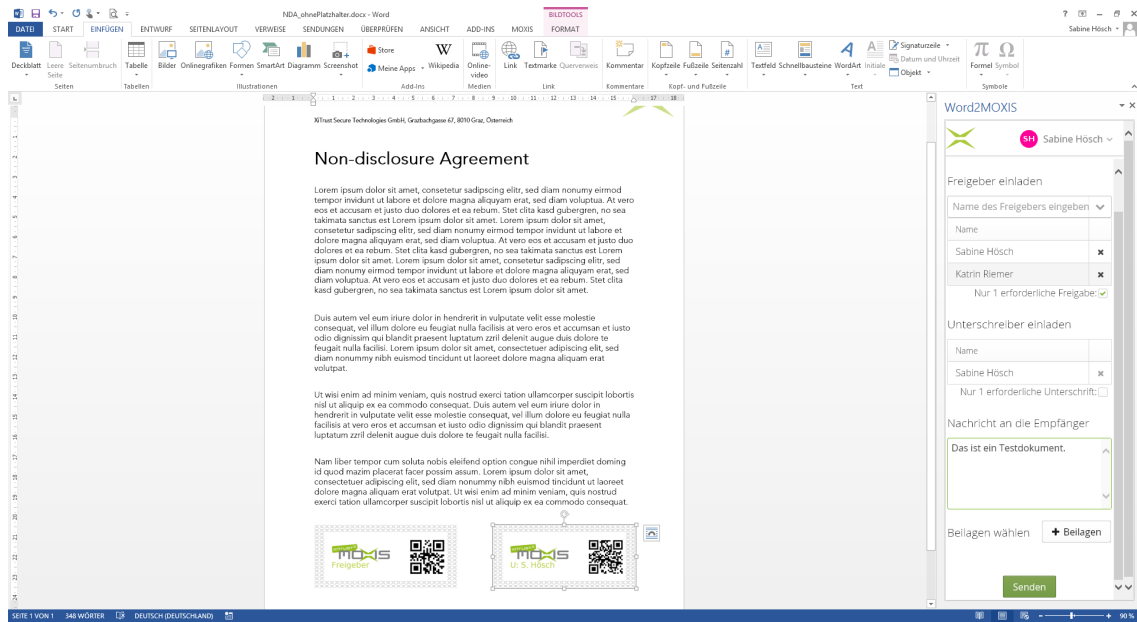


Abbildung 28: Prototyp 2, Auftrag abschließen (Gesamtansicht)

### 4.3 Analyse

In diesem Abschnitt werden die durch die Umsetzung gewonnenen Erkenntnisse aufbereitet, sowie Vor- und Nachteile herausgearbeitet. Während der Dienstleistungsimplementierung wurde kontinuierlich die Zielsetzung mit dem Ist-Zustand verglichen. Von besonderem Interesse ist die Eignung von Vaadin als Office Add-in Framework. Die Anwendung wurde abschließend mit verschiedenen üblichen Kombinationen von Betriebssystemen und Office Varianten getestet und für stabil befunden.

Das Ergebnis wurde bei der XiTrust unter Beisein externer Marketing Partner und eines Kunden vorgestellt und an das hausinterne Entwickler Team zur Wartung und Weiterentwicklung übergeben. In dieser Form wird der zweite Prototyp schon als Dienstleistung mit MOXIS gebündelt.

**Positiv** Da der Prototyp letztendlich ein fertiges Produkt im Sinne der Dienstleistungskonzeption ist und an einen Kunden ausgeliefert wird, kann man von einem Erfolg sprechen. Im Folgenden werden besonders auffallende Aspekte dargelegt.



**Benutzererfahrung** Die Benutzerführung und das subjektive Erlebnis entsprechen den Erwartungen. Anzumerken ist, dass die Geschwindigkeit der Anwendung nicht unter dem vermehrten Kommunikationsaufwand leidet. Der erleichterte Zugang zum Add-in wird erwartungsgemäß zu einer höheren Nutzungsrate führen. Mehrere Add-in Commands verschaffen den Benutzern zusätzliche Einstiegspunkte und beschleunigen ihre Arbeit.

**Entwicklungsvereinfachung** Die Schaffung der Wissensbasis über die Verbindung von JavaScript Komponenten mit dem Vaadin Server war sehr wertvoll. Die Verwendung der in Abschnitt 4.2.1 skizzierten Connectoren wird in einem weiteren Teilprodukt von MOXIS vorbereitet und in anderen in Erwägung gezogen.

**Testvereinfachung** Auch die für MOXIS-PE geschriebenen Tests der Vaadin Komponenten können wiederverwendet werden.

**Design** Die geteilten CSS Stile machen das Anpassen des Basisdesigns über alle Module einfacher. Lediglich die Punkte an denen das Responsive Design seinen Inhalt umpositioniert, müssen geändert werden.

**Auslieferung** Es muss nur mehr das Manifest XML mit den korrekten URLs versehen werden und an den Kunden übergeben werden. Das Ausrollen der Datei ist entweder über SharePoint oder Windows Group Policies einfach durchzuführen.

**Probleme** Jedes Softwareentwicklungsprojekt wird von mehreren kleinen Problemen begleitet, doch im Zusammenspiel der in dieser Arbeit verwendeten Technologien traten zwei schwerwiegendere Hindernisse zu Tage. Diese konnten zwar gelöst werden, aber lassen hinsichtlich der Entwicklungserfahrung zu wünschen übrig.

**GET Request Parameter** Das Add-in wird von der Host Anwendung in einem eingebetteten Frame angezeigt. Bei diesem initialen Aufruf fügt Word die folgenden GET Parameter an die URL:

```
"GET /myapp/?_host_Info=Word|Win32|16.01|en-US HTTP/1.1" 200 2101
```

Dieser Parameter ist aufgrund der direkt enthaltenen „|“ (Senkrechter Strich) keine gültige URL-Kodierung. Eigentlich sollte dieser als „%7C“ enkodiert sein. Dadurch wird im Vaadin Framework ein Error ausgelöst, der nicht direkt in diesem behebbbar ist, da er in einem nicht konfigurierbaren Initialisierungsschritt bei AJAX-Aufrufen von der Client-Seite passiert.

Die implementierte Problemlösung basiert auf dem Einsatz eines Servlet-Filters, bei dem der betreffende Parameter aus der URL gefiltert wird. Der Zeitaufwand zum Lösen dieses Problems war sehr hoch, denn Dokumentation zu diesem Verhalten von Word ist nicht vorhanden. Auch der Grund für den Einsatz dieser Enkodierung ist fragwürdig.

Diese Eigenheit scheint auch in Verbindung mit anderen Frameworks problematisch zu sein, denn ein Benutzer berichtet von ähnlichen Problemen mit Angular.js <sup>17</sup>

**Caching** Vaadin stellt die Initialisierung und das Verbinden mit dem Client-Side Code über verschiedene JavaScript Hilfsbibliotheken sicher. Eines davon lädt alle auf der Seite benötigten zusätzlichen Skript Dateien nach. Dieses Lademodul wird in manchen Word-Umgebungen für bis zu 3 Wochen im Cache des eingebetteten Browser Fensters gespeichert. Bei Änderungen an den zu ladenden JavaScript werden die veränderten Dateien selbst solange nicht nachgeladen, bis das Ladeskript erneuert wird.

Dies ist sowohl beim Entwickeln als auch im Auslieferungszustand problematisch, da hier notwendige Updates verzögert werden könnten. Viel schlimmer ist das nicht einheitliche Verhalten in Bezug auf die Plattformen. Eine Möglichkeit, den Cache direkt zu leeren, wurde nicht gefunden, aber es ist ohnehin wenig wünschenswert, dies auf Kundensystemen durchzuführen.

Zur Lösung des Problems wird ein weiterer Servlet Filter eingesetzt, der das Caching der betreffenden Ressource verhindert.

**Session Behandlung** Die von MOXIS standardmäßig verwendete Sicherheitseinstellung erlaubt es nicht, mit einem Account zeitgleich in mehr als einer Session eingeloggt zu sein. Diese Limitierung behindert aktuell noch die Benutzererfahrung, da der Benutzer automatisch in der Webanwendung ausgeloggt wird, sobald er das Word Add-in benutzt.

Ist zurzeit ruhend gelegt, aber vom Team jederzeit durch eine Konfigurationsänderung behebbar.

**Öffnen von MOXIS aus Add-in** Ergibt sich aus dem Session Problem. Beim Öffnen muss der Benutzer aus dem Add-in ausgeloggt sein, um kein Session Timeout zu erhalten. Zugleich kann das Öffnen des Browser Fensters nicht durchgeführt werden, wenn Vaadin nicht mehr dessen Inhalt steuert.

---

<sup>17</sup><https://stackoverflow.com/questions/37388541/office-add-in-development-malformed-get-url-host-info>, zuletzt zugegriffen am 01.10.2016

# Kapitel 5

## Conclusio

### 5.1 Fazit

Eingebettet in die rechtlichen, technischen und ökonomischen Rahmenbedingungen wurde ein Innovationsprozess einer IKT basierten Dienstleistung umgesetzt. Mittels der in diesem Prozess verwendeten Methodik wurde ein beidseitiger Wissenstransfer mit dem Unternehmen durchgeführt. Damit wurde das technologische Leistungspotential nachhaltig gestärkt.

Die Forschungsfrage:

*Platzhalter mit QR Code, wie kann man das Konzept der Unterschriftenplatzhalter im XiTrust MOXIS innovieren?*

wurde mit einer Analyse der maschinenlesbaren visuellen Datenrepräsentation gestartet. Daraus wurden Vorschläge hinsichtlich der Syntax und dem Aussehen der Barcodes erarbeitet und der Geschäftsführung vorgelegt. Das Instrument der Ideensteckbriefe erwies sich als robuste Methode der Ideenfindung und Aufbereitung von Informationen. Eine Integration in das Produkt MOXIS wurde gestartet.

Mithilfe des 9 Fenster Operators konnte eine umfassende Sichtweise auf die zweite Forschungsfrage:

*Wie kann man das Modul Word2MOXIS mit Hilfe des neuen Unterschriftenplatzhalterkonzeptes verbessern?*

gewonnen werden. Zusammen mit dem in der Firmenphilosophie gut verankerten Customer Co-Creation Prozess wurden die Bedürfnisse der Kunden ausführlich dokumentiert.

Durch die Verwendung des iterativen Service Engineering Verfahren wurde die Dienstleistung strategisch geplant. Mit experimentellem Prototyping wurde das Evaluierungsrisiko der Neuentwicklung minimiert, und rechtzeitig Korrekturmaßnahmen durchgeführt. Diese Werkzeuge erlaubten eine zielgerichtete Entwicklung des Prototypen.

Die resultierende Dienstleistung ist durch die Kundenorientierung ein Erfolg und hilft gleichzeitig dem Unternehmen selbst, Zeit und Kosten zu sparen. Zum Abschluss dieser Arbeit wurde das Add-in als Modul in MOXIS integriert und in den Testbetrieb eines Kunden überführt.

Durch die verwendete Technologie kann die XiTrust GmbH schnell auf geänderte Anforderungen reagieren und die Produktpalette gemeinsam mit MOXIS-PE kontinuierlich verbessern.

## 5.2 Ausblick und weitere Forschungsfragen

Im Rahmen dieser Arbeit konnten aus Zeitgründen nicht alle vorgeschlagenen Komponenten implementiert werden. Diese Umsetzung muss mit den in der Firma gelebten Service Engineering Grundsätzen geleitet und überprüft werden. Hinsichtlich des Funktionsumfangs des Word Add-in existieren einige Ideen, welche noch nicht im Detail geplant und umgesetzt worden sind. Unter anderem:

**Bidirektionale Repräsentation der Platzhalter** Die neue Word.js Schnittstelle erlaubt es, Regionen im Dokument permanent mit einem Add-in zu verbinden. So könnte Word2MOXIS immer den Status und die Position der Platzhalter ermitteln, und noch einfacheres Einfügen und Ersetzen anbieten.

**Früherer Scan** Der Scanvorgang könnte auf die Client Seite verlegt werden, wodurch die Benutzerführung vereinfacht wird, da eventuell schon vorher am Dokument angebrachte Dokumente beim Start des Add-in gefunden werden, und nicht erst beim Upload. Innerhalb des Word Formates kann dies auch rein mit OOXML Identifikation umgesetzt werden.

Obwohl die Dienstleistungsentwicklung die an sie gestellten Anforderungen erfüllt, kann der aktuelle Status des Office.js Ökosystems neu evaluiert werden. Im Vergleich zum Projektstart ist die angebotene Vielfalt an Bibliotheken und Design Ressourcen enorm gestiegen. Eine Nutzung der in Unterabschnitt 5.2.1 beschriebenen Technologien könnte eine weitere Effizienzsteigerung bewirken.

In einem weiteren Schritt kann das Add-in und MOXIS selbst auf die Office 365 oder mobile Anwendungen maßgeschneidert werden, und eine Veröffentlichung im Office Add-in Store umgesetzt werden. Diese Öffnung könnte die Kundensicht erweitern.

Die Prozessintegration der Platzhalter selbst ist weiterhin eine zentrale Forschungsfrage. Mit aktuellen Trends und Funktionsumfang des MOXIS PE ergeben sich neue Möglichkeiten der Automatisierung:

- *Gibt man den Kunden die Möglichkeit, selbst mittels einem Web BPMN-Editors, Prozesse zu erstellen? Wie sieht ein dazu passendes Geschäftsmodell aus?*
- *In welche Produktivitätsumgebungen ist eine Integration von MOXIS ähnlich des Word Add-ins noch interessant?*
- *Wie kann man die gesamte Dienstleistung für Kunden ohne eigene IT Infrastruktur als Cloud Service konzipieren?*
- *Wie gestaltet man die Kommunikation zwischen einzelnen MOXIS Instanzen? Was und vor allem wie wird es automatisiert ausgetauscht?*

### 5.2.1 Entwicklung der Office.js Plattform

Zu Beginn der Dienstleistungskonzeption im Februar 2016 war die Dokumentation für Office.js noch nicht in vollem Umfang veröffentlicht. Die Unterscheidung zwischen den unterstützten Schnittstellen der einzelnen Office Anwendungen, sowie Beispiele ihrer Nutzung waren nicht dokumentiert. Dies betraf auch die Funktion, OOXML Inhalte direkt in das Dokument einzufügen. Im Bezug auf die Ribbon Integration sowie OOXML Einfügemodi wurde auf ein Updateversprechen<sup>18</sup> von Microsoft gesetzt, weshalb sofort mit der Planung und Entwicklung des Services inklusive dieser Features begonnen wurde.

Dies wurde in einem Update im Juli 2016 von Microsoft nachgeholt, in welchem auch die angekündigten Add-In Commands sowie weitere Word spezifische APIs integriert wurden. Mit diesem wurde auch die Dokumentation qualitativ verbessert, da ein eigenes Development Center für Office bereitgestellt wurde<sup>19</sup>. Auch das Ausliefern von Add-ins und die Produktivpipeline wurden erörtert.

Der Fokus wurde von Microsoft auch in den Entwurf und das Veröffentlichen von Office Design-Patterns gelegt. Diese sollen im Add-in verwendete Elemente ästhetisch an die Office Oberflächen angleichen, sowie schnellere Entwicklung erlauben<sup>20</sup>.

---

<sup>18</sup><https://blogs.office.com/2015/09/28/whats-new-in-office-2016-for-developers/>, zuletzt zugegriffen am 04.10.2016

<sup>19</sup><http://dev.office.com/docs/add-ins/overview/office-add-ins>, zuletzt zugegriffen am 30.09.2016

<sup>20</sup><https://github.com/OfficeDev/Office-Add-in-UX-Design-Patterns>, zuletzt zugegriffen am 30.09.2016

Zuletzt wurde ein offizielles Front-End Framework für Office und Office 365 Anwendungen namens **Office UI Fabric** veröffentlicht <sup>21</sup>. Dieses bietet robuste wiederverwendbare Komponenten und stellt eine komplette Design Sprache mit Stilen und Icons zur Verfügung. Neben einer Basisvariante in purem JavaScript werden auch die beiden modernen Frameworks React und Angular unterstützt.

Diese aktuellen Entwicklungen sind für das Ökosystem als Ganzes positiv, und stellen eine Option für zukünftige Erweiterungen von Word2MOXIS dar. Insbesondere, da alle Ressourcen lizenzfrei angeboten und als Open Source Projekte weiterentwickelt werden.

---

<sup>21</sup><https://dev.office.com/fabric>, zuletzt zugegriffen am 30.09.2016

# Anhang A

## Begriffsbestimmung

### A.1 Abkürzungsverzeichnis

<b>MOXIS</b>	Mobile XiTrust Signaturlösung
<b>QR Code</b>	Quick Response Code
<b>BPMN</b>	Business Process Model and Notation
<b>IKT</b>	Informations- und Kommunikationstechnologien
<b>GWT</b>	Google Web Toolkit
<b>KVP</b>	Kontinuierlicher Verbesserungsprozess
<b>RPC</b>	Remote Procedure Call
<b>DMC</b>	Data Matrix Code
<b>PDF417</b>	Portable Data File 417
<b>UPC</b>	Universal Product Code
<b>ACI</b>	Automatic Car Identification
<b>EAN</b>	European Article Number
<b>GTIN</b>	Global Trade Item Number
<b>HCCB</b>	High Capacity Color Barcode
<b>PH</b>	Platzhalter
<b>PKI</b>	Public Key Infrastruktur

**Unterschriften-PH** Unterschriftenplatzhalter

**XiTrust** XiTrust Secure Technologies GmbH

**eIDAS** Verordnung über elektronische Identifizierung und Vertrauensdienste für elektronische Transaktionen im Binnenmarkt und zur Aufhebung der Richtlinie 1999/93/EG

**SigG** Signaturgesetz

**SigV** Signaturverordnung

**SGV** Signatur- und Vertrauensgesetz

**VSTO** Visual Studio Tools for Office



# Literaturverzeichnis

- [Asc09] Helmut Aschbacher. Entwicklung eines service engineering prozessmodells für software solution provider im kmu sektor unter nutzung eines webbasierten informationssystems. Master's thesis, Technische Universität Graz, 2009.
- [Asc14] Helmut Aschbacher. *Framework für das agile Entwickeln von IKT-basierten Dienstleistungen unter Nutzung von Smart Services*. PhD thesis, Technische Universität Graz, 2014.
- [BL09] Bendik Bygstad and Gjermund Lanestedt. Ict based service innovation—a challenge for project management. *International Journal of Project Management*, 27(3):234–242, 2009.
- [Bro01] David L Brock. Integrating the electronic product code (epc) and the global trade item number (gtin). *White Paper available at [www. autoidcenter. org/pdfs/MIT-WUTOID-WH-004. pdf](http://www.autoidcenter.org/pdfs/MIT-WUTOID-WH-004.pdf)*, 25, 2001.
- [BS06] Hans-Jörg Bullinger and August-Wilhelm Scheer. Service engineering—entwicklung und gestaltung innovativer dienstleistungen. In *Service Engineering*, pages 3–18. Springer, 2006.
- [Cra11] I. Cranstone. A guite to aci (automatic car identification)/kar trak labels. <http://www.nakina.net/other/aci/aci.html>, 2011. Zuletzt zugegriffen am 28.07.2016.
- [DB94] Stan Davis and Jim Botkin. The conning of knowledge-based business. *Harvard Business Review*, 1994.
- [Dos10] Martin Dostál. User acceptance of the microsoft ribbon user interface. In *Proceedings of the 9th WSEAS International Conference on Data Networks, Communications, Computers, DNCOCO'10*, pages 143–149, Stevens Point, Wisconsin, USA, 2010. World Scientific and Engineering Academy and Society (WSEAS).

- [Fox11] M. Fox. Alan haberman, who ushered in the bar code, dies at 81. [http://www.nytimes.com/2011/06/16/business/16haberman.html?\\_r=1&hp](http://www.nytimes.com/2011/06/16/business/16haberman.html?_r=1&hp), June 2011. Zuletzt zugegriffen am 28.07.2016.
- [fS06] International Organization for Standardization. Information technology – automatic identification and data capture techniques – data matrix bar code symbology specification. software, International Organization for Standardization, 2006.
- [fS12] International Organization for Standardization. Information technology – document description and processing languages – office open xml file formats – part 1: Fundamentals and markup language reference. software, International Organization for Standardization, 2012.
- [fS15a] International Organization for Standardization. Information technology – automatic identification and data capture techniques – pdf417 bar code symbology specification. software, International Organization for Standardization, 2015.
- [fS15b] International Organization for Standardization. Information technology – automatic identification and data capture techniques – qr code bar code symbology specification. software, International Organization for Standardization, 2015.
- [Grö14] Marko Grönroos. Book of vaadin: Vaadin 7 edition, revision 2nd ed. *Vaadin Ltd*, 2014.
- [GS96] Heinz Lothar Grob and Sabine Seufert. Vorgehensmodelle bei der entwicklung von cal-software. *Online im Internet: URL: http://www.wi.uni-muenster.de/aw/calcat/ab5/[Letzter Zugriff 2002-08-13]*, 1996.
- [HL11] Martin Hilbert and Priscila López. The world’s technological capacity to store, communicate, and compute information. *science*, 332(6025):60–65, 2011.
- [KF06] Michael Kleinaltenkamp and Janine Frauendorf. Wissensmanagement im service engineering. In *Service Engineering*, pages 359–376. Springer, 2006.
- [KMS<sup>+</sup>97] Henning Klodt, Rainer Maurer, Axel Schimmelpfennig, et al. Tertiarisierung in der deutschen wirtschaft. Technical report, Kiel Institute for the World Economy (IfW), 1997.
- [KT05] Hiroko Kato and KT Tan. 2d barcodes for mobile phones. In *2005 2nd Asia Pacific Conference on Mobile Technology, Applications and Systems*, pages 8–pp. IEEE, 2005.

- [LB07] Tobias Langlotz and Oliver Bimber. Unsynchronized 4d barcodes. In *International Symposium on Visual Computing*, pages 363–374. Springer, 2007.
- [MD<sup>+</sup>07] Emmanuel Muller, David Doloreux, et al. The key dimensions of knowledge-intensive business services (kibs) analysis: a decade of evolution. Technical report, Fraunhofer Institute for Systems and Innovation Research (ISI), 2007.
- [Mic16] Microsoft. Privacy and security for office add-ins. <http://dev.office.com/docs/add-ins/develop/privacy-and-security>, 2016. Zuletzt zugegriffen am 05.10.2016.
- [Mil05] Ian Miles. Knowledge intensive business services: prospects and policies. *foresight*, 7(6):39–63, 2005.
- [OEC15] OECD. Oecd digital economy outlook 2015. *OECD Publishing*, 2015.
- [PJ08] D. Parikh and G. Jancke. Localization and segmentation of a 2d high capacity color barcode. In *Applications of Computer Vision, 2008. WACV 2008. IEEE Workshop on*, pages 1–6, Jan 2008.
- [PotC99] European Parliament and of the Council. Signature directive 1999/93/ec, 1999.  
<http://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=CELEX:31999L0093:en:HTML>,.
- [PotC14] European Parliament and of the Council. Regulation (eu) no 910/2014 -eidas, 2014.  
<http://eur-lex.europa.eu/legal-content/EN/ALL/?uri=CELEX:32014R0910>,.
- [Rad12] Tijds Rademakers. *Activiti in Action: Executable business processes in BPMN 2.0*. Manning Publications Co., 2012.
- [RB99] RIS-BKA. Signaturgesetz - sigg, 1999.  
<https://www.ris.bka.gv.at/GeltendeFassung.wxe?Abfrage=Bundesnormen&Gesetzesnummer=10003685>,.
- [SCJF10] M. L. Snyder, A. Carter, K. Jenkins, and C. R. Fantz. Patient misidentifications caused by errors in standard bar code technology. *Clinical Chemistry*, 56(10):1554–1560, aug 2010.
- [Sei11] T Seidemann. Barcodes sweep the world. [http://www.barcoding.com/information/barcode\\_history.shtml](http://www.barcoding.com/information/barcode_history.shtml), Zuletzt zugegriffen am 28.07.2016 2011.

- [Ste10] Das Land Steiermark. Innolightfaden, 2010.  
[https://www.campus02.at/uploads/innolightfaden\\_2010\\_rev04\\_2285\\_DE.pdf](https://www.campus02.at/uploads/innolightfaden_2010_rev04_2285_DE.pdf), zuletzt zugegriffen am 02.10.2016,.
- [SZS12] E Schricke, A Zenker, and T Stahlecker. Knowledge-intensive (business) services in europe. *Brussels: European Commission*, 2012.
- [UH12] Iris Uitz and Michael Harnisch. Der qr-code – aktuelle entwicklungen und anwendungsbereiche. *Informatik-Spektrum*, 35(5):339–347, 2012.
- [Uni14] International Telecommunication Union. *Measuring the Information Society Report 2014*. International Telecommunication Union, 2014.
- [VGKW12] Rohit Verma, Anders Gustafsson, Per Kristensson, and Lars Witell. Customer co-creation in service innovation: a matter of communication? *Journal of Service Management*, 23(3):311–327, 2012.
- [Whi04] Stephen A White. Introduction to bpmn. *IBM Cooperation*, 2(0):0, 2004.