

# Bachelorarbeit

Sofia Knap

Untersuchung des Zusammenspiels von RPA und BPMS  
am Beispiel eines automatisierten Bewerbungsprozesses

Sofia Knap

# Untersuchung des Zusammenspiels von RPA und BPMS am Beispiel eines automatisierten Bewerbungsprozesses

Bachelorarbeit eingereicht im Rahmen der Bachelorprüfung  
im Studiengang *Bachelor of Science Wirtschaftsinformatik*  
am Department Informatik  
der Fakultät Technik und Informatik  
der Hochschule für Angewandte Wissenschaften Hamburg

Betreuender Prüfer: Prof. Dr. Ulrike Steffens  
Zweitgutachter: Prof. Dr. Rüdiger Weißbach

Eingereicht am: 18. Juli 2024

**Sofia Knap**

## **Thema der Arbeit**

Untersuchung des Zusammenspiels von RPA und BPMS am Beispiel eines automatisierten Bewerbungsprozesses

## **Stichworte**

Robotic Process Automation, RPA, Business Process Management System, BPM, BPMS, Integration

## **Kurzzusammenfassung**

Diese Bachelorarbeit untersucht das Zusammenspiel von Robotic Process Automation (RPA) und Business Process Management Systemen (BPMS) anhand eines automatisierten Bewerbungsprozesses. Die Arbeit beleuchtet die Vorteile und Herausforderungen der Verbindung dieser Technologien und zeigt, wie sie zusammen eingesetzt werden können, um Geschäftsprozesse effizienter zu gestalten.

Im Rahmen der Untersuchung wird ein Konzept für die Integration von RPA und BPMS entwickelt und ein Teil des Konzepts in einen Prototypen umgesetzt. Dieser Prototyp ermöglicht die Automatisierung von sowohl RPA- als auch BPM-Aufgaben eines Prozesses. Durch die Integration beider Technologien wird gezeigt, wie repetitive und regelbasierte Aufgaben durch RPA automatisiert und Prozessabläufe durch BPMS gesteuert werden können.

Die Ergebnisse zeigen, dass durch die Verbindung von RPA und BPMS eine signifikante Optimierung von Geschäftsprozessen und deren Ausführung möglich ist. Der Prototyp demonstriert, wie manuelle Arbeitsaufwände reduziert und die Prozessgenauigkeit verbessert werden können. Insbesondere wird verdeutlicht, dass RPA und BPMS zusammenarbeiten können, um die Effizienz, Genauigkeit und Konsistenz von Prozessen zu steigern. Zukünftige Arbeiten sollten sich auf die Verbesserung der Benutzerfreundlichkeit und Performance des Systems sowie die Integration fortschrittlicher Analysetools und künstlicher Intelligenz zur weiteren Prozessoptimierung konzentrieren. Durch die Erweiterung der Funktionalitäten und die Verfeinerung der Technologien könnten noch größere Effizienzgewinne und präzisere Prozesssteuerungen erzielt werden.

---

**Sofia Knap**

**Title of Thesis**

Exploration of the interaction between RPA and BPMS using the example of an automated application process

**Keywords**

Robotic Process Automation, RPA, Business Process Management System, BPM, BPMS, Integration

**Abstract**

This bachelor thesis examines the interaction of Robotic Process Automation (RPA) and Business Process Management Systems (BPMS) using an automated application process. The thesis highlights the benefits and challenges of combining these technologies and shows how they can be used together to make business processes more efficient.

As part of the study, a concept for the integration of RPA and BPMS is developed and part of the concept is implemented in a prototype. This prototype enables the automation of both RPA and BPM tasks in a process. By integrating both technologies, it is shown how repetitive and rule-based tasks can be automated by RPA and process flows can be controlled by BPMS.

The results show that a significant optimization of business processes and their execution is possible by combining RPA and BPMS. The prototype demonstrates how manual workloads can be reduced and process accuracy improved. In particular, it shows that RPA and BPMS can work together to increase the efficiency, accuracy and consistency of processes.

Future work should focus on improving the user-friendliness and performance of the system as well as integrating advanced analysis tools and artificial intelligence to further optimize processes. Through the functionalities and the refinement of technologies, even greater efficiency gains and more precise process controls could be achieved.

# Inhaltsverzeichnis

<b>Abbildungsverzeichnis</b>	<b>vii</b>
<b>Tabellenverzeichnis</b>	<b>viii</b>
<b>1 Einleitung</b>	<b>1</b>
1.1 Motivation . . . . .	1
1.2 Zielsetzung . . . . .	2
1.3 Aufbau der Arbeit . . . . .	3
<b>2 Grundlagen</b>	<b>4</b>
2.1 Business Process Management (BPM) . . . . .	4
2.1.1 BPM-Lebenszyklus . . . . .	4
2.1.2 Business Process Management System (BPMS) . . . . .	7
2.1.3 Vor- und Nachteile eines BPMS . . . . .	8
2.2 Robotic Process Automation (RPA) . . . . .	9
2.2.1 RPA-Lebenszyklus . . . . .	10
2.2.2 Vor- und Nachteile einer RPA . . . . .	10
<b>3 Verwandte Arbeiten</b>	<b>12</b>
3.1 Potential eines BPM-RPA Systems . . . . .	12
3.2 Herausforderungen eines BPM-RPA Systems . . . . .	14
<b>4 Konzept</b>	<b>15</b>
4.1 Anforderungsanalyse . . . . .	15
4.1.1 Anforderungen des Gesamtsystems . . . . .	15
4.1.2 Anforderungsanalyse für die Process Engine . . . . .	17
4.2 Systemarchitektur . . . . .	19
4.2.1 Der Prozess Modeller . . . . .	19
4.2.2 Der RPA-Parser . . . . .	21
4.2.3 Die Engines . . . . .	21

<b>5</b>	<b>Anwendungsbeispiel</b>	<b>23</b>
<b>6</b>	<b>Umsetzung</b>	<b>25</b>
6.1	Auswahl der Engines . . . . .	25
6.2	Implementierung der Engines . . . . .	26
6.3	Die REST API . . . . .	28
<b>7</b>	<b>Evaluation</b>	<b>31</b>
7.1	Vergleich mit den Anforderungen . . . . .	31
7.2	Diskussion . . . . .	33
<b>8</b>	<b>Schlussbetrachtung</b>	<b>36</b>
8.1	Fazit . . . . .	36
8.2	Ausblick . . . . .	37
	<b>Literaturverzeichnis</b>	<b>39</b>
	<b>Selbstständigkeitserklärung</b>	<b>43</b>

# Abbildungsverzeichnis

2.1	BPM-Lebenszyklus nach Dumas u. a. (2021) . . . . .	5
2.2	BPMS Architektur von König u. a. (2020) . . . . .	7
3.1	Pareto-Verteilung von Prozessen nach van der Aalst u. a. (2018) . . . . .	14
4.1	Systemarchitektur des BPM-RPA Systems . . . . .	20
4.2	RPA-Aktivität im Modeller . . . . .	20
4.3	UML Komponentendiagramm der Engines . . . . .	21
5.1	BPMN des Bewerbungsprozesses . . . . .	24
6.1	UML Sequenzdiagramm der Prozessausführung . . . . .	30

# Tabellenverzeichnis

4.1	Nicht-funktionale Systemanforderungen . . . . .	16
4.2	Funktionale Systemanforderungen . . . . .	17
4.3	Funktionale Anforderungen für die ausführende Komponente . . . . .	18
4.4	Nicht-funktionale Anforderungen für die ausführende Komponente . . . . .	19

# 1 Einleitung

## 1.1 Motivation

In der heutigen, sich rasant entwickelnden Geschäftswelt stehen Unternehmen vor der fortwährenden Herausforderung, ihre Betriebsabläufe kontinuierlich zu optimieren, um wettbewerbsfähig zu bleiben. Diese Optimierung erfordert nicht nur die Steigerung der Effizienz bestehender Prozesse, sondern auch die strategische Integration neuer Technologien, die diese Transformation unterstützen können. (Penttinen u. a., 2018)

Zwei dieser zukunftsweisenden Technologien sind Business Process Management Systems (BPMS) und Robotic Process Automation (RPA). BPMS bieten Unternehmen die Werkzeuge zur Modellierung, Ausführung und Überwachung komplexer Geschäftsprozesse. Sie ermöglichen eine systematische Verbesserung und Anpassung von Abläufen, indem sie Transparenz schaffen und die Zusammenarbeit zwischen verschiedenen Abteilungen fördern (Dumas u. a., 2021). Auf der anderen Seite revolutioniert RPA die Automatisierung von repetitiven, regelbasierten Aufgaben durch Software-Roboter, die menschliche Interaktionen in digitalen Systemen nachahmen können. Diese Technologie verspricht nicht nur Kosteneinsparungen und Zeitgewinne, sondern auch eine Reduzierung von Fehlern und eine Erhöhung der Skalierbarkeit in Unternehmen (van der Aalst u. a., 2018).

Der weltweite Markt für RPA-Software verzeichnet ein beachtliches Wachstum, das durch die zunehmende Automatisierung von Geschäftsprozessen getrieben wird. So wird der Markt für RPA-Software bis 2027 voraussichtlich ein Volumen von über 11,3 Milliarden US-Dollar erreichen, mehr als das dreifache des Marktwertes von 2023.<sup>1</sup> Dies verdeutlicht das steigende Interesse und die wachsenden Investitionen in diese Technologie, da Unternehmen die Vorteile der Automatisierung erkennen und nutzen wollen.

---

<sup>1</sup><https://www.statista.com/statistics/1309384/worldwide-rpa-software-market-size/>

Parallel dazu entwickelt sich der Markt für Business Process Management (BPM) dynamisch weiter. BPM-Lösungen sind entscheidend für die Effizienzsteigerung und Flexibilität von Unternehmen, da sie ein breites Anwendungsspektrum von der Prozessautomatisierung bis hin zur Optimierung der Geschäftsabläufe bieten. In den kommenden Jahren wird der BPM-Markt weiter wachsen<sup>2</sup>, angetrieben durch die Nachfrage nach skalierbaren Lösungen zur Verwaltung und Optimierung komplexer Unternehmensprozesse.

Die Integration von BPMS und RPA stellt somit einen evolutionären Schritt in Richtung der umfassenden Optimierung und Automatisierung von Geschäftsprozessen dar. Durch die Kombination dieser Technologien können Unternehmen nicht nur ihre operativen Effizienzen steigern, sondern auch neue Möglichkeiten zur Innovation und zum Wachstum erschließen. Der Fokus liegt dabei auf der Schaffung einer flexiblen und agilen Organisation, die in der Lage ist, sich schnell auf sich ändernde Marktbedingungen und Kundenanforderungen anzupassen.

Diese Bachelorarbeit widmet sich genau diesem Spannungsfeld und untersucht das Potenzial und die praktischen Anwendungen der Verbindung von BPMS und RPA anhand eines realen Anwendungsbeispiels: einem automatisierten Bewerbungsprozess. Durch eine detaillierte Analyse und Fallstudien wird die Forschungsfrage verfolgt, inwiefern BPMS und RPA miteinander verbunden werden können und welche konkreten Vorteile sich aus dieser synergetischen Nutzung ergeben. Ziel dieser Untersuchung ist es, fundierte Erkenntnisse zu gewinnen, die Unternehmen dabei unterstützen, ihre eigenen Prozesse gezielt zu optimieren und dadurch ihre Wettbewerbsposition nachhaltig zu stärken und zukunftsfähig aufzustellen.

## 1.2 Zielsetzung

Das Ziel dieser Bachelorarbeit ist die Untersuchung von Methoden, welche notwendig wären, um ein BPMS mit einer RPA zu verbinden. Der Fokus soll dabei auf der Suche nach einer Lösung sein, die das Ausführen eines Bewerbungsprozesses ermöglicht, welcher sowohl BPM- als auch RPA-Elemente enthält.

Die Bachelorarbeit kann als erfolgreich angesehen werden, wenn sie verschiedene Kriterien erfüllt. Ein wichtiger Faktor für den Erfolg der Arbeit ist eine umfassende Literaturrecherche zu den Themengebieten des BPM, BPMS und der RPA, da diese Themen mit der Beantwortung der Forschungsfrage verbunden sind. Daneben muss der Stand der

---

<sup>2</sup><https://www.fortunebusinessinsights.com/business-process-management-bpm-market-102639>

Forschung so erfasst werden, sodass konkrete Anforderungen an das System erschlossen werden können. Anschließend soll ein konkretes Konzept entwickelt und umgesetzt werden. Schlussendlich soll das Konzept und die Umsetzung in einem letzten Schritt bewertet werden. Falls das Konzept nicht in der Praxis anwendbar ist, müssen die Gründe dafür klar ausgelegt werden. Wenn diese Kriterien erfüllt sind, gilt diese Bachelorarbeit als erfolgreich.

### 1.3 Aufbau der Arbeit

Zum Beginn der Arbeit werden die im Kapitel 2 die relevanten Grundlagen erklärt. Es werden dort die Begrifflichkeiten der Themenbereiche beschrieben und die Systeme erläutert, die für die Beantwortung der Forschungsfrage notwendig sind. In dem nachfolgenden Kapitel 3 wird der Stand der Wissenschaft zu dem Thema BPM-RPA vorgestellt. Das Kapitel 4 enthält eine Anforderungen, welche ein BPM-RPA System erfüllen sollte, spezifiziert diese auf die ausführenden Komponenten des Systems und entwickelt die Systemarchitektur. Auf dies stützt sich das Kapitel 6, welches sich mit der genauen Umsetzung der ausführenden Komponente befasst. In Kapitel 7 wird die Umsetzungslösung mit den Anforderungen verglichen und das Ergebnis diskutiert. Das letzte Kapitel 8 schließt die Arbeit mit einem Fazit und Ausblick ab.

## 2 Grundlagen

In diesem Kapitel werden die für diese Arbeit relevanten Grundlagen geklärt. Das Kapitel dient zur Verständlichkeit der nachfolgenden Kapitel.

### 2.1 Business Process Management (BPM)

Laut Dumas u. a. (2021) ist Business Process Management (BPM)

„die Kunst und Wissenschaft, die Arbeit in einer Organisation so zu gestalten, dass konsistente Ergebnisse sichergestellt und Verbesserungspotenziale genutzt werden.“

Dabei wird unter der Formulierung „Arbeit der Organisation“ sogenannte Geschäftsprozesse verstanden. Diese Prozesse sind eine Abfolge von Aktivitäten, Entscheidungen und Ereignissen, die ausgeführt werden, um ein definiertes Geschäftsergebnis für einen bestimmten Kunden oder Markt zu erzielen (Weske, 2019; Dumas u. a., 2021).

Eine Aktivität ist dabei ein Arbeitsschritt, welcher Zeit zur Ausführung benötigt. Die Art und Weise, wie diese Aktivität ausgeführt wird, ob durch einen Menschen, mit Hilfe eines Systems oder vollständig durch ein System, spielt dabei keine Rolle (Weske, 2019). Ist eine Aktivität atomar, so kann sie auch als Aufgabe bezeichnet werden.

Entscheidungen und Ereignissen sind Zeitpunkte im Prozess, sind also atomar und zeitlos. Dabei bestimmen Entscheidungspunkte den weiteren Prozessverlauf.(Dumas u. a., 2021)

#### 2.1.1 BPM-Lebenszyklus

BPM umfasst eine Reihe von Methoden, Techniken und Werkzeugen zur Identifizierung, Erhebung, Analyse, Verbesserung, Ausführung und Überwachung von Geschäftsprozessen (Dumas u. a., 2021). Diese Werkzeuge ermöglichen es Organisationen, ihre Prozesse

kontinuierlich zu optimieren und an neue Anforderungen anzupassen, um wettbewerbsfähig zu bleiben.

Eines dieser Werkzeuge ist der BPM-Lebenszyklus. In der Literatur sind viele Ausprägungen des BPM-Lebenszyklus zu finden, die jedoch alle das gleiche Ziel verfolgen. Nousias u. a. (2023) beschreibt dies wie folgt:

„BPM lifecycles constitute schematic diagrams that systematize the methodology and steps of a BPM project, in an effort to effectively manage the organizational operations. This viewpoint aims at improving overall performance by ensuring that business activities are better scheduled, executed, monitored, and coordinated“

Dabei wird der Lebenszyklus in unterschiedliche Phasen eingeteilt. Dabei lassen sich die Phasen eines jeden Lebenszykluses in zwei Gruppen einteilen. Auf der einen Seite sind die Design-, Entwicklungs- und Automatisierungsphasen und auf der anderen die Betriebsphasen. (Uskenbayeva R. K. u. a., 2022)

Zur Veranschaulichung wird hier der BPM-Lebenszyklus von Dumas u. a. (2021) vorgestellt, welche in Abbildung 2.1 dargestellt ist. Dieser wurde ausgewählt, da er die am häufigsten vorkommenden Phasen in sich vereint und somit einen detaillierten Gesamtüberblick ermöglicht (Nousias u. a., 2023).

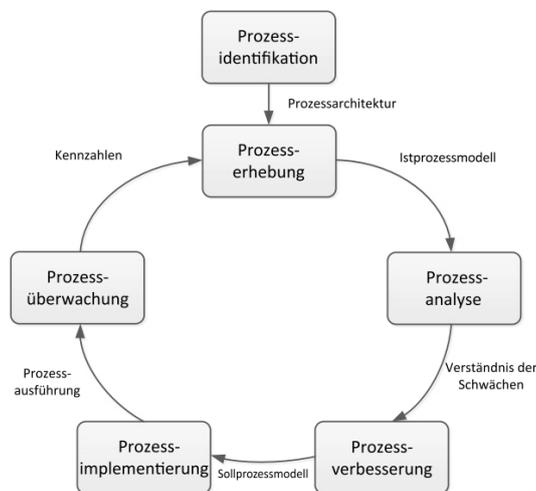


Abbildung 2.1: BPM-Lebenszyklus nach Dumas u. a. (2021)

Die erste Phase des BPM-Lebenszyklus ist die *Prozessidentifikation*. Eine Organisation, die bisher keine BPM-Initiativen durchgeführt hat und keine Dokumentation über ihre

Geschäftsprozesse besitzt, identifiziert ein Problem, welches durch BPM gelöst werden soll.

Anschließend werden die relevanten Prozesse identifiziert und abgegrenzt, was zu der Entstehung einer Prozessarchitektur führt, welche die wichtigsten Prozesse der Organisation beschreibt. Außerdem werden Kennzahlen ermittelt, an denen der Erfolg der BPM-Initiative gemessen werden kann. Diese Zahlen betreffen die Kategorien Zeit, Kosten und Qualität.

Anschließend wird mit der *Prozesserhebung* begonnen. In dieser Phase wird das Ist-Prozessmodell entwickelt, was den aktuellen Prozess darstellt. Diese werden in einer Prozessmodellierungssprache wie BPMN, ULM oder EPK als Modell dargestellt. Dadurch wird der Prozess in eine verständliche Form gegossen, wodurch er für alle Stakeholder greifbar wird.

Nach der Prozesserhebung folgt laut Dumas u. a. (2021) die *Prozessanalyse*. In dieser Phase werden weitere Probleme identifiziert und Verbesserungsmöglichkeiten entwickelt. Dabei können Probleme in unterschiedlichen Formen vorkommen, wie etwa zu hohen Durchlaufzeiten, Engpässen oder Medienbrüchen.

Diese Erkenntnisse fließen dann in die nächste Phase mit ein: der *Prozessverbesserung*. Hier wird nun ein neues Prozessmodell, der Soll-Prozess, entwickelt, welche die Verbesserungsvorschläge der vorherigen Phase umsetzt.

Anschließend wird auf Basis des Sollprozesses die *Prozessimplementierung* eingeleitet. Hier werden alle Maßnahmen getroffen, damit der neue und verbesserte Prozess ausgeführt werden kann. Diese umfassen nicht nur die Automatisierung des Prozesses, also die Einführung und Anpassung der IT-Systeme, welche den Prozess unterstützen, sondern auch das organisationsbezogene Änderungsmanagement, was beispielsweise Schulungen der Mitarbeiter oder Definition von Übergangsregeln umfasst.

Ab der ersten Ausführung des neuen Prozesses beginnt die letzte Phase von dem BPM-Lebenszyklus nach Dumas u. a. (2021), die *Prozessüberwachung*. Hierbei werden laufend Daten über die Ausführung des Prozesses gesammelt und die Ausführung überwacht. Anhand dieser Daten wird festgestellt, ob die während der Prozessidentifikation festgelegten Kennzahlen erfüllt werden und ob weitere Probleme auftauchen.

Ist dies der Fall, so wird erneut eine Prozesserhebung erstellt. Mit dem Motto:

„Every good process eventually becomes a bad process.“ Hammer (2015)

wird der BPM-Lebenszyklus von neuem startet und so eine kontinuierliche Verbesserung des Prozesses gewährleistet.

### 2.1.2 Business Process Management System (BPMS)

Ein Business Process Management System (BPMS) ist ein Informationssystem, welches die Entwicklung und Verwaltung von Geschäftsprozessen unterstützt. Es enthält Werkzeuge, welche anhand von Prozessdarstellungen die Ausführung von Prozessen koordinieren (Weske, 2019).

„Ein BPMS verfolgt den Zweck, einen automatisierten Geschäftsprozess so zu koordinieren, dass alle Arbeiten zur richtigen Zeit von der richtigen Ressource ausgeführt werden.“ Dumas u. a. (2021)

Um dieses Ziel zu erfüllen, stellt ein BPMS neben der Prozessausführung und -koordination auch Werkzeuge bereit, welche die Gestaltung, Analyse und Überwachung von Geschäftsprozessen ermöglichen (Dumas u. a., 2021).

Wie auch bei den BPM-Lebenszyklen bestehen auch hier unterschiedliche Architekturmodelle eines BPMS, wobei für diese Arbeit die Architektur von König u. a. (2020) ausgewählt wurde, die in Abbildung 2.2 näher beleuchtet wird.

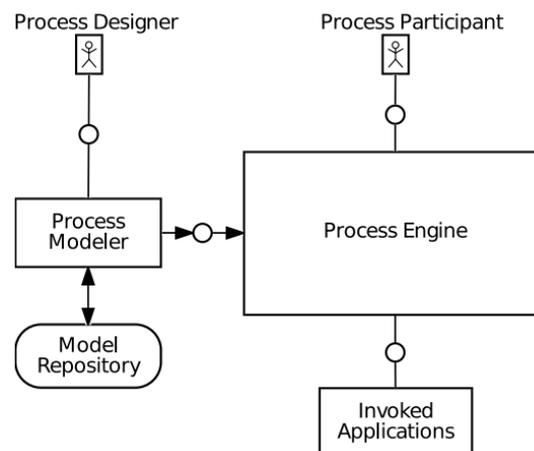


Abbildung 2.2: BPMS Architektur von König u. a. (2020)

Diese Architektur basiert auf den Angaben von Weske (2019), der die einzelnen Komponenten wie folgt beschreibt: Der *Prozess Modeller* stellt die Funktion bereit, vorhandene Prozessmodell mit technischen Details zur Ausführung zu versehen. Die fertigen Prozessmodelle werden anschließend im *Model Repository* gespeichert und zur Ausführung an die *Prozess Engine* übergeben. Wird ein Prozess gestartet, erstellt die Engine eine Prozessinstanz auf Basis des jeweiligen Prozesses und führt diese aus. Bei der Ausführung orientiert

sich die Engine an der im Prozess definierten Reihenfolge der Aktivitäten. Ob die Aktivität von der Engine selbst ausgeführt, diese an ein anderes System (*Invoked Application*) über ein passendes Programmierschnittstelle (engl.: application programming interface (API)) übergeben oder sie manuell über eine Benutzerschnittstelle (engl.: graphical user interface (GUI)) von einem Prozessbeteiligten durchführen wird, wird ebenfalls vom Prozess definiert. Alle zur Ausführung gehörenden Daten werden von der Engine gesammelt und dem Monitoring und Werkzeugen zur Prozessanalyse zur Verfügung gestellt.

### 2.1.3 Vor- und Nachteile eines BPMS

Ein BPMS bietet zahlreiche Vorteile, die zur Effizienz und Transparenz in der Organisation beitragen. Durch die Zuweisung der Aufgaben an Prozessbeteiligte und Systeme wird sichergestellt, dass diese auch zeitnahe ausgeführt wird. Die flexible Systemintegration erlaubt die unabhängige Aktualisierung und Modifikation von Anwendungen und Prozessen, was die Anpassungsfähigkeit an sich ändernde Geschäftsanforderungen erhöht. Zudem bietet die Ausführtransparenz Einblicke in den Ablauf und die Leistung von Prozessen durch operative und historische Informationen, die als Grundlage für die Prozessüberwachung dienen. Durch die feste Implementierung und Definition von Regeln wird deren strikte Durchsetzung gewährleistet, sodass Abweichungen vermieden werden. (Dumas u. a., 2021)

„While these features are still relevant, it seems fair to say that many organizations have found it difficult to overcome technical and organizational challenges that relate to adopting this technology [...]“ Reijers (2021)

Unter der technischen Herausforderung wird vor allem die die Integration von älteren Systemen, welche meist keine geeigneten APIs implementieren. So muss eine spezielle API entwickelt werden, was durch eine lückenhafte Dokumentation des Systems erschwert wird (Dumas u. a., 2021). Ist gar kein System vorhanden, welches das BPMS unterstützt, so muss dieses von Grund auf aufgebaut werden (Agostinelli u. a., 2020). Dies führt zu einem gewaltigen Mehraufwand, was die Kosten der Automatisierung erhöht.

Die organisatorische Herausforderung betrifft vor allem die Kommunikation mit den Prozessbeteiligten. Da Menschen von Natur aus Gewohnheitstiere sind, sind die meist skeptisch gegenüber Änderungen. Außerdem können sich im Bezug auf die Automatisierung auch Job- und Überwachungsängste bilden. Diese Aspekte müssen auf die richtige Art angegangen werden, sodass das BPMS von den Prozessbeteiligten akzeptiert wird. Eine

weitere Herausforderung ist, dass durch eine lange Entwicklungszeit der neue entwickelte Prozess zum Zeitpunkt Einführung wieder veraltet ist. (Dumas u. a., 2021)

Da die Einführungskosten für ein BPMS sehr hoch sind, werden meist nur die Prozesse voll automatisiert, die oft wiederholt werden und standardisiert sind. Bei seltenen Prozessen übersteigen jedoch oft die Kosten der Einführung die erwarteten Einsparungen (Weske, 2019; van der Aalst, 2020).

## 2.2 Robotic Process Automation (RPA)

Robotic Process Automation (RPA) ist eine softwarebasierte Lösung, die eine Reihe von Werkzeugen zur Verfügung stellt, welche den Vorgang der Prozessautomatisierung schnell und simpel umsetzen können (Czarnecki und Fettke, 2021). Dafür nutzt die RPA Software-Roboter, auch Bots genannt, die Geschäftsregeln und vordefinierte Aktivitäten nutzen, um diese Prozessen autonom in verschiedenen Softwaresystemen auszuführen, ohne dass Änderungen an bestehenden Anwendungen erforderlich sind (Willcocks u. a., 2015; Niculescu u. a., 2023). Dies wird anhand manuelle Eingaben durch Bots auf den vorhandenen GUIs erreicht, wodurch Interaktionen mit Systemen auf die gleiche Weise wie ein Mensch ermöglicht werden (Mendling u. a., 2018). Somit werden für die Anwendung keine API benötigt, was die RPA damit von traditionellen Prozessautomatisierungen durch BPMS oder Schnittstellenprogrammierung unterscheidet (Hofmann u. a., 2020; van der Aalst, 2021; Morelli und Karkos, 2022). Dabei ist das Aufgabenspektrum der Bots enorm.

„Examples of tasks that bots perform include data transfer between applications through screen scraping, automated email query processing, and collation of payroll data from different sources.“Syed u. a. (2020)

Anhand der von Syed u. a. (2020) aufgelisteter Beispiele können Merkmale von Prozessen und Aufgaben definiert werden, die für die Automatisierung durch RPA geeignet sind. Dazu gehören Prozesse, die auf mehrere Systeme zugreifen müssen, jedoch über keine passende API verfügen, wie es bei älteren Systemen der Fall ist. Auch Prozesse mit hochstandardisierten Aufgaben ohne Ausnahmen, sowie Aufgaben die kein subjektives Urteil und keine Kreativität erfordern sind geeignet. Aufgaben mit hohen Volumina, die häufig ausgeführt werden und einem hohen Risiko für menschliche Fehler profitieren besonders von der Automatisierung. (Uskenbayeva R. K. u. a., 2022)

### 2.2.1 RPA-Lebenszyklus

Wie auch das BPMS besitzt die RPA einen Lebenszyklus, der von Enriquez u. a. (2020) in sechs Phasen aufgeteilt ist.

Die erste Phase ist die *Analysephase*. Hier werden die Prozesse auf ihr Automatisierungspotential durch die RPA untersucht und eine Machbarkeitsanalyse durchgeführt. In der zweiten *Designphase* wird der durch die RPA zu automatisierende Prozess in seinem Ist-Zustand beschrieben und modelliert. Anschließend beginnt die *Konstruktionsphase*. Hier werden die zu automatisierenden Abschnitte des Prozesses mit der benötigten Ausführungslogik versehen. In der *Bereitstellungsphase* müssen die in der Konstruktionsphase erstellten Roboter in einer Umgebung ausgeführt werden, da sie ähnlich wie ein menschlicher Mitarbeiter eine Arbeitsumgebung benötigen. In der RPA-Umgebung entspricht diese Umgebung normalerweise einem Computer, auf dem ein oder mehrere Informationssysteme installiert sind. Jeder Roboter muss in seiner eigenen Ausführungsumgebung betrieben werden, da er direkt den menschlichen Mitarbeiter ersetzt. Anschließend beaufsichtigt die *Kontroll- und Überwachungsphase* kontinuierlich die Leistung jedes Roboters. Hierbei wird der Roboter ausgeführt, der Fortschritt der Aufgaben überwacht, bis alle Aufgaben abgeschlossen sind und potenziell schwerwiegende Fehler führen zum Stoppen der Ausführung. Die abschließende Phase des Prozesses umfasst die Evaluation und Beurteilung der Leistung der Roboter (*Evaluierungs- und Leistungsphase*).

### 2.2.2 Vor- und Nachteile einer RPA

Die Vorteile von RPA sind vielfältig und lassen sich direkte und indirekte Aspekte unterteilen.

Direkt profitiert RPA von der Fähigkeit, 24/7 zu arbeiten, was durch die automatisierten Bots ermöglicht wird, die Aufgaben schneller, präziser und fehlerfreier ausführen können (Santos u. a., 2020; Uskenbayeva u. a., 2019). Dies führt zu einer Reduktion von menschlichen Fehlern und einer konsistenten Qualität der Prozesse, während Mitarbeiter sich auf wichtigere Aufgaben konzentrieren können (Santos u. a., 2020).

Ein weiteres Vorteil ist, dass die RPA keine umfangreiche Programmierung erfordert, da RPA-Bots mit Low-Code- oder No-Code-Ansätzen entwickelt und über grafische Benutzeroberflächen konfiguriert werden (Flehsig u. a., 2022). Die Automatisierung ist dadurch schnell und ohne tiefgehende Programmierkenntnisse möglich, was die zügige Implementierung neuer Funktionen begünstigt und eine nahtlose Integration über die

Benutzeroberfläche in bestehende Systeme ermöglicht (Völker u. a., 2021; Santos u. a., 2020). Zusätzlich bietet RPA eine kostengünstige Skalierbarkeit, da entwickelte Roboter wiederverwendet werden können (Uskenbayeva u. a., 2019).

In Summe erhöht der Einsatz von RPA die betriebliche Effizienz durch die Reduktion von Human-, Finanz- und Zeitressourcen.

Indirekte Vorteile von RPA umfassen die Erhöhung der Zufriedenheit sowohl interner als auch externer Kunden durch eine verbesserte Servicequalität und Zuverlässigkeit. Durch die Automatisierung von Prozessen können Unternehmen ihre Dienstleistungen effizienter und konsistenter bereitstellen, was die Kundenbindung fördert und die Wettbewerbsfähigkeit stärkt. Darüber hinaus trägt RPA zur Kontinuität der Unternehmensleistung bei, indem es eine stabile und zuverlässige Arbeitsweise gewährleistet, die weniger anfällig für menschliche Fehler oder Schwankungen in der Leistung ist. (Stravinskienė, 2022)

Allerdings bringt der Einsatz von RPA auch Herausforderungen mit sich.

„A common criticism is that the integrations offered by RPA are less robust than those that are by nature embedded into core systems“ Syed u. a. (2020)

Infolge dessen entstehen Stabilitäts- und Wartungsprobleme. Geschehen noch zusätzlich Änderungen in den zugrundeliegenden IT-Systemen oder der GUI, kommen zeitaufwändige und kostspielige Anpassung der RPA hinzu. (Syed u. a., 2020; Flechsig u. a., 2019)

„[...] RPA vendors often only support if/else or switch constructs to steer the process.“ Flechsig u. a. (2019)

Dadurch können nur mit viel Aufwand komplexere Konstrukte implementiert werden, weshalb die Zahl der durch die RPA automatisierbaren Prozesse beschränkt wird. Da sich die Modellierungsphase im RPA-Lebenszyklus nur auf den Ist-Prozess beschränkt, besteht die Gefahr dass fehlerhafte und ineffiziente Prozesse automatisiert werden, sodass Fehler aufkommen und das volle Automatisierungspotential nicht ausgeschöpft werden kann. Zudem bietet die RPA keine umfassende Überwachung und Kontrolle der automatisierten Prozesse an, was die Identifikation von Optimierungspotentialen und das Fehlerhandling erschwert. (Flechsig u. a., 2019)

## 3 Verwandte Arbeiten

Die Verbindung von Business Process Management Systems (BPMS) und Robotic Process Automation (RPA) birgt ein enormes Potenzial zur Optimierung und Automatisierung von Geschäftsprozessen. Diese Synergien ermöglichen es Unternehmen, ihre Prozesslandschaft effizienter und flexibler zu gestalten, wobei die jeweiligen Stärken von BPMS und RPA voll ausgeschöpft werden können. Trotz des enormen Potenzials gibt es bislang nur eine begrenzte Anzahl an wissenschaftlichen Arbeiten, die sich explizit mit der Integration von BPMS und RPA auseinandersetzen.

### 3.1 Potential eines BPM-RPA Systems

Ein wesentliches Potenzial der Integration von einem BPM-RPA System liegt in der verbesserten Prozessoptimierung. Durch die umfassende Plattformen zur Modellierung, Analyse und Optimierung von Geschäftsprozessen, die ein BPMS bietet, ist die systematische Untersuchung und Neugestaltung der Prozesse möglich.

„It [the RPA-to-BPM integration] allows to use existing techniques of BPM in order to design, configure, enact, and evaluate RPA processes.“ König u. a. (2020)

Somit können ineffiziente Schritte identifiziert und eliminiert werden. Diese Kombination von BPMS und RPA sorgt außerdem dafür, dass nicht nur bestehende Prozesse automatisiert, sondern auch deren Effizienz und Effektivität maximiert werden. (Flehsig u. a., 2019; Signavio, 2019)

Darüber hinaus bietet die Integration von BPMS und RPA eine verbesserte Überwachung und Kontrolle der Geschäftsprozesse. BPMS-Systeme sind darauf ausgelegt, eine umfassende Transparenz über die laufenden Prozesse zu bieten und diese kontinuierlich zu überwachen. Durch die Integration von RPA kann die Überwachung auch auf automatisierte Prozesse ausgedehnt werden. Dies ermöglicht eine schnelle Erkennung und

Behebung von Problemen sowie eine kontinuierliche Anpassung (Flechsig u. a., 2019) Ein weiteres Potenzial der Kombination von BPMS und RPA liegt in der erhöhten Flexibilität und Skalierbarkeit. Während BPMS eine langfristige und strategische Prozessoptimierung ermöglicht, kann RPA schnell und flexibel auf kurzfristige Anforderungen reagieren. Unternehmen können somit auf veränderte Marktbedingungen oder neue gesetzliche Vorgaben reagieren, indem sie ihre Prozesse entsprechend anpassen und die Automatisierung durch RPA schnell skalieren. Dies ist besonders in dynamischen und sich schnell verändernden Geschäftsumfeldern von Vorteil, wo Flexibilität und Reaktionsfähigkeit entscheidend sind. (Flechsig u. a., 2019)

Ein weiteres Potential eines BPM-RPA Systems wurde von van der Aalst u. a. (2018) erkannt und durch Flechsig u. a. (2019) aufgegriffen. Dieses sagt aus, dass sich die Menge der potentiell automatisierbaren Prozesse durch die Integration von RPA in BPMS erheblich erhöht. Erklärt wird das durch die Pareto-Verteilung, welche in der Abbildung 3.1 als Graph dargestellt ist. Die x-Achse zeigt die Falltypen an. Prozesse, die auf die selbe Art behandelt werden, gehören zur selben Falltypus. Die y-Achse stellt die Frequenz der Fälle dar, also die die Anzahl der Fälle in einem begrenzten Zeitraum. Meist folgt die Verteilung dem Prinzip der Pareto-Verteilung. Die sagt aus, dass 80% der Fälle durch 20% der Falltypen erklärt werden kann. Diese Typen sind ideal für eine kostengünstige und schnelle Automatisierung durch BPMS, da sie meist standardisiert, strukturiert und in hoher Frequenz vorkommen. Die restlichen 80% der Falltypen werden meist manuell durch Menschen durchgeführt und beinhalten eintönige Arbeit oder Aufgaben, bei denen Entscheidungen getroffen werden müssen. Ein großer Teil dieser 80% kann, wenn es möglich und wirtschaftlich sinnvoll ist, von der RPA automatisiert werden, wodurch die Arbeitslast der Menschen reduziert werden kann.

Ein praktisches Beispiel für das Potenzial der Integration von BPMS und RPA liefert die Fallstudie von Flechsig u. a. (2019). In dieser Studie wurde ein administrativer Prozess zunächst mittels BPMS analysiert und optimiert, bevor die optimierten Prozessschritte durch RPA automatisiert wurden. Das Ergebnis war eine signifikante Reduzierung der Prozesskosten und eine Erhöhung der Prozessgeschwindigkeit. König u. a. (2020) unterstützten diese Ergebnisse durch ihre Untersuchung eines optimierten Beschaffungsprozesses, der durch RPA automatisiert wurde. Auch hier führten die Integration und Automatisierung zu einer deutlichen Reduktion der Bearbeitungszeit und einer Erhöhung der Prozessgenauigkeit.

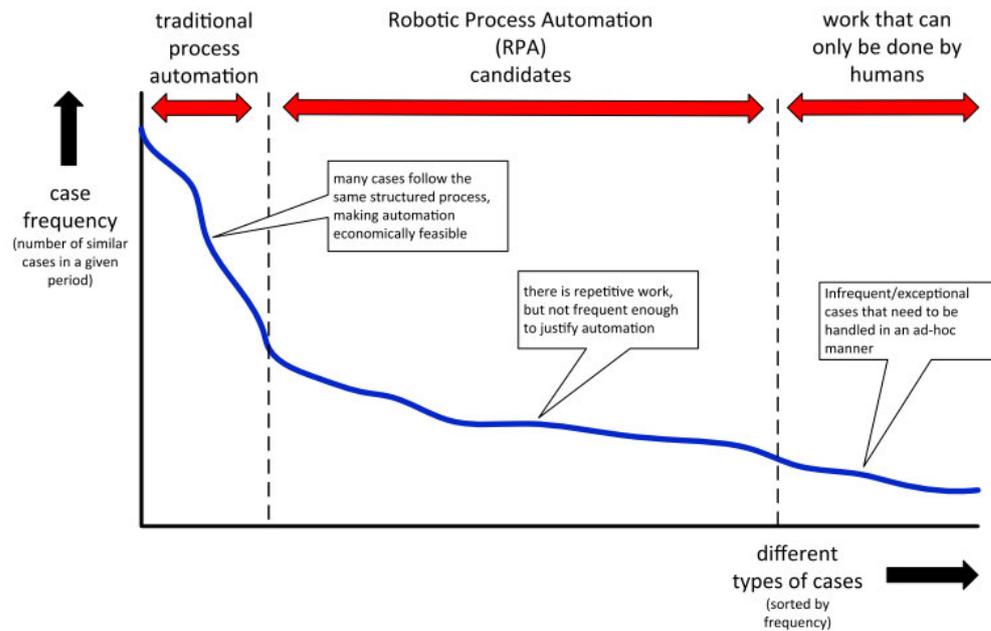


Abbildung 3.1: Pareto-Verteilung von Prozessen nach van der Aalst u. a. (2018)

### 3.2 Herausforderungen eines BPM-RPA Systems

Trotz der zahlreichen Vorteile und des erheblichen Potenzials gibt es auch Herausforderungen bei der Integration von BPMS und RPA. Eine der größten Herausforderungen ist die technische Komplexität der Integration. Die Abstimmung zwischen den beiden Systemen erfordert eine sorgfältige Planung und Koordination, um sicherzustellen, dass die Prozesse nahtlos ineinandergreifen. Darüber hinaus müssen die Mitarbeiter entsprechend geschult und in den Veränderungsprozess eingebunden werden, um die Akzeptanz und den erfolgreichen Einsatz der neuen Technologien zu gewährleisten. (Flehsig u. a., 2019)

Flehsig u. a. (2019) und König u. a. (2020) weisen auch auf die Abhängigkeit von der bestehenden IT-Infrastruktur hin. Die erfolgreiche Implementierung von BPMS und RPA setzt voraus, dass die IT-Systeme modern und standardisiert sind. Insbesondere bei älteren oder stark fragmentierten Systemlandschaften kann die Integration auf technische Schwierigkeiten stoßen. Unternehmen müssen daher möglicherweise ihre IT-Infrastruktur anpassen und modernisieren, um die Vorteile der Kombination von BPMS und RPA voll ausschöpfen zu können.

## 4 Konzept

Nach dem die Grundlagen behandelt worden sind, wird nun das Konzept, der in dieser Arbeit entwickelten Process-Engine, erstellt. Dafür müssen Anforderungen definiert werden, an denen sich die Entwicklung orientiert. Anschließend wird die Systemarchitektur entwickelt, welche auf Basis der Anforderungen entwickelt wird und als Startpunkt für die Implementierung dient.

### 4.1 Anforderungsanalyse

Um die Anforderungen für die Process-Engine zu definieren, ist es von Vorteil erst die Anforderungen für das BPM-RPA System zu definieren und daraus anschließend die Anforderungen für die Engine abzuleiten.

#### 4.1.1 Anforderungen des Gesamtsystems

Das Grundkonzept des BPM-RPA Systems basiert auf den Analysen von Flechsig u. a. (2022) und König u. a. (2020). Diese Forschung definiert Anforderungen, die ein BPM-RPA System erfüllen sollte. Diese Anforderungen werden hier in funktionale (SF1 - SF5) und nicht-funktionale Systemanforderungen (SNF1 - SNF3) unterteilt und werden in den beiden Tabellen 4.1 und 4.2 zusammengefasst.

Gemäß deren Erkenntnissen sollte das BPM-RPA System auf einem BPMS basieren, um Prozesse effektiv zu automatisieren, zu steuern und zu überwachen (SF1). Diese Basis ermöglicht den Einsatz ausgefeilter Verfahren zur Prozessdokumentation, -analyse und -orchestrierung sowie Schnittstellen zur Integration externer Anwendungen, wie etwa Process-Mining-Tools. Dadurch kann auch die Leistungsfähigkeit des RPA-Systems verbessert werden, da der RPA-Prozess durch die Verfahren und Integrationen des BPMS optimiert und überwacht wird. (Flechsig u. a., 2022; König u. a., 2020) Die Wirksamkeit

dieses Ansatzes hängt jedoch von der Reife des BPMS ab, um die Verfahren und Tools in zufriedenstellender Qualität auf die Prozesse anzuwenden (SNF1). Fehlt diese Reife, so können die Vorteile von BPMS nicht voll genutzt werden, was sich negativ auf das BPM-RPA System auswirkt. Außerdem fördert die Verwendung eines BPMS als Basis die richtige Mentalität der Benutzer gegenüber RPA (SNF2). Da das RPA-System als eine Erweiterung zum BPMS gesehen wird und nicht als kostengünstiger Ersatz, wird auch hier gewährleistet, dass beide Systeme ihr volles Potenzial ausschöpfen können. (Flehsig u. a., 2022)

Anforderungs-ID	Anforderung
SNF1	Das System soll eine gewisse BPM-Reife besitzen.
SNF2	RPA soll als Ergänzung zum BPM dienen.
SNF3	Das System sollte benutzerfreundlich und attraktiv gestaltet sein.

Tabelle 4.1: Nicht-funktionale Systemanforderungen

Um dies zu erreichen, sollte sowohl die Koordination zwischen BPM-RPA System und User als auch die zwischen BPMS und RPA-Bots möglichst reibungslos verlaufen (SF2). Dies kann durch eine konzentrierte Aufgabenorchestrierung erreicht werden, die transparente Schnittstellen, fundierte Entscheidungslogik und vordefinierte Regeln für Anfragen bietet, sowie Funktionalitäten für die synchrone und asynchrone Mensch-Bot-Zusammenarbeit bereitstellt. (Flehsig u. a., 2022) Zudem sollten sowohl das BPMS als auch die RPA ihre volle Funktionalität ausnutzen können (SF3). Folglich sollte die Verbindung der Systeme so erfolgen, dass es diese in ihrer Funktionalität nicht einschränkt und somit ihr volles Potential ausgeschöpft wird. (König u. a., 2020)

Flehsig u. a. (2022) beschreibt außerdem, dass eine konsistente Prozessmodellierungssprache benötigt wird (SF3), um standardisierte Designs von BPMS- und RPA-Prozessen zu ermöglichen und automatisch zugehörige Flussdiagramme zu generieren. Dies erleichtert die nahtlose Integration der RPA in das BPMS. Da BPMN 2.0 in vielen qualitativ hochwertigen BPMS, wie SAP<sup>1</sup> oder Camunda<sup>2</sup>, verwendet wird, bietet sich BPMN 2.0 als Prozessmodellierungssprache für das BPM-RPA System an.

Schließlich sollte das System nur über eine begrenzte Komplexität verfügen (SF4). Dies bedeutet, dass es idealerweise eine Low- oder No-Code-Plattform für die Prozessimplementierung sein sollte. (Flehsig u. a., 2022) Außerdem sollten keine Engine-spezifischen Konfigurationen während der Implementierung nötig sein. (König u. a., 2020)

---

<sup>1</sup><https://www.sap.com/germany/products/technology-platform/process-orchestration.html>

<sup>2</sup><https://camunda.com/de/platform/>

Zusätzlich zu den Anforderungen von Flechsig u. a. (2022) und König u. a. (2020) sollte das BPM-RPA System attraktiv und benutzerfreundlich gestaltet werden (SNF3). Damit soll gewährleistet werden, dass das System auch von den Prozessbeteiligten benutzt wird und somit die Vorteile einer solchen Verbindung ideal ausgeschöpft werden.

Anforderungs-ID	Anforderung
SF1	Das BPMS soll für das Modellieren, Analysieren und Dokumentieren des Prozesses, sowie für das Starten, Steuern und Überwachen der Ausführung verantwortlich sein.
SF2	Eine zentrale Aufgabenorchestrierung zwischen Anwender, Bot und BPMS soll gewährleistet sein.
SF3	Die beiden Systeme sollen sich in ihrer Funktionalität nicht gegenseitig einschränken.
SF4	BPMN 2.0 soll als konsistente Prozessmodellierungssprache genutzt werden.
SF5	Im Bezug auf die Darstellung und Implementation soll die Komplexität begrenzt werden (low-code/ no-code).

Tabelle 4.2: Funktionale Systemanforderungen

Da nun die grundlegenden Anforderungen für das BPM-RPA System erörtert wurden, beschäftigt sich der nächste Abschnitt mit der genauen Spezifikation der Anforderungen für die Process-Engine.

#### 4.1.2 Anforderungsanalyse für die Process Engine

Basierend auf den allgemeinen Anforderungen für das BPM-RPA System, wie sie im Abschnitt 4.1.1 beschrieben sind, lassen sich nun spezifische Anforderungen für den in dieser Arbeit zu entwickelnde Process-Engine-Prototyp ableiten, der die ausführende Komponente des RPA-BPM Systems darstellt. Diese werden anschließend als Grundlage zur Implementierung des Prototypen genutzt und werden abschließend im Kapitel 7 mit dem Istzustand des Prototypen gegenübergestellt. Wie auch im vorherigen Abschnitt lassen sich die Anforderungen in funktionale (F1 - F4) und nicht-funktionale (NF1 - NF5) unterteilen und sind in den Tabelle 4.3 und 4.4 zusammengefasst.

Eine der entscheidenden funktionalen Anforderungen ist die Prozessausführung (F1). Als Kernfunktion des RPA-BPM Systems muss die Engine in der Lage sein, sowohl die traditionellen BPMN-Prozesse als auch solche mit RPA-Funktionalität automatisch auszuführen.

Dabei muss der Prototyp alle grundlegenden Funktionen und Aufgaben ausführen können, die im vordefinierten BPMN 2.0-Modell enthalten sind und diese korrekt interpretieren (F2). Hierbei ist entscheidend, dass die Engine die Prozesse und RPA-Aufgaben versteht und ausführt.

Des Weiteren muss eine rudimentäre Synchronisation zwischen RPA- und Nicht-RPA-Aufgaben implementiert werden (F3), um die fehlerfreie Ausführung des Prozesses zu gewährleisten. Da es trotz sorgfältiger Planung jedoch immer wieder zu Fehlern kommen kann, muss ein einfacher Mechanismus zur Erkennung und Behandlung von Fehlern während der Programmausführung implementiert werden (F4). Dieser Mechanismus soll in Form eines Prozessabbruchs und der anschließenden Protokollierung der Fehlermeldung im Log realisiert werden.

Anforderungs-ID	Anforderung
F1	Prozessausführung
F2	Interpretation von BPMN-Diagrammen
F3	Synchronisation von RPA- und Nicht-RPA-Aufgaben
F4	Fehlerbehandlung und Ausnahmehandling

Tabelle 4.3: Funktionale Anforderungen für die ausführende Komponente

Als nicht-funktionale Anforderungen können Robustheit und Zuverlässigkeit des Prototyps definiert werden (NF1). Damit wird sichergestellt, dass die Prozesse korrekt und ohne Ausfälle ausgeführt werden. Ebenso sollte der Prototyp benutzerfreundlich und portabel sein (NF2), was bedeutet, dass dieser einfach zu installieren und auszuführen ist, ohne dass komplexe Konfigurationen nötig sind.

Zudem sollte der Prototyp plattformunabhängig funktionieren, was die Nutzung auf verschiedenen Betriebssystemen ermöglicht. Eine weitere wichtige Anforderung betrifft die Wartbarkeit des Prototyps (NF3), was auch die Langlebigkeit und Versionsunabhängigkeit der beiden Engine voraussetzt.

Die Implementierung sollte gut strukturiert sein, um den Wartungsaufwand zu minimieren. Es soll sichergestellt werden, dass der Prototyp leicht aktualisiert und erweitert werden kann, um zukünftige Anforderungen zu erfüllen und die Process-Engine mit möglichen anderen Komponenten des RPA-BPM Systems zu integrieren. Dies erfordert eine angemessene Dokumentation des Prototyps, um die Verwendung und Wartung zu erleichtern (NF4). Abschließend muss der Prototyp eine bestimmte Leistungsfähigkeit aufweisen (NF5). Seine Performance sollte ausreichend sein, um Prozesse in angemessener Zeit auszuführen, ohne dabei übermäßig Ressourcen zu beanspruchen.

Anforderungs-ID	Anforderung
NF1	Zuverlässigkeit und Robustheit
NF2	Benutzerfreundlichkeit und Portabilität
NF3	Wartbarkeit und Erweiterbarkeit
NF4	Dokumentation
NF5	Performance

Tabelle 4.4: Nicht-funktionale Anforderungen für die ausführende Komponente

## 4.2 Systemarchitektur

Da nun die spezifischen Anforderungen für die Engine definiert wurden, kann auf dieser Basis die Systemarchitektur des BPM-RPA Systems entwickelt werden. Wie bereits in Abschnitt 4.1.1 erwähnt, basiert die in dieser Arbeit entwickelte Idee auf den Artikeln von Flechsig u. a. (2022) und König u. a. (2020). Diese Artikel erkannten das Potential eines Systems, das klassische BPMS und einfache RPA kombiniert, und entwickelten entsprechende Konzepte zur Umsetzung.

König u. a. (2020) entwickelte einen Adapter, der generisch ein BPMS und ein RPA-System miteinander verbindet. Dies ermöglicht eine vielfältige und plattformunabhängige Anwendung, wobei die Implementierung jedoch getrennt bleibt.

Flechsig u. a. (2022) führte Experteninterviews durch, die eher eine ganzheitliche Lösung vorschlugen. Diese Lösung sah insbesondere eine einheitliche Modellierungssprache vor, weshalb Flechsig u. a. (2022) ein System konzipierte, in der die RPA fest in die BPMS integriert ist.

In diese Arbeit das Konzept von Flechsig u. a. (2022) aufgegriffen und weiterentwickelt. Die entwickelte Systemarchitektur wird in Abbildung 4.1 dargestellt und besteht aus vier Komponenten: dem Prozess Modeller, dem RPA Parser, der BPM-Engine und der RPA-Engine.

### 4.2.1 Der Prozess Modeller

Der Modeller stellt eine Benutzeroberfläche zur Verfügung, auf der sowohl BPM- als auch RPA-Prozesse modelliert und miteinander verbunden werden können. Dabei wird ausschließlich die Prozessmodellierungssprache BPMN 2.0 genutzt. Flechsig u. a. (2022) sieht hier ein Drei-Schichten-Modell vor, das den Prozess in unterschiedlichen Granularitäten darstellt. Die von der RPA auszuführenden Aktivitäten sollen stets in der dritten und feinsten Stufe modelliert werden.

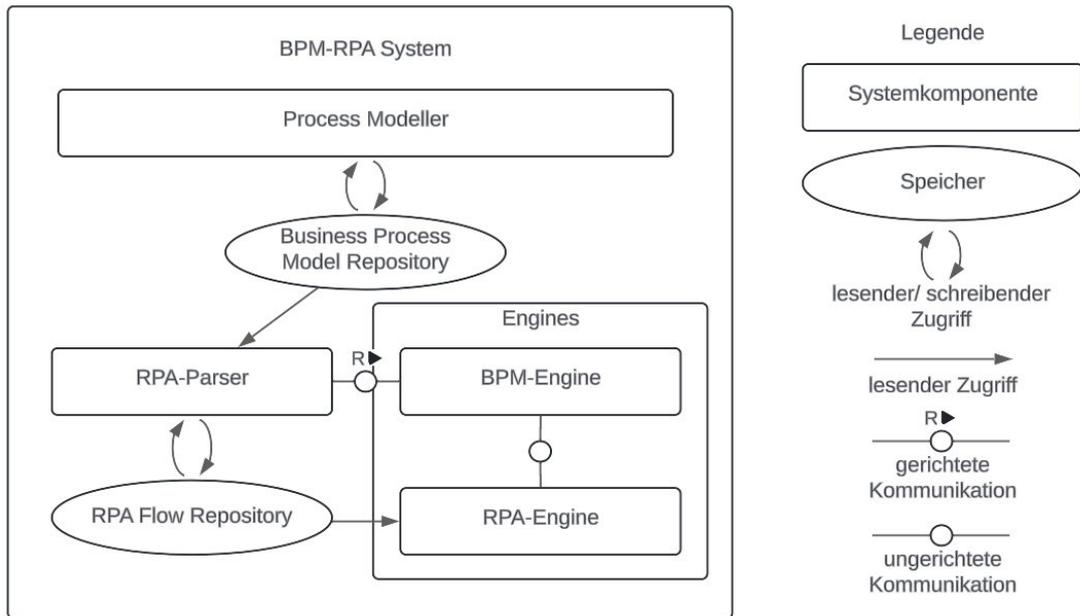


Abbildung 4.1: Systemarchitektur des BPM-RPA Systems

Zwar ist es sinnvoll die RPA-Aktivitäten in einer Subprozess zu bündeln, da diese im Vergleich zu einer BPM-Aktivität technischer und detaillierter sind, jedoch ist dieses Konzept sehr einschränkend. Prozesse könnten unnötig komplex gestaltet werden, um dem von Flehsig u. a. (2022) vorgeschlagenen Modell zu entsprechen. Ein einfacher Prozess, der keine Unterteilung in drei Schichten benötigt, müsste künstlich mit Subprozessen gestreckt werden, während ein sehr komplexer Prozess auf maximal drei Stufen beschränkt wäre, was die Verständlichkeit und Übersichtlichkeit beeinträchtigt.

Um diese Problem zu beheben, verzichtet der hier beschriebene Prozess Modeller auf eine solche Beschränkung. Stattdessen verfügt er über eine eigene Aktivität, die RPA-Aktivitäten als solche identifiziert. (siehe Abbildung 4.2)



Abbildung 4.2: RPA-Aktivität im Modeller

Im XML des modellierten BPMN wird ein eigener Namensraum verwendet, um die Aktivität zu kennzeichnen.

```
<rpa:Task id="Activity_ow5qe824" name="RPA-Aktivität" .../>
```

Nach der Modellierung und Implementierung wird das BPMN 2.0 im Business Process Model Repository als eine .bpmn- oder bpmn20.xml-Datei gespeichert.

### 4.2.2 Der RPA-Parser

Nachdem der Prozess fertig modelliert wurde, wird die Datei deployed und der Engine zur Verfügung gestellt. Zuvor übersetzt jedoch der RPA Parser die in der Datei enthaltenen RPA-Aktivität in ein Format, welches der RPA-Engine die Ausführung ermöglicht. Bei mehreren aufeinander folgenden RPA-Aktivitäten werden diese durch den RPA Parser in eine RPA-Datei zusammengefasst und im RPA Flow Repository gespeichert.

Anstelle der vom Modeller definierten RPA-Aktivität und dem Namensraum in der BPMN-Datei wird eine Service-Aktivität gesetzt. Diese enthält unter anderem eine Referenzen auf die Schnittstellenklasse der BPM-Engine zur RPA-Engine, die RPA-Datei und die für Ausführung benötigten Prozessvariablen. Dadurch wird gewährleistet, dass die RPA-Engine alle erforderlichen Informationen für die Ausführung erhält.

Die nun überarbeitete BPMN-Datei wird zusätzlich von der BPM-Engine inspiziert, geparsed und in der internen Datenbank der BPM-Engine gespeichert.

### 4.2.3 Die Engines

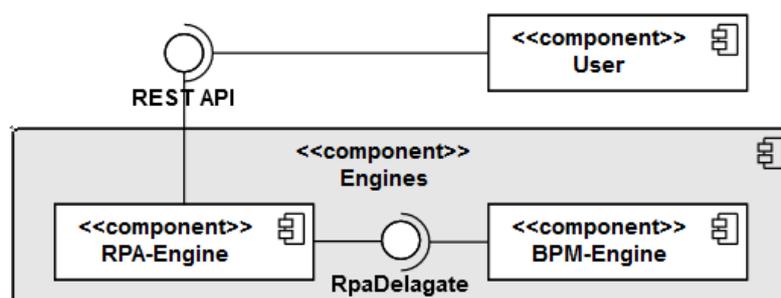


Abbildung 4.3: UML Komponentendiagramm der Engines

Die letzten beiden Komponenten des BPM-RPA Systems sind die Engines für die Ausführung der BPM- und der RPA-Aktivitäten. Diese werden noch einmal separat in Abbildung 4.3 dargestellt und bilden die Grundlage für die Implementierung.

Während die RPA-Engine lediglich den Prozess ausführt und die Informationen über die Ausführung an die Schnittstelle der BPM-Engine weitergibt, muss die BPM-Engine neben der eigenen Ausführung den Prozess starten, RPA-Aktivitäten erkennen, sie an die RPA-Engine weiterleiten und das Fehlerhandling übernehmen. Diese Logik wird in der Schnittstellenklasse `RpaDelegate` implementiert. Mögliche Tools zur Prozessanalyse, -optimierung und -überwachung werden ebenfalls von der BPM-Engine verwaltet, sowie ein REST API, welche die Kommunikation zwischen User und Engine ermöglicht.

## 5 Anwendungsbeispiel

In diesem Kapitel wird nun das Anwendungsbeispiel definiert. Anhand dessen soll die in Kapitel 6 entwickelten BPM-RPA Engine umfassend getestet werden. Dieser Prozess entspricht im Ansatz den Prozessen, welche bei einer Bewerbung an einem Institut ausgelöst werden, ist jedoch an die Anforderungen dieser Arbeit angepasst. Das entsprechende BPMN ist in Abbildung 5.1 dargestellt.

Wird der Prozess gestartet, überprüft er zunächst, ob eine Bewerbung eingegangen ist. Ist dies nicht der Fall, wird der Prozess beendet. Andernfalls wird der Bildungsabschluss des Bewerbers eingeordnet und die Abschlussnote bei einem höheren Abschluss verbessert. So führt beispielsweise ein Masterabschluss zu einer Verbesserung der Note um 0,4 Punkte, da er höher eingeordnet wird als eine Hochschulreife.

Sowohl die Prüfung einer vorhandenen Bewerbung als auch die Einordnung werden durch eine Serviceaktivität realisiert, deren Implementierung in einer Delegate-Klasse erfolgt. Bleibt nach der Einordnung die Note des Bewerbers zu schlecht, so wird die Bewerbung direkt abgelehnt. Bei einer überdurchschnittlich guten Note wird der Bewerber automatisch akzeptiert. Liegt die Note im mittleren Bereich, so entscheidet ein Mensch über die Annahme oder Ablehnung der Bewerbung.

Anschließend soll die Mailadresse aus der Bewerbung extrahiert und dem Bewerber eine Absage oder Zusage geschickt werden. Dies soll von einer RPA übernommen werden. Abschließend wird der Prozess beendet.

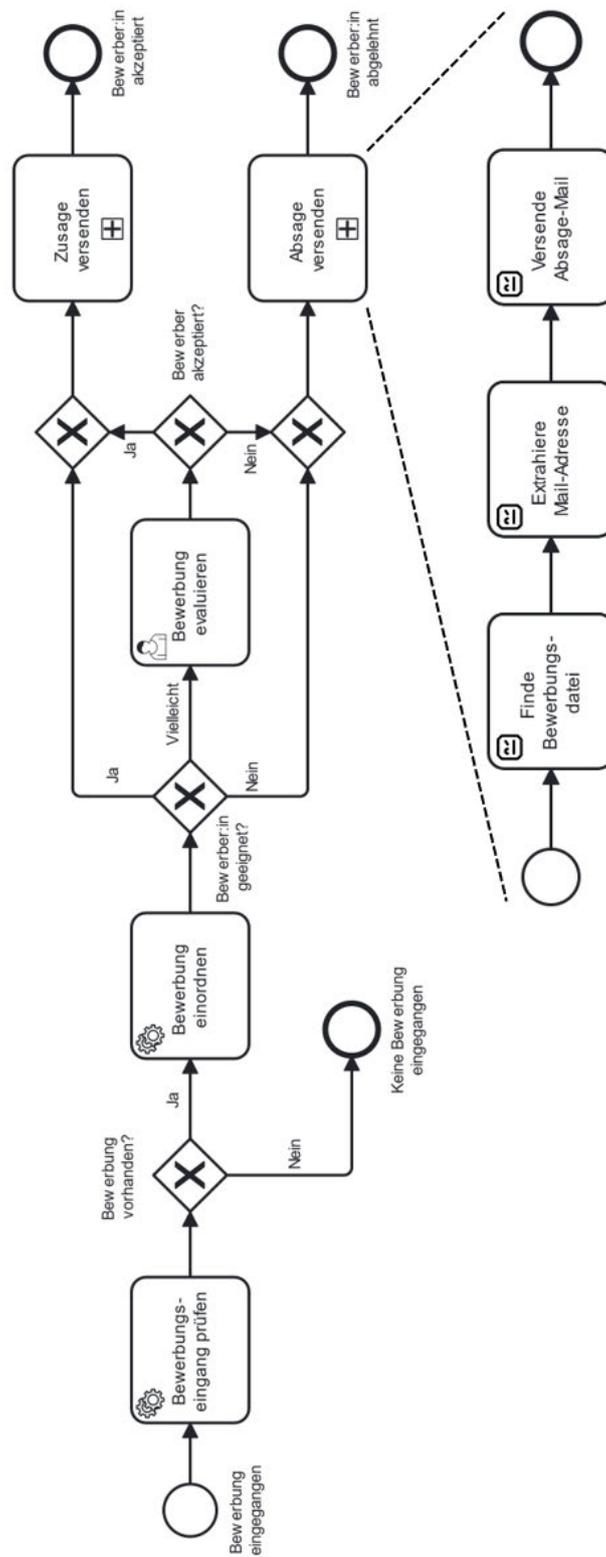


Abbildung 5.1: BPMN des Bewerbungsprozesses

## 6 Umsetzung

Ausgehend von den in Kapitel 4 formulierten Anforderungen und der in Abbildung 4.3 dargestellten Systemarchitektur wird im Folgenden die Umsetzung der beiden ausführenden Komponenten sowie einer rudimentären REST-API als Benutzerschnittstelle beschrieben.<sup>1</sup> Zur Verdeutlichung der Ausführung wird die Implementierung anhand des Bewerbungsprozesses illustriert, welcher in Kapitel 2 beschrieben und entwickelt wurde. Das dazugehörige BPMN ist in der Abbildung 5.1 dargestellt.

### 6.1 Auswahl der Engines

Die Auswahl der geeigneten Engines für die Prozessausführung ist ein entscheidender Schritt in der Umsetzung. Für die Definition der Kriterien wurden die Anforderungen aus dem Kapitel 4.1.2 berücksichtigt.

Die BPM-Engine sollte Open Source sein, da dies den Vorteil bietet, dass der Quellcode frei zugänglich und anpassbar ist. Eine aktive Community sorgt für eine kontinuierliche Weiterentwicklung, Fehlerbehebung und umfangreiche Dokumentation, was die Langlebigkeit der Software sicherstellt. (Red Hat, Inc. (2023); Vollmer und Schmitt (2020)) Außerdem sollte die Engine in der Lage sein BPMN 2.0 Diagramme zu automatisieren, um den oben definierten Anforderungen zu entsprechen.

Auf Basis dieser Kriterien fiel die Wahl auf die BPM-Engine von Activiti. Activiti<sup>2</sup> ist eine BPM Plattform welche in BPMN 2.0 dokumentierte Geschäftsprozesse automatisieren kann. Die Engine ist vollständig in Java geschrieben und kann sowohl als Cloud- als auch als On-Premise-Lösung genutzt werden. Für diese Arbeit wurde die On-Premise-Version gewählt.

---

<sup>1</sup><https://github.com/sofia-knap/BPM-RPA-System>

<sup>2</sup><https://www.activiti.org/>

Die RPA-Engine sollte Open Source sein und eine gute Dokumentation besitzen. Das Robot Framework, das von Unternehmen wie Cisco und Nokia verwendet wird Robot Framework<sup>3</sup>, erfüllte diese Kriterien. Dieses Framework ist Open-Source und hat sich auf dem Bereich Bot-Testing und Robotic Process Automation spezialisiert. Es besitzt eine sehr große und aktive Community, welche das Framework immer weiter ausbaut, verbessert und dokumentiert. Für die Automatisierung nutzt das Framework eine eigene auf Stichwörtern basierte Skriptsprache, die eine hohe Verständlichkeit und einfache Implementierung der Bots ermöglicht. Die in der RobotFramework spezifischen Sprache verfassten Dateien werden als .robot-Dateien gespeichert und sind so vom RobotFramework ausführbar.

## 6.2 Implementierung der Engines

Wie schon im Kapitel 4.2.3 erwähnt erfolgt die Kommunikation zwischen der BPM-Engine und der RPA-Engine über eine Schnittstellenklasse, die von der BPM-Engine zum richtigen Zeitpunkt aufgerufen wird. Dies wird durch die Darstellung der RPA-Aktivität als Serviceaktivität mit Verweis auf die Klasse `RpaDelegate` in der BPMN-Datei gewährleistet. Diese Klasse implementiert das Interface `JavaDelegate`, welches eine Methode `execute()` definiert.

Um die RPA Aktivität ausführen zu können, muss der genaue Speicherort zur .robot Datei ermittelt werden, welche zusammen mit anderen .robot-Dateien im RPA Flow Repository gespeichert ist. Dieser Pfad wird in der Variable `robotPath` im BPMN gelesen und lokal in der Variable `robotFilePath` gespeichert. Anschließend wird eine Instanz von der Klasse `ProcessBuilder` erstellt, die den folgende Befehl definiert, welcher dann anschließend in der für das Betriebssystem spezifischen Konsole ausgeführt wird.

```
ProcessBuilder pb = new ProcessBuilder(
    "robot", "--exitonfailure",
    "--variable", "PROCESS_INSTANCE_ID:" + processInstanceId,
    "--variable", "CUSTOM_VARIABLE:" + robotVariable, robotFilePath
);
Process robotProcess = pb.start();
```

---

<sup>3</sup><https://robotframework.org/>

Dabei gibt `robotFilePath` den Pfad zur auszuführenden `.robot`-Datei an. Da das `RobotFramework` sowohl zum Testen mit Bots als auch zur RPA-Ausführung genutzt wird, kann nicht vorausgesetzt werden, dass der Bot nach einer fehlerhaften Ausführung eines Befehls, die gesamte Ausführung beendet. Um sicherzustellen, dass bei einem Fehler die folgenden RPA-Aktivitäten nicht ausgeführt werden und weitere Fehler verursachen, wird die Option `--exitonfailure` dem Befehl hinzugefügt.

```
--variable PROCESS_INSTANCE_ID: [processInstanceId]
--variable CUSTOM_VARIABLE: [robotVariable]
```

Zusätzlich wird die Prozessinstanz-ID als Variable übergeben, sodass der Bot auf die von der Instanz erstellten Dateien zugreifen kann. Weitere Prozessvariablen können ebenfalls als Parameter übergeben und in der RPA-Ausführung verfügbar gemacht werden, wobei mehrere Prozessvariablen in einer `Map<String, ? extends Object>` zusammengefasst werden müssen.

```
*** Settings ***
[...]

*** Variables ***
${PROCESS_INSTANCE_ID}    ${PROCESS_INSTANCE_ID}
${CUSTOM_VARIABLE}       ${CUSTOM_VARIABLE}

*** Tasks ***
[...]
```

Listing 6.1: Deklaration der Prozessvariablen in `.robot`

Wird während der Ausführung des Bots eine Prozessvariable geändert, so wird diese Änderung im Log des Bots dokumentiert. Die `RpaDelegate`-Klasse überwacht den ausgehenden Datenstrom nach dem Starten des Bots und dokumentiert alle relevanten Informationen wie den Status der Ausführung und Änderungen der Prozessvariablen. Läuft die RPA-Aktivität auf einen Fehler, wird dieser ebenfalls von der Schnittstellenklasse über den `InputStream` erfasst.

Ist die RPA-Ausführung auf einen Fehler gelaufen, wird in der Klasse `RpaDelegate` die Prozessvariable `taskStatus` auf `FAILURE` gesetzt und der Prozess wird abgebro-

chen. War die Ausführung erfolgreich, wird die Variable auf SUCCESS gesetzt und die Prozessausführung in der BPM-Engine fortgesetzt.

### 6.3 Die REST API

Nach der Implementierung der Verbindung zwischen den beiden ausführenden Komponenten kann der Bewerbungsprozess automatisiert werden. Der Ablauf dieses Prozesses wird in Abbildung 6.1 dargestellt.

Für das Starten des Prozesses und das Ausführen der User-Aktivität wird eine Benutzerschnittstelle benötigt, die durch eine REST-API bereitgestellt wird. Diese API bietet eine Auswahl von POST- und GET-Anfragen, um mit der BPM-Engine zu kommunizieren. Da die Activiti Engine über einen Login verfügt, muss jede Anfrage mit einer Basisauthentifizierung in Form von Benutzername und Passwort mitschicken.

Um den Bewerbungsprozess zu starten, sendet der Benutzer die folgende Anfrage an die API:

```
GET http://localhost:8080/start-process/{processDefinitionKey}
```

Dabei entspricht die Variable `processDefinitionKey` dem Schlüssel des Bewerbungsprozesses. War die Anfrage erfolgreich so wird der Prozess gestartet und die ID der entstandenen Prozessinstanz zurückgegeben.

Ist keine Bewerbung vorhanden oder befindet sich die Bewerbung nicht im mittleren Bereich, kann der Prozess ohne weiteren Input des Users durchgeführt und beendet werden. Liegt die Bewerbung im mittleren Bereich, muss ein Mensch mittels einer User-Aktivität entscheiden. Diese anstehende User-Aktivität lässt sich mit der folgenden Anfrage darstellen:

```
GET http://localhost:8080/get-task/{processInstanceId}
```

Zur Einordnung wird die ID der Prozessinstanz benötigt, die beim Starten des Prozesses zurückgegeben wurde. Diese kann anstelle von `processInstanceId` in die URL eingegeben werden. Zurückgegeben wird eine Reihe von Informationen über die Aktivität, die dem Benutzer helfen sollen, diese zu erledigen, wie die Task-ID oder ein im BPMN hinterlegtes Formular. Im Falle des Bewerbungsprozesses sieht das wie folgt aus:

```
{  
  "id": "f6596bba-3bcb-11ef-9205-84c5a6a77b01",
```

```
"name": "Bewerbung evaluieren",
"processInstanceId": "f64f0b70-3bcb-11ef-9205-84c5a6a77b01",
"formProperties": [
  {
    id": "accepted",
    "name": "Bewerbung akzeptieren? (J/N)",
    "required": true,
    "formValues": []
  }
]
}
```

Listing 6.2: Ausgabe einer User-Aktivität über die REST API

Nachdem entschieden wurde, die Bewerbung anzunehmen, muss die Entscheidung der BPM-Engine mitgeteilt werden. Dafür wird POST-Anfrage an die BPM-Engine geschickt, wobei die zu übergebenden Informationen im JSON-Format übermittelt werden:

```
POST http://localhost:8080/complete-task/{taskId}

Header: {Content-Type: application/json}
Body: {"accepted": "J"}
```

Anschließend wird der Prozess weiter ausgeführt. Die RPA-Engine generiert und versendet die Zusagemail und der Prozess wird am End-Event beendet. Um zu überprüfen, ob und wie der Prozess beendet wurde, kann der User eine Liste der abgeschlossenen Prozessinstanzen aufrufen:

```
GET http://localhost:8080/get-completed-processes
```

Zusätzlich zu den bereits beschriebenen Anfragen wurden noch drei weitere GET-Anfragen implementiert, die dem User einen besseren Überblick über die Ausführungen verschaffen sollen. Die Anfrage `/processes` gibt Details über alle deployten und ausführbaren Prozesse zurück, während die Anfragen `/process-instances` und `/get-all-tasks` Listen zurückgibt, die entweder alle laufenden Prozessinstanzen oder alle offenen User-Aktivitäten enthalten.

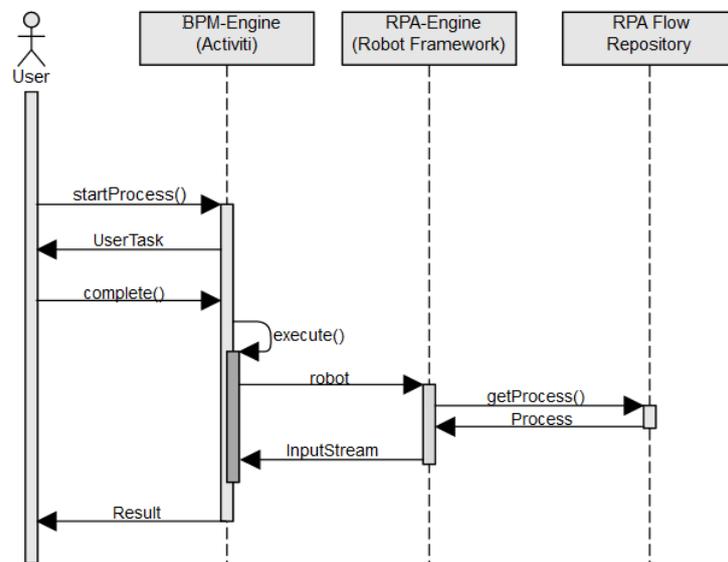


Abbildung 6.1: UML Sequenzdiagramm der Prozessausführung

# 7 Evaluation

In diesem Kapitel wird das entwickelte System und BPM-RPA Konzept den Anforderungen gegenübergestellt und überprüft, ob und in welchem Umfang diese umgesetzt worden sind. Anschließend werden die Ergebnisse diskutiert.

## 7.1 Vergleich mit den Anforderungen

Die Anforderungen, die als Basis für das Konzept und die Umsetzung dienten, sind in Kapitel 4.1 aufgelistet. Diese sind in vier Gruppen unterteilt: funktionale Anforderungen sowohl für das entwickelte System (F1 - F4) als auch für das Gesamtsystem (SF1 - SF5), sowie nicht-funktionale Anforderungen für die Engines (NF1 - NF5) und für das gesamte System (SNF1 - SNF3).

Bei der Untersuchung des entwickelten Systems in Bezug auf die Erfüllung der Anforderungen zeigt sich, dass viele Anforderungen durch die sorgfältige Auswahl der beiden Engines erfüllt worden sind. Das Kriterium, dass die Engines Open-Source sein sollen, gewährleistet beispielsweise die Wartbarkeit des Systems (NF3). Open-Source Anwendung können an die eigene Entwicklung angepasst werden und sind meist modular aufgebaut, was eine Versionsunabhängigkeit gewährleistet. Zudem ist die Software unabhängig von Unternehmen, die das System vernachlässigen oder einstellen könnten.

Sowohl Robot Framework, als auch Activiti verfügen über eine sehr detaillierte Dokumentation, und auch die Schnittstelle wurde ausführlich dokumentiert (NF4). Dies erleichtert die Wartung und ermöglicht die Erweiterung des Systems durch gut dokumentierte Schnittstellen (NF3). Zwar kann das BPMS Activiti BPMN-Diagramme interpretieren und ausführen, aber das Robot Framework ist dazu nicht in der Lage (F2). Dies ist jedoch vernachlässigbar, da im Gesamtsystem der RPA-Parser das BPMN für die RPA-Engine übersetzt.

Sowohl Robot Framework als auch Activiti, die auch kommerziell genutzt werden, verfügen über umfassende Mechanismen zur Fehlerbehandlung und zum Ausnahmehandling.

Diese Mechanismen stellen sicher, dass Fehler zwar erkannt und gemeldet werden, aber die Anwendung dennoch stabil weiterläuft (F4). Fehler, die durch die Kommunikation der beiden Engines über die Schnittstelle oder durch Interaktionen mit dem Benutzer über die API entstehen, werden abgefangen und in der Konsole protokolliert. Zudem wird die Zuverlässigkeit und Robustheit des Systems durch die Auswahl Engines sichergestellt (NF1).

Da beide Engines in einer Spring-Boot Anwendung eingebettet sind, ist die Anforderung der Portabilität und Benutzerfreundlichkeit teilweise erfüllt (NF2). Das Spring-Boot Framework reduziert die Konfiguration eines Systems, stellt eine Vielzahl von Bibliotheken zur Verfügung und ermöglicht den Start der Anwendung mit einem Befehl. Jedoch ist das System durch eine rudimentäre REST API und einem fehlenden User Interface (UI) nicht sehr benutzerfreundlich.

Die erfolgreiche Prozessausführung wurde anhand des Bewerbungsprozesses in Abschnitt 6.2 demonstriert, was eine weitere Anforderung erfüllt (F1). Die Basis dieser Ausführung ist die Synchronisation von RPA- und Nicht-RPA-Aufgaben (F3), welche durch die Konzipierung der BPMN 2.0 Struktur und der Schnittstelle umgesetzt wurde.

Zur Evaluierung der Performance (NF5) wurde untersucht, wie sich die Ausführungszeit des Bewerbungsprozesses durch die Integration von BPM- und RPA-Engine verändert hat. Der Bewerbungsprozess wurde in BPM- und RPA-Teile unterteilt und sowohl synchron als auch asynchron ausgeführt, wobei die Ausführungszeiten gemessen wurden. Die durchschnittlichen Zeiten der jeweiligen Abschnitte wurden summiert und mit der durchschnittlichen Ausführungszeit des gesamten Bewerbungsprozesses verglichen. Es konnte kein signifikanter Unterschied zwischen den Zeiten festgestellt werden. Auffällig war jedoch, dass die RPA-Engine etwas mehr als zwei Sekunden zum Starten benötigt, bevor sie mit der eigentlichen Ausführung beginnt, die in der Regel etwa 6,54 Sekunden dauert. Im Vergleich dazu benötigt die BPM-Engine nur wenige Millisekunden, um ihre Aktivitäten auszuführen.

Die Anforderungen für ein Gesamtsystem, welches BPM und RPA verbindet, werden zum Teil bereits durch die implementierten Engines erfüllt. Die Funktionsfähigkeit der Systeme durch das Zusammenspiel der Komponenten nicht eingeschränkt (SF3), da alle Funktionalitäten der Engine verfügbar sind und die Ausführungszeit nicht beeinträchtigt wird. Da zwar die RPA-Engine in der Lage ist, Prozesse selbständig auszuführen, aber im Konzept dieser Arbeit nur durch die BPM-Engine aufrufbar ist, kann die RPA als Ergänzung der BPM-Engine gesehen werden (SNF2).

Für die Systemarchitektur wurde die Architektur der BPMS als Basis genommen, sodass

sowohl die BPM-Engine als auch der Prozess Modeller und das Business Process Model Repository als Bestandteile des BPMS angesehen werden. Somit ist das BPMS sowohl für das Starten, Steuern und Überwachen der Ausführung verantwortlich als auch für das Modellieren und das Dokumentieren (SF1). Alle essenziellen koordinativen Aufgaben werden somit durch das BPMS gehandhabt (SF2). Durch mögliche Schnittstellen kann das BPMS Tools für die Analyse Implementieren und somit sowohl den BPM- als auch den RPA-Prozess optimieren. Dieser Aspekt ist jedoch nicht explizit im Konzept vermerkt.

Indirekt impliziert aber nicht explizit im Konzept festgehalten, ist außerdem die Reife des BPMS (SNF1). Diese Anforderung ist jedoch durch die Auswahl der BPM-Engine im Prototyp zumindest in Bezug auf die Engine erfüllt. Durch den RPA-Parser wird die Prozessmodellierungssprache auf BPMN 2.0 beschränkt (SF4). Dadurch muss der User keine weitere Modellierungssprache erlernen, was die Modellierung vereinfacht. Da zum Implementieren von Service- und Skript-Aktivitäten jedoch Kenntnis von weiteren Sprachen benötigt werden, bleibt die Modellierung ein komplexer Prozess (SF5). Da das Konzept des Gesamtsystems BPM-RPA keine Benutzerschnittstellen berücksichtigt, kann die Benutzerfreundlichkeit und Attraktivität des Systems nicht beurteilt werden (SF5).

Zusammenfassend lässt sich sagen, dass das entwickelte BPM-RPA-System viele der gestellten Anforderungen erfolgreich erfüllt. Die sorgfältige Auswahl von Open-Source-Engines wie Activiti und Robot Framework trägt maßgeblich zur Wartbarkeit, Dokumentation und Robustheit des Systems bei. Während einige Aspekte wie die Benutzerfreundlichkeit durch fehlende UI-Komponenten noch Verbesserungspotenzial aufweisen, zeigen die durchgeführten Tests, dass die Prozessausführung zuverlässig und performant ist. Das System verbindet BPM und RPA nahtlos und ermöglicht durch die Implementierung von BPMN 2.0 eine konsistente Prozessmodellierung. Insgesamt legt der entwickelte Prototyp eine solide Grundlage für zukünftige Erweiterungen und Optimierungen im Bereich BPM-RPA-Integration.

## 7.2 Diskussion

Die Konzeption und Umsetzung der Verbindung zwischen BPM und RPA zeigt sowohl Stärken als auch Bereiche mit Verbesserungspotenzial. Die folgenden Aspekte sollen die wichtigsten Erkenntnisse und Herausforderungen dieser Integration beleuchten.

Ein zentraler Vorteil des entwickelten Systems ist die nahtlose Integration der BPM-Engine und der RPA-Engine. Durch die Verwendung der BPMN 2.0 konnte eine konsistente und verständliche Prozessmodellierung erreicht werden, was die Erstellung und Anpassung von Prozessen erleichtert.

Ein weiterer positiver Aspekt ist die Nutzung von Open-Source-Engines wie Activiti und Robot Framework. Diese Wahl bietet zahlreiche Vorteile hinsichtlich Wartbarkeit und Flexibilität. Beide Engines sind gut dokumentiert, was die Erweiterung und Anpassung des Systems erleichtert.

Trotz dieser Stärken weist das System auch einige Schwächen auf. Eine der größten Herausforderungen ist die fehlende Benutzerfreundlichkeit, die hauptsächlich durch das Fehlen einer grafischen UI bedingt ist. Die aktuelle Implementierung über eine rudimentäre REST API macht die Bedienung des Systems unnötig kompliziert und könnte potenzielle Nutzer abschrecken. Ein benutzerfreundliches Interface erleichtert nicht nur die Interaktion mit dem System, sondern kann auch die Attraktivität des Systems erhöhen.

Ein weiterer kritischer Punkt ist die Leistung des Systems, insbesondere die Startzeit der RPA-Engine. Während die BPM-Engine durch das einmalige Starten der Anwendung initialisiert wird und anschließend jederzeit einsatzbereit ist, wird für jede RPA-Ausführung eine neue Instanz der RPA-Engine erstellt. Dies führt zu langen Startzeiten bei jeder RPA-Ausführung. In Szenarien, in denen schnelle Reaktionen erforderlich sind, kann dies problematisch sein. Optimierungen, wie die Bereithaltung mehrerer vorkonfigurierter Bots, könnten die Effizienz des Gesamtsystems erheblich steigern.

Zusätzlich besteht das Problem der Unflexibilität in der Auswahl der Komponenten. Die gegenwärtige Lösung, die auf den Erkenntnissen von Flechsig u. a. (2022) basieren, setzt stark auf die spezifischen ausgewählten Engines, was die Anpassungsfähigkeit des Systems einschränkt. In diesem Kontext ist das Konzept von König u. a. (2020) flexibler, da es einen Adapter vorsieht, der die Engines flexibel miteinander verbindet und es somit erlaubt, verschiedene Engines je nach den spezifischen Anforderungen der Nutzer auszutauschen.

Ein weiterer Punkt ist die Abhängigkeit der RPA-Komponente von der BPM-Engine. Obwohl dies sicherstellt, dass die RPA als Ergänzung zur BPM fungiert, könnte eine größere Unabhängigkeit der RPA-Engine die Flexibilität des Systems erhöhen. Dies könnte beispielsweise durch die Implementierung einer bidirektionalen Kommunikation zwischen den Engines erreicht werden, sodass die RPA-Engine auch eigenständig Prozesse initiieren kann, wenn dies notwendig ist.

Die Integration von BPMN 2.0 als Prozessmodellierungssprache bietet zwar viele Vortei-

le, birgt jedoch das Potential den Prozess der Modellierung zu verkomplizieren. Dieses Potential wird durch das Konzept der Übersetzung vom BPMN zu RPA-Sprache durch den RPA-Parser beeinflusst.

Ohne integrierte Verbesserungstools bleibt das Potenzial des Systems ebenfalls begrenzt. Zwar können Prozesse modelliert und ausgeführt werden, doch ohne geeignete Analyse- und Optimierungstools wird es schwierig, die Prozesse kontinuierlich zu verbessern. Dies beeinträchtigt die Effizienz und Effektivität des Systems.

Zusammenfassend lässt sich sagen, dass das BPM-RPA Konzept viele der gestellten Anforderungen erfolgreich erfüllt. Dennoch erfordert die vollständige Realisierung des Potenzials dieser Integration weitere Verbesserungen in Bereichen wie Benutzerfreundlichkeit, Performance, Flexibilität der Komponentenwahl und der Integration von Verbesserungstools.

## 8 Schlussbetrachtung

Dieses letzte Kapitel dieser Arbeit beinhaltet das Fazit der Untersuchung der gesamten Arbeit und teilt mit, wie an dieser Arbeit weiter gearbeitet werden kann. Zu Anfang wird die Idee vorgestellt und die Anforderungen zusammengefasst. Danach wird die Umsetzungslösung beschrieben. Dabei wird insbesondere auf die Ergebnisse der Umsetzung geachtet. Schließlich wird das Ergebnis der Arbeit beschrieben und es wird ausgewertet, ob die gesamte Arbeit nach den Kriterien aus 1.2 erfolgreich ist. Der Ausblick wird in einem separaten Unterkapitel ausgeführt.

### 8.1 Fazit

Ziel dieser Arbeit war es, zu untersuchen, inwiefern ein Business Process Management System (BPMS) und Robotic Process Automation (RPA) miteinander verbunden werden können und welche Vorteile sich dadurch ergeben. Hierbei wurde der Bewerbungsprozess als praktisches Anwendungsbeispiel gewählt, um die theoretischen Konzepte in die Praxis umzusetzen und zu evaluieren.

Die Arbeit begann mit einer umfassenden Literaturrecherche, um die Grundlagen von BPM, BPMS und RPA zu verstehen und den aktuellen Stand der Forschung in diesem Bereich zu ermitteln. Darauf aufbauend wurden die Anforderungen an ein System definiert, welches BPM- und RPA-Elemente integriert.

Ein zentrales Ergebnis dieser Arbeit war die Entwicklung eines Konzepts, das die Verbindung von BPMS und RPA ermöglicht. Dieses Konzept umfasst ein Prozess Modeller, ein RPA-Parser und die ausführende Komponente, die Prozess Engines für BPM und RPA. Die Prozess Modeller ermöglicht die Modellierung und Optimierung von Geschäftsprozessen, während der RPA-Parser die Übersetzung von BPMN in RPA-Skripte übernimmt. Dies stellt sicher, dass die Automatisierung sowohl durch BPM- als auch durch RPA-Komponenten erfolgt. Die Ausführung des Prozesses übernimmt die BPM-Engine, welche RPA-Prozesse zur Ausführung an die RPA-Engine weiterleitet und den Prozessablauf

überwacht. Ein Prototyp der ausführenden Komponente wurde entwickelt.

Die Evaluation des entwickelten Systems zeigte, dass viele der gestellten Anforderungen erfolgreich erfüllt wurden. Der Bewerbungsprozess konnte automatisiert werden, was die Effizienz und Effektivität der Prozessausführung deutlich steigerte. Zudem konnte gezeigt werden, dass die Integration von BPMS und RPA die Flexibilität und Anpassungsfähigkeit von Geschäftsprozessen erhöht.

Allerdings wurden auch einige Herausforderungen identifiziert, die eine vollständige Realisierung des Potenzials dieser Integration noch begrenzen. Insbesondere die Benutzerfreundlichkeit und die Performance des Systems bieten Verbesserungspotential. Zudem fehlt es an integrierten Werkzeugen zur kontinuierlichen Verbesserung und Analyse der Prozesse, was die langfristige Effizienz des Systems beeinträchtigen kann.

Zusammenfassend lässt sich sagen, dass die Arbeit erfolgreich gezeigt hat, dass die Verbindung von BPMS und RPA viele Vorteile mit sich bringt und einen bedeutenden Schritt zur Optimierung von Geschäftsprozessen darstellt. Die entwickelte Lösung erfüllt die meisten Anforderungen und bietet eine solide Basis für weitere Forschungen und Entwicklungen in diesem Bereich.

Insgesamt trägt diese Arbeit wesentlich zum Verständnis der Integration von Business Process Management und Robotic Process Automation bei und zeigt auf, wie diese Technologien effektiv kombiniert werden können, um Geschäftsprozesse effizienter und flexibler zu gestalten.

## 8.2 Ausblick

Die Integration von BPMS und RPA bietet erhebliche Potenziale zur Optimierung und Automatisierung von Geschäftsprozessen. Die Ergebnisse dieser Bachelorarbeit haben bereits vielversprechende Ansätze und erste Erfolge bei der Automatisierung des Bewerbungsprozesses aufgezeigt. Dennoch gibt es zahlreiche Bereiche, die weiter erforscht und entwickelt werden können, um die Effizienz und Effektivität der Systeme weiter zu steigern.

Ein zentraler Aspekt der zukünftigen Forschung und Entwicklung in diesem Bereich sollte die Verbesserung der Benutzerfreundlichkeit und Performance des integrierten Systems sein. Obwohl die aktuelle Implementierung viele Anforderungen erfüllt, bleibt noch erhebliches Potenzial zur Steigerung der Systemleistung und zur Vereinfachung der Benutzeroberfläche. Dies ist insbesondere wichtig, um die Akzeptanz und Nutzung der Technologie

in der Praxis weiter zu erhöhen.

Ein weiterer wichtiger Schritt ist die Entwicklung und Integration von Werkzeugen zur kontinuierlichen Verbesserung und Analyse der Prozesse. Aktuell fehlen solche integrierten Tools, was die langfristige Effizienz des Systems beeinträchtigen kann. Durch die Implementierung von Analyse- und Optimierungstools könnte die Anpassungsfähigkeit und Flexibilität der Geschäftsprozesse weiter gesteigert werden, was einen erheblichen Mehrwert für Unternehmen darstellt.

Darüber hinaus sollten zukünftige Forschungsarbeiten den Prototyp verbessern und erweitern. So sollte die Optimierung der Ausführung der RPA-Engine und Entwicklung des RPA-Parsers und des Prozess Modellers in Angriff genommen werden. Auch Methoden wie Process Mining und Künstlichen Intelligenz (KI) könnten eingesetzt werden, um Prozesse automatisch zu überwachen, zu analysieren und Optimierungspotenziale zu identifizieren. Durch den Einsatz von Machine Learning (ML) Algorithmen könnten Anomalien und Engpässe in Echtzeit erkannt und Handlungsempfehlungen zur Prozessverbesserung generiert werden.

Insgesamt hat diese Arbeit gezeigt, dass die Integration von BPMS und RPA einen vielversprechenden Ansatz zur Optimierung von Geschäftsprozessen darstellt. Die entwickelte Lösung bildet eine solide Grundlage für weiterführende Forschungen und Entwicklungen, die das volle Potenzial dieser Technologien ausschöpfen und deren Einsatz in der Praxis weiter vorantreiben können.

# Literaturverzeichnis

- [Agostinelli u. a. 2020] AGOSTINELLI, Simone ; MARRELLA, Andrea ; MECELLA, Massimo: Towards Intelligent Robotic Process Automation for BPMers. In: *CoRR* abs/2001.00804 (2020). – URL <http://arxiv.org/abs/2001.00804>
- [Czarnecki und Fettke 2021] CZARNECKI, Christian (Hrsg.) ; FETTKE, Peter (Hrsg.): *Robotic process automation: Management, Technology, Applications*. Berlin and Boston : De Gruyter, 2021 (De Gruyter STEM). – ISBN 9783110676693
- [Dumas u. a. 2021] DUMAS, Marlon ; LA ROSA, Marcello ; MENDLING, Jan ; REIJERS, Hajo A.: *Grundlagen des Geschäftsprozessmanagements*. Berlin : Springer, 2021. – ISBN 9783662587355
- [Enriquez u. a. 2020] ENRIQUEZ, J. G. ; JIMENEZ-RAMIREZ, A. ; DOMINGUEZ-MAYO, F. J. ; GARCIA-GARCIA, J. A.: Robotic Process Automation: A Scientific and Industrial Systematic Mapping Study. In: *IEEE Access* 8 (2020), S. 39113–39129
- [Flechsigt u. a. 2019] FLECHSIG, Christian ; LOHMER, Jacob ; LASCH, Rainer: Realizing the Full Potential of Robotic Process Automation Through a Combination with BPM. In: *Logistics Management*. Cham : Springer International Publishing, 2019, S. 104–119. – ISBN 978-3-030-29821-0
- [Flechsigt u. a. 2022] FLECHSIG, Christian ; VÖLKER, Maximilian ; EGGER, Christian ; WESKE, Mathias: Towards an Integrated Platform for Business Process Management Systems and Robotic Process Automation. In: *Business Process Management: Blockchain, Robotic Process Automation, and Central and Eastern Europe Forum*. Cham : Springer International Publishing, 2022, S. 138–153. – ISBN 978-3-031-16168-1
- [Hammer 2015] HAMMER, Michael: *What is Business Process Management?* S. 3–16. In: BROCKE, Jan vom (Hrsg.) ; ROSEMAN, Michael (Hrsg.): *Handbook on Business Process Management 1: Introduction, Methods, and Information Systems*. Berlin, Heidelberg : Springer Berlin Heidelberg, 2015. – URL [https://doi.org/10.1007/978-3-642-45100-3\\_1](https://doi.org/10.1007/978-3-642-45100-3_1). – ISBN 978-3-642-45100-3

- [Hofmann u. a. 2020] HOFMANN, Peter ; SAMP, Caroline ; URBACH, Nils: Robotic process automation. In: *Electronic Markets* 30 (2020), Nr. 1, S. 99–106. – ISSN 1422-8890
- [König u. a. 2020] KÖNIG, Maximilian ; BEIN, Leon ; NIKAJ, Adriatik ; WESKE, Mathias: Integrating Robotic Process Automation into Business Process Management. In: *Business Process Management: Blockchain and Robotic Process Automation Forum*. Cham : Springer International Publishing, 2020, S. 132–146. – ISBN 978-3-030-58779-6
- [Mendling u. a. 2018] MENDLING, Jan ; DECKER, Gero ; HULL, Richard ; REIJERS, Hajo ; WEBER, Ingo: How do Machine Learning, Robotic Process Automation, and Blockchains Affect the Human Factor in Business Process Management? In: *Communications of the Association for Information Systems* 43 (2018), 01, S. 297–320
- [Morelli und Karkos 2022] MORELLI, Frank ; KARKOS, Fabian: Entwicklung eines Vorgehensmodells zur Nutzung von Process Mining für die Auswahl und Anwendung von RPA-Lösungen zur Optimierung von Geschäftsprozessen. In: *Anwendungen und Konzepte der Wirtschaftsinformatik* (2022), Nr. 16, S. 14. – URL <https://akwi.hswlu.ch/article/view/3404>. – ISSN 2296-4592
- [Niculescu u. a. 2023] NICULESCU, Virginia ; OSMAN, Cristina-Claudia ; CHISĂLIȚĂ-CREȚU, Camelia ; STERCA, Adrian: Business Process Representation Analysis in the RPA Context. In: *0730-3157* (2023), S. 1600–1606. – URL <https://www.computer.org/csdl/proceedings-article/compsac/2023/269700b600/1PhCLRa6p4k>. – ISSN 0730-3157
- [Nousias u. a. 2023] NOUSIAS, Nikolaos ; TSAKALIDIS, George ; VERGIDIS, Kostas: BPM Lifecycles and Their Core Cycle Steps: Identification, Processing and Clustering. In: MATSATSINIS, Nikolaos F. (Hrsg.) ; KITSIOS, Fotis C. (Hrsg.) ; MADAS, Michael A. (Hrsg.) ; KAMARIOTOU, Maria I. (Hrsg.): *Operational Research in the Era of Digital Transformation and Business Analytics*. Cham : Springer International Publishing and Imprint: Springer, 2023 (Springer Proceedings in Business and Economics), S. 125–132. – ISBN 978-3-031-24293-9
- [Penttinen u. a. 2018] PENTTINEN, Esko ; KASSLIN, Henje ; ASATIANI, Aleksandre: How to Choose Between Robotic Process Automation and Back-End System Automation? In: *European Conference on Information Systems 2018* (2018). – URL <http://publications.aston.ac.uk/id/eprint/33685/>

- [Red Hat, Inc. 2023] RED HAT, INC.: *Was ist Open Source? Definition, Funktionsweise und Vorteile*. 2023. – URL <https://www.redhat.com/de/topics/open-source/what-is-open-source>. – Zugriffsdatum: 05.07.2024
- [Reijers 2021] REIJERS, Hajo A.: Business Process Management: The evolution of a discipline. In: *Computers in Industry* 126 (2021), S. 103404. – ISSN 01663615
- [Santos u. a. 2020] SANTOS, Filipa ; PEREIRA, Rúben ; VASCONCELOS, José B.: Toward robotic process automation implementation: an end-to-end perspective. In: *Business Process Management Journal* 26 (2020), Nr. 2, S. 405–420. – URL <https://www.emerald.com/insight/content/doi/10.1108/BPMJ-12-2018-0380/full/pdf>. – ISSN 1463-7154
- [Signavio 2019] SIGNAVIO ; SIGNAVIO GMBH (Hrsg.): *Putting the ‘P’ in RPA: Overcoming the Challenges of RPA Implementation*. 2019
- [Stravinskienė 2022] STRAVINSKIENĖ, Inga: The Link Between the Business Process Management Capabilities and the Benefits Created by Robotic Process Automation in an Organisation. In: *Vilnius University Open Series* (2022), S. 64–71. – URL <https://www.zurnalai.vu.lt/open-series/article/view/29610>. – ISSN 2669-0535
- [Syed u. a. 2020] SYED, Rehan ; SURIADI, Suriadi ; ADAMS, Michael ; BANDARA, Wasana ; LEEMANS, Sander J. ; OUYANG, Chun ; HOFSTEDÉ, Arthur H. ter ; VAN DE WEERD, Inge ; WYNN, Moe T. ; REIJERS, Hajo A.: Robotic Process Automation: Contemporary themes and challenges. In: *Computers in Industry* 115 (2020), S. 103162. – URL <https://www.sciencedirect.com/science/article/pii/S0166361519304609>. – ISSN 01663615
- [Uskenbayeva u. a. 2019] USKENBAYEVA, Raissa ; KALPEYEVA, Zhyldyz ; SATYBALDIYEVA, Ryskhan ; MOLDAGULOVA, Aiman ; KASSYMOVA, Aizhan: Applying of RPA in Administrative Processes of Public Administration. In: BECKER, Jörg (Hrsg.) ; NOVIKOV, D. A. (Hrsg.): *21st IEEE Conference on Business Informatics*. Los Alamitos, California : Conference Publishing Services, IEEE Computer Society, 2019, S. 9–12. – ISBN 978-1-7281-0650-2
- [Uskenbayeva R. K. u. a. 2022] USKENBAYEVA R. K. ; KUANDYKOV, A. A. ; NALGOZHINA, N.Zh. ; BERKLAIYEVA, M. A.: RPA APPROACH IN BUSINESS PROCESS MANAGEMENT LIFE CYCLE. In: *Journal der Universität für Energietechnik und Kommunikation in Almaty* 1 (2022), Nr. 1, S. 126–132. – URL <https://>

- [//vestnik.aues.kz/index.php/none/article/view/339](http://vestnik.aues.kz/index.php/none/article/view/339). – ISSN 1999-9801
- [van der Aalst 2020] VAN DER AALST, Wil: On the Pareto Principle in Process Mining, Task Mining, and Robotic Process Automation. In: *Proceedings of the 9th International Conference on Data Science, Technology and Applications*, SCITEPRESS - Science and Technology Publications, 2020
- [van der Aalst u. a. 2018] VAN DER AALST, Wil M. P. ; BICHLER, Martin ; HEINZL, Armin: Robotic Process Automation. In: *Business & Information Systems Engineering* 60 (2018), Nr. 4, S. 269–272. – URL <https://link.springer.com/article/10.1007/s12599-018-0542-4>. – ISSN 2363-7005
- [van der Aalst 2021] VAN DER AALST, Wil M.: Hybrid Intelligence: to automate or not to automate, that is the question. In: *International Journal of Information Systems and Project Management* 9 (2021), Nr. 2, S. 5–20. – URL <https://revistas.uinho.pt/index.php/ijispm/article/view/3540>
- [Völker u. a. 2021] VÖLKER, Maximilian ; SIEGERT, Simon ; WESKE, Mathias: Adding Decision Management to Robotic Process Automation. In: WEBER, Ingo (Hrsg.) ; VAN DE WEERD, Inge (Hrsg.) ; PLEBANI, Pierluigi (Hrsg.) ; FETTKE, Peter (Hrsg.) ; DEBOIS, Søren (Hrsg.) ; ENRÍQUEZ, José G. (Hrsg.): *Business Process Management: Blockchain and Robotic Process Automation Forum: BPM 2021 Blockchain and RPA Forum, Rome, Italy, September 6-10, 2021, Proceedings* Bd. 428. Cham : Springer, 2021, S. 23–37. – ISBN 978-3-030-85866-7
- [Vollmer und Schmitt 2020] VOLLMER, Anna M. ; SCHMITT, Anna: Was ist Open Source Software? Definition, Vor- und Nachteile, Lizenzen. In: *Fraunhofer IESE* (2020). – URL <https://www.iese.fraunhofer.de/blog/open-source-software/>. – Zugriffsdatum: 05.07.2024
- [Weske 2019] WESKE: *Business Process Management*. Berlin, Heidelberg : Springer Berlin Heidelberg, 2019. – ISBN 978-3-662-59431-5
- [Willcocks u. a. 2015] WILLCOCKS, Leslie P. ; LACITY, Mary ; CRAIG, Andrew: The IT function and robotic process automation. (2015)

## Erklärung zur selbständigen Bearbeitung

Hiermit versichere ich, dass ich die vorliegende Arbeit ohne fremde Hilfe selbständig verfasst und nur die angegebenen Hilfsmittel benutzt habe. Wörtlich oder dem Sinn nach aus anderen Werken entnommene Stellen sind unter Angabe der Quellen kenntlich gemacht.



---

Ort

Datum

Unterschrift im Original