



Höhere Technische Bundeslehranstalt für Informatik, Perg

Diplomarbeit

Prozessautomatisierungs-Software für die Synchronisation von Arbeitspaketen und Automatisierung einer Query Software

eingereicht von:

Alexandra Bauer
Mert Ibisler
Simon Praher

Eingereicht am:

03. April 2024

Betreuer:

Prof. Dipl.-Ing. (FH) Johannes Oppitz, MSc

In Zusammenarbeit mit:

voestalpine group-IT GmbH
Dipl.-Ing. (FH) Robert Tremel

Bearbeitungszeitraum:

Juli 2023 bis März 2024

Eidesstattliche Erklärung

Die unterfertigten Kandidaten / Kandidatinnen haben gemäß § 34 (3) SchUG in Verbindung mit § 22 (1) Zi. 3 lit. b der Verordnung über die abschließenden Prüfungen in den berufsbildenden mittleren und höheren Schulen, BGBl. II Nr. 70 vom 24.02.2000 (Prüfungsordnung BMHS), die Ausarbeitung einer Diplomarbeit mit der umseitig angeführten Aufgabenstellung gewählt.

Die Kandidaten / Kandidatinnen nehmen zur Kenntnis, dass die Diplomarbeit in eigenständiger Weise und außerhalb des Unterrichtes zu bearbeiten und anzufertigen ist, wobei Ergebnisse des Unterrichtes mit einbezogen werden können.

Die Abgabe der Diplomarbeit hat bis spätestens 04.04.2024 beim zuständigen Betreuer / der zuständigen Betreuerin zu erfolgen.

Die Kandidaten / Kandidatinnen nehmen weiters zur Kenntnis, dass gemäß § 9 (6) der Prüfungsordnung BMHS nur der Schulleiter bis spätestens Ende des vorletzten Semesters den Abbruch einer Diplomarbeit anordnen kann, wenn diese aus nicht beim Prüfungskandidaten (bei den Prüfungskandidaten) gelegenen Gründen nicht fertiggestellt werden kann.

Kandidaten / Kandidatinnen inkl. Unterschrift:

St. Valentin, 03.04.2024

Ort, Datum



Alexandra Bauer

St. Valentin, 03.04.2024


Ort, Datum



Mert Ibisler

Ried in der Riedmark, 03.04.2024

Ort, Datum



Simon Praher

1 DANKSAGUNG

Wir bedanken uns bei all jenen, die uns bei unserer Diplomarbeit in jeglicher Hinsicht unterstützt haben.

Besonderer Dank gilt unserem Betreuungslehrer Dipl.-Ing. (FH) Johannes Oppitz für seine Unterstützung und Hilfestellung beim Verfassen der Diplomarbeit.

Weiters bedanken wir uns bei Dipl.-Ing. (FH) Robert Treml, unserer Kontaktperson bei der Voestalpine group-IT. Bei Fragen ist er uns immer zur Verfügung gestanden und hat uns auch tatkräftig bei der Ideenfindung der Diplomarbeit unterstützt.

Weiterer Dank gilt Dipl.-Ing. (FH) Romana Bauer, die den Erstkontakt zu unserem Ansprechpartner aufbaute und uns somit die Diplomarbeit überhaupt ermöglicht hat.

1	DANKSAGUNG	2
2	EINFÜHRUNG.....	6
2.1	KURZFASSUNG.....	6
2.2	ABSTRACT	7
2.3	AUFTRAGGEBER	8
2.4	PROJEKTTEAM	9
2.5	MOTIVATION	10
2.6	ZIELSETZUNG.....	11
2.7	LEISTUNGSUMFANG	12
2.8	GLIEDERUNG	14
3	GRUNDLAGEN	15
3.1	RPA	15
3.1.1	<i>Softwareroboter und Prozesse</i>	<i>15</i>
3.1.2	<i>Vorteile</i>	<i>15</i>
3.1.3	<i>Sicherheit.....</i>	<i>16</i>
3.1.4	<i>Wann sollte man Automatisieren</i>	<i>17</i>
3.1.5	<i>Welche Prozesse eignen sich zur Automatisierung</i>	<i>19</i>
3.1.6	<i>Wie soll die Automatisierung realisiert werden.....</i>	<i>21</i>
3.1.7	<i>Vollautomatisierung und Teilautomatisierung.....</i>	<i>22</i>
3.1.8	<i>Verschiedenen Aspekte von RPA</i>	<i>22</i>
3.1.9	<i>IPA.....</i>	<i>23</i>
3.1.10	<i>Anwendungsfälle.....</i>	<i>24</i>
3.2	UIPATH.....	26
3.2.1	<i>UiPath Studio.....</i>	<i>26</i>
3.2.2	<i>UiPath Orchestrator.....</i>	<i>26</i>
3.2.3	<i>Vorteile gegenüber händischem Programmieren.....</i>	<i>27</i>
3.3	JIRA	28
3.4	SHAREPOINT.....	29
3.4.1	<i>Workflows.....</i>	<i>29</i>
4	AUSWIRKUNG AUF DIE ARBEITSWELT	30
4.1	GRUNDSATZ	30
4.2	ANFÄNGE DER AUTOMATISIERUNG.....	31
4.3	WIRTSCHAFTLICHE VORTEILE	33
4.4	ANGST VOR JOBVERLUST.....	33
4.5	NEUE ARBEITSPLÄTZE.....	34
4.5.1	<i>RPA Developer</i>	<i>34</i>
4.5.2	<i>RPA-Strategist</i>	<i>34</i>
4.5.3	<i>RPA-Architekt</i>	<i>35</i>
4.6	QUALIFIKATIONEN	36
4.7	MENSCHLICHE ÜBERWACHUNG UND KONTROLLE	37
4.8	ZUKUNFTSAUSSICHTEN DER AUTOMATISIERUNG.....	38
5	DATENVERARBEITUNG	40
5.1	PROBLEME	40
5.2	VORTEILE	40
6	DATENSCHUTZ UND RISIKEN.....	42
6.1	ALLGEMEINE RISIKEN.....	42
6.2	DATENSCHUTZ-RICHTLINIEN	43
6.3	RISIKEN	44
6.3.1	<i>Beschaffungsrisiken.....</i>	<i>44</i>
6.3.2	<i>Tool Risiken</i>	<i>44</i>
6.3.3	<i>Stakeholder.....</i>	<i>44</i>
6.3.4	<i>Strategie</i>	<i>44</i>
6.3.5	<i>Über-Automatisierung.....</i>	<i>45</i>

6.4	MAßNAHMEN FÜR SICHERHEIT UND DATENSCHUTZ.....	46
6.4.1	<i>Rechte und Governance</i>	46
6.4.2	<i>Logs und Protokolle</i>	46
6.4.3	<i>Sicherheitsabteilung</i>	47
7	TOOLS UND ANBIETER.....	48
7.1	AUTOMATION ANYWHERE	48
7.1.1	<i>Discovery-Bot</i>	48
7.1.2	<i>IQ-Bot</i>	48
7.1.3	<i>Bot-Überwachung</i>	48
7.2	BLUE PRISM.....	49
7.3	OPENTEXT	50
7.3.1	<i>LiquidOffice</i>	50
7.3.2	<i>SAP S/4HANA</i>	50
7.4	PEGASYSTEMS INC.....	51
7.4.1	<i>Low-Code-Plattform</i>	51
7.4.2	<i>Pega-Plattform</i>	51
7.5	MICROSOFT POWER AUTOMATE	52
7.5.1	<i>Microsoft Power Automate</i>	52
8	KÜNSTLICHE INTELLIGENZ.....	53
8.1	DEEP LEARNING	53
8.1.1	<i>Beispiel für Deep Learning: Menschen erkennen</i>	54
8.1.2	<i>Künstliche neuronale Netze</i>	54
8.2	MACHINE LEARNING	55
8.3	KI IM ZUSAMMENHANG MIT RPA	56
8.3.1	<i>Treffen von Entscheidungen</i>	56
8.3.2	<i>Mustererkennung und Vorhersage</i>	56
8.3.3	<i>Maschinelles Lernen</i>	56
8.4	CHAT GPT	57
8.4.1	<i>Chat GPT-API</i>	58
8.5	ANDERE CHAT BOTS.....	58
9	BENUTZUNG.....	60
9.1	SYNCHRONISATION ZWISCHEN SHAREPOINT UND JIRA	60
9.2	SIMPLEQUERY	61
10	PROJEKTSTRUKTUR.....	62
10.1	ZUSAMMENHANG JIRA UND SHAREPOINT	62
10.1.1	<i>Felder Problem</i>	63
11	IMPLEMENTIERUNG	64
11.1	SYNCHRONISATION ZWISCHEN SHAREPOINT UND JIRA	64
11.1.1	<i>Technologieauswahl</i>	64
11.1.2	<i>UiPath Installation</i>	65
11.1.3	<i>Schnittstelle Jira</i>	69
11.1.4	<i>Programmbeschreibung</i>	73
11.2	SIMPLE QUERY.....	100
11.2.1	<i>Technologieauswahl</i>	100
11.2.2	<i>Spring Installation</i>	101
11.2.3	<i>SimpleQuery API</i>	102
11.2.1	<i>Programmaufbau</i>	102
11.2.2	<i>Query Zusammenstellung</i>	103
12	RISIKOMANAGEMENT.....	120
12.1	RISIKEN	120
12.2	EINGETRETENE RISIKEN	123
12.3	FAZIT AUS DEN PROBLEMEN	123

13	WEITERE BEISPIELPROGRAMME	124
13.1	E-MAIL BOT	124
13.1.1	<i>Benutzung</i>	124
13.1.2	<i>Projektstruktur</i>	125
13.1.3	<i>Implementierung</i>	126
13.2	DATEI-ORGANISATION	136
13.2.1	<i>Prinzip</i>	136
13.2.2	<i>Benutzung</i>	136
13.2.3	<i>Projektstruktur</i>	137
13.2.4	<i>Implementierung</i>	138
13.3	WINDOWS VIRTUALISIERUNG	140
13.3.1	<i>Problem</i>	140
13.3.2	<i>Benutzung</i>	140
13.3.3	<i>Implementierung</i>	140
14	RESÜMEE	142
14.1	ALEXANDRA BAUER	142
14.2	MERT IBISLER	143
14.3	SIMON PRAHER	144
15	LITERATURVERZEICHNIS	145
16	ABBILDUNGSVERZEICHNIS	149
17	TABELLENVERZEICHNIS	152

2 EINFÜHRUNG

2.1 Kurzfassung

OptiFlow ist ein Automatisierungsprojekt, das rein für die Voestalpine group-IT entwickelt worden ist. Die Anwendung wird in der Abteilung PROMES im Bereich PPS und mit der Abteilung PPS eingesetzt. Bei unserem Programm handelt es sich, um eine Synchronisation zwischen dem Voestalpine internen SharePoint und dem personalisierten Jira der Voest. Da die Abteilung von PROMES im SharePoint und die Abteilung von PPS im Jira arbeitet, bekommen die beiden Abteilungen wichtige Änderungen nicht mit. Zum Beispiel im Hinblick auf Projektanforderungen, zuständige Mitarbeiter, und alle anderen relevanten Informationen. Die Programme sollen aus diesem Grund regelmäßig synchronisiert werden. Allerdings dauert dieser Prozess für die Mitarbeiter der group-IT mindestens vier Wochen oder länger pro Durchführung einer solchen Synchronisation, wenn diese händisch durchgeführt werden soll. Während die Synchronisation noch nicht durchgeführt wurde, sollte dann auch nicht an den PPS-Projekten gearbeitet werden, da dies zu führt weitere Änderungen zu erhalten, die Synchronisation erfordern. Da eine ganze Abteilung natürlich nicht stillstehen kann, ist der Synchronisationsprozess also eine kontinuierliche und ununterbrochene Tätigkeit, die von den Voest Mitarbeitern am besten ohne Unterbrechung durchgeführt werden soll. Dies ist somit eine Stelle, wo sehr viele Ressourcen, wie Zeit, Kosten und Mitarbeiterverfügbarkeit unnötigerweise und ohne Unterbrechung verbraucht werden.

Mithilfe von OptiFlow soll die Synchronisation nicht nur vereinfacht, sondern auch die gesamten unnötig verbrauchten Ressourcen des Unternehmens an anderen Stellen eingesetzt werden. Mittels der UiPath Studio Anwendung kann dieser langwierige Prozess also auch deutlich verkürzt und innerhalb von Sekunden abgehandelt werden. Die Mitarbeiter können die Synchronisation nun per Knopfdruck durchführen.

2.2 Abstract

OptiFlow is an automation project that has been developed purely for the company Voestalpine group-IT. This application is used by the PPS section of the PROMES department and by the PPS department. Our program aims to provide a synchronization between the Voestalpine internal SharePoint and the personalized Jira of Voestalpine. Since the PROMES department works with SharePoint and the PPS department works with Jira, the two departments fail to be updated on important changes regarding project requirements, responsible employees and all other relevant information. For this reason, the programs have to be synchronized regularly. However, if carried out manually, this process takes the group-IT employees at least four weeks or longer per synchronization. As long as the synchronisation has not been completed, no work is supposed to be done on the PPS projects, as any further changes need to be synchronized as well. Since an entire department cannot stand still, the synchronisation process is a continuous and uninterrupted activity that should be carried out by the Voest employees without interruption. This is therefore an issue where a lot of resources, such as time, costs and employee availability are consumed unnecessarily and without interruption.

With the help of OptiFlow, the synchronization of SharePoint and Jira should consequently not only be simplified, but all of the company's unnecessarily used resources can also be used elsewhere. Using the UiPath Studio application, this lengthy and tedious process can be significantly shortened and completed within seconds. The employees can now carry out synchronization at the touch of a single button.

2.3 Auftraggeber

Diese Diplomarbeit wurde in Zusammenarbeit mit der Voestalpine group-IT GmbH entwickelt. Die group-IT ist die sozusagen die IT-Abteilung des führenden Stahl- und Technologiekonzerns Voestalpine. Die

Voestalpine ist Weltmarktführer bei Bahninfrastruktursystemen, Werkzeugstahl und Spezialprofilen. Zusätzlich ist die Voest in der Automobil-, Haushalts-, Luftfahrt-, Öl und Gasindustrie tätig. (Voestalpine, 2024) ¹ Die Voestalpine group-IT befasst sich schon seit einigen Jahren mit der Prozessautomatisierung und möchte diese auch stetig immer mehr und mehr nutzen. Auch Optiflow ist eine weitere Prozessautomatisierung, die in der Voest eingeführt wurde und dabei hilft, viele Ressourcen des Großkonzerns einzusparen.

The logo for Voestalpine, featuring the word "voestalpine" in a lowercase, blue, sans-serif font.

ONE STEP AHEAD.

Abbildung 1: Voestalpine

¹ Voestalpine group-IT Logo: <https://www.voestalpine.com/group-it/de/>

2.4 Projektteam

Das Projektteam besteht aus Bauer Alexandra, Ibisler Mert und Praher Simon. Die drei Schüler der HTL Perg sind allesamt sehr interessiert an Automatisierungen. Deswegen ist es naheliegend, dass sie sich für ein Projektthema entschieden haben, das stark auf Automatisierung abzielt. Als Projektleiterin hat Alexandra die Verantwortung, die verschiedenen Aufgaben im Blick zu behalten und sicherzustellen, dass alles nach Plan verläuft. Mert Ibisler und Simon Praher bringen weitere Fähigkeiten und Kenntnisse in das Projekt ein, um die Automatisierung effektiv umzusetzen.



Abbildung 2: Projektteam

2.5 Motivation

Die Auswahl des Projekts wurde auf Basis der Erfahrungen und der Interessen der Teammitglieder an der Automatisierung getroffen. Einige Mitglieder haben bereits in ihrer Freizeit und vorherigen Projekten Automatisierungen entwickelt.

Durch das Interesse am Thema Automatisierung und die Möglichkeit, diese Fähigkeiten in einem Projekt anzuwenden, sind der Grund für die Auswahl dieses Projektes. Diese Projekte bieten eine gute Möglichkeit, die Kenntnisse und Fähigkeiten der Mitglieder in der Prozessautomatisierung zu vertiefen und anzuwenden. Bei einer Automatisierung der Synchronisation von Aufgabenpaketen zwischen SharePoint und Jira, sowie auch beim Vergleich von Query Ausgaben, können die Projektmitglieder ihre Kenntnisse in der Praxis anwenden.

Die Motivation besteht also sowohl darin individuelle Fähigkeiten weiterzuentwickeln, aber auch Effizienzsteigerung und Ressourceneinsparung bei den Abläufen, welche der Firma langfristig viele Kosten spart. Das Projekt verspricht, die erworbenen Kenntnisse in der Automatisierung in einem realen Szenario anzuwenden und gleichzeitig einen Mehrwert für die Voest zu schaffen.

2.6 Zielsetzung

Die Ziele dieses Projekts lassen sich in zwei Hauptkategorien unterteilen: Geschäftsziele und spezifische Projektziele.

Im Fokus der Geschäftsziele steht die Synchronisation der Aufgabenpakete zwischen SharePoint und Jira. Das Ziel besteht darin, diese Synchronisation auf Knopfdruck durchzuführen, um Zeitverschwendung für Mitarbeiter zu minimieren. Durch eine wöchentliche Synchronisation kann die Effizienz gesteigert werden, und auch menschliche Fehler können vermieden werden. Aber auch das Abgleichen der Versionen auf den Servern ist ein Ziel, welches verfolgt wird. Dies ist wichtig, um sicherzustellen, dass beide Systeme auf dem neuesten Stand sind und korrekte Informationen anzeigen. Durch diese Abstimmung können Probleme vermieden werden, die auftreten könnten, wenn die Versionen nicht übereinstimmen. Die Einsparung von Mitarbeitern, die derzeit manuelle Abgleiche durchführen, ist ein großer Vorteil, da sich diese dann wichtigeren Aufgaben widmen können.

Der Fokus des Projekts liegt sowohl auf der Prozessautomatisierung für die Synchronisation der Aufgabenpakete. Das Projektteam will einen Softwareroboter zu implementieren, der die Verbindung zwischen den Arbeitspaketen in SharePoint und Jira herstellt. Dabei wird SharePoint als hauptsächlich führendes System genutzt, mit Ausnahmen für bestimmte Datenfelder. Eine besondere Herausforderung besteht darin, die Einträge des Progress Logs im SharePoint als Kommentar im Jira darzustellen, wobei sicherzustellen ist, dass der Kommentar nicht im Progress Log landet. Weiters ist die Verknüpfung von IDs und URLs zwischen den Plattformen sowie die Anpassung von Statusänderungen zu implementieren. Das bedeutet, dass bei bestimmten Ereignissen, wie dem Setzen des SharePoint-Status auf "Completed", der Jira-Status auf "Done" gesetzt wird. Zusätzlich soll genehmigten Arbeitspaketen im SharePoint der Status "In Review" im Jira zugewiesen werden.

Weiters hat auch der Versionsabgleich eine wichtige Bedeutung. Bei diesem besteht die Schwierigkeit darin, nach dem Zusammenbauen einer Query, die große Anzahl an Daten, welche man von diesem System zurückbekommt, richtig auszulesen und in einen Zustand zu bringen, in dem die Versionsdaten verglichen werden können.

2.7 Leistungsumfang

Der Leistungsumfang ist sowohl für das Projektteam als auch den Kunden relevant. Sie umfassen alle erforderlichen Aufgaben und Verantwortlichkeiten beider Seiten.

Leistungsumfang hinsichtlich der Organisation:

In Scope	Out of Scope
Einteilung der Projektmitglieder in verschiedene Aufgabenbereiche	Nachträgliche organisatorische Änderungen oder Erweiterungen
Abstimmung mit dem Auftraggeber	Veränderungen in der Zuständigkeit der Ansprechpartner
Lieferung eines synchronisierenden Programms	
Meeting-Abstimmung mit Kunden bei festgelegten Terminen	

Tabelle 1: Organisation

Leistungsumfang hinsichtlich der Kommunikation:

In Scope	Out of Scope
Webex oder Outlook für Kommunikation und Meetings	Treffen in der Schule für Kommunikation und Meetings
Treffen auf dem Voestgelände für Kommunikation und Meetings	

Tabelle 2: Kommunikation

Leistungsumfang hinsichtlich der Applikation:

In Scope	Out of Scope
Entwicklung eines Programms zur Synchronisation	Neue Anwendungen oder Funktionen außerhalb des definierten Rahmens
Nutzung von Jira API und SharePoint Excel Liste	Erweiterungen der Anwendungen über die festgelegten Ziele hinaus
Erstellung einer Excel-Liste als Log-File im SharePoint	Automatische Versionsänderung auf den Servern
Zusammenbauen von Queries welche verglichen werden sollen	
Vergleich der query Resultate	

Tabelle 3: Applikation

Leistungsumfang hinsichtlich der Technologie:

In Scope	Out of Scope
GitLab zur Versionsverwaltung	Änderungen der grundlegenden Technologien außerhalb des Projekts
Verwendung von Postman für Jira API	
Verwendung von UiPath zum Implementieren der Automation	
Java Applikation zur Versionsverwaltung	

Tabelle 4: Technologie

Leistungsumfang hinsichtlich der Daten:

In Scope	Out of Scope
Priorisierung von Daten im SharePoint	Priorisierung von Daten im Jira
Synchronisation von Daten zwischen SharePoint und Jira	Verwendung von Daten aus anderen Quellen

Tabelle 5: Daten

Leistungsumfang hinsichtlich der Location:

In Scope	Out of Scope
Entwicklung auf dem Voestgelände	Entwicklung an anderen Standorten
Entwicklung zu Hause	
Entwicklung in der Schule	

Tabelle 6: Location

Leistungsumfang hinsichtlich der Sicherheit:

In Scope	Out of Scope
Schutz der Datenintegrität während der Synchronisation	Sicherheitsaspekte die Nötig für einen betrieb außerhalb des Intranets wären
Zugriffsverwaltung für GitLab und Projektdokumente	

Tabelle 7: Sicherheit

2.8 Gliederung

Diese Arbeit ist in 13 Kapitel unterteilt, deren Inhalt hier kurz erläutert wird. Kapitel 1, Einführung, enthält eine Kurzfassung der Arbeit sowohl in deutscher als auch englischer Sprache, sowie eine kurze Beschreibung des Auftraggebers. Zusätzlich wird auch über das Projektteam, sowie über die Motivation, Zielsetzung informiert. Auch der Leistungsumfang befindet sich im Einführungskapitel. Kapitel 2, Grundlagen, werden nicht nur die verwendeten Technologien erläutert, sondern auch das Thema RPA näher gebracht. In Kapitel 3, Auswirkungen auf die Arbeitswelt, geht es, wie der Titel bereits aussagt, über die Auswirkungen und die Zukunftsaussichten rund um RPA. Im 4. Kapitel, Datenverarbeitung geht es um die Probleme und Vorteile von RPA. Kapitel 5, Datenschutz und Risiken, handelt um die Risiken, die die Benutzung von RPA verursacht werden, als auch über Datenschutzrichtlinien der RPA, die eingehalten werden müssen, sowie deren Maßnahmen. Darüber hinaus wird auch über die Maßnahmen berichtet, mit denen man Sicherheit gewährleistet. In Kapitel 6, Tools und Anbieter, werden die bekanntesten Tools angeführt, mit denen man automatisieren kann. In Kapitel 7, Künstliche Intelligenz wird zunächst allgemein über KI erzählt, sowie im Zusammenhang mit RPA. Des Weiteren wird auch über die zurzeit sehr bekannte KI ChatGPT geschrieben, als auch weitere Chat Bots näher gebracht.

Im Kapitel 8, Benutzung, geht es darum, wie man die *Synchronisation zwischen SharePoint und Jira* und *SimpleQuery* benutzt. Das Kapitel 9, Projektstruktur, bietet einen allgemeinen Einstieg in den Zusammenhang zwischen Jira und SharePoint. In Kapitel 10, Implementierung werden die beiden Programme *Synchronisation zwischen SharePoint und Jira* und *SimpleQuery* beschrieben. Zusätzlich wird auch erklärt warum das Projektteam die jeweilige Technologie ausgewählt hat und welche Programme man wie installieren muss, um die jeweiligen Programme zum Laufen zu bringen. Auch die APIs werden erklärt, sowie welche Probleme aufgetreten sind. Im Kapitel 11, Risikomanagement, werden die möglichen Risiken, die eintreten könnten, angeführt, sowie die jeweiligen Maßnahmen dazu. Noch dazu werden auch die Risiken und Maßnahmen angeführt, die wirklich eingetreten sind. Im Kapitel 12, weitere Beispielprogramme, werden weitere Programme neben den anderen beiden Programmen für den Auftraggeber angeführt, um einen weiteren kleinen Einblick in RPA zu geben. Abschließend beendet sich in Kapitel 13, Resümee, ein Rückblick auf die Erfahrungen, die bei der Erstellung dieser Arbeit gewonnen wurden.

3 GRUNDLAGEN

3.1 RPA

RPA (Robotic Process Automation ²) ist ein sehr schnell aufstrebendes Thema, das zur Automatisierung von umständlichen und zeitaufwändigen Prozessen dient, welche ständig wiederholt und manuell durchgeführt werden. Diese Prozesse können den Menschen von sogenannten Softwarerobotern abgenommen werden. Diese Prozesse können die verschiedensten Dateneingaben, Transaktionsdurchführungen und Verarbeitungen, Prozessauslösungen bei bestimmten Events und noch vieles mehr automatisieren. Selbst in der Kommunikation spielt RPA heutzutage eine große Rolle, da man mit den verschiedensten Systemen oder auch Menschen kommunizieren kann.

3.1.1 Softwareroboter und Prozesse

Der Softwareroboter ermöglicht nun das eigentliche Automatisieren und Starten von Prozessen. Zuerst wird eine manuelle, repetitive Aufgabe gefunden, die in Zukunft automatisiert ablaufen soll. Vor allem in großen Firmen hat man üblicherweise kein Problem solche Aufgaben zu finden. Um den Geschäftsprozess zu automatisieren, kann man einen Menschen, der alle durchzuführenden Aktionen nacheinander ausführt, von einem RPA-Bot aufzeichnen. Den aufgezeichneten Prozess kann man nun mit den verschiedensten Ausnahmen, Bedingungen und noch vielen weiteren Aktionen versehen. Anschließend kann der Prozess auch schon in die Produktion übernommen werden. Damit der Prozess allerdings nicht nur einmal oder nur bei einem Klick durchgeführt wird, kann man den Softwareroboter so kalibrieren, dass dieser bei bestimmten Events oder zu einer bestimmten Uhrzeit durchgeführt werden soll.

3.1.2 Vorteile

Die Prozessautomatisierung wird auch von Firmen immer öfter verwendet. Mit den vielen Vorteilen, die diese Technologie bietet, ist dies allerdings mehr als nur verständlich.

Wenn eine sich wiederholende Tätigkeit von der neuartigen Technologie automatisiert wird, hat ein Mitarbeiter mehr Zeit für andere Aufgaben, wodurch der unnötig viele Aufwand dem Mitarbeiter abgenommen wird. Dies führt wiederum zu einer großen Mitarbeiterentlastung und ebenfalls zu einer hohen Effizienzsteigerung. Wenn die Mitarbeiter, den wiederkehrenden Prozess nicht mehr manuell durchführen müssen, werden die menschlichen Fehler beseitigt und somit die Fehleranfälligkeit verringert. Außerdem führt der Einsatz von RPA-Bots zu einer außergewöhnlich hohen Genauigkeit. Zudem wird die Skalierung deutlich beschleunigt, wenn der Mitarbeiter diese sich wiederholende Tätigkeit nicht mehr durchführen muss, da die Aufgabe, für die der Mensch vielleicht sogar mehrere Stunden lang braucht, mithilfe eines Softwareroboters binnen Sekunden erledigt wird. Weiters kann er auch immer im Hintergrund

² RPA: https://en.wikipedia.org/wiki/Robotic_process_automation

fungieren, da dieser beim Auslösen von bestimmten Events weiß, wann er einen Prozess starten soll. Der RPA-Bot kann zudem auch zu jeder Tageszeit arbeiten und eingesetzt werden.

3.1.3 Sicherheit

Auch wenn RPA ein vielversprechendes Thema ist, gibt es auch hier einige Sicherheitslecks da die Technologie eine neue, große Angriffsfläche für Cyberattacken bietet. Dadurch dass sich die Bots meist auf einer Plattform anmelden müssen, wird dieser Zugriff meistens in den Code des Roboters fest einprogrammiert. Wenn also ein Cyberangriff auf genau diesen Bot durchgeführt wird, kann der Angreifer die Anmeldedaten stehlen und sich Zugriff zum jeweiligen System verschaffen. Die sensiblen Daten kann er dann ins Internet stellen, wo diese für die Öffentlichkeit zugänglich sind. Auch wenn man die Daten nicht direkt in das Skript integriert, sondern auf eine Datenbank oder der gleichen zugreift, kann der Angreifer immer noch mit Leichtigkeit auf die geheimen Daten zugreifen. Zudem hat der Angreifer Zugriff auf die gesamte Datenbank. Wie kann man sich also dagegen schützen? Die einfachste und wohl sicherste Lösung für solch ein Problem ist ein simples Anmeldefeld, wo der Benutzer seine Daten vor dem Zugriff des Softwareroboters eingeben muss. Weiters könnte man den Benutzer, mit dem sich der Bot anmeldet, beschränken, so dass er nur seine eigene Aufgabe machen kann. Doch auch das kann zu einer Sicherheitslücke führen. Eine weitere Möglichkeit sich davor zu schützen, ist die Sicherung des Zugriffs durch die Isolierung und Überwachung der Softwareaktivitäten. Wenn der Bot seine Tätigkeit auf unerklärliche Weise unterbricht, kann man den Zugriff beispielsweise sperren, um somit größere Schäden zu verhindern und das Risiko von Cyberattacken zu minimieren.

3.1.4 Wann sollte man Automatisieren

Einen Prozess sollte man dann automatisieren, wenn die jeweilige Aufgabe sehr repetitiv ist. Auch sollte der Prozess ohne menschliche Eingriffe funktionieren können, wenn dieser automatisiert wird. Dabei muss man jedoch zwischen Vollautomatisierung und Teilautomatisierung unterscheiden.

Des Weiteren sollte man dann RPA anwenden, wenn der Vorgang anfällig für menschliche Fehler ist. Menschliche Fehler können bei Analyse und sich wiederholenden Aufgaben durch Müdigkeit und Monotonie entstehen. Die Einsparungen an Zeit und Geld machen die Automatisierung auch zu einer lohnenden Überlegung.

Zeitaufwändige Aufgaben wie Administrationsaufgaben können und sollen durch die Automatisierung des langen Prozesses verkürzt werden. Wenn man in Betracht zieht, dass im Jahr 2019 deutsche Firmen wegen zu zeitaufwändigen Administrationsprozessen einen Geldverlust von 30 Milliarden Euro verzeichneten, sind Automatisierungen bei zeitaufwändigeren Aufgaben sehr sinnvoll. (Persiono, 2024) Durch die Automatisierung können Mitarbeiter im Unternehmen effizienter und gezielter eingesetzt werden. Die Bewältigung monotoner Aufgaben stellt dank RPA-Lösungen kein großes Problem mehr dar.

Wenn man darüber nachdenkt, wann es am besten ist, einen Prozess zu automatisieren, sollte man die Wirtschaftlichkeit berücksichtigen. Eine Kosten-Nutzen-Analyse ist hierbei hilfreich, um sowohl kurzfristige als auch langfristige Auswirkungen auf die Wirtschaft zu ermitteln. Besonders sinnvoll ist es, Prozesse zu automatisieren, die sehr regelbasiert sind. RPA-Bots können diese Regeln präziser einhalten, während Menschen eher dazu neigen, Regeln zu übersehen oder zu vergessen. Damit bietet die Automatisierung eine effektive Lösung für Aufgaben, die eine hohe Genauigkeit und Konsistenz erfordern.

Während der Arbeit in Unternehmen, stößt man oft auf kleine Aufgaben, welche beispielsweise jeden Tag durchgeführt werden müssen. Auch wenn diese händischen Prozesse nicht sehr zeitaufwändig sind, sammeln sich Zeitaufwände. Diese Minutenarbeiten könnten viel besser genutzt werden. Dabei rentiert sich die Automatisierung dieses kleinen Prozesses sehr-kleine Prozesse sollten Angestellte nicht von ihrer eigentlichen Arbeit aufhalten. (Visualmakers, 2023)

Automatisierungen eignen sich auch dann, wenn es Aufgaben gibt, wo eine schnelle Rückmeldung oder eine hohe Reaktionsfähigkeit gefordert ist. Als Beispiel könnte man die Überwachung von IT-Systemen in Unternehmen nehmen. Wenn ein Ausfall oder ein Fehler innerhalb des Systems auftritt, sollte schnell reagiert werden. Durch eine zu langsame Reaktion oder Fehlerbehebung könnte es dadurch zu einem hohen Kostenverlust und Zeitverlust kommen. Die Prozessautomatisierung kann man in dem Fall bei der automatischen Benachrichtigung oder der automatischen Fehlererkennung nutzen. Auch kann die Automatisierung dabei hilfreich sein Fehler generell mit einer schnellen Reaktion zu beheben, falls diese Fehlerarten schon bekannt sind.

Händische Prozesse, welche eine hohe Komplexität besitzen, sollten besonders zum Automatisieren. Es kann Aufgaben geben, welche viele vereinzelte Schritte haben und diese sehr Zeitaufwändig und Komplex gelöst werden müssen, diese können auch durch, wie schon beschrieben, die hohe Komplexität und den hohen Zeitaufwand, zu vereinzelten oder mehreren und schwerwiegenden Problemen und Fehlern durch die menschliche Arbeit und Beteiligung am Projekt oder Aufgabe.

Noch dazu sollte man dann Automatisieren, wenn die Konsistenz nicht leicht eingehalten werden kann. Wenn die Einhaltung von Standards und Regeln nicht durch menschliche Arbeit allein eingehalten werden kann oder sich die Menge an Standards und Regeln sehr stapelt, sollte eine Prozessautomatisierung in Betracht gezogen werden und automatisiert werden. Dabei kann es auch sein, dass sich Regeln mit der Zeit ändern können und deswegen verschiedene Angestellte umgeschult werden müssen, um eine perfekte Konsistenz liefern zu können. RPA-Bots müssen in dem Fall nur einmal angepasst werden und können dann eine gute Konsistenz und Einhaltung von Standards und Regeln liefern. Dadurch kann die Reputation und Vertrauen der Kunden geschützt werden, was vielleicht durch die menschliche Arbeit nicht so gut funktionieren kann falls wie schon beschrieben, viele Vorschriften einzuhalten sind. RPA-Bots können so implementiert sein, um regelmäßig Tests durchzuführen, um auf eine Einhaltung zu prüfen. Auch durch die Sammlung von verschiedenen Daten und Bewertung dieser, kann eine Einhaltung geliefert werden.

Eine Automatisierung sollte auch dann verwendet werden, wenn eine Kommunikation zwischen zwei verschiedenen Systemen und eine Verbindung zwischen diesen aufgebaut werden muss. Auch beispielsweise bei der Migration von SAP-ERP Systemen auf die neuere Version der Business Software S/4HANA.

3.1.5 Welche Prozesse eignen sich zur Automatisierung³

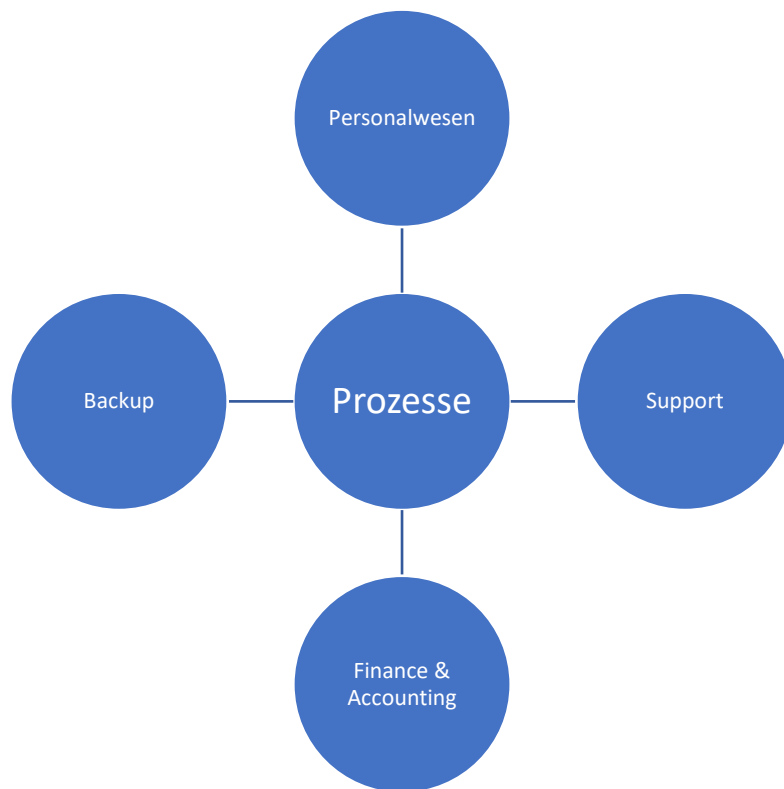


Abbildung 3: geeignete Prozesse für Automatisierung

- **Personalwesen**
Prozesse welche im Human Resources / Personalwesen durchgeführt werden, eignen sich besonders für Automatisierungen. Dabei gibt es Aufgaben wie beispielsweise Abwesenheitsverwaltungen, Reporting, HR-Analysen und Gehaltsabrechnungen. Da viele Aufgaben des Personalwesens sehr repetitiv sind und diese auch bestimmte standardisierte Abläufe besitzen, ist die Automatisierung dieser Prozesse besonders passend.
- **Support**
Im Support oder auch Kundenservice hört man sehr oft von Prozessautomatisierungen in diesem Bereich. Dabei werden automatische Antworten auf Fragen, welche gestellt werden, durch Erkennung von auffallenden Wörtern im Text oder voreingestellte und auswählbare Fragen, gegeben. Häufig gestellte Fragen werden dem RPA-Bot mit den jeweiligen Antworten bereitgestellt. Auch können durch Automatisierungen, Probleme, nach einem Ticketsystem angeordnet und die wichtigeren an den menschlichen Support übergeben werden.

³ Prozesse: <https://nativdigital.com/rpa-prozesse/>

- Finance & Accounting
Prozesse im Finance & Accounting haben meist auch repetitive und stark regelbasierte Abläufe. Dazu gehören Aufträge wie beispielsweise das Buchen von Rechnungen oder das Schreiben und Erstellen von Berichten. Rechnungswesensprozesse haben noch dazu viele Regeln, welche eingehalten werden müssen, da ein Mensch jedoch nicht immer in der Lage sein kann diese einzuhalten bietet sich eine Automatisierung dieser perfekt an. Das Buchen von Rechnungen, müssen durch Automatisierungen, nicht von Menschen erstellt werden, damit können Fehler vermieden werden und viel Zeit und Kosten eingespart werden. Berichte werden oftmals wiederholend nacheinander erstellt, diese können durch einen RPA-Bot binnen Sekunden geschrieben werden.
- Backup
Backups im Allgemeinen haben viele Prozesse, welche automatisiert werden können. Dabei gibt es als Beispiel die Continuous Data Protection, welche eine kontinuierliche Erfassung aller Änderungen liefert und diese in einer Backup Datei speichert. Um beispielsweise mit einem Backup von einer Datenbank, die Datenbank wiederherzustellen, müssen Recovery Prozesse ausgeführt werden. Ein Recovery, ist der Ablauf, wo die Daten des Backups angewendet werden und die Datenbank mit den Informationen der Backup Datei geändert wird. Bei einem Point in Time Recovery welcher beispielsweise die Datenbank, zu einem bestimmten Zeitpunkt wiederherstellen kann, bietet sich eine Automatisierung dieses Recovery-Prozesses an. Durch eine Automatisierung der Continuous Data Protection und das Recovery dieses Backups, kann viel Zeit gespart werden. Auch Recovery-Tests bieten sich für eine Automatisierung an, diese führen ein Recovery zu einem Backup aus und überprüfen auf Fehler, die dabei passieren könnten. Diese können durch ein automatisiertes schreiben eines Fehlerberichts, an Zuständige geschickt werden, welche diese dann bearbeiten und analysieren.

3.1.6 Wie soll die Automatisierung realisiert werden

Es gibt viele Tools und Softwareprodukte, die verwendet werden können, um die Automatisierung effektiv durchzuführen. Um eine Flexibilität gewährleisten zu können, besteht die Möglichkeit die Prozessautomatisierung Cloud-basiert umzusetzen. Diese Lösung hat nebenbei auch noch den Vorteil einer guten Skalierbarkeit, und kann meist zumindest eine Überlegung wert sein.

Als aller erstes sollte den Personen, welche diese Arbeit derzeit vollrichten, über die Automatisierung informiert werden. Dabei sollte glaubwürdig vermittelt werden, dass die Software tatsächliche Vorteile bietet und die Mitarbeitenden unterstützen soll. Denn System und dessen Funktionen, sollte in Zusammenarbeit mit diesen Mitarbeitern erstellt werden.

Die Realisierung dieser Automatisierung erfordert einen strukturierten Ansatz, der verschiedene Schritte umfasst. Zunächst ist die Bestimmung der Prozessart entscheidend, denn die Erfolgsaussichten bei der Automatisierung hängen von der Komplexität ab. Es ist ratsam eine detaillierte Skizzierung der Prozesse zu erstellen, sowie die Definition von Aufwand und Aufgabenstellung sind entscheidend, um ein Verständnis für die Komplexität und die damit verbundenen Kosten der Automatisierung zu erhalten. Außerdem sollte eine Kosten-Nutzen-Analyse durchgeführt werden. Nach der Auswahl eines geeigneten Prozesses folgt die Entscheidung für ein passendes Automatisierungssystem. Sollte keine eindeutige Entscheidung möglich sein, gibt es auch noch die Möglichkeit einer Teilautomatisierung.

Die Überwachung der Automatisierung nach dem Abschluss der Entwicklung, sollte auch durchgeführt werden, um zu überprüfen, ob das System wie erwartet läuft und um potenzielle Ausfälle zu vermeiden. Um die Funktionalität und Effektivität des automatisierten Prozesses zu überprüfen, sind regelmäßige Kontrollen ratsam. Falls sich die Rahmenbedingungen oder Anforderungen im Laufe der Zeit ändern, können Anpassungen vorgenommen werden. (vgl. Lanuschy, 2022)

3.1.7 Vollautomatisierung und Teilautomatisierung

Bei der Vollautomatisierung ist kein Eingriff von einem Menschen notwendig. Der Software-Roboter führt die vorgefertigten und kalibrierten Vorgänge durch und beendet diese binnen Sekunden. Dabei hat die Vollautomatisierung eine hohe Effizienz und Genauigkeit beim Ausführen von Prozessen. Die Herausforderung dabei ist, dass die Automatisierungen einen reibungslosen und vollständigen Ablauf liefern muss, da es sonst zu Problemen kommen könnte.

Teilautomatisierung hingegen automatisiert, wie es im Wort schon vorkommt, nur einen Teil des Prozesses. Der Mensch ist hierbei in dem Prozess miteingebunden, der andere Teil des Prozesses wird also durch Menschen gesteuert. Die Automatisierung wird in dem Fall von dem Benutzer als Hilfsmittel hergenommen und dann gestartet, wenn es nötig ist. Das bedeutet die Teilautomatisierung beschreibt die Zusammenarbeit zwischen Menschen und Robotern. Die humane Arbeit wird bei der Teilautomatisierung nicht vernachlässigt und schafft sogar mehr Arbeitsmöglichkeiten. Allerdings kann es dazu kommen, dass Prozesse im Vergleich zu der Vollautomatisierung anfälliger für Fehler sind.

Vollautomatisierungen sind also effizienter und schneller als Teilautomatisierungen. Wobei Teilautomatisierungen im engen Kontakt mit menschlichen Entscheidungen stehen und dadurch der Mensch nicht komplett durch Bots ersetzt wird. Die Wahl welche Automatisierungsart gewählt wird, hängt ganz und allein von den spezifischen Anforderungen und Problemstellungen des Prozesses ab.

3.1.8 Verschiedenen Aspekte von RPA

RPA hat verschiedene Typen in der das Thema unterteilt werden kann. Diese Typen kann man wiederum in Haupt- und Nebentypen unterteilen. Unter den Haupttypen versteht man das regelbasierte RPA, das cognitive RPA, das IPA und die Hyperautomation.

Das regelbasierte RPA ist das herkömmliche RPA, das sich auf Aufgaben mit einer klar definierten Struktur konzentriert. Dies könnte zum Beispiel ein sich immer wiederholender Prozess sein, der sich vom Ablauf aus gesehen nicht verändert.

Die Art des cognitiven RPAs ist schon sehr viel weiterentwickelter und verwendet schon Technologien wie Natural Language Processing (NLP), Machine Learning (ML) und Artificial Intelligence (AI) verwendet. Dies ermöglicht dem cognitive RPA auch komplexe Aufgaben zu automatisieren. Die cognitive RPA kann mit dieser Technik Entscheidungen treffen oder den Menschen interpretieren. Dieser RPA-Typ kann von seinen Erfahrungen lernen und auch seine Leistungen nach und nach verbessern.

IPA (Intelligent Process Automation) ist eine große Erweiterung zum cognitive RPA, da IPA nicht „nur“ komplexe Aufgaben bewältigen kann, sondern seine kognitiven Fähigkeiten auch für nicht-strukturierte Abläufe einsetzen kann. IPA lernt nicht nur mit seinen gemachten Erfahrungen, sondern kann sich auch selbst seine benötigten Daten erlernen. Seine Informationsquelle kann hierbei alles Mögliche sein. IPA kann seine gesammelten Informationen dann auch analysieren und diese wie der Bot es wünscht weiterverarbeiten. Beispielsweise nutzt ChatGPT auch diese Technik, durch das Deep-Learning welches angewendet wird, kann dieser Bot lernen und Entscheidungen zu bestimmten Fällen treffen.

Die Hyperautomation ist der nächste Evolutionsschritt der Prozessautomatisierung. Bei dieser Methode werden verschiedene Technologien, wie maschinelles Lernen, KI, Entscheidungsmanagement kombiniert. Daraus ergibt sich eine Ende-zu-Ende Automatisierung, womit sich gesamte Geschäftsprozesse auf Unternehmensebene automatisieren lassen. Im Prinzip kann man damit jede sich wiederholende Tätigkeit automatisieren.

3.1.9 IPA

IPA (Intelligent Process Automation) ⁴ ist eine große Erweiterung zur Prozessautomatisierung. Der Unterschied zu der herkömmlichen RPA ist es, dass IPA mit Hilfe von einer KI (Künstlicher Intelligenz) und ML (Maschinelles Lernen) dazu lernen und wichtige Entscheidungen während Prozessen auf Basis von gesammelten Daten treffen kann. Beispielsweise kann eine IPA als Kunden Chat-Bot fungieren, indem es Entscheidungen nach den Wörtern im Text des Kunden trifft und antwortet. Auch könnte Chat-GPT in eine RPA-Anwendung integriert werden. Chat-GPT ist eine Künstliche Intelligenz, welche als Chat bot agiert, diese benutzt Deep Learning welche das Lernen durch Daten im Internet und vom User geschriebenen Sachen ermöglicht.

Unterschied RPA-IPA

RPA wird für gleiche Prozesse und Aufgaben benutzt, welche immer wiederholt werden. IPA allerdings wird für komplexere Prozesse verwendet wo Entscheidungen und humanoide Aufgaben gelöst werden müssen.

⁴ IPA: https://en.wikipedia.org/wiki/Intelligent_automation

3.1.10 Anwendungsfälle

Automatisierung S/4HANA ⁵

S/4HANA ist die vierte Produktgeneration der von SAP entwickelten SAP-Software-Suite, welches ein System für das zentrale Datenmanagement von Unternehmen ist. Sehr viele Systeme von Firmen basieren immer noch auf der Vorgänger Version SAP-ERP. Mit der neuesten Generation kommen sehr viele ausschlaggebende Änderungen hinzu, wie beispielsweise eine neue Benutzeroberfläche, eine vereinfachtere Datenstruktur und eine In-Memory-Datenbank, welche den Arbeitsspeicher für die Speicherung nutzt und somit für eine bessere Effizienz und Leistung sorgt, wenn es zu der Verarbeitung und Behandlung von großen Datenmengen (Big Data) kommt. Da viele Unternehmer ihr System auf die neueste Version S/4HANA aktualisieren möchten, kommt es zu Migrationsproblemen von Daten, welche in die In-Memory-Datenbank migriert werden müssen. Weil vor allem diese Migration sehr kostspielig und zeitaufwändig ist, kann man Robotic Process Automation sehr oft im Zusammenhang mit S/4HANA hören. Somit kann man den Prozess des manuellen Datentransfers, die Eingabe und Analyse von Big Data automatisieren. Auch wird die UI-Path Automatisierungs-Software benutzt, um große Test-Anwendungsfälle mit der neuen SAP-UI durchzuführen. Diese können binnen Sekunden ausgeführt und getestet werden, um somit viel Zeit zu sparen. Auch kann man mit RPA die individuellen, firmeneigenen Codierungen von SAP-Systemen auf Syntax Fehler überprüfen und analysieren.

Kundenservice

Der menschliche Kundenservice ist meistens eine kostspielige und zeitaufwändige Arbeit, wobei oftmals normale Fragen gestellt werden. Dabei werden viele Mitarbeiter gebraucht, welche sich zu jeder Zeit um Kunden kümmern müssen. Das Personal unterstützt diese mit hilfreichen Lösungen zu ihren Anliegen und Problemen. Der Kundenservice ist ein sehr wichtiger Bestandteil eines Unternehmens, da dieses den Kontakt zwischen Abnehmer und der Firma herstellt. Durch den Einsatz von Robotic Process Automation wird durch automatisierte Antworten und Tätigkeiten, welche ein Mitarbeiter des Kundenservices durchführen muss, viel Geld und Zeit gespart. Bezüglich Fragen-Bots werden verschiedene Automation Tools verwendet, wobei vorgefertigte Antworten auf zuvor definierte Fragen gegeben werden. Auch wird oftmals die Automatisierung eines Chat-bots, mit Hilfe von künstlicher Intelligenz verbessert, dadurch muss es nicht nur automatisierte Antworten auf Auswahl-Fragen geben, sondern es können auch intelligente und spezifischere Antworten und Lösungen herangeführt werden. Darüber hinaus besteht der Kundenservice nicht nur aus einfachen Fragen und Antworten, sondern auch beispielsweise aus einer Kundenfeedback-Analyse, einer Vertragsverwaltung oder aus einem Beschwerdemanagement. Alle diese Prozesse können automatisiert durchgeführt werden und müssen grundsätzlich nicht von einem Menschen bearbeitet werden. (vgl. Safar, 2024)

⁵ S/4HANA: <https://rz10.de/knowhow/sap-s-4hana/>

Rechnungsstellungen

Rechnungen auszustellen kann für einen Hersteller viel kostbare Zeit in Kauf nehmen. Natürlich dauert eine Rechnungsausstellung nicht stundenlang, jedoch wenn mehrere Rechnungen ausgestellt werden müssen, häufen sich diese und man verliert viel Zeit, welche man besser hätte investieren können. Natürlich gibt es in einem Unternehmen bestimmte Menschen, die dafür zuständig wären, jedoch geht dies mit einer Automatisierung viel schneller und kosteneffizienter. Dadurch braucht man das Personal nicht mehr beim Rechnungen ausstellen, sondern kann diese speziell und effizienter in anderen Bereichen einsetzen. Durch die Automatisierung der Rechnungsstellung werden zudem auch Fehler vermieden, welche das menschliche Auge auf den ersten Blick vielleicht nicht deutlich erkennen kann. Auch können durch Automatisierungsprozesse, eine Analyse und Überprüfung der Rechnungen durchgeführt werden, um alle Richtlinien und Vorschriften zu beachten. Außerdem können die Rechnungen direkt mittels Automatisierung in ein Buchhaltungssystem integriert werden. Dies ist ein weiterer Punkt, wo man durch RPA viel Geld einsparen kann. Die Automatisierung ist jedoch auch schon so weit fortgeschritten, dass diese sowohl Rechnungen als auch Mahnungen an Kunden verschicken kann, Da der Automatisierungs-Roboter meistens alle notwendigen Daten bereits besitzt, kann dieser Prozess sehr leicht durchgeführt werden. Generell durch die Verwendung von RPA in der Rechnungsstellung wird eine schnellere Zahlungsabwicklung und eine bessere Effizienz erzielt, womit man auch auf deutliche Kosteneinsparungen kommt. (vgl. Gurd, 2022)

3.2 UiPath



Abbildung 4: UiPath Logo

UiPath⁶ ist ein rumänisch-amerikanisches Softwareunternehmen, das uns ermöglicht mit den verschiedensten Tools einen immer wiederkehrenden Geschäftsprozess zu automatisieren. UiPath steht für "User Interface Path", was so viel bedeutet wie "dem Arbeitspfad des Benutzers zu folgen".

3.2.1 UiPath Studio

UiPath Studio dient zur Entwicklung eines Softwareroboters. In diesem Programm hat man die verschiedensten Aktivitäten, die zum Beispiel Applikationen öffnen/schließen können oder mit der Applikation auf verschiedenste Weise interagieren können, wie es der Applikationsbenutzer tun kann. Die Aktivitäten kann man bei einem Layout-Diagramm, welches sich in einem Workflow befindet, mittels Drag & Drop hineinziehen und die Aktivitäten dort nach Belieben bearbeiten. Das Bearbeiten der Aktivitäten ist üblicherweise sehr Code-arm gestaltet. Die verschiedenen Layout-Diagramme, wie zum Beispiel eine Sequence und ein Flowchart kann man auch verknüpfen, um diese zu einem einzigen Prozess zusammenzufassen. Bei der Entwicklung von einem UiPath Studio Prozess steht einem auch eine sogenannte "Aufzeichnen"-Funktion zur Verfügung. Wie der Name schon verraten lässt, kann man einen Prozess aufzeichnen. Der von UiPath Studio aufgezeichnete Prozess kann manchmal auch 1:1 verwendet werden. Allerdings sollte man sich den durch die Aufzeichnung generierten Prozess unbedingt noch einmal anschauen, bevor man diesen in die Entwicklung weitergibt, da dieser aufgezeichnete Prozess nur als Grundgerüst für eine vereinfachte Entwicklung von einem UiPath Studio Projekt dienen soll.

3.2.2 UiPath Orchestrator

Der Orchestrator von UiPath ist ein Knotenpunkt der auch als Kontrollzentrum bezeichnet wird. Das Programm verwaltet die Benutzer, Roboter, Maschinen, Softwarepakete, Warteschlangen und noch vieles mehr. Bei den Softwarepaketen handelt es sich zum Beispiel um die Prozesse, die man mit UiPath Studio erstellt hat und in die UiPath Cloud geladen hat. Zusätzlich kann man im Orchestrator die vorhandenen Roboter und Prozesse manuell starten. Die hierbei entstehenden Ergebnisse werden mitprotokolliert und können auch später noch einmal eingesehen werden. Wenn man eine spezielle Ergebnisausgabe haben möchte, kann man dies auch im Projekt zum Beispiel mittels eines simplen Logs erhalten.

⁶ UiPath (Home Page): <https://www.uipath.com>

UiPath (Wikipedia): <https://en.wikipedia.org/wiki/UiPath>

UiPath Logo: <https://logowik.com/ui-path-logo-vector-svg-pdf-ai-eps-cdr-free-download-14931.html>

3.2.3 Vorteile gegenüber händischem Programmieren

Der größte Vorteil von UiPath ist es, dass man Prozesse 1:1 übernehmen kann und der Softwareroboter dem ihm gegebenen Ablauf immer befolgt. Zusätzlich muss man deutlich weniger Code schreiben, da einem dies die Benutzeroberfläche größtenteils abnimmt. Allerdings kann auch eine eigene oder selbst programmierte Aktivität erstellt werden. Zusätzlich können alle Prozesse, die man erstellt und in UiPath Orchestrator hochgeladen wurden, mittels einer API ausgeführt werden.

3.3 Jira

Jira ⁷ ist eine Webanwendung des Unternehmens Atlassian, womit Software Development Projekte verwaltet und verfolgt werden können. Es wird hauptsächlich in der Softwareentwicklung genutzt und es beschäftigt sich mit dem Projektmanagement, der Fehlerverwaltung und der Statusverfolgung.



Abbildung 5: Jira Logo

Jira bietet einen Überblick über den Stand des Projektes und wer mit welcher Arbeit beschäftigt ist. Dem Team ermöglicht es somit eine umfangreiche Planung während eines laufenden Projekts zu gestalten. Das Programm hilft dem zuständigen Projektteam bei der Planung von Vorgängen, einer gezielten Realisierung von Sprints und der Aufgabenverteilung im Team.

Die Webanwendung stellt unter anderem auch vorgefertigte Workflows zur Verfügung. Diese können allerdings auch eigens angefertigt werden. Zudem stellt Jira ein Product Backlog zur Verfügung bei welchem alle User Stories, Vorgänge und Bugs auf einmal überblickt und mit Drag & Drop bearbeiten werden können. Alle Änderungen können sofort von den einzelnen Teammitgliedern eingesehen werden.

Zusätzlich können mehrere Issues in einem Projekt erstellt werden. Diese repräsentieren einzelne Aufgaben, die noch zu erledigen sind. Üblicherweise sind diese Aufgaben größere Features, vorhandene Software Bugs oder Benutzeranforderungen. Die Issues können allerdings pro Projekt auf die verschiedenste Weise variieren, da jedes Projekt andere Anforderungen aufweist. Im Übrigen können Aufgaben und Issues auch mit einem Fälligkeitsdatum versehen werden. Jedes Teammitglied kann einsehen, wie viel Zeit bleibt, bis die jeweilige Aufgabe erledigt werden muss und natürlich, auch falls diese Zeit überschritten worden ist. Außerdem kann jedes Teammitglied in einem Jira-Projekt seine erledigten Dateien hochladen, so dass diese für jedes Teammitglied sichtbar ist.

⁷ Jira (Atlassian): www.atlassian.com/de/software/jira

Jira (Wikipedia): [https://de.wikipedia.org/wiki/Jira_\(Software\)](https://de.wikipedia.org/wiki/Jira_(Software))

Jira Logo: https://commons.wikimedia.org/wiki/File:Jira_Logo.svg

3.4 SharePoint

SharePoint ist ein Tool von Microsoft welches auch in Microsoft-365 enthalten ist. Es ermöglicht die einfache Zusammenarbeit und die Verwaltung von Dokumenten und Projekten. Mit Microsoft SharePoint können verschiedenste Personen gleichzeitig an Dokumenten in Echtzeit arbeiten. Somit hat man eine einzige Version eines Dokuments, welches von jedem gesehen und bearbeitet werden kann. Damit so eine Zusammenarbeit erst möglich ist, müssen die Dateien freigegeben werden. SharePoint kann entweder lokal auf eigenen Servern in der Firma oder auf der von Microsoft zur Verfügung gestellten Cloud laufen. Auch können Aufgabenlisten und Kalender für alle Teammitglieder erstellt und verwaltet werden. Durch SharePoint können Firmen eine bessere Organisation ermöglichen.

3.4.1 Workflows

SharePoint hat die Möglichkeit verschiedene Geschäftsprozesse zu Automatisieren. Diese Funktion gibt es seit 2010 - damals noch mit dem Namen Microsoft Workflow. Heute gibt es eine neue und verbesserte Version des Workflow Tools mit dem Namen Microsoft Flow. Beispielsweise kann man eine Workflow Automatisierung zum Verschieben oder Genehmigen von Dokumenten erstellen.

4 AUSWIRKUNG AUF DIE ARBEITSWELT

4.1 Grundsatz

Mit der Einführung von Robotic Process Automation in Unternehmen, verändert sich auch die Arbeitswelt. Die Mitarbeiter können sich mehr auf ihre Kernaufgaben fokussieren, während die repetitiven Geschäftsprozesse durch die Automatisierung Nebensache werden. Die Angst vor Jobverlust ist zwar gerechtfertigt, jedoch passiert dies in den wenigsten Fällen, da RPA für die Erleichterung der Arbeit geeignet ist, anstatt für die Ersetzung von Arbeitskräften. Das Zentrum für Europäische Wirtschaftsforschung (ZEW)⁸ in Mannheim hat durch ihre Forschungen herausgefunden, dass es durch die Automatisierung zum Jobabbau kommen wird, jedoch werden dadurch viele neue Berufe dazu gewonnen. Beispielsweise der RPA-Entwickler, welcher sich hauptsächlich um die Automatisierung und Verbesserung von Geschäftsprozessen im Unternehmen kümmert. Der RPA-Berater ist eine weitere Arbeitsmöglichkeit. Er hat die Aufgabe herauszufinden, wo eine Automatisierung im Unternehmen sinnvoll wäre. Er ist auch für die Kommunikation und Problemlösung zwischen dem Kunden und dem Prozessautomatisierungsteam zuständig.

⁸ ZEW: <https://www.zew.de/>

4.2 Anfänge der Automatisierung

Die Automatisierung ist kein neues Konzept. Schon vor tausenden Jahren wurden Versuche unternommen Arbeitsschritte zu automatisieren. Doch erfolgreich war erst eine Erfindung aus Ägypten, von dem Mathematiker und Ingenieur Heron von Alexandria. Dieser entwickelte ein Jahrhundert nach Christus den ersten Verkaufsautomaten, welcher Wein und Milch spenden konnte.

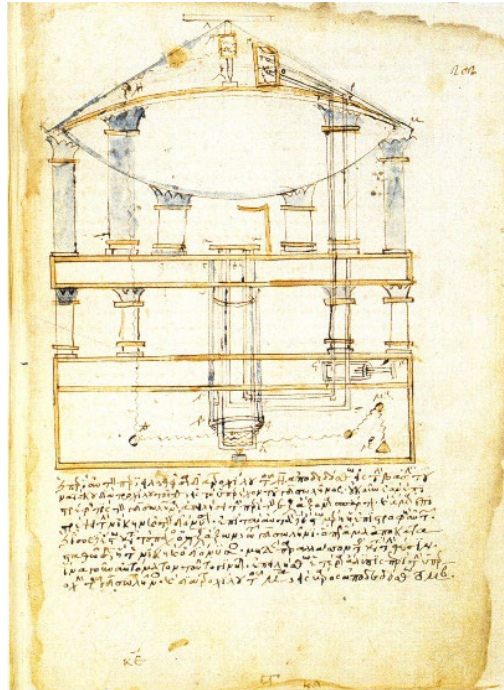


Abbildung 6: Automata 13, Heron von Alexandria⁹

Doch so richtig Fahrt hat die Automatisierung erst in der Zeit der Industriellen Revolution aufgenommen. Gegen Ende des 18. Jahrhunderts kam es in der Textilindustrie zur Verwendung einer bahnbrechenden dampfbetriebenen Webmaschine, die zu einer erheblichen Steigerung der Produktion führte. Ein bedeutender Meilenstein wurde dann durch die Nutzung von Elektrizität erreicht, als Henry Ford im Jahr 1913 das erste Automodell namens "Tin-Lizzy" in der Fließbandfertigung produzierte. (Künzli, 2019) Ein weiterer entscheidender Schritt in der Automatisierungsgeschichte ereignete sich im Jahr 1956, als der Erfinder George Devol und der Ingenieur sowie Unternehmer Joseph Engelberger sich zufällig auf einer Cocktailparty trafen und eine Partnerschaft bildeten. Gemeinsam erstellen sie ein Konzept für den ersten Industrieroboter, der den Namen "Unimate" trug. Schließlich wurde drei Jahre später ein Prototyp dieser bahnbrechenden Erfindung bei General Motors getestet. Im Jahr 1961 fand der "Unimate"¹⁰ schließlich seinen Einsatz an der Fertigungsstraße, wo er für das Schweißen von Kfz-Karosserien verwendet wurde. Die Bedeutung der Autoproduktion für die Innovation in der Geschichte der Automatisierung setzte sich auch nach den vorherigen Meilensteinen fort. Im Jahr 1974 präsentierte das schwedische Unternehmen Asea (heute ABB) einen bahnbrechenden Roboter. Dieser wird als der erste vollständig elektrische kommerzielle Industrieroboter mit Mikroprozessorsteuerung angesehen. (vgl. AutomationNEXT, 2017)

⁹ Heron von Alexandria:

https://de.wikipedia.org/wiki/Datei:Heron_of_Alexandria_Automata_Venice_Gr_516.jpg

¹⁰ Unimate: <https://de.wikipedia.org/wiki/Unimate>

Jedoch nicht nur in der Produktion wurde Automation eingeführt. Als Ender der siebziger Jahre der Personal Computer immer verbreiteter wurde, bekam die Automation einen völlig neuen Anwendungsbereich. Die Auswertung von Daten. Diese wurde besonders durch die Einführung von Tabellenkalkulationssoftware vorangetrieben.

ITEM	NO.	UNIT	COST
MUCK RAKE	4	12.50	50.00
BUCK CUT	1	100.00	100.00
TONER	25	49.00	1225.00
EYE SNUFF	2	4.00	8.00
SUBTOTAL			1315.00
9.75% TAX			128.38
TOTAL			14438.16

Abbildung 7: VisiCalc ¹¹

Die Entwicklung des Internets Ende der 90er Jahre verstärkte diesen Trend und es wurde eine erhebliche Zunahme der Datenanalyse verzeichnet. Mit dem Aufkommen von mobilen Endgeräten, die ebenfalls Daten produzierten, stieg die Datenmenge kontinuierlich an. Infolgedessen wurde die automatisierte Auswertung dieser Daten immer wichtiger, und es wurden intelligente Technologien dafür entwickelt.

Insgesamt hat die fortschreitende Digitalisierung, insbesondere mit dem Aufkommen des Internets, die automatisierte Datenanalyse und Datenverarbeitung maßgeblich vorangetrieben, wodurch intelligente Technologien und die Bewältigung von Big Data zu zentralen Elementen in verschiedenen Bereichen der Technologie und Wirtschaft geworden sind. (vgl. Künzli, 2019)

¹¹ Visacalc: <https://de.wikipedia.org/wiki/Visicalc>

4.3 Wirtschaftliche Vorteile

Mit der Einführung von modernen IT-Infrastrukturen können ganze Geschäftsprozesse automatisiert werden, was zu einer hohen Zeitersparnis führt. Mit dieser Entwicklung sind Effizienzsteigerung und Kostensenkung zu einem wichtigen Faktor für die Automatisierung geworden. Denn je effizienter diese Systeme sind, desto mehr Geld kann eingespart werden. (vgl. Künzli, 2019)

Doch nicht nur im IT-Bereich sind diese Vorteile spürbar, sondern auch in anderen Sektoren der Industrie kann die Automatisierung zu einer erheblichen Kosteneinsparung führen. Die Nutzung von Robotern in Fabriken ist ein Beispiel für die Produktivitätssteigerung, da diese in der Lage sind, monotone Aufgaben ohne Unterbrechungen durchzuführen und dass nicht nur schneller, sondern auch zuverlässiger als menschliche Arbeitskräfte. Die erhöhte Zuverlässigkeit führt zu einer Steigerung der Produktqualität. Hinzu kommt der Faktor, dass Roboter im Vergleich zu Mitarbeitern wesentlich kostengünstiger sind. (vgl. Draese, 2023)

4.4 Angst vor Jobverlust

Auch wenn es so klingt, dass durch die Einführung von Automatisierungsprozessen in Unternehmen viele Angestellte ihre Jobs verlieren werden, dient dies nicht zur Sorge. Die ZEW¹² hat herausgefunden, dass zwischen dem Zeitraum von 2016 bis 2021, 560.000 neue Berufsmöglichkeiten erschaffen worden sind. Wie bereits beschrieben kann jedoch nicht alles automatisiert werden. Ein Mensch mit einem klaren Denkvermögen, muss trotzdem diese Automatisierungsprozesse im Auge behalten. Eine Automatisierung dient nur dem Abschaffen von Arbeiten, welche sich ständig wiederholen und wo kein klares Denken nötig ist. Arbeitsplätze mit Teilautomatisierungen kommen in dem Fall sehr in Frage, Menschen können hierbei die Arbeiten, welche zuvor händisch erledigt worden sind, automatisieren und müssen sich nur auf das logische Denken und das Treffen von Entscheidungen konzentrieren. Natürlich könnte diesen Entscheidungsprozess in der späteren Laufbahn auch eine künstliche Intelligenz übernehmen, wobei künstliche Intelligenzen noch nicht auf diesem Level des klaren Denkens sind, um den Menschen komplett zu ersetzen. Also könnten Berufe durch RPA erleichtert und sich viel mehr auf das Logische konzentriert werden. Dadurch kann man auch sehr viel Zeit sparen und viel mehr Arbeitsabläufe durchführen ohne viel Zeit zu investieren.

¹² ZEW-Studie: <https://www.sueddeutsche.de/wirtschaft/digitalisierung-automatisierung-arbeitsplaetze-zew-studie-1.4501771>

4.5 Neue Arbeitsplätze

4.5.1 RPA Developer

Als RPA Developer hat man nicht nur die Aufgabe des Entwickelns, sondern auch das Designen von Prozessen und Entwerfen von Lösungen. Der Entwickler muss hierbei auf mehrere Eigenschaften wie die Fähigkeit zu analysieren, zu konzeptionieren und zu entwickeln zurückgreifen. Als RPA-Entwickler hat man auch eine große Auswahl an Tools, welche man zum Automatisieren verwenden kann. Diese können je nach Art der Automatisierung, also eine Teilautomatisierung oder eine Vollautomatisierung entschieden oder schon vorgegeben sein. (Safar, Weissenberg, 2024) Als RPA Developer ist es vor allem wichtig sich in allen notwendigen Entwicklungswerkzeugen auszukennen. Wenn man diesen Beruf ausüben möchte, sollte man natürlich Erfahrungen in der Softwareentwicklung und eine entsprechende Ausbildung haben. Dies ist sehr wichtig, wenn man bedenkt, dass auch wenn Tools, wie beispielsweise UI-Path, eher einen Low-code Ansatz benutzen, trotzdem sehr oft auf normale Softwareentwicklung und Programmiersprachen zurückgegriffen wird. Die Robotic Process Automation entwickelt sich täglich weiter, daher ist es sehr wichtig in der Lage zu sein immer etwas Neues zu erlernen. Die Einhaltung von Sicherheitsmaßnahmen und Datenschutzrichtlinien ist als Entwickler im Mittelpunkt. RPA umfasst sehr oft sensible Daten, welche nicht an die Öffentlichkeit gelangen dürfen, weswegen man sich immer bewusst sein sollte die Richtlinien einzuhalten. (vgl. Marko, 2020)

4.5.2 RPA-Strategist

Der Robotic Process Automation Strategist ist nicht unbedingt an der direkten Entwicklung von Prozessen beteiligt. Dieser behält einen Überblick über alle wichtigen Geschäftsprozesse des Unternehmens und entscheidet anhand von diesen, neue Automatisierungskonzepte, welche dem Unternehmen viel Geld und Zeit einsparen können. Der RPA Strategist muss perfekt mit der Kombination und den Abläufen von Automatisierungs-Prozessen vertraut sein, um eine optimale Verbindung herzustellen. Die Kernaufgabe ist dabei eine ausführliche Strategie für Automatisierungen innerhalb des Unternehmens für wichtige Geschäftsprozesse zu entwickeln. Dabei ist es sehr bedeutsam kreative und innovative Konzepte zu erzeugen, um die Effizienz und Produktivität zu steigern. Im Zusammenhang mit dem RPA Developer sieht man, dass der Strategist eine Gesamtstrategie für den Developer entwickelt, welcher dann wiederum direkt an der Umsetzung der Strategie beteiligt ist. Jedoch ist es trotzdem notwendig, als Automation Strategist ein tiefgehendes Wissen über die Automatisierung zu besitzen. (vgl. Marko, 2020)

4.5.3 RPA-Architekt

RPA-Architekten sind zuständig für die technische Infrastruktur und Implementierung von Automatisierungs-Werkzeugen. Diese Infrastrukturen und Werkzeuge werden durch die Developer benötigt, weswegen die RPA-Architekten eine sehr große Rolle bei den Automatisierungen in einem Unternehmen spielen. Architekten integrieren Tools und Werkzeuge in die RPA-Plattformen und sorgen sich um die perfekte Funktionalität untereinander. Dabei werden auch verschiedenste Technologien beachtet und diese auch in das Unternehmen eingebracht. Alle Entscheidungen im Zusammenhang mit der Automatisierungspolitik und Strategien werden durch den RPA-Architekten entschieden und getroffen. Der Architekt ist also nicht nur für die technische Infrastruktur und Implementierung zuständig, sondern auch für die unterschiedlichsten Entscheidungen rundum Automatisierung. Auch wird das Kostenmanagement in Bezug auf RPA durch den RPA-Architekten geleitet. Diese entscheiden durch ständige Analysen, zu welchen Kosten die Einführung von jeweiligen Automatisierungsprozessen und Werkzeugen oder Tools führt. Natürlich steckt eine sehr enge Mitarbeit mit verschiedensten Teams dazwischen. Ein RPA-Architekt muss nicht nur die bereits beschriebenen Charakteristiken besitzen, sondern auch Wissen in mehreren Programmiersprachen, Kenntnisse in Quellcode, Kenntnisse im Projektmanagement und Erfahrungen mit Finanzen haben. (vgl. Marko, 2020)

4.6 Qualifikationen

Wenn man in der Robotic Process Automation Branche tätig sein möchte, sollte man ein tiefes Verständnis über die Automatisierung von Prozessen haben und sich in der Technik oder in Programmiersprachen sehr gut auskennen. Es ist außerdem maßgeblich einen bestimmten Ausbildungszweig und eine Ahnung von Softwareentwicklung haben. Ein Verständnis über verschiedene Automatisierungs-Tools zu besitzen ist auch erforderlich. Es kann sein, dass Unternehmen verschiedene Werkzeuge verwenden, weshalb es notwendig ist ein Wissen über alle angeeignet zu haben. Wie auch in anderen Branchen muss man Fähigkeiten in Kommunikationen und Teamarbeit besitzen. Wie schon beschrieben ist die RPA-Entwicklung keine 1-Mann Arbeit, dabei arbeiten mehrere Teams ständig miteinander und müssen immer in der Lage sein gegenseitig zu kommunizieren. Natürlich ist es auch von Nöten eine Dokumentationsfähigkeit zu besitzen, Dokumentation und Qualitätssicherung ist in Bezug auf Robotic Process Automation ein sehr elementarer Prozess. Ohne Dokumentationen können beispielsweise die anderen Teammitglieder nicht weiterarbeiten. Durch die Dokumentation können auch Fehler besser gefunden und falls Mitglieder in das Automatisierungs-Projekt eingearbeitet werden müssen, kann dies auch leichter gemacht werden. Auch wie schon klargestellt, sollte man immer bereit sein Neues zu erlernen, das kann durch verschiedene Schulungen vom Unternehmen oder externen Anbietern passieren.

4.7 Menschliche Überwachung und Kontrolle

Wenn eine Automatisierung durchgeführt wird, ist es nicht unbedingt notwendig, ständig eine menschliche Überwachung oder Kontrolle während den Automatisierungsprozessen zu haben. Solange Automatisierungen gut und fehlerfrei entwickelt sind, können diese fehlerfrei arbeiten. Trotz dessen ist es klug eine Kontrolle von Zeit zu Zeit durchzuführen, um auf mögliche Fehler und Risiken zu kommen. Zudem kümmern sich RPA-Manager auch um die Überwachung und Kontrolle von Robotic Process Automation Prozessen. Dabei werden die Performance, das Fehlermanagement, die Skalierbarkeit, die Sicherheit und Verbesserungsmaßnahmen besonders beachtet. Der RPA-Manager hat zwar nicht die Aufgabe eine ständige Überwachung und Kontrolle durchzuführen, ist jedoch trotzdem für den reibungslosen Ablauf und Einhaltung von Geschäftsforderungen zuständig. Die Observation kann beispielsweise durch Echtzeitüberwachung während bestimmter Automatisierungs-Prozesse oder durch Analyse und Protokollierung mit Hilfe von Hilfs-Tools abgewickelt werden. Falls jedoch die menschliche Überwachung nicht ausreichen sollte oder komplett scheitern sollte, gibt es spezielle Tools für die Kontrolle von RPA-Bots. Eines dieser Werkzeuge wäre Camunda, welcher einen Lifecycle der gesamten Prozesse, die ein RPA-Bot durchführt, liefert und analysiert. (vgl. Camunda, 2023) (vgl. Safar, Weissenberg, 2024)

4.8 Zukunftsaussichten der Automatisierung

Die Zukunft verändert sich mit all ihren guten und schlechten Aspekten stetig weiter. Vor allem bei der Erforschung neuer Technologien wie der KI steht die Welt nicht still. Natürlich betrifft dies auch die Prozessautomatisierung und hat bereits zu einem enormen Anstieg von Forschungen im Bereich der No-Code-RPA¹³ geführt. No-Code-RPA ermöglicht die Entwicklung von Arbeitsbereichen, die ohne viel Programmiercode automatisiert werden können, beispielsweise durch Drag-and-Drop-Funktionen. Ein weiterer Fortschritt ist das Self-Learning RPA System, das verschiedene Recorder-Tools umfasst, welche eigenständige Bots verwenden, um Tastatur- und Mausinteraktionen zu verfolgen und aufzuzeichnen. Dieses System wird kontinuierlich weiterentwickelt, um in Zukunft eine automatisierte Nutzung der vom Bot erstellten Automatisierung ohne Überprüfung zu ermöglichen. Mit dem Aufkommen der KI gewinnt auch die cognitive Automation, oder Intelligent Process Automation (IPA), an Bedeutung. Die Zusammenarbeit zwischen KI und RPA zielt darauf ab, Bots autonomer zu machen und auch unstrukturierte Prozesse problemlos zu bewältigen. Die KI und Machine Learning können dabei helfen, Automatisierungen durch Varianzen zu ermöglichen, indem sie in Echtzeit automatische und adaptive Korrekturen vornehmen.

Wie bereits angesprochen, wird die KI in den kommenden Jahren auch in der RPA-Branche immer präsenter sein. Die KI soll nicht nur bei der Erstellung automatisierter Prozesse helfen, sondern auch flexibel und intelligent auf unerwartete Ereignisse reagieren können. Obwohl Softwareroboter dies derzeit noch nicht können, wird durch die Zusammenarbeit von KI und Robotern erwartet, dass immer mehr Bots lernen, wie sie auf verschiedene Situationen reagieren sollen. Dieser Prozess wird zwar einige Zeit in Anspruch nehmen und anfangs von Menschenhand unterstützt werden müssen, aber es ist absehbar, dass die Softwarebots in der Lage sein werden, eigenständig auf Ausnahmesituationen zu reagieren, da KIs immer intuitiver werden und sich zunehmend selbstständig weiterbilden. Dennoch werden die Bots vorerst noch von Menschen angeleitet.

Die Zukunft der Automation verspricht zudem eine Vielzahl von neuen Entwicklungen und Herausforderungen. Laut einer Studie der Boston Consulting Group¹⁴ wird die Robotikbranche von professionellen Servicerobotern dominiert werden und einen erheblichen Umsatzanstieg im Vergleich zu konventionellen Robotern und Logistikrobotern verzeichnen. Prognosen gehen davon aus, dass der globale Robotikmarkt von 25 Milliarden US-Dollar im Jahr 2022 auf 160 bis zu 260 Milliarden US-Dollar bis 2030 anwachsen wird. Ein bedeutender Faktor für die steigende Bedeutung der Robotik liegt in den veränderten Verbraucherpräferenzen und gesellschaftlichen Trends. Die Nachfrage nach schnelleren Lieferungen kundenspezifischer Produkte wird die Roboterkapazitäten in der Fertigungsindividualisierung und in Logistikanwendungen erweitern.

Zusätzlich werden in der Zukunft viele schlecht bezahlte und weniger qualifikationsintensive Jobs von Robotern übernommen. Nach einer Studie des McKinsey Global Institute¹⁵ könnten bis 2030 durch die Automatisierung rund 800 Millionen Arbeitsplätze verloren gehen. Vor allem sind Routineaufgaben, welche einfach automatisiert werden können. Die Softwareroboter können zwar auch Menschen in der Übernahme von langweiligen Routinearbeiten ersetzen,

¹³ No-Code-RPA: Automatisierung ohne, dass Programmierfähigkeiten erforderlich sind

¹⁴ Studie der Boston Consulting Group: https://web-assets.bcg.com/img-src/20190325_BCG%20Presse_Advanced%20Robotics_tcm9-216999.pdf

¹⁵ Studie des McKinsey Global Institute: <https://www.mckinsey.com/featured-insights/future-of-work/ai-automation-and-the-future-of-work-ten-things-to-solve-for>

doch dies sollte nicht als negativ betrachtet werden. Vielmehr eröffnen sich dadurch neue Möglichkeiten für den Menschen. Auch für die Unternehmen ist dies vorteilhaft, da sie ihre Ressourcen effizienter einsetzen können, Zeit und Geld sparen und gleichzeitig eine „neue“ Mitarbeiterkraft in einer anderen Abteilung gewinnen können, die wichtige Aufgaben erledigt. Die Nutzung von RPA wird voraussichtlich zu einer weiteren Verbreitung führen, da in Zukunft nicht nur Routineaufgaben, sondern auch komplexere Prozesse automatisiert werden können. Dies wird höchstwahrscheinlich zu einer breiteren Akzeptanz von RPA führen und die globale Wahrnehmung von KI und RPA beeinflussen. (vgl. Talentra, 2023), (vgl. Hamberger, 2022), (vgl. Thiele, 2024)

5 DATENVERARBEITUNG

Inmitten der stetig wachsenden Flut an Daten, die aus unterschiedlichsten Quellen und in vielfältigen Formaten stammen, steht die effektive Datenverarbeitung vor einer grundlegenden Herausforderung. Der Umfang und die Diversität dieser Daten erfordern eine Lösung, die über herkömmliche manuelle Prozesse hinausgeht. Hier setzt die Datenautomatisierung als essenzieller Prozess an, der es Unternehmen ermöglicht, nicht nur große Datenmengen, sondern auch die Vielfalt der Datenquellen und Datentypen effizient zu bewältigen. Der Fokus liegt darauf, Daten nahtlos zu erfassen, zu bearbeiten und zu verarbeiten, ohne dabei auf manuelle Eingriffe angewiesen zu sein.

In der Bewältigung dieser Herausforderung spielen verschiedene Strategien eine entscheidende Rolle. Ziel ist es, die Daten zu transformieren und daraus geschäftlich relevante Erkenntnisse zu gewinnen. Besonders wichtig ist dabei die Verarbeitung unstrukturierter Rohdaten, um saubere und hochwertige Informationen zu generieren. Dafür wird die Datenautomatisierung genutzt, indem sie wiederholende und zeitaufwändige Aufgaben automatisiert. Von der Erfassung über die Umwandlung, Validierung und Bereinigung bis hin zur Integration und Analyse von Daten ermöglicht die Datenautomatisierung Unternehmen, ihre Daten optimal zu nutzen.

Der Prozess der Datenautomatisierung gliedert sich in drei Hauptelemente: Extrahieren, Transformieren und Laden (ETL). Dieser ETL-Prozess wird in drei klaren Schritten unterteilt. In einem ersten Schritt erfolgt die Extraktion von Daten aus einer oder mehreren Quellen. Anschließend durchlaufen die Daten eine Transformation, bei der sie in das erforderliche Format des Zielsystems überführt werden. Diese Transformation beinhaltet verschiedene Maßnahmen wie Sortieren und Filtern. Schließlich werden die bearbeiteten Daten in das Zielsystem geladen, sei es eine Datenbank oder ein Data Warehouse.

5.1 Probleme

Trotz der fortgeschrittenen Möglichkeiten der Datenautomatisierung sind jedoch einige Herausforderungen und potenzielle Probleme zu beachten, die im Zuge der Datenverarbeitung auftreten können.

Die Vielfalt der Datenquellen und Datentypen stellt eine Herausforderung dar. Unterschiedliche Datenformate erfordern flexible Ansätze in der Verarbeitung. Die Automatisierung muss in der Lage sein, mit den verschiedensten Datentypen umzugehen. Daten können auch aus Bildern oder Audio-Dateien extrahiert werden. Dies erfordert jedoch fortschrittliche Algorithmen und Technologien, um eine umfassende und genaue Datenverarbeitung sicherzustellen.

Schließlich bleibt trotz der Automatisierung menschliches Eingreifen bei der Fehlerbehebung unerlässlich. Bei komplexen Fehlern müssen Menschen die Ursache eines Problems verstehen und geeignete Lösungen zu implementieren. Eine effektive Kommunikation und Zusammenarbeit zwischen automatisierten Systemen und menschlichen Entscheidungsträgern ist daher entscheidend.

5.2 Vorteile

Die Verarbeitung riesiger Datenmengen aus verschiedenen Quellen wird für Menschen zunehmend schwierig. Computer können jedoch diese komplexe und zeitaufwändige Aufgabe effizient bewältigen. Die Datenautomatisierung ermöglicht eine schnelle Standardisierung und

Validierung der Daten, bevor sie in ein einheitliches System geladen werden. Dadurch wird die Analyse beschleunigt, und Entscheidungen können auf fundierten Informationen basieren.

Weiters würde eine manuelle Verarbeitung großer Datenmengen das Risiko menschlicher Fehler mitbringen. Durch die Datenautomatisierung wird diese Fehleranfälligkeit reduziert, indem sie ein konsistentes und strukturiertes Laden der Daten gewährleistet. Die resultierende verbesserte Datenqualität ist von entscheidender Bedeutung für präzise und zuverlässige Datenverarbeitung.

Die Verbindung von Datenautomatisierung und Data Mining ermöglicht es Unternehmen, in kürzerer Zeit präzise Erkenntnisse aus großen Datensätzen zu gewinnen. Durch die automatisierte Verarbeitung und Analyse beschleunigt sich der Prozess der Mustererkennung erheblich. Diese Automatisierung gewährleistet nicht nur den Zugang zu umfangreichen Daten, sondern auch zu hochpräzisen Informationen. Die resultierende Genauigkeit fördert die Qualität der Data Mining-Ergebnisse, da Algorithmen auf fundierten und aussagekräftigen Daten basieren. Insgesamt entsteht eine leistungsstarke Kombination, bei der Datenautomatisierung und präzise Daten die Effektivität von Data Mining maximieren und Unternehmen ermöglichen, tiefgehende Einblicke in ihre Daten zu gewinnen. (vgl. Naeem, 2024), (vgl. databricks, 2024)

6 DATENSCHUTZ UND RISIKEN

6.1 Allgemeine Risiken

Automatisierungsprozesse werden oft mit dem Zugriff auf wichtige und sensible Daten verknüpft, wodurch Robotic Process Automation zu einem interessanten Ziel für Cyberkriminelle wird, die darauf abzielen, private und sensible Informationen zu erlangen. Ein Mangel an Sicherheitsvorkehrungen erhöht das Risiko eines unbefugten Zugriffs auf diese Daten erheblich und macht sie anfälliger für Datenschutzverletzungen. (vgl. Aich, 2023), (vgl. Savanovich, 2023) Dafür gibt es spezielle Richtlinien der DSGVO (Die Datenschutz-Grundverordnung), welche eingehalten werden müssen. Als Risiko der RPA kann auch die unzureichende Konfiguration oder Programmierung des Prozesses gezählt werden, Geld Verluste und Rufschädigung können dabei sehr leicht auftreten. Durch die immer wachsende Komplexität von Automatisierungs-Prozessen, erhöhen sich auch die Datenschutzanforderungen. Der Datenschutz bei großen und komplexen Prozessen muss deswegen sehr im Auge behalten werden.

6.2 Datenschutz-Richtlinien

Das Datenschutzgesetz ist insbesondere bei RPA wichtig, wenn sensible oder personenbezogene Daten verarbeitet werden. Personenbezogene oder sensible Daten umfassen Informationen wie Vorname, Nachname und Privatanschrift. Dabei kommt EU-weit die Datenschutz-Grundverordnung (DSGVO) ¹⁶ ins Spiel. Diese regelt Richtlinien und Datenschutzanforderungen für den Schutz und die Verwaltung von personenbezogenen Informationen. In den USA würde in diesem Fall der CCPA (California Consumer Privacy Act) beachtet werden. Die Hauptrichtlinie der DSGVO besteht darin, sensible Daten zu schützen. IT-Administratoren und Programmierer müssen Konfigurationen durchführen, um sicherzustellen, dass nur autorisiertes Personal oder autorisierte Bots Zugriff auf diese Details haben. (vgl. Anywhere, 2024)



Abbildung 8: DSGVO Logo

Um Datenschutz zu gewährleisten, sollten bei der Implementierung Verschlüsselungen, Zugriffsbeschränkungen, Authentifizierungsmechanismen und Überwachungssysteme berücksichtigt werden. Schon bei der Umsetzung des Automatisierungsprozesses sollten Datenschutz- und Sicherheitsrichtlinien beachtet und angewendet werden, um die Prozesse frühzeitig gemäß den Richtlinien zu gestalten und zukünftige Probleme zu erkennen. Es ist auch wichtig, vor der Implementierung Datenschutzrisiken zu identifizieren und zu bewerten, um bereits vor der Entwicklung Sicherheitsmaßnahmen und Risikominderung zu implementieren. Die Technologien und Anforderungen der Automatisierung sollten gemäß den Datenschutzgesetzen und Risiken implementiert werden.

Vor allem sollten Entwickler, die RPA-Lösungen umsetzen, regelmäßige und gründliche Schulungen erhalten, in denen sie über Datenschutzrichtlinien und Risikominderungsstrategien informiert werden. Es sollten nicht nur theoretische Aspekte des Datenschutzes vermittelt werden, sondern auch praktische Themen, wie beispielsweise die Sicherstellung von Sicherheit und Lizenzierung im Code.

¹⁶ DSGVO: <https://dsgvo-gesetz.de/>

DSGVO Logo: <https://www.kmu-center.at/datenschutz-grundverordnung/>

6.3 Risiken

6.3.1 Beschaffungsrisiken

Entscheidet sich ein Unternehmen dazu, einen Prozess zu automatisieren, jedoch allein intern und ohne externe Hilfe, können dadurch möglicherweise hohe Kosten entstehen. Ein Mangel an Fachwissen oder begrenztes Personal birgt das Risiko von Kostenerhöhungen. Gleiches gilt, wenn externe RPA-Berater erst zu einem späteren Zeitpunkt einbezogen werden. (vgl. Roboyo, 2019)

6.3.2 Tool Risiken

Die Verwendung ungeeigneter Tools für die Automatisierung von Prozessen kann zu erheblichen Instandhaltungsproblemen und Kosten führen. Dieses Verhalten wird oft auch als RPA-Washing bezeichnet. RPA-Washing beinhaltet die Nutzung von Prozessautomatisierungen, um dem aktuellen Markttrend zu folgen. Unternehmen automatisieren häufig kleinere und unnötige Prozesse, nur um behaupten zu können, dass sie RPA einsetzen. Dadurch können sie ihre Produkte unter dem Etikett "mit RPA-Unterstützung" besser vermarkten. Diese Strategie kann jedoch auch zum Verhängnis werden, da sie Instandhaltungsprobleme und hohe Kosten mit sich bringt und somit Risiken birgt. (vgl. Roboyo, 2019)

6.3.3 Stakeholder

Beim Implementieren eines Prozesses ist es wichtig, Stakeholder wie in jedem herkömmlichen Projekt zu berücksichtigen. Wenn jedoch nicht alle erforderlichen Stakeholder vollständig eingebunden sind, kann dies zu Problemen bei Entscheidungen und der Bedeutung des RPA-Projekts führen. Eine unzureichende Einbindung externer Stakeholder, die dazu führen kann, dass Bedürfnisse abgelehnt oder ignoriert werden, birgt die Gefahr, dass die Umsetzung des RPA-Projekts gefährdet wird. (vgl. Violino, 2020)

6.3.4 Strategie

Wenn Firmen Automatisierungen nur isoliert verwenden und diese für Kostensenkungen implementiert werden, können diese zu Problemen bei Verwendung der richtigen Strategien führen. Isolierung von Prozessautomatisierungen, können die Zusammenarbeit von verschiedenen Unternehmensbereichen und Projekten erschweren. Außerdem wird durch Automatisierung nur für Kostensenkung und nicht für Verbesserung von Arbeitsabläufen und Innovation, der Ruf des Unternehmens erheblich geschädigt. Auch durch zu schnelle Übernahme von Automatisierungen im Unternehmen ohne eine genaue Strategie und Vorgehensweise, trägt das Risiko, der Firma stark zu schädigen. (vgl. Roboyo, 2019) (vgl. Kosmopoulos, 2021)

6.3.5 Über-Automatisierung

Wenn Firmen Automatisierungen nur isoliert verwenden und diese ausschließlich zur Kostensenkung implementiert werden, können sie Probleme bei der Anwendung der richtigen Strategien verursachen. Die Isolierung von Prozessautomatisierungen kann die Zusammenarbeit zwischen verschiedenen Unternehmensbereichen und Projekten erschweren. Außerdem wird durch Automatisierung zur Kostensenkung und nicht zur Verbesserung von Arbeitsabläufen und Innovationen, der Ruf des Unternehmens sehr geschädigt. Auch eine zu schnelle Übernahme von Automatisierungen im Unternehmen ohne eine richtige und genaue Strategie und Vorgehensweise birgt das Risiko, die Firma stark zu schädigen. (vgl. Kosmopoulos, 2021)

6.4 Maßnahmen für Sicherheit und Datenschutz

6.4.1 Rechte und Governance

Es ist von entscheidender Bedeutung, dass ein Unternehmen eine genaue Governance-Struktur aufbaut, um mehr Sicherheit beim Erstellen der RPA-Bots zu gewährleisten. Der Governance-Rahmen muss mit klaren Rollenverteilungen und Verantwortlichkeiten geschaffen werden. Die Einrichtung und Sicherung dieser Struktur dürfen ausschließlich von autorisierten Personen durchgeführt werden. Zugriffsberechtigungen im Rahmen eines RPA-Entwicklungsteams spielen ebenfalls eine große Rolle. Die Zugriffsrechte jedes Mitglieds müssen eingeschränkt werden, damit jeder nur die ihm zugewiesenen Aufgaben erledigen kann. Eine solche Einschränkung kann durch Benutzeranmeldungen erreicht werden. Die Benutzer erhalten dann spezifische Rechte und können nur auf die Teilbereiche der Automatisierung zugreifen, für die sie verantwortlich sind. Zusätzlich können auch Bot-Rechte eingeschränkt werden. Bots können dann nur die ihnen zugewiesenen Aufgaben ausführen. Zum Beispiel könnte ein Bot, der nur zwei Excel-Dokumente vergleichen und bearbeiten muss, Lese- und Schreiboperationen durchführen. Der Bot hätte jedoch keine Berechtigung, diese Excel-Dokumente zu löschen (Bremmer, 2021).

RPA-Entwickler müssen sich aller Risiken und Sicherheitslücken bewusst sein, insbesondere wenn sie neue Robotic Process Automation Bots entwickeln. Bot-Entwickler sollen daher Schulungen und Weiterbildungen erhalten, um eine bessere Sicherheit und Datenschutz zu gewährleisten. (vgl. SAFAR, Weissenberg, 2024)

6.4.2 Logs und Protokolle

Logs und Protokolle, die beim Ausführen der RPA-Bots erstellt werden, sind streng zu schützen. Diese enthalten sensible Informationen, die dem Bot zuvor bereitgestellt wurden, um bestimmte Prozesse auszuführen. Dabei können Daten wie Benutzernamen, E-Mail-Adressen oder andere Informationen enthalten sein. Logs und Protokolle sollten idealerweise in einem separaten System gespeichert werden. Dies ist besonders wichtig, wenn die Prozesse noch nicht vollständig entwickelt sind und Schwachstellen aufweisen. Denn Sicherheitslücken können durch Protokolle leicht aufgedeckt werden. Ebenso sollte die Vollständigkeit von Logs und Protokollen sichergestellt werden. Die Vollständigkeit ist wichtig, da es sonst zu Problemen bei der Untersuchung der Logs und Protokolle kommen kann. (vgl. SAFAR, Weissenberg, 2024)

6.4.3 Sicherheitsabteilung

Zusätzlich zur Sicherheit ist es notwendig, eine Sicherheitsabteilung einzubinden. Diese ist entscheidend, um sicherzustellen, dass alle Abläufe in der RPA-Entwicklung den Sicherheitsstandards entsprechen. Die Sicherheitsabteilung muss mit den RPA-Entwicklern zusammenarbeiten, um für eine bessere Sicherheit zu sorgen. In den meisten Fällen werden Sicherheitsabteilungen erst gegen Ende der Implementierung von den Entwicklern dazu gezogen, um eine Genehmigung der Abteilung zu erhalten. Jedoch durch eine frühere Einbindung können Sicherheitsstandards und Richtlinien besser befolgt und Risiken früher erkannt und entsprechend reagiert werden. Dadurch können viele Ressourcen wie Geld eingespart werden.

Außerdem hilft die Zusammenarbeit mit der Sicherheitsabteilung dabei, die Sicherheitsanforderungen und -risiken im Zusammenhang mit RPA umfassend zu bewerten. Dadurch wird sichergestellt, dass sensible Daten geschützt und die Integrität gewahrt bleibt. Letztendlich trägt die aktive Einbindung der Sicherheitsabteilung dazu bei, dass RPA-Projekte reibungsloser ablaufen und das Risiko von Sicherheitsvorfällen minimiert wird.

7 TOOLS UND ANBIETER

Im Bereich der Robotic Process Automation gibt es einige Anbieter, mit denen man Prozesse automatisieren kann. Hierbei ist UiPath zwar der führende Anbieter, da dieses Unternehmen auch neuen Produktnutzern eine ausführliche Erklärung mit den verschiedensten Tutorials zur Verfügung stellt, allerdings gibt es noch viele mehr von denen die meisten RPA Developer noch nichts wussten. Hier werden wir allerdings nicht nur die beliebten, sondern auch von dem leistungsstärksten Anbieter mit deren Tools veranschaulichen.

7.1 Automation Anywhere

¹⁷ Automation Anywhere ist ein amerikanisches Softwareunternehmen, das sich auf die Prozessautomatisierung spezialisiert hat. Mittels der gleichnamigen Software kann man somit, wie in UiPath seine Geschäftsprozesse automatisieren und mithilfe der



Abbildung 9: Automation Anywhere Logo

Softwareroboter diese Automatisierung ausführen. Automation Anywhere stellt neben dem RPA-Arbeitsbereich auch einen Discovery-Bot, einen IQ-Bot und die Überwachung von seinen eigens erstellten Bots zur Verfügung. (vgl. Baitech Data, Was ist Automation Anywhere, 2023)

7.1.1 Discovery-Bot

Der Discovery-Bot von Automation Anywhere, ist dient dazu mittels Task Mining den Benutzeraktionen, wie Maus- oder Tastaturinteraktionen zu folgen und so eine fertige Automatisierung zu erstellen. Dies kann nützlich sein für kleinere Automatisierungen, allerdings sollte man immer überprüfen, ob die vom Bot erstellte Automatisierung auch korrekt aufgenommen wurde und durchgeführt werden kann.

7.1.2 IQ-Bot

Der IQ-Bot ermöglicht es nicht sichtbare oder unstrukturierte Daten, die man zum Beispiel in E-Mails oder anderen Dokumenten finden kann, zu extrahieren und zu klassifizieren. Diese können mit Künstlicher Intelligenz, Maschinellem Lernen, Computer Vision und Natural Language Processing behandelt werden, um die Automatisierung des Geschäftsprozesses zu erleichtern.

7.1.3 Bot-Überwachung

Bei der Überwachung seiner Bots kann man die Aktivitäten der Bots in Echtzeit analysieren und verwendete Daten mit den gefundenen Informationen, die für den Prozess, den der Bot gerade ausführt, verwendet werden überprüfen. Aus den hier gesammelten Daten kann man mithilfe von Automation Anywhere Vorhersagen über die verschiedensten Daten treffen.

¹⁷ Automation Anywhere: <https://www.automationanywhere.com/de>

Automation Anywhere Logo: <https://www.prnewswire.com/news-releases/itc-infotech-und-automation-anywhere-leisten-pionierarbeit-bei-der-digital-workforce-885534183.html>

7.2 Blue Prism

Blue Prism ¹⁸ wurde vom britischen Softwareunternehmen Blue Prism Group entwickelt und ist ebenfalls eine führende Software im Bereich von RPA.

Die Hauptkomponenten von Blue Prism sind die Prozessdiagramm, das Process Studio, das Object Studio und der Application Modeler. Ein Prozessdiagramm ist nichts anderes als eines Workflows, der dazu dient, die automatisierten Prozesse darzustellen und dient auch als Softwareprogramm. Process Studio ist ähnlich wie UiPath Studio und ist eine Anwendung, in dem die Workflows erstellt und erweitert werden. Im Object Studio hingegen werden Visual Business Objects erstellt. Diese Objekte können allerdings im Process Studio verwendet werden. Der Application Modeler existiert im Object Studio und erstellt Anwendungsmodelle. (vgl. Baitech Data, Was ist Blue Prism, 2023)

Für eine bessere Veranschaulichung der vier Hauptkomponenten könnte man als Beispiel die automatische Datenextraktion einer E-Mail. Diese Daten sollen dann automatisch in eine Datenbank abgespeichert werden.

- **Prozessdiagramm:** Der Prozess wird hier vom Developer erstellt und grafisch dargestellt.
- **Process Studio:** Mittels dem Process Studios kann man nun das Prozessdiagramm verwenden und den eigentlichen Prozess automatisieren, indem der Developer die verschiedensten Aktionen in das Prozessdiagramm zieht. Ohne dieser Anwendung hätte man nur ein leeres Prozessdiagramm.
- **Object Studio:** Wenn man nun speziellere Objekte und Aktionen benötigt, verwendet man das Object Studio. Diesen erstellten Aktionsblock kann man, wie die bereits vorhandenen und vorgefertigten Blöcke im Process Studio verwenden.
- **Application Modeler:** Der Application Modeler ist nun für die Benutzeroberfläche zuständig. Hierbei kann man Elemente wie Textfelder und Buttons einfügen und darstellen. Diese können dann im Object Studio verwendet werden, um die Interaktionen mit den dargestellten Elementen mit der Anwendung zu verknüpfen und auch zu automatisieren.



Abbildung 10: Blue Prism Logo

¹⁸ Blue Prism: <https://www.blueprism.com/de/>

Blue Prism Logo: <https://azuremarketplace.microsoft.com/en-us/marketplace/apps/blueprismlimited-4827145.blueprismweb?tab=overview>

7.3 OpenText

OpenText¹⁹ ist ein kanadisches Unternehmen, das sich auf Enterprise Information Management spezialisiert hat und verschiedenste Softwarelösungen zur Verfügung stellt, die Unternehmen dabei helfen soll ihre Daten korrekt zu verwalten und zu verarbeiten. Mithilfe dieser



Abbildung 11: OpenText LiquidOffice Logo

Programme können also Daten und Dokumente gesichert, gesammelt, geordnet und abgerufen werden. Das Unternehmen OpenText hat dies nun so weit gebracht, dass sie eine Software zur Formularautomatisierung anbieten. Diese Software heißt LiquidOffice. (vgl. Wiefel, 2013)

7.3.1 LiquidOffice

LiquidOffice ist also eine Technologie, mit der man PDF- und Web-Formulare problemlos erstellen kann. Die in den Formularen eingetragenen Informationen sind nach dem Ausfüllen direkt abrufbar und können eingesehen und weiterverarbeitet werden. LiquidOffice besitzt ebenfalls einen Workflow Designer mit der man wie mit anderen Automatisierungssoftware einen Prozess modellieren kann. Mithilfe von dieser Prozessmodellierung kann man nach Ausfüllen des Formulars einen weiteren Prozess abhandeln. Diese können Benachrichtigungen sein, dass das Formular abgeschickt wurde oder auch die Zuweisung von verschiedenen Aufgaben an eine Abteilung zum Beispiel. Mithilfe von LiquidOffice können allerdings auch externe Benutzer auf die verschiedenen Formulare Zugriff erhalten um diese einsehen oder sogar bearbeiten zu können.

7.3.2 SAP S/4HANA

In S/4HANA wird LiquidOffice ebenfalls eingesetzt. Es dient also dazu die Formulare und Dokumente zu verwalten, die man in seinem SAP-System anlegt und verwendet. Die Funktionen, wie Workflows, Genehmigen von Prozessen und Versenden von Benachrichtigungen, verbessern die Geschäftsprozesse und erhöhen die Effizienz von S/4HANA im Allgemeinen.

¹⁹ OpenText: <https://www.opentext.com>

OpenText LiquidOffice Logo: <https://www.optiform.com/news/opentext-liquidoffice-16-4-new-release-notes/>

7.4 Pegasystems Inc.

Das amerikanische Unternehmen Pegasystems Inc.²⁰ ist auf die Entwicklung von Kundenbeziehungsmanagement und Geschäftsprozessmanagement spezialisiert und entwickelten die Low-Code-Plattform Pega.



Abbildung 12: Pega Logo

7.4.1 Low-Code-Plattform

Wenn man von einer Low-Code-Plattform redet, kann man sich eine Software vorstellen, die den Prozess der Softwareentwicklung erheblich beschleunigt und vereinfacht, da sie nur sehr wenig herkömmliches Codeschreiben benötigt. Low-Code bezieht sich deshalb auf die Entwicklung von Anwendungen mit nur wenigem Code-Aufwand und ermöglicht es Menschen ohne professionelles Fachwissen, Anwendungen zu erstellen oder anzupassen. Allerdings können komplexere Aufgaben trotzdem dazu führen ein gewisses Level an Fachwissen besitzen zu müssen. (vgl. GBTEC, 2023)

7.4.2 Pega-Plattform

Pega ist also eine Plattform, die dazu dient mittels einer KI-Entscheidungen zu treffen und Workflowautomatisierungen erstellen und durchzuführen. Die von Pegasystems entwickelte Software hat, wie viele andere Automatisierungssoftware, ebenfalls eine visuelle Entwicklungsumgebung, wo man mittels Drag-and-Drop-Funktionalität seine Prozesse automatisieren kann. Pega kann mit den KI-unterstützten Elementen der Software auch nicht nur herkömmlich statische Prozesse, sondern auch selbstständig komplexere Prozesse erkennen und automatisieren. Die hierbei verwendeten Prozesse für die Entscheidungsfindung für die Automatisierung antrainierten Daten der KI umfassen die verschiedensten Analysen und Regeln. (vgl. Pega, 2024)

²⁰ Pega: <https://www.pegacom.de>

Pega Logo: <https://www.capgemini.com/de-de/unternehmen/technologie-partner/pega/>

7.5 Microsoft Power Automate

Auch das weltweit bekannte Unternehmen Microsoft hält sich nicht von der Prozessautomatisierung fern und entwickelte eine Anwendung namens „Microsoft Flow“, die sie vor kurzem zu „Microsoft Power Automate“²¹ umbenannt haben.

7.5.1 Microsoft Power Automate

Früher konnte man das von Microsoft entwickelte Programm nicht nur auf dem PC lokal als auch in Webbrowser nutzen. Man konnte die damalige Microsoft Anwendung sogar am Handy nutzen. Mittlerweile ist Microsoft Power Automate eine rein cloudbasierte Automatisierungsplattform.



Microsoft Power Automate bietet früher als auch jetzt an, dass Benutzer, ohne großartig viel Programmierkenntnisse zu benötigen, ebenfalls ganz leicht einen Prozess zwischen verschiedenen Anwendungen automatisieren können. Allerdings bietet Microsoft mit ihrer umbenannten Software einige neue als auch verbesserte Integrationen an, sowie das Erstellen einer Automatisierung mithilfe deren eigener KI für Power Automate. Außerdem bietet Microsoft nicht nur das Automatisieren von Microsoft Produkten, sondern auch von vielen anderen anerkannten Drittanbietern an.

Darüber hinaus gibt es in Microsoft Automate auch zwei neue Funktionen, die es in Microsoft Flow früher nicht gab. Diese wären der Power-Berater und auch der KI-Baukasten. Mithilfe des Power-Beraters können vor allem Nutzer, denen das Automatisieren und Programmieren fremd ist, eine passende und einfache Automatisierungslösung finden, da der Power-Berater den Nutzern Empfehlungen gibt, wie sie einen Workflow automatisieren können (Natürlich basiert dies ebenfalls auf Künstlicher Intelligenz und Maschinellem Lernen.). Mithilfe des KI-Baukastens werden dann auch die Formularverarbeitung erleichtert, indem man mit Microsoft Power Automate in einem Dokument die verschiedenen Objekte erkennen kann und mit einer einfach gestalteten Benutzeroberfläche diese extrahieren. (vgl. Baitech Data, 2024)

²¹ Microsoft Power Automate: <https://www.microsoft.com/de-at/power-platform/products/power-automate>
Microsoft Power Automate Logo: <https://www.cognitofirms.com/product/microsoft-power-automate>

8 KÜNSTLICHE INTELLIGENZ

Eine Künstliche Intelligenz (KI) ²² ist prinzipiell ein sehr komplexes Programm, das Menschen nicht nur nachahmen, sondern auch in manchen Bereichen sogar übertreffen kann. Es kann Schlussfolgerungen treffen, selbstständig handeln, die menschliche Fantasie und Gefühle nachahmen, sich an die verschiedensten Umgebungen anpassen und auch verschiedenste Details wahrnehmen. Die KI basiert prinzipiell auf Deep Learning und Machine Learning, womit sie sich selbst Wissen anlernen kann. Desto mehr Daten die KI aufgenommen hat, desto schlauer wird sie auch. Auch wenn es jetzt noch für viele Menschen so aussehen mag, dass die KIs dumm sind und uns nur nachahmen und niemals ersetzen könnten, wird dies wohl nicht mehr allzu lange dauern bis dieser Fall wirklich eintritt...

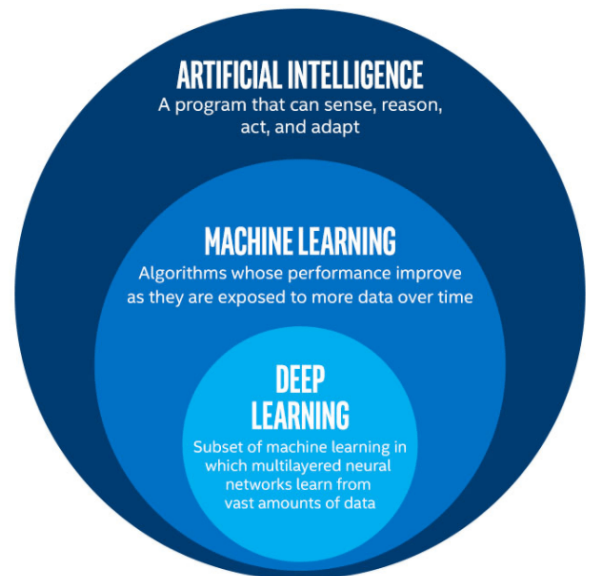


Abbildung 13: Künstliche Intelligenz, Machine Learning und Deep Learning

8.1 Deep Learning

Das prinzipielle Konzept von Deep Learning ²³ ist die Informationsverarbeitung von großen Datenmengen. Man verwendet dieses Konzept hauptsächlich zum Erkennen von Bildern, zum Verstehen von Texten (mündlich als auch schriftlich) oder zur besseren Entscheidungsfindung, was die KI zum Beispiel antworten soll. Hierbei wird das menschliche Gehirn nachgeahmt (KNN: künstliche neuronale Netze). Die Daten werden also so vom KNN verarbeitet, wie wir Informationen mit unserem Gehirn verarbeiten. Die bereitgestellten Daten werden somit zuerst von der Datenquelle aufgenommen, danach werden sie analysiert und im Anschluss wird eine Schlussfolgerung gezogen, was die Daten aussagen wollen.

Da normalerweise Big Data (also eine riesige Datenmenge) für die Trainingsdaten eingesetzt wird, ist Deep Learning ein sehr rechenintensives Konzept. Bei einem ausführlichen Training einer funktionstüchtigen KI werden also in den meisten Fällen einige Wochen und Monate oder sogar Jahre benötigt, um immer das richtige Ergebnis zu erzielen. Dafür ist die KI beim endgültigen Einsatz umso schneller und kann innerhalb von wenigen Millisekunden erkennen, was man von der KI will oder was die KI erkennen soll. Deshalb ist es bei Deep Learning immer wichtig eine größtmögliche Datenmenge zur Verfügung zu stellen, um die KI so genau wie möglich zu trainieren.

²² Künstliche Intelligenz: <https://www.iks.fraunhofer.de/de/themen/kuenstliche-intelligenz.html>

Abbildung 12: Künstliche Intelligenz, Machine Learning und Deep Learning: <https://datasolut.com/machine-learning-vs-deep-learning/>

²³ Deep Learning: <https://datasolut.com/was-ist-deep-learning/>

8.1.1 Beispiel für Deep Learning: Menschen erkennen

Um einer KI beizubringen, dass ein Mensch vor ihr steht, werden verschiedenste Trainingsdaten verwendet, um der KI zu zeigen, wie ein Mensch aussieht. Hierbei werden nicht nur die verschiedensten Bilder von Menschen verwendet, sondern auch von Tieren oder anderen Dingen wie Lebensmittel. Mit den nichtmenschlichen Gegenständen und Tieren hilft es der KI also zu lernen, wie ein Mensch aussieht und welche Merkmale dieser hat.

8.1.2 Künstliche neuronale Netze

Wie vorhin angesprochen wird bei Deep Learning auch künstliche neuronale Netze (KNN) verwendet. Hierbei wird unser menschliches Gehirn nachgeahmt, um Muster zu erkennen, Texte zu interpretieren und auch Objekte auf Bildern (wie zum Beispiel einen Menschen auf einem Bild) zuzuordnen und zu klassifizieren um welches Objekt es sich handelt. Bei einem KNN gibt es 3 verschiedene Schichten. Diese 3 Schichten sind die Eingangsschicht (Input Layer), die versteckte Schicht (Hidden Layer) und die Ausgangsschicht (Output Layer).

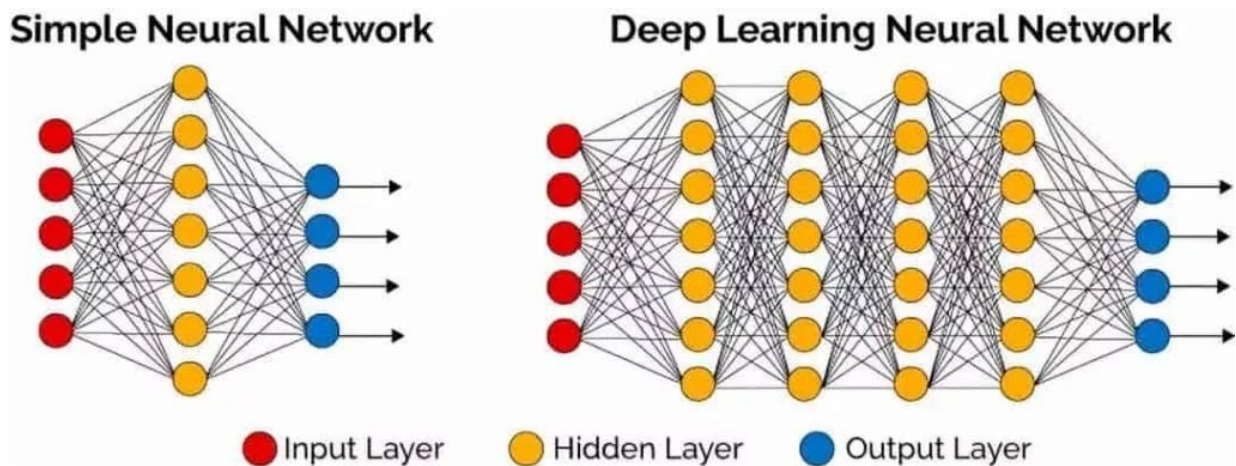


Abbildung 14: künstlich neuronale Netze (links: einfaches KNN; rechts: KNN mit Deep Learning kombiniert) ²⁴

- **Eingangsschicht (Input Layer):**
 - In der Eingangsschicht repräsentiert jeder rote Punkt ein festgelegtes Merkmal und die eingegebenen Daten werden direkt in die versteckte Schicht übergeben (unterteilt in die Merkmale).
 - Bei beispielsweise einem Bild von einer Katze wird jeder Pixel einem Merkmal zugeordnet (desto größer das Bild, desto mehr Merkmale) und der versteckten Schicht übergeben.
- **Versteckte Schicht (Hidden Layer):**
 - In der versteckten Schicht oder auch Zwischenschicht bekannt werden nun die Merkmale von der Eingabeschicht extrahiert und analysiert, um ein Muster zu finden. Da die Daten der Eingabeschicht nicht gleich sind wird das KNN immer besser darin diese Muster zu erkennen und zu verstehen.
 - Wenn man wieder an das Beispiel mit dem Bild zurückdenkt, haben wir von der Eingabeschicht nun unsere Merkmale bekommen. Diese Merkmale werden in der versteckten Schicht also analysiert und es wird ein Muster gefunden.
- **Ausgangsschicht (Output Layer):**

²⁴ Abbildung Simple und Deep Learning Neural Network: <https://datasolut.com/was-ist-deep-learning>

- In der Ausgangsschicht bekommt man nun die analysierten Daten der versteckten Schicht. Das von der versteckten Schicht analysierte Muster wird nun klassifiziert und ausgegeben.
- Bei unserem Bild wird hier also festgestellt, dass das Bild eine Katze darstellt und dies wurde auch entsprechend ausgegeben.

8.2 Machine Learning

Machine Learning ²⁵ ist ebenfalls eine wichtige Anwendung im Bereich der Künstlichen Intelligenz. Es soll das automatische Erkennen von Mustern und Datenzusammenhängen für eine KI vereinfachen ohne das ein Mensch eingreifen muss. Die KI soll sich also mithilfe von Machine Learning das Anlernen von neuem Wissen, wie das Erkennen von Mustern, Identifizieren von Zusammenhängen oder Antrainieren von Algorithmen selbstständig und automatisch und ohne menschliche Hilfe durchführen. Damit die KI all das allein durchführen kann muss derzeit jedoch am Anfang immer noch ein Mensch mithelfen und den Algorithmus des Lernens der KI beibringen. Dies wird mit Big Data Trainingsdaten und Verfahren wie dem Deep Learning gemacht. Desto länger das man mit der KI-Trainingsprozesse, wie das Deep Learning durchführt, desto besser kann die KI später Entscheidungen und Vorhersagen treffen. Das Hauptziel hierbei ist es also immer die KI auf die Unabhängigkeit des Menschen zu trainieren.

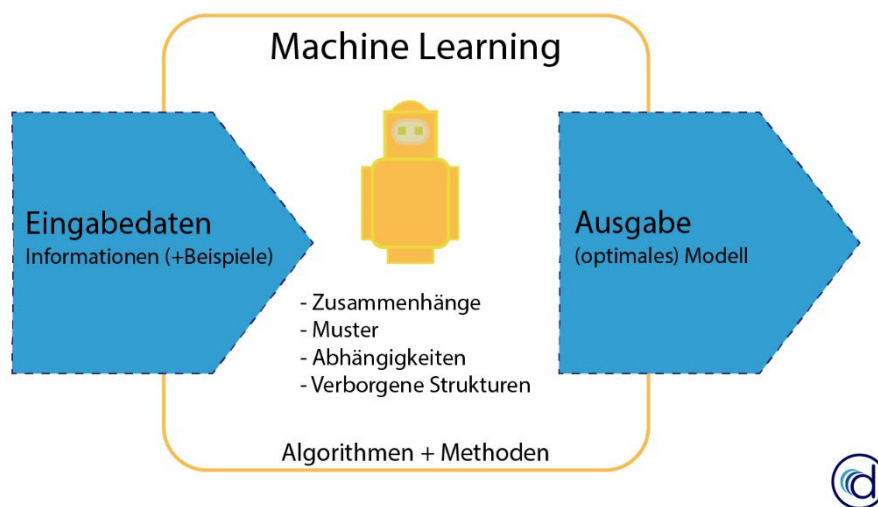


Abbildung 15: Machine Learning ²⁶

²⁵ Machine Learning: https://en.wikipedia.org/wiki/Machine_learning

²⁶ Abbildung Machine Learning: <https://datasolut.com/was-ist-machine-learning/>

8.3 KI im Zusammenhang mit RPA

Wie wir bereits bei den Tools des Öfteren erwähnt haben, wird die KI oft im Zusammenhang mit RPA verwendet, da dies das Automatisieren auch um einiges leichter macht. In der Szene der Prozessautomatisierung wird Künstliche Intelligenz momentan in den folgenden Bereichen eingesetzt:

8.3.1 Treffen von Entscheidungen

KIs werden sehr häufig dazu eingesetzt, um uns kleinere Entscheidungen abzunehmen. Diese Entscheidungen können beispielsweise das Beantworten von E-Mails sein oder die Informationsentnahme eines Dokuments, um diese uns in einer übersichtlicheren Form darzustellen. Zurzeit ist es zwar schon so, dass die KI nicht alles beantworten oder perfekt extrahieren kann, allerdings lässt der technologische Fortschritt nicht auf sich warten und die KI kann langsam aber stetig immer komplexere Fragen beantworten.

8.3.2 Mustererkennung und Vorhersage

Die Künstliche Intelligenz wird natürlich auch in Bereichen der Mustererkennung und Vorhersage eingesetzt. Hierbei kann man der KI eine extrem große Datenmenge geben und die KI erkennt in den vorhandenen Daten, dann ein Muster worauf diese dann auch Vorhersagen treffen kann, was in Zukunft mit großer Wahrscheinlichkeit passieren wird. In der RPA wird diese Fähigkeit vor allem zur Prozessoptimierung eingesetzt, da potenzielle Risiken oder Probleme somit frühzeitig erkannt und umgangen werden können.

8.3.3 Maschinelles Lernen

Wie wir bereits gehört haben, gehört natürlich auch das Maschinelle Lernen zu einer KI. Mit dieser Eigenschaft kann sich eine Prozessautomatisierung stetig verbessern und sich auf immer mehr und schwierigerer Situationen anpassen, was die Prozessautomatisierung, um einen gewaltigen Schritt effizienter und weniger Problemanfällig gestaltet.

In Zukunft wird die Integration der KI natürlich immer weiter zunehmen, was dazu führen wird, dass die Automatisierungen immer intelligenter und effektiver werden. Vor allem wird dies wahrscheinlich einen großen Einfluss auf Firmen haben. Unternehmen erkennen zwar bereits jetzt nach und nach, dass es gut ist seine Prozesse zu automatisieren, die nur unnötig Zeit aufwenden. Doch desto weiter fortgeschritten die Technologie mithilfe der KI wird, wird das Automatisieren von Workflows immer einfacher und sie können auch die bislang nur sehr schwer zu automatisierenden Prozesse automatisieren. Des Weiteren werden Firmen sich wahrscheinlich immer mehr Firmen auf die Prozessautomatisierung spezialisieren und das Automatisieren auch als Dienstleistung anbieten.

8.4 Chat GPT

Der Chatbot "ChatGPT" ist ein Chat Generative Pre-trained Transformer, entwickelt von der Firma OpenAI, die im Dezember 2015 gegründet wurde. Das Ziel der Firma ist die Forschung und Entwicklung von Künstlicher Intelligenz und, wie sie selbst sagen, "safe and beneficial", also sicher und hilfreich.

OpenAI, Inc:

Ursprünglich wurde das Unternehmen als Open-Source-Non-Profit-Organisation geführt und von Spenden finanziert. Als jedoch 2019 finanzielle Probleme entstanden, wurde eine Profit

orientierte Tochterfirma gegründet. Von dieser Tochterfirma aus fließt seitdem finanzielle Unterstützung in Form von überschüssigen Erträgen in das Unternehmen. Die Tochterfirma erwirtschaftet dieses Geld durch das Anbieten eines Abonnements auf ihrer Website ²⁸, mit dem man weitere Funktionen freischaltet. Die jüngeren KI-Modelle waren, wie es der Name der Firma erwarten lässt, noch Open Source, was wiederum bei den neueren Modellen wie GPT 3.5 oder GPT 4 nicht mehr der Fall ist, was der Firma einiges an Kritik eingebracht hat.



27

Abbildung 16: OpenAI Logo

Training:

ChatGPT eine ist ein „Large Language Model“, welches mit zahlreichen Trainingsdaten angereichert wurde. Das Training wurde in drei verschiedenen Phasen abgewickelt. In der ersten Phase dieses Trainings (Pre-Training), wurde mit zahlreichen Büchern, literarischen Textsammlungen, Wikipedia-Einträge und vielen weiteren Webseiten trainiert. Dieses Training diente dazu, dass das System sprachliche Muster in den Texten erkennt. In der zweiten Phase wurde mittels überwachten Lernen, die eigentliche Funktion des Chatbots Trainiert: Auf gestellte Fragen antworten zu generieren. Diese Antworten wurden immer von Testpersonen bewertet. In der letzten Phase wurde mit Hilfe von bestärkendem Lernen durch menschliche Rückkopplung (RLHF) weitere Optimierungen vorgenommen. Dafür wurde ein Reward-Modell verwendet, um die Antworten von ChatGPT qualitativ zu bewerten. Dieses Reward-Modell wurde trainiert in dem es Antworten der KI, in Form einer von Menschen gefertigten Rangliste zu bewerten. Diese Bewertungen wurden verwendet, um das Reward-Modell zu verbessern. Die Weiterentwicklung stoppte jedoch noch nicht, der nächste Schritt war die Analyse sowie eine Beschreibung von Bildern. Diese Funktion ist ab der Version GPT-4 verfügbar. (vgl. Wikipedia, 2024) (vgl. Wikipedia, 2023)

²⁷ OpenAI Logo: https://en.wikipedia.org/wiki/OpenAI#/media/File:OpenAI_Logo.svg

²⁸ Website: <https://chat.openai.com/#pricing>

8.4.1 Chat GPT-API

Mittels der API von OpenAI, kann die KI auch in andere Programme eingebunden werden, in dem alle Fragen an die Künstliche Intelligenz weitergeleitet werden, und man von dieser dann eine Antwort erhält, welche im Programm angezeigt werden kann. Das bietet eine breite Fläche für Anwendungsmöglichkeiten.

Mit dieser Methode können Kundensupports automatisiert werden, indem ChatGPT integriert wird. Dadurch ergibt sich der Vorteil einer sehr schnellen Antwortzeit. Weiters ist die API auch gut für Softwareentwickler nutzbar, die entweder die API in ihre Programme einbauen wollen, oder auch in der Robotic Process Automation, bei der es sehr hilfreich ist, in Randfällen, die nicht von der regulären Automatisierung abgedeckt sind. (vgl. Wikipedia, 2024) (vgl. Wikipedia, 2023)

8.5 Andere Chat Bots

Es gibt jedoch nicht nur ChatGPT, auch wenn dieser der Bekannteste ist. Zurzeit ist Claude 3 mit dem Opus-Modell in den meisten Aufgaben überlegen. Diese Angaben stammen vom 04.03.2024, und da sich in diesem Bereich zurzeit sehr viel ändert, könnte bereits ein besseres Modell vorhanden sein.

	Claude 3 Opus	Claude 3 Sonnet	Claude 3 Haiku	GPT-4	GPT-3.5	Gemini 1.0 Ultra	Gemini 1.0 Pro
Undergraduate level knowledge <i>MMLU</i>	86.8% 5-shot	79.0% 5-shot	75.2% 5-shot	86.4% 5-shot	70.0% 5-shot	83.7% 5-shot	71.8% 5-shot
Graduate level reasoning <i>GPQA, Diamond</i>	50.4% 0-shot CoT	40.4% 0-shot CoT	33.3% 0-shot CoT	35.7% 0-shot CoT	28.1% 0-shot CoT	—	—
Grade school math <i>GSM8K</i>	95.0% 0-shot CoT	92.3% 0-shot CoT	88.9% 0-shot CoT	92.0% 5-shot CoT	57.1% 5-shot	94.4% Maj1@32	86.5% Maj1@32
Math problem-solving <i>MATH</i>	60.1% 0-shot CoT	43.1% 0-shot CoT	38.9% 0-shot CoT	52.9% 4-shot	34.1% 4-shot	53.2% 4-shot	32.6% 4-shot
Multilingual math <i>MGSMT</i>	90.7% 0-shot	83.5% 0-shot	75.1% 0-shot	74.5% 8-shot	—	79.0% 8-shot	63.5% 8-shot
Code <i>HumanEval</i>	84.9% 0-shot	73.0% 0-shot	75.9% 0-shot	67.0% 0-shot	48.1% 0-shot	74.4% 0-shot	67.7% 0-shot
Reasoning over text <i>DROP, F1 score</i>	83.1 3-shot	78.9 3-shot	78.4 3-shot	80.9 3-shot	64.1 3-shot	82.4 Variable shots	74.1 Variable shots
Mixed evaluations <i>BIG-Bench-Hard</i>	86.8% 3-shot CoT	82.9% 3-shot CoT	73.7% 3-shot CoT	83.1% 3-shot CoT	66.6% 3-shot CoT	83.6% 3-shot CoT	75.0% 3-shot CoT
Knowledge Q&A <i>ARC-Challenge</i>	96.4% 25-shot	93.2% 25-shot	89.2% 25-shot	96.3% 25-shot	85.2% 25-shot	—	—
Common Knowledge <i>HellaSwag</i>	95.4% 10-shot	89.0% 10-shot	85.9% 10-shot	95.3% 10-shot	85.5% 10-shot	87.8% 10-shot	84.7% 10-shot

Abbildung 17: Chatbot Vergleich Text ²⁹

²⁹ Chatbot Vergleich Text: <https://www.anthropic.com/news/claude-3-family>

In dieser Grafik werden Mathematische Fähigkeiten, Programmierfähigkeiten und weitere Text basierte Resultate verglichen. Alle dieser Modelle ermöglichen einen Zugriff über eine eigene API, wodurch alle im Kapitel Chat GPT-API genannten Anwendungsmöglichkeiten möglich sind.

	Claude 3 Opus	Claude 3 Sonnet	Claude 3 Haiku	GPT-4V	Gemini 1.0 Ultra	Gemini 1.0 Pro
Math & reasoning <i>MMMU (val)</i>	59.4%	53.1%	50.2%	56.8%	59.4%	47.9%
Document visual Q&A <i>ANLS score, test</i>	89.3%	89.5%	88.8%	88.4%	90.9%	88.1%
Math <i>MathVista (testmini)</i>	50.5% CoT	47.9% CoT	46.4% CoT	49.9%	53.0%	45.2%
Science diagrams <i>AI2D, test</i>	88.1%	88.7%	86.7%	78.2%	79.5%	73.9%
Chart Q&A <i>Relaxed accuracy (test)</i>	80.8% 0-shot CoT	81.1% 0-shot CoT	81.7% 0-shot CoT	78.5% 4-shot CoT	80.8%	74.1%

Abbildung 18: Chatbot Vergleich Bild ³⁰

Weites wurden auch noch die Bilderkennungsfähigkeiten getestet, wobei in diesem Test sich das Spitzenmodell der Firma Anthropic PBC, Opus, auch nicht schlecht schlägt, wobei auch die Gemini 1.0 Ultra-Version von Google beachtliche Ergebnisse liefert.

³⁰ Chatbot Vergleich Bild: <https://www.anthropic.com/news/claude-3-family>

9 BENUTZUNG

9.1 Synchronisation zwischen SharePoint und Jira

Grundsätzlich gibt es kein Frontend oder eine Benutzeroberfläche für das Ausführen der Synchronisation. Die Automatisierung erfolgt in UiPath Studio und wird daher nur durch einen Klick auf einen Button ausgelöst.

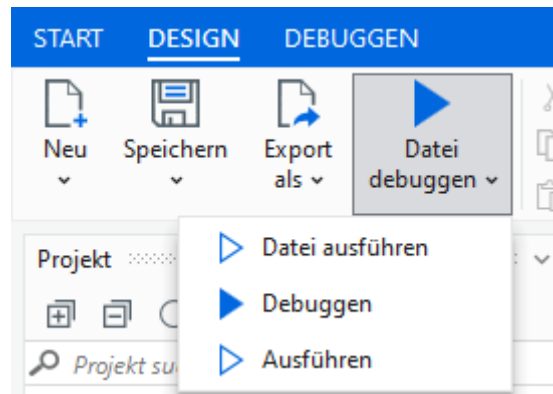


Abbildung 19: UiPath - Datei Ausführen

In UiPath Studio kann man das Projekt entweder ausführen oder Debuggen.

Jedoch durch den UiPath Orchestrator, welcher dazu dient, um alle Prozessautomatisierungen zu verwalten, kann das Projekt auch gestartet werden. Dazu muss man auf die UiPath Studio Website und dann auf den Punkt Orchestrator.

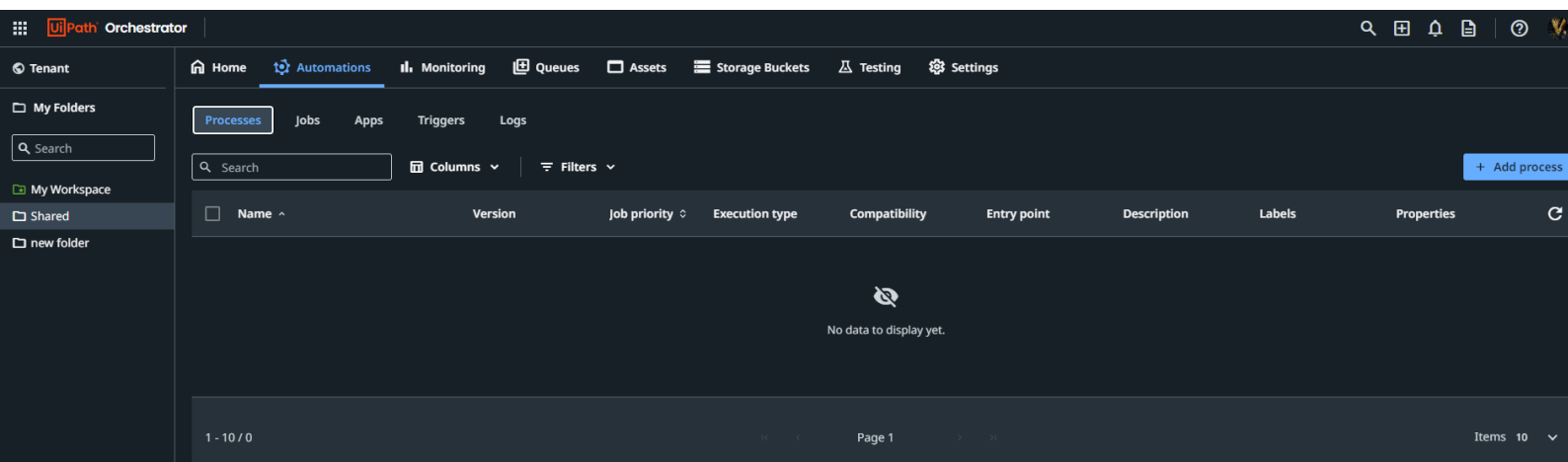


Abbildung 20: UiPath Orchestrator

Hierbei gibt es verschiedene Folder, wo auch verschiedene Automations vorhanden sein können, die Automatisierungen können dadurch übersichtlicher und besser eingesehen werden und auf einmal gestartet werden.

9.2 SimpleQuery

Die Nutzung des Programmes ist recht einfach. Zuerst den Start Button drücken und danach kann auch schon begonnen werden mit der Zusammenstellung der Query.

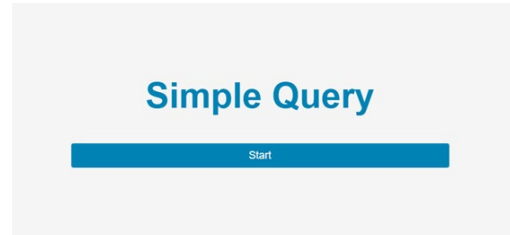


Abbildung 21: Benutzung Simple Query Start Site

Abbildung 25: Benutzung Simple Query Choose Data Site

Dabei wird einfach der gewünschte Server ausgewählt sowie die zu verwendenden Komponenten, Objekt Typen, Konditionen und die gewünschten Ergebnisse.

Nachdem die Query fertig konfiguriert ist, wird diese mit dem Submit-Button bestätigt und es werden die Daten geladen.

Print	Raw	WSDL	Software Component Version	Key
	R		Z DK0, 1.0 of voestalpine.com	External Definition ED_RoiSachnummern_DKI I

Abbildung 23: Benutzung Simple Query DataSite

Diese Daten können danach noch verglichen werden und es werden alle Daten, die nicht doppelt vorkommen, also jene die nicht die in der selben Version sind oder die auf einem Server gar nicht vorhanden sind.

Print	Raw	WSDL	Software Component Version	Key
-------	-----	------	----------------------------	-----

Abbildung 24: Benutzung Simple Query Compare Data Site

10 PROJEKTSTRUKTUR

10.1 Zusammenhang Jira und SharePoint

Die Voestalpine group-IT hat zwei Abteilungen namens PROMES und PPS. Hierbei verwendet die PROMES-Abteilung das interne SharePoint der Voestalpine und die PPS-Abteilung verwendet währenddessen Jira. Die PROMES-Abteilung hat ebenfalls Mitarbeiter, die an PPS arbeiten. Dies führt zu dem Problem, dass Mitarbeiter in den jeweiligen Abteilungen unabhängig voneinander arbeiten. Trotz des Eintragens in der jeweiligen Plattform, die sie nutzen, kann ein Mitarbeiter der jeweils anderen Abteilung die Änderungen eines Tasks weder einsehen noch davon wissen, was und ob die jeweils andere Abteilung etwas verändert hat. Aus diesem Grund haben die Mitarbeiter die Tasks im SharePoint und Jira zwar bis lang so gut und oft es geht synchronisiert, allerdings ist dauert diese Synchronisierung viel zu lange, weshalb dieser Prozess automatisiert werden musste.

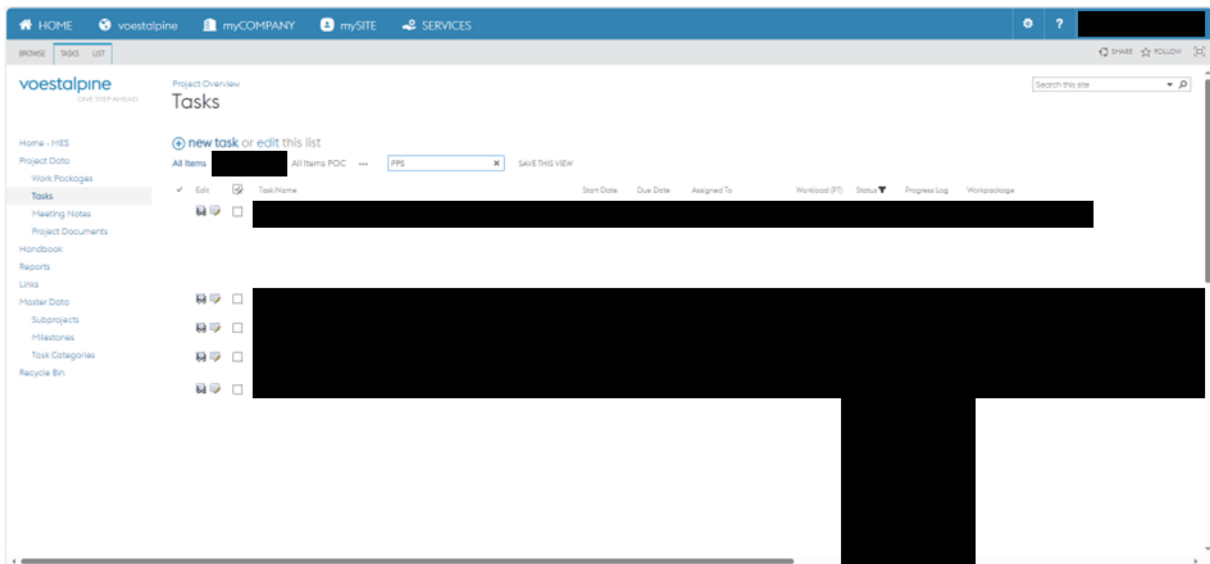


Abbildung 26: SharePoint (wegen Datenschutzrichtlinien geschwärzt)

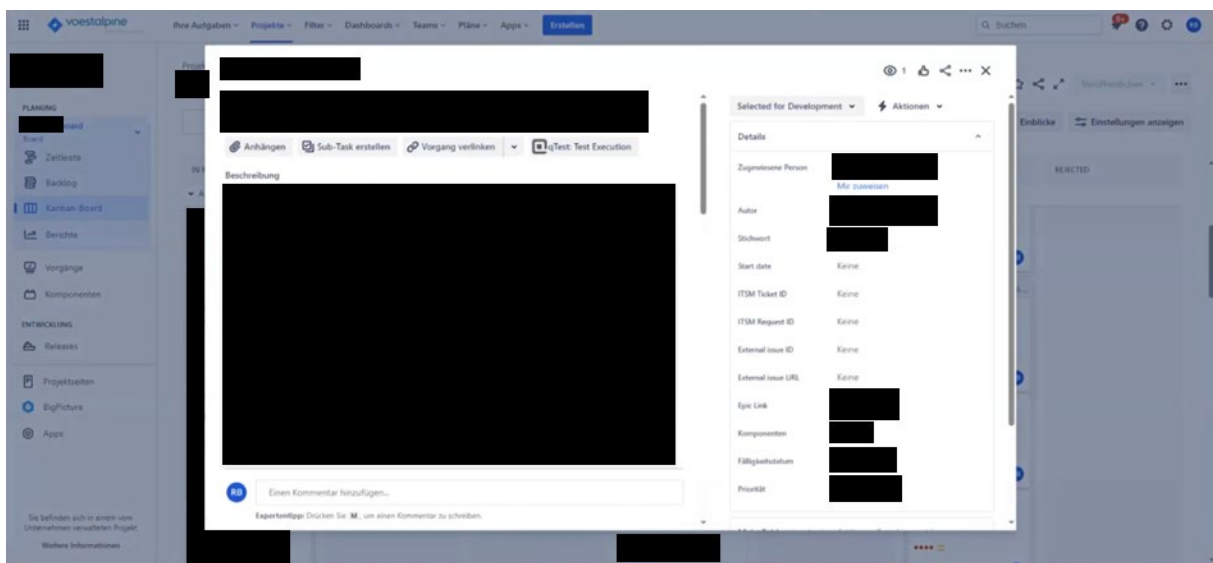


Abbildung 27: Jira (wegen Datenschutzrichtlinien geschwärzt)

10.1.1 Felder Problem

In Jira und SharePoint gibt es des Weiteren auch das Problem, dass die relevanten und zu synchronisierenden Felder in der jeweils anderen Plattform andere Namen hatten. In der nachfolgenden Tabelle werden deshalb die Namen der Felder gegenübergestellt und in die zusätzlichen Felder mitsamt dem Inhalt beschrieben:

Feldname im SharePoint	Feldname im Jira
Task Name	Zusammenfassung
WorkPackage	Stichwort
Start Date	Start Date
Due Date	Fälligkeitsdatum
Status	Status
Assigned To	Zugewiesene Person
Workload	Ursprüngliche Schätzung
Description	Beschreibung
Progress Log	Kommentar

Tabelle 8: Feldernamen Sharepoint und Jira

Zusätzlich gibt es noch die Felder *Property-ID5* in dem ein Link zum betreffenden Jira Task abgespeichert werden soll. Im Jira wiederum soll die SharePoint Task ID im Feld *External issue ID* und die SharePoint Task URL im Feld *External issue URL* abgespeichert werden.

11 IMPLEMENTIERUNG

11.1 Synchronisation zwischen SharePoint und Jira

11.1.1 Technologieauswahl

Den Teammitgliedern wurde die Technologieauswahl freigestellt. Somit konnten sich diese aussuchen, ob mit einer regulären Programmiersprache, wie Java implementiert, oder eine UI basierte Automatisierungssoftware, wie UiPath verwendet wird. Die Entscheidung ist auf UiPath gefallen, da eines der Teammitglieder bereits praktische Erfahrungen mit den UiPath Programmen während vorherigen Praktika gesammelt hat. Dieses Teammitglied hat den anderen vorgeschlagen, dass UiPath in diesem Fall verwendet werden soll, da es einiges erleichtern würde. Als die Idee dem Auftraggeber vorgeschlagen wurde, hat dieser die Idee abgesegnet. Danach haben die Teammitglieder zusammengesehen, dass sich alle Mitglieder auskennen und Fragezeichen und Unklarheiten seitens der UiPath Programme geklärt werden. Während der Arbeit mit UiPath wurde zu dem auch bei Fehlern oder Problemen zusammengesehen.

Zusätzlich zu UiPath hat das Projektteam Postman verwendet, um die HTTP Requests von Jira auszuarbeiten. Dies war wichtig damit die HTTP Request korrekt ausgearbeitet werden können, um zu verhindern das Probleme innerhalb des Programms seitens der HTTP Requests stammen können. Dadurch ist die Fehleranfälligkeit erheblich gesunken und hat dem Team die Möglichkeit gegeben, die Requests ausführlich zu testen und die Rückmeldungen der HTTP Requests zu analysieren und zu bewerten. Nachdem sichergestellt wurde, dass diese Requests fehlerfrei und perfekt funktionieren, konnten sie im eigentlichen Programm eingesetzt werden.

11.1.2 UiPath Installation

Um UiPath Studio kostenlos zu verwenden, kann man sich mithilfe von <https://cloud.uipath.com/testoyggnavi/portal/home> die UiPath Studio Community Edition herunterladen. Den Downloadbutton kann man oben auf der rechten Seite finden:

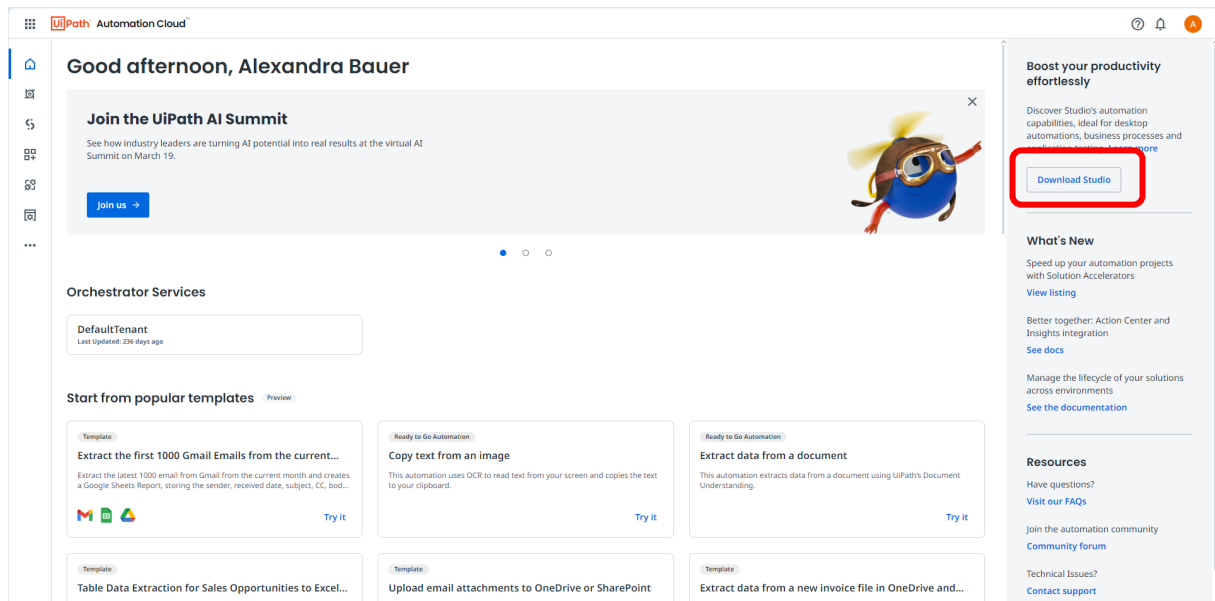


Abbildung 28: UiPath Studio Download

Wenn man nun die gerade heruntergeladene .exe Datei aufmacht macht sich folgendes Installationsfenster auf:

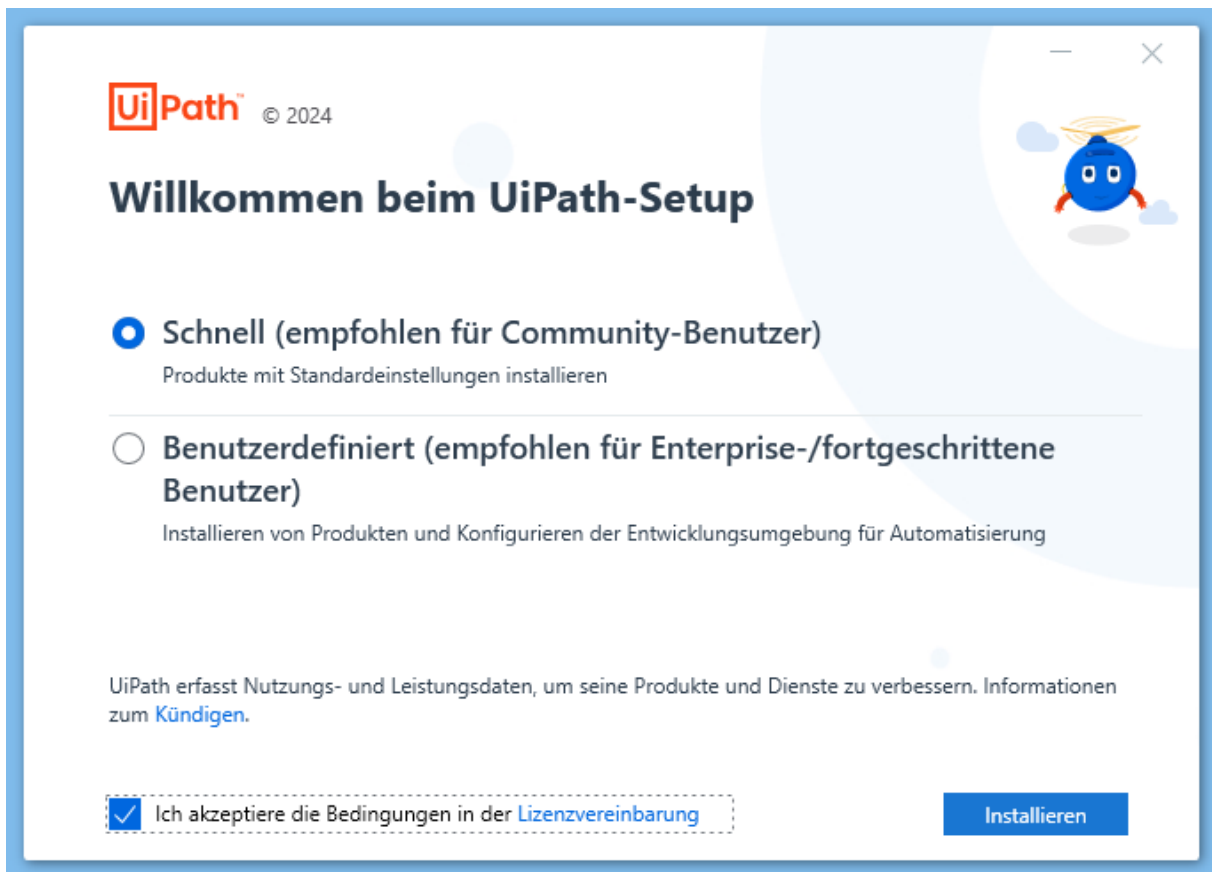


Abbildung 29: UiPath Setup

Hierbei kann man (vor allem wenn man neu in der Prozessautomatisierungsszene ist) einfach „Schnell“ auswählen, damit UiPath alle notwendigen Pakete installiert. Bei der Auswahl „Benutzerdefiniert“ wird man, wenn man auf „Installieren“ drückt auf eine weitere Installationsseite geleitet, wo man alle benötigten Pakete spezifisch auswählen kann, die UiPath mit installieren soll. Wenn UiPath fertig installiert hat, muss man sich schließlich nur mehr anmelden. Die Anmeldung wird mit cloud.uipath.com durchgeführt.

Nachdem man das Setup abgeschlossen hat, kann man zwischen zwei Profilen wählen. UiPath Studio ist in der Diplomarbeit verwendet worden, da es sich besser für die Implementierung des Prozesses erwiesen hat. Aktivitäten innerhalb eines Prozesses kann man mithilfe von UiPath Studio deutlich besser konfigurieren als in UiPath StudioX, da UiPath StudioX für Menschen ohne Programmierkenntnisse vorgesehen ist und deswegen eine vereinfachtere Version der UiPath Anwendung ist.

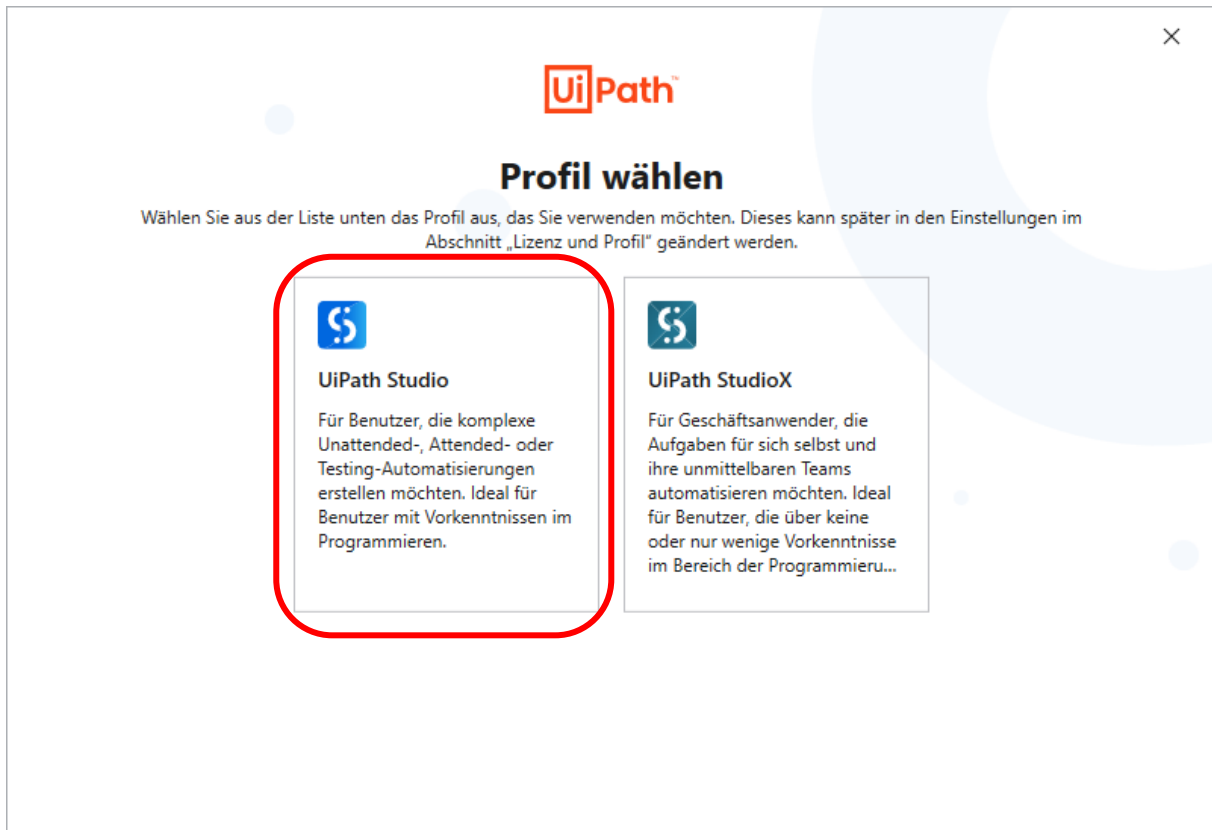


Abbildung 30: UiPath Studio & StudioX

11.1.2.1 *Browser Extensions*

Im Programm werden verschiedene Browser Aktivitäten verwendet. Um UiPath die Möglichkeit zu geben beispielsweise Daten aus dem Browser zu extrahieren, verwendet man eine Browser Extension. In UiPath muss man hierfür unter Tools/UiPath Extension gehen und die jeweils gewollte Browser Extension installieren (in diesem Fall wurde die Browser Extension für Edge installiert). Hierfür muss der jeweilige Browser geschlossen werden.

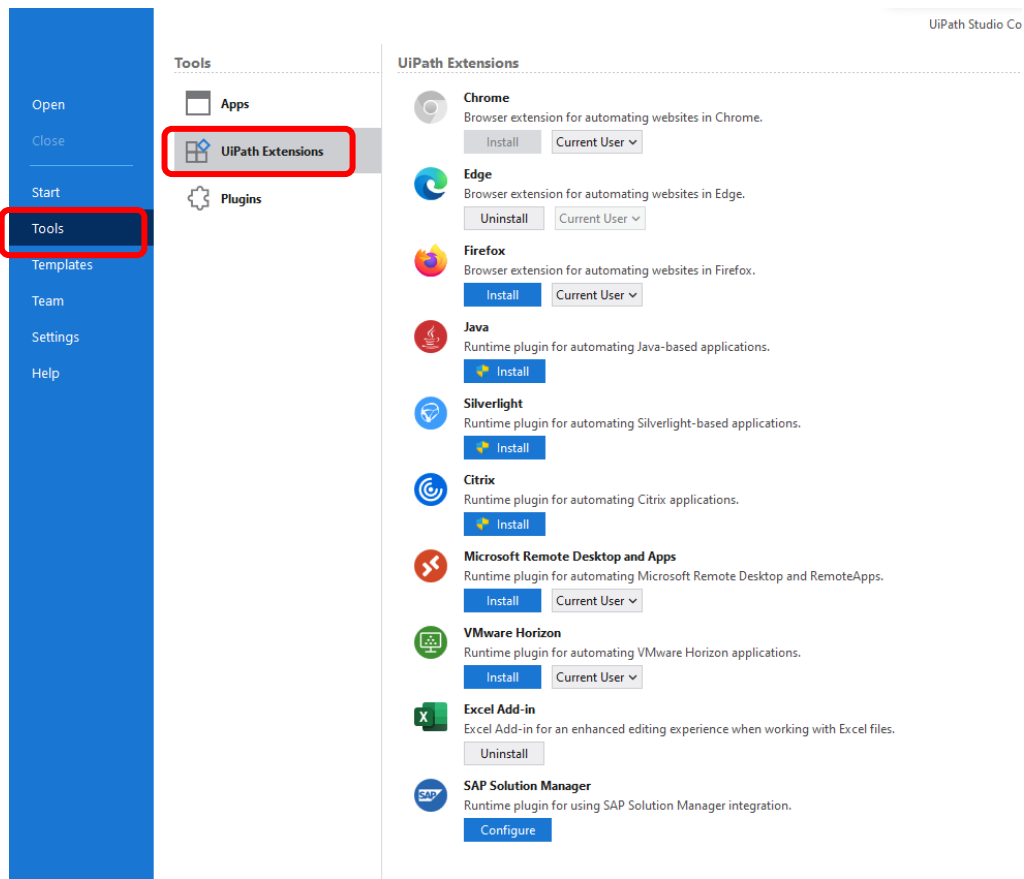


Abbildung 31: UiPath Extensions

Um die Browser Extension auch auf Seiten des Browsers zu installieren, muss man dies in den Erweiterungen tun. Für jeden Browser gibt es hierbei einen anderen Link.

- Edge: `edge://extensions`
- Firefox: `about:addons`
- Brave: `brave://extensions`

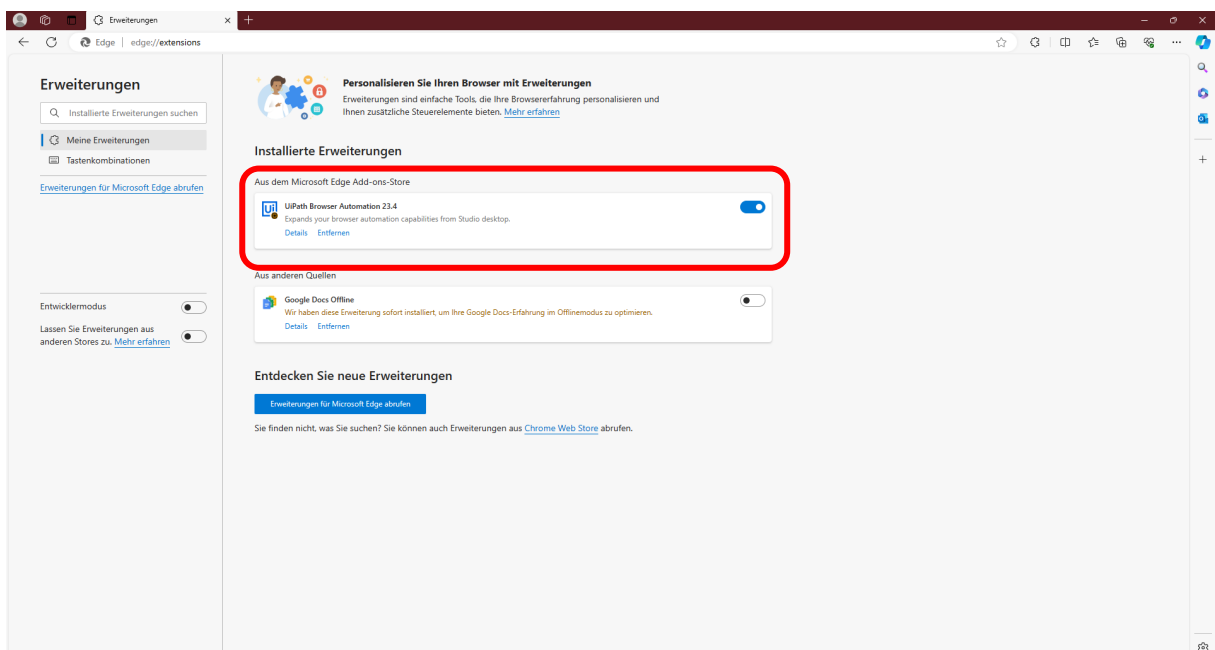


Abbildung 32: UiPath Extension in Edge

11.1.3 Schnittstelle Jira

Für den Zugriff auf Jira haben wir mit der öffentlichen API von Atlassian gearbeitet. Die folgenden HTTP Requests haben wir im Zuge von unserem Programm verwendet. Um diese Requests durchführen zu können, wurde uns ein User mit Passwort gegeben, der nur für diese HTTP Requests verwendet wird.

Alle PPS-Projekte

GET `https://voestalpine.atlassian.net/rest/api/3/search?jql=project=PPS`

Diese HTTP GET Request haben wir verwendet, um alle PPS-Projekte von Jira zu bekommen. Wir haben den Request in unserem Programm eingesetzt, um den Status mit den PPS-Tasks im SharePoint zu vergleichen (Also ob die Tasks zum Beispiel noch *In Progress* waren oder bereits auf *Completed* gesetzt gehören).

Ein bestimmter PPS-Task

GET `https://voestalpine.atlassian.net/rest/api/3/issue/<taskID>`

Um nur einen bestimmten PPS-Task von Jira zu bekommen, haben wir diesen HTTP GET Request verwendet. Statt `<taskID>` setzt man dann die ID des Tasks ein, dessen Informationen man abrufen will. Dieser http Request wird bei uns tatsächlich häufiger eingesetzt. Er kommt bei uns deshalb bei folgenden Aktivitäten vor:

- Wenn wir die External Issue ID beziehungsweise uns die External Issue URL holen.
- Wenn wir überprüfen, ob der in SharePoint angelegte PPS-Task überhaupt existiert.

Erstellen eines PPS-Tasks

POST `https://voestalpine.atlassian.net/rest/api/3/issue`

Body (JSON):

```
{
  "fields": {
    "project": {
      "id": "14984",
      "key": "PPS"
    },
    "issuetype": {
      "name": "Task"
    },
    "customfield_12205": "<external issue ID>",
    "customfield_12200": "<external issue URL>",
    "summary": "<Task Name>",
    "duedate": "<Fälligkeitsdatum>",
    "assignee": {
      "accountId": "<assignee account ID>"
    },
  },
}
```

```

    "labels": [
      "<label1>",
      "<label2>"
    ],
    "description": [
      {
        "set": {
          "version": 1,
          "type": "doc",
          "content": [
            {
              "type": "paragraph",
              "content": [
                {
                  "type": "text",
                  "text": "<description>"
                }
              ]
            }
          ]
        }
      ]
    ]
  }
}

```

Falls ein PPS-Task in Jira gar nicht existiert, verwenden wir diesen HTTP POST Request, um ihn zu erstellen. Für diesen Request ist auch ein JSON-Body nötig der mitgegeben wird. Dieser Body beinhaltet bereits die gesammelten Daten von SharePoint und werden somit vom SharePoint in Jira übertragen (Für die jeweiligen Platzhalter wird natürlich der jeweils gewollte Wert eingesetzt.).

Standardänderungen

PUT <https://voestalpine.atlassian.net/rest/api/3/issue/<taskID>>

Body:

```

{
  "fields": {
    "summary": "<Task Name>",
    "duedate": "<Fälligkeitsdatum>",
    "assignee": {
      "accountId": "<assignee account ID>"
    },
    "labels": []
  },
  "update": {
    "description": [
      {
        "set": {

```

```

    "version": 1,
    "type": "doc",
    "content": [
      {
        "type": "paragraph",
        "content": [
          {
            "type": "text",
            "text": "<description>"
          }
        ]
      }
    ]
  }
}

```

Mit diesem HTTP PUT Request werden die Standardänderungen eines PPS-Tasks durchgeführt. Wie man an der URL entnehmen kann, sieht diese zwar gleich aus und wird auch gleich gehandhabt, wie die HTTP GET REQUEST URL, wenn man einen bestimmten PPS-Task holt, allerdings hat dieser Request einen Body, um die verschiedenen Änderungen durchführen zu können (Diese Standardänderungen eines PPS-Task im Jira wird natürlich nur dann durchgeführt, falls diese nicht schon beim Erstellen durchgeführt wurden.).

Alle Jira User

GET <https://voestalpine.atlassian.net/rest/api/3/users/search>

Um die ID des Zuständigen (Assignee) zu bekommen wird diesen HTTP GET Request verwendet, um alle Jira User anzuzeigen und die gewollte ID des gesuchten Zuständigen (assigneeID) zu bekommen.

Mögliche Übergänge eines PPS-Tasks (Status)

GET <https://voestalpine.atlassian.net/rest/api/3/issue/<taskID>/transitions>

Wenn man den Status auf etwas anderes ändern möchte, braucht man die ID der Transition (Übergangs ID). Um diese ID zu erhalten, nutzt man diesen HTTP GET Request und sucht somit nach dem gewollten Status. Mit der ID kann man den HTTP POST Request ausführen, um den Status in Jira zu ändern.

Status des PPS-Tasks ändern

POST <https://voestalpine.atlassian.net/rest/api/3/issue/<taskID>/transitions>

Body:

```
{
  "transition": {
    "id": "<Transition ID>"
  }
}
```

Wie bereits bei den Transitions erklärt, kann man nun mit der Transition ID, die man sich von den möglichen Transitions herausgesucht hat. Dies macht man mit genau diesem HTTP POST Request, indem man beim JSON-Body den Platzhalter für die Transition mit der richtigen ID ersetzt.

Erstellen eines neuen Kommentars

POST <https://voestalpine.atlassian.net/rest/api/3/issue/<taskID>/comment>

Body:

```
{
  "body": {
    "content": [
      {
        "content": [
          {
            "text": "<comment1>",
            "type": "text"
          }
        ],
        "type": "paragraph"
      },
      {
        "content": [
          {
            "text": "<comment2>",
            "type": "text"
          }
        ],
        "type": "paragraph"
      }
    ],
    "type": "doc",
    "version": 1
  }
}
```

Mit diesem HTTP POST Request kann man nun einen neuen Kommentar für einen PPS-Task in Jira erstellen. Beim JSON-Body muss man nur die jeweiligen Platzhalter austauschen. Außerdem kann man beliebig viele Kommentare übergeben, da man die Kommentare mithilfe eines Arrays übergibt und muss somit auch nicht denselben Request mehrere Male ausführen, wenn man mehr als einen Kommentar hat.

11.1.4 Programmbeschreibung

In UiPath Studio gibt es Workflow Dateien, die man zusätzlich zum Haupteintrittspunkt des Programms erstellen und dann mithilfe von sogenannten „Invoke Workflow File“-Aktivitäten aufrufen und natürlich auch ausführen kann. Diese Workflow Dateien kann man also in der herkömmlichen Programmierwelt mit Klassen vergleichen.

Für diese Automatisierung haben wir sechs verschiedenen Workflow Dateien erstellt. Der Workflow für den Programmstart ist hauptsächlich für die Datenbeschaffung in SharePoint zuständig. Natürlich führt dieser Workflow auch alle anderen Workflow Dateien aus, um Daten von Jira zu holen aber auch diese umzuändern. Folgende Workflow Dateien haben wir also eingebaut:

- Main.xaml
- CheckJiraTaskExistence.xaml
- GetAssigneeAccountID.xaml
- DefaultJiraChanges.xaml
- CreateNewComment.xaml
- ChangeStatus.xaml

SharePoint

Die Main.xaml markiert den Programmstart. Wie vorhin bereits gesagt, werden hier die Daten vom SharePoint geholt und auch die anderen Workflow Dateien aufgerufen.

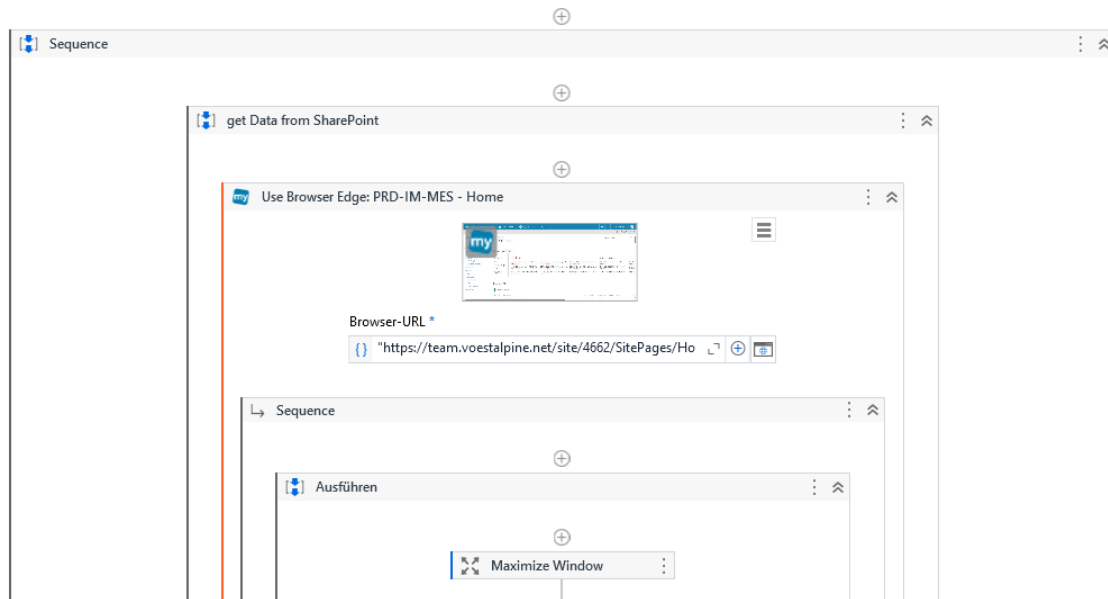


Abbildung 33: UiPath – SharePoint – User Browser

Mithilfe einer „Use Browser“-Aktivität verwenden wir den gewünschten Browser und geben der Aktivität eine Browser-URL, damit UiPath weiß welche Webseite verwendet werden soll. In diesem Fall ist der gewünschte Browser Edge, da dieser Browser standartgemäß von der Voestalpine verwendet wird. Mithilfe der „Maximize Window“-Aktivität stellen wir sicher, dass das Fenster für den Roboter sichtbar ist und vergrößert wird damit dieser die gesuchten UI-Elemente finden kann.

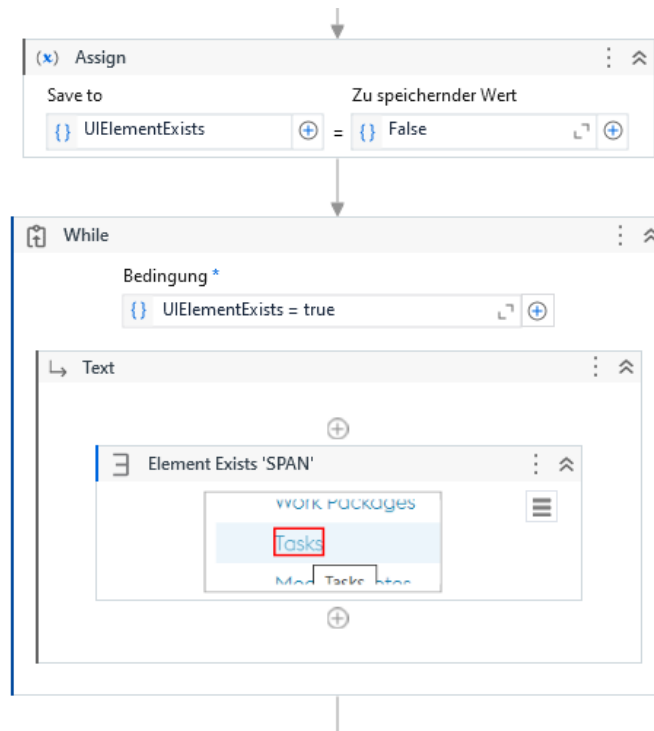


Abbildung 34: UiPath – SharePoint - UiElementExists

Dieser Part ermöglicht, dass wenn die Webseite zwar geladen worden ist aber die UI-Elemente noch nicht vollständig angezeigt werden, solange gewartet wird, bis dieses Element existiert. Dabei wird eine Variable „UIElementExists“ erstellt und als *false* gespeichert. In der *while-Schleife*, wird die Variable dann auf *true* gesetzt, wenn der Button „Tasks“ vorhanden ist. Wenn „UIElementExists“ auf *true* gesetzt wurde, wird die *while-Schleife* abgebrochen. Das „Element Exists SPAN“-Element, setzt die Variable UiElementExists automatisch auf *true*, wenn der Button vorhanden ist.

Eigenschaften von „Element Exists SPAN“-Element:

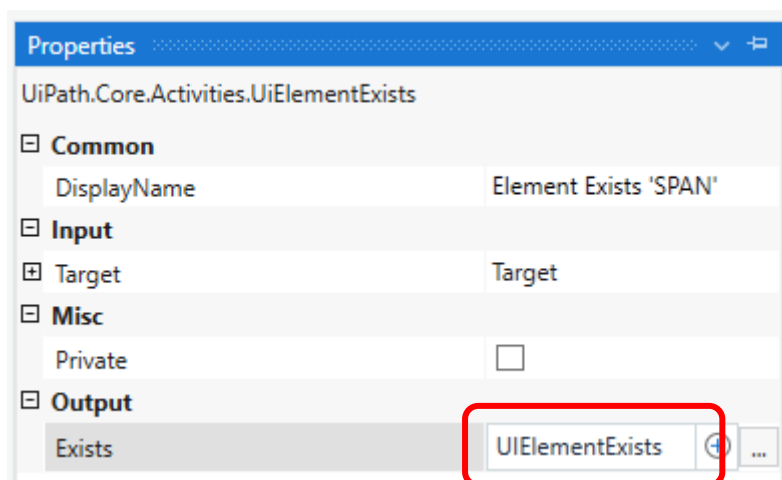


Abbildung 35: UiPath – SharePoint - Element Exists Eigenschaften

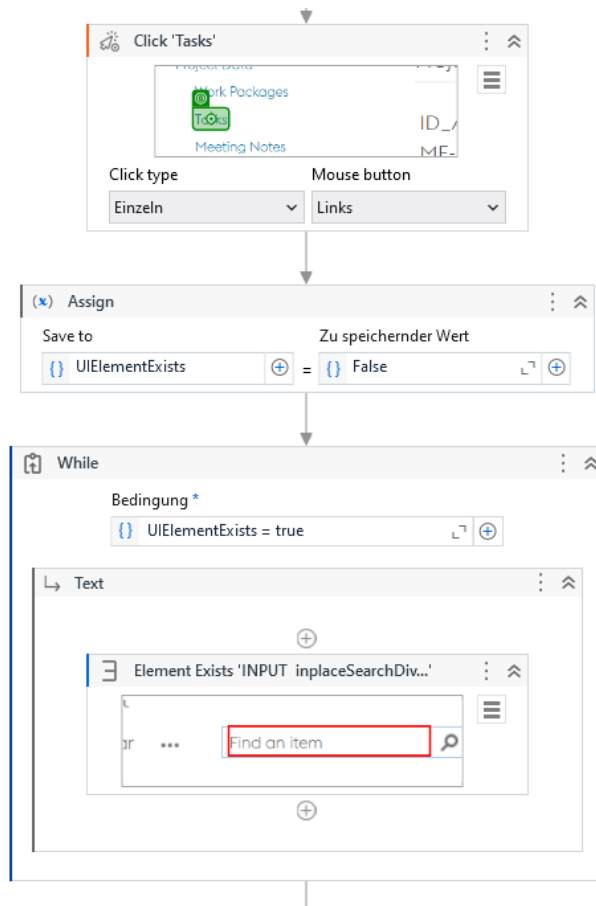


Abbildung 36: UiPath – SharePoint – Nächste Seite

Als nächstes wird auf den Button „Tasks“ mit einem „Click“-Element gedrückt, um auf die nächste Seite zu gelangen. Dabei wird der Klick Prozess durch eine Aufnahme, dem RPA-Bot bereitgestellt. Mit dem „INPUT“-Element können Aufgaben gesucht werden, jedoch muss wieder mit einer *while-Schleife*, auf das Vorhandensein der Suchleiste geprüft werden.

Mit dem „Type Into“-Element wird in die Suchleiste „PPS“ eingegeben. Das „[k(Enter)]“ ist hierbei da, um Enter auszuführen.



Abbildung 37: UiPath – SharePoint – Type Into (PPS)

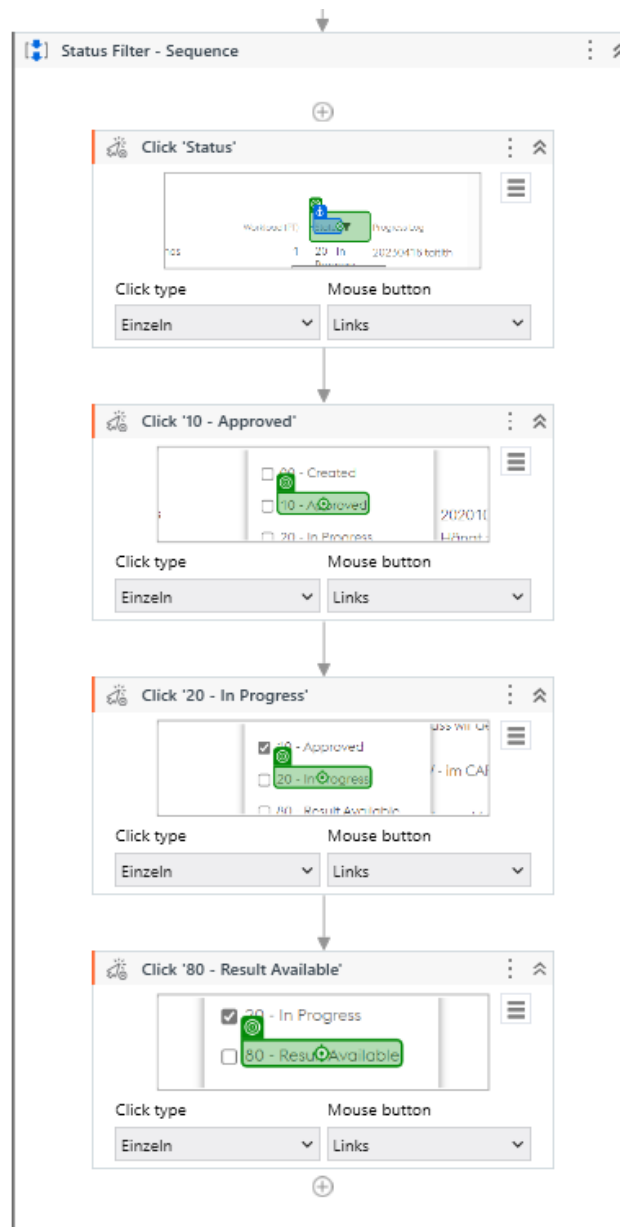


Abbildung 38: UiPath – SharePoint – Filter Status

Hier wird auf die jeweiligen Status gefiltert, welche von der Voestalpine gewünscht worden sind.

Status:

- Approved
- In Progress
- Available

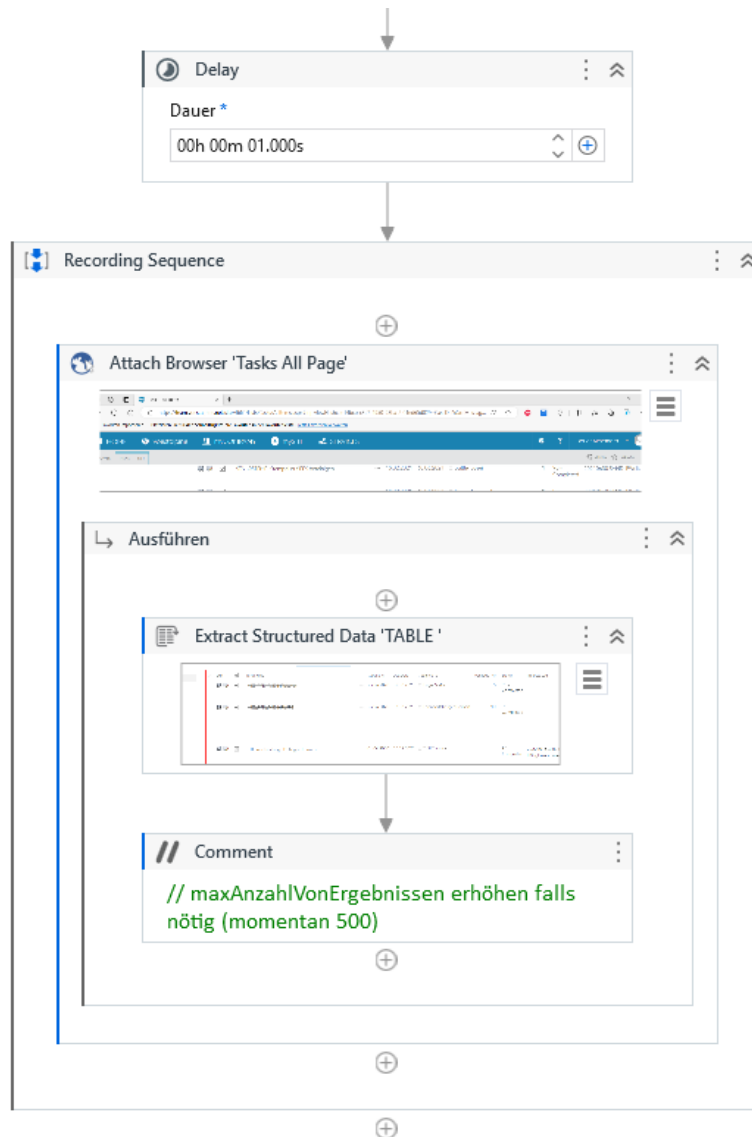


Abbildung 39: UiPath – SharePoint – Extract Structured Data Table

Danach wird eine Sekunde gewartet, bis sich die Daten anhand von dem Filter neu geladen haben. Dieser Schritt ist zur Vermeidung von Fehlern, falls eine Aktualisierung noch nicht stattgefunden hat.

Mit dem Element „Extract Structured Data ‘Table‘“ wird die Tabelle angeklickt, welche die Daten enthält, die aus der Website extrahiert werden sollen. Die Daten werden in eine Variable „ExtractDataTable“ gespeichert. Dabei sind die Elemente „Recording Sequence“ und „Attach Browser“ automatisch erzeugt worden. Diese geben an, dass eine Aufnahme in der Website gestartet wird, in der man sich befindet

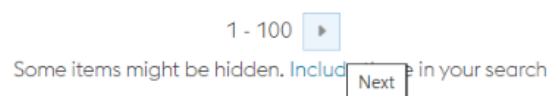


Abbildung 40: UiPath – SharePoint – next-Button

Durch das Klicken auf den „Next“ Button, wird dem Element „Attach Browser“ gesagt, dass die gleichen Aktionen auch auf der nächsten Seite der Website durchgeführt werden sollen.

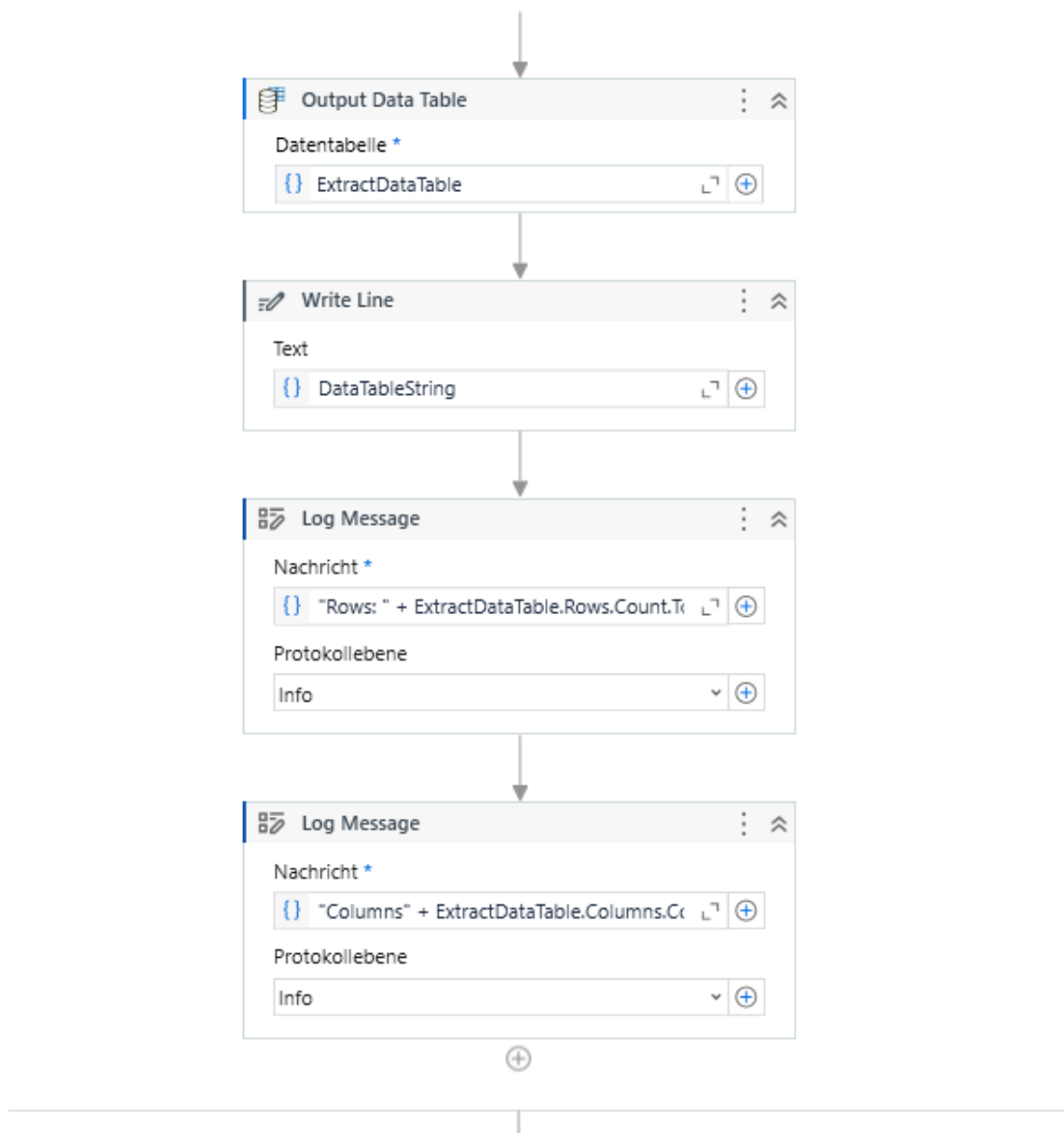


Abbildung 41: UiPath – SharePoint – Output Data Table

Die Daten, welche in der Variable „ExtractDataTable“ sind, werden durch das Element „Output Data Table“ in einen String „DataTableString“ konvertiert und mit dem Element „Write Line“ in der Konsole und dem Output Panel ausgegeben.

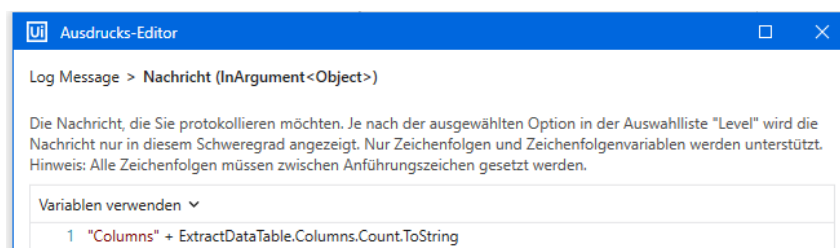


Abbildung 42: UiPath – SharePoint – Log Message

In den zwei „Log Message“ Elementen, werden die Anzahl der Reihen und Zeilen in einer Log Nachricht angezeigt.

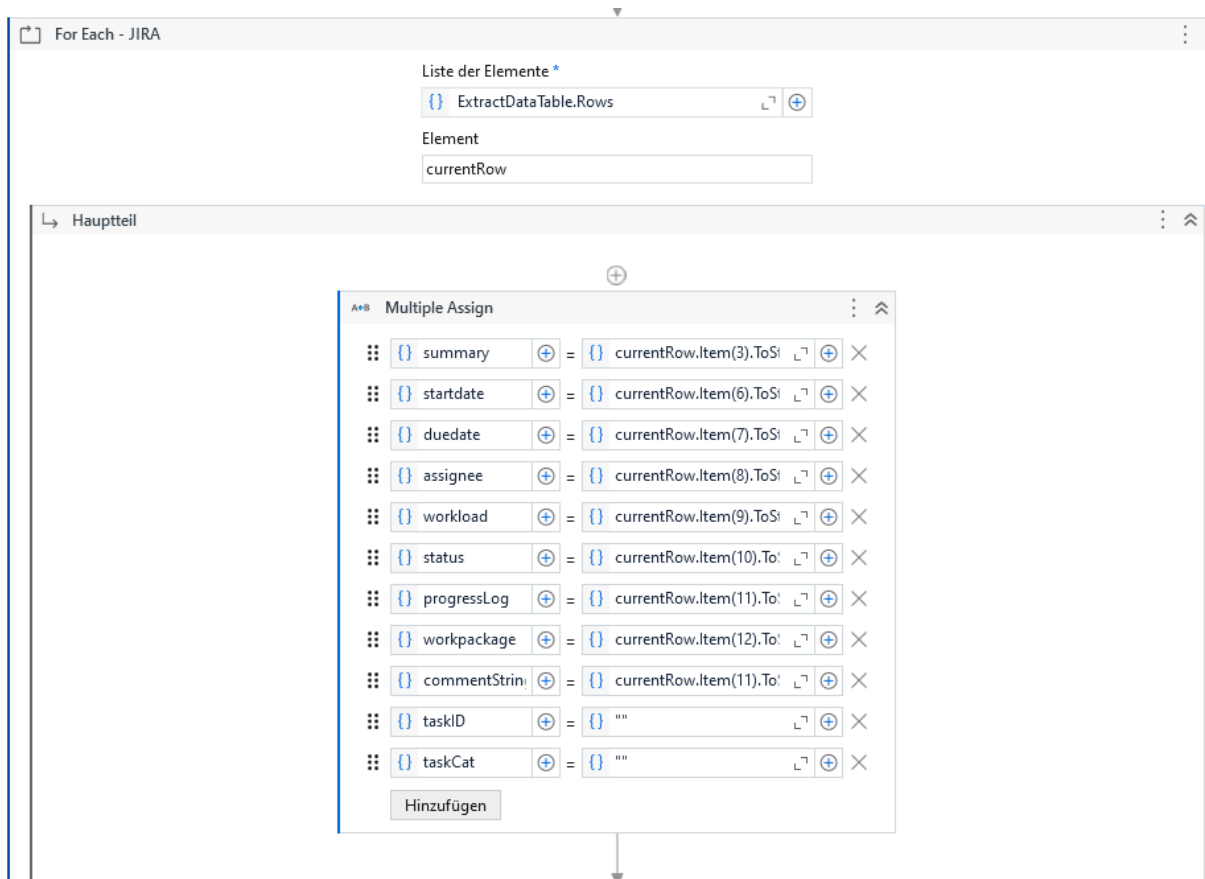


Abbildung 43: UiPath – SharePoint – Multiple Assign

Der eigentliche Hauptteil fängt hier an, dieser besteht aus dem SharePoint und dem Jira Teil. Die Datentabelle, welche zuvor aus der Website extrahiert worden ist, muss einzeln in Variablen gespeichert werden, um diese in Jira zu aktualisieren. Dabei wird mit einer „For- Each“ Aktivität dafür gesorgt, dass jede Zeile nach der Reihe behandelt wird. Dies kann man anhand des festgelegten Worts „currentRow“ feststellen.

Im Hauptteil der Sequenz werden zunächst mehrere Zuweisungen mit dem Element „Multiple Assign“ durchgeführt. Diese Zuweisungen speichern die Daten aus der Tabelle in manuell erstellte Variablen.

Variablen:

- summary

- startdate
- duedate
- assignee
- workload
- status
- progressLog
- workpackage
- commentString
- taskID
- taskCat

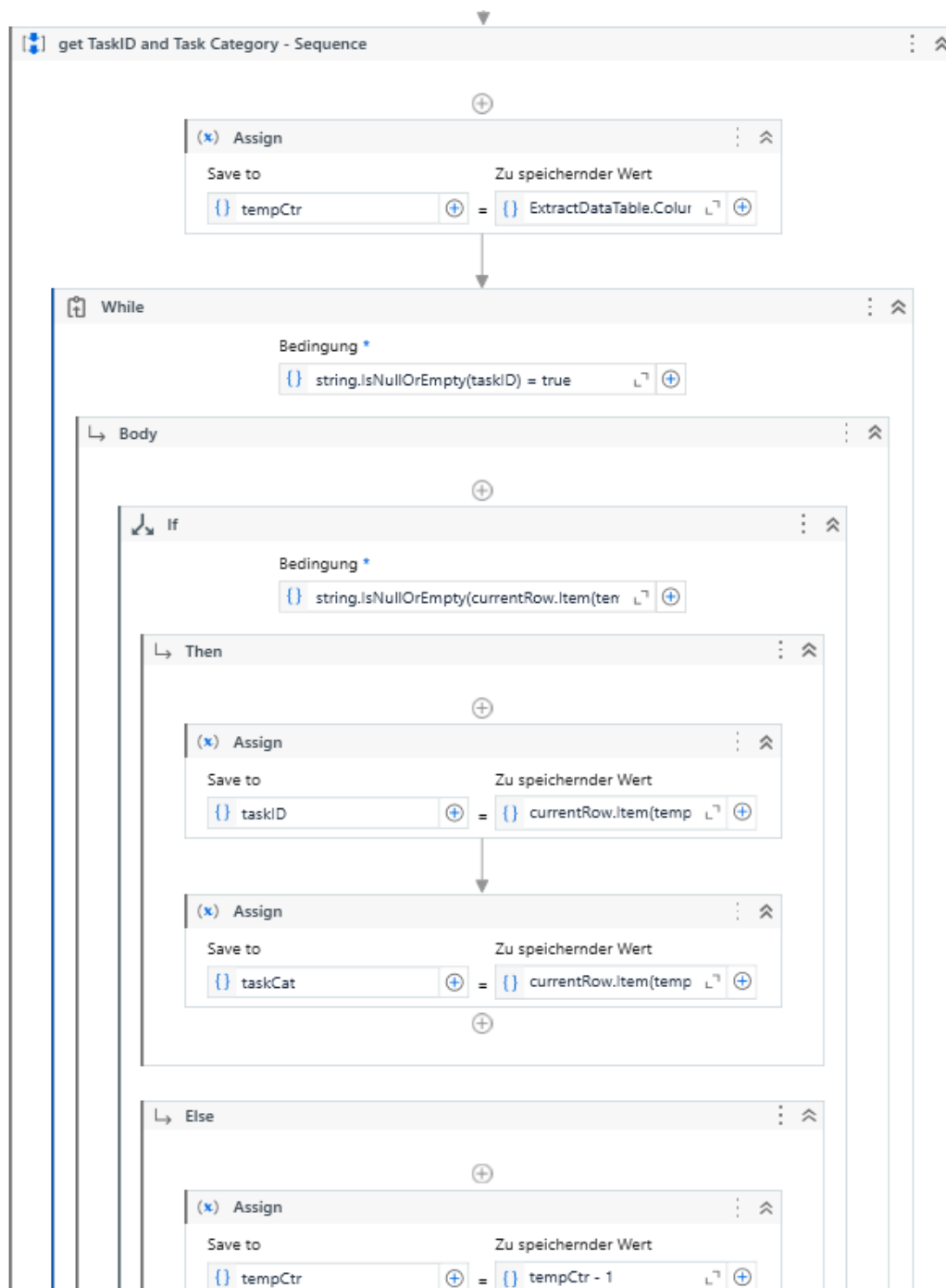


Abbildung 44: UiPath – SharePoint – taskID & taskCat

Hier wird anhand einer Zählvariable, die in der *while-Schleife* mit jedem Durchgang addiert wird, die taskID und taskCat bestimmt. Dabei muss die Tabelle von hinten durchgegangen werden, da es in der DataTable eine unbegrenzte Anzahl von Spalten für Mitarbeiter geben kann, und die taskID und taskCat ganz am Ende der Tabelle gespeichert sind. Zuerst wird die taskID und dann, die taskCat mit „-1“ geholt. Die taskCat befindet sich genau eine Spalte hinter der taskID. Wenn der letzte Eintrag in der Tabelle nicht NULL ist, soll er die taskID und die taskCat zuordnen.

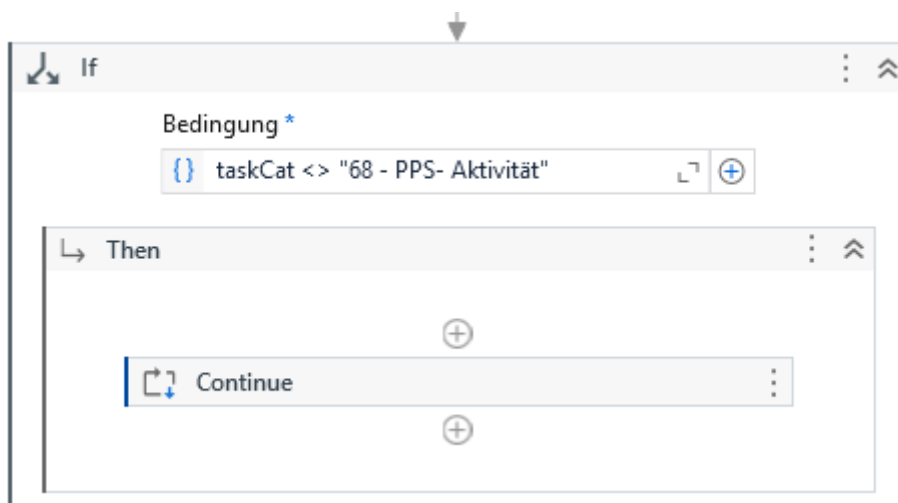


Abbildung 45: UiPath – SharePoint – Bedingung taskCat

Wenn die Variable taskCat jedoch ungleich „68 – PPS- Aktivität“ ist, dann wird der aktuelle Schleifendurchgang beendet. Dieser Fall ist jedoch sehr unwahrscheinlich, da zuvor mit dem Filter gearbeitet worden ist.

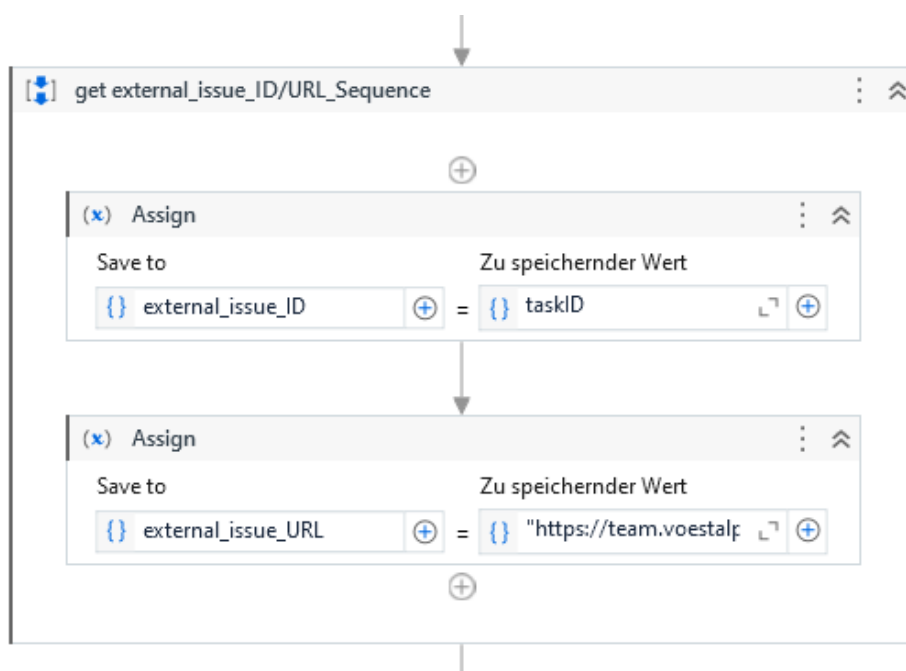


Abbildung 46: UiPath – SharePoint – external_issue_ID & URL

Die external_issue_ID und die external_issue_Url werden für die Verbindung zwischen dem Jira-Arbeitspaket und dem SharePoint-Arbeitspaket in Jira gespeichert. Dieser Schritt erfolgt jedoch später im zweiten Teil der Automatisierung. Dabei wird der external_issue_ID die taskID zugewiesen und der external_issue_URL wird der Link zu dem Arbeitspaket in SharePoint.

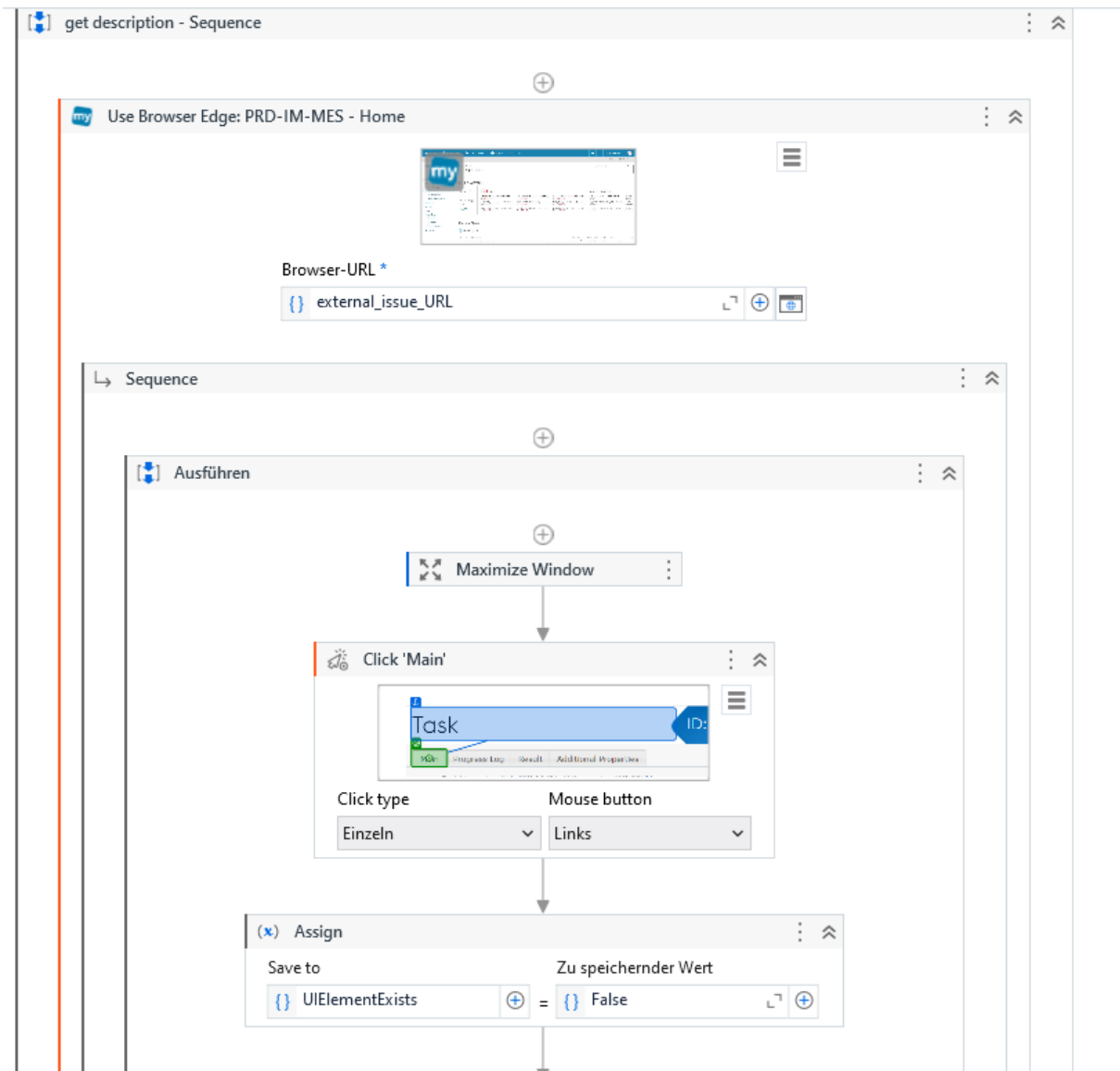


Abbildung 47: UiPath – SharePoint – Arbeitspaket aufruf external_issue_URL

Dieser nächste Schritt besteht aus dem Aufrufen von dem Arbeitspaket durch die external_issue_URL und dem Speichern der Beschreibung der Task. Dabei wird, wenn man auf der Website ist, das Fenster vergrößert und auf das Element Task auf der Website geklickt. Danach ist eine *boolean* Variable UIElementExists auf *false* gesetzt worden.

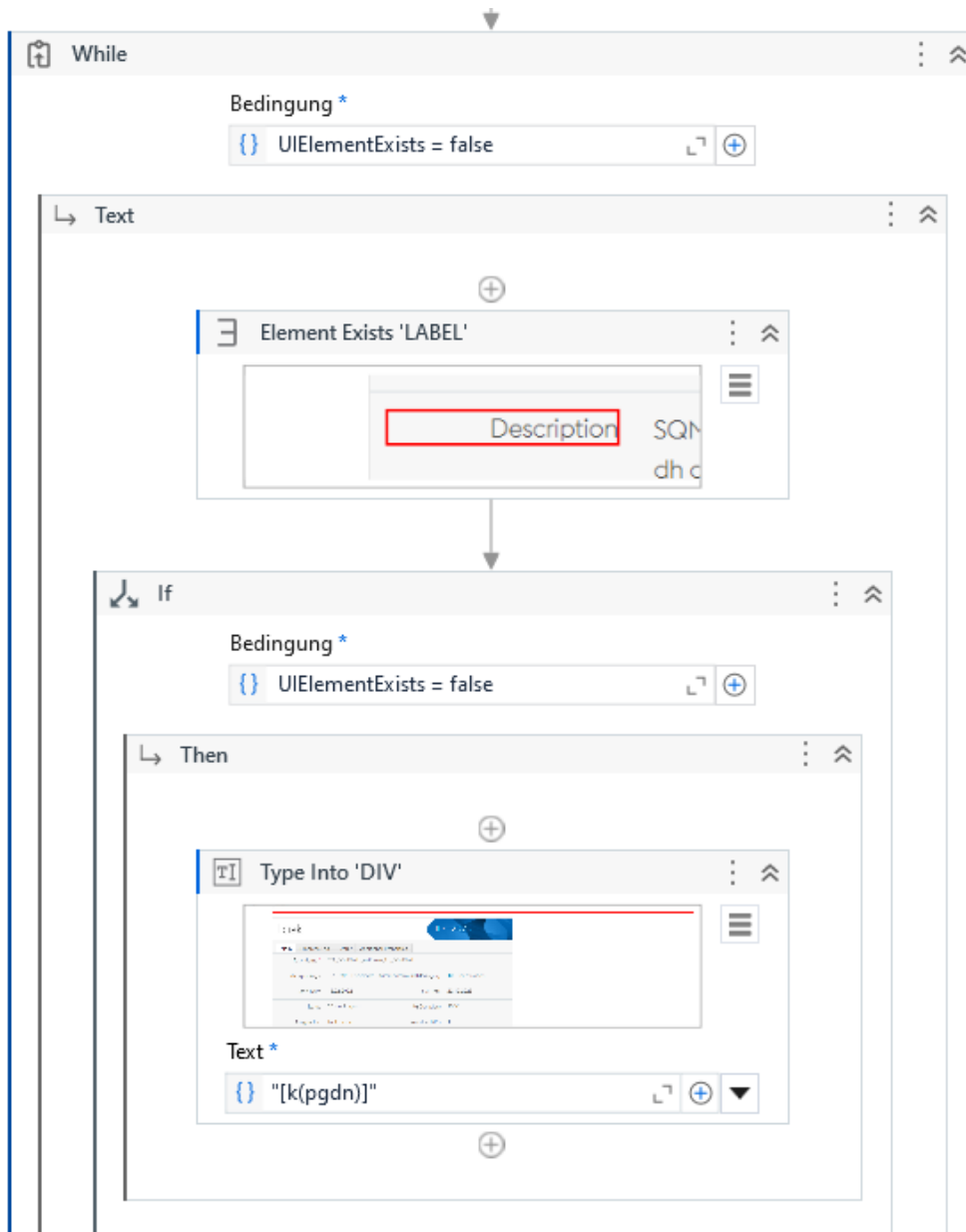


Abbildung 48: UiPath – SharePoint – Element Exists description

Mit der *boolean* Variable `UIElementExists` welche zuvor auf *false* gesetzt wurde, führen wir eine *while-Schleife* so lange aus, bis das Element `Description` auf der Website gefunden wurde. Dies wird durch das UiPath Element „Element Exists“ erzielt. Wenn „Description“ gefunden wurde, dann wird die *boolean* Variable auf *true* gesetzt und die Schleife abgebrochen. Falls dies nicht der Fall ist, wird durch das Element `Type Into` „DIV“ mit dem Befehl „[k(pgdn)]“ was automatisch durch das Klicken der Pfeiltaste nach unten erstellt wurde, so lange nach unten gescrollt, bis `Description` gefunden wird.

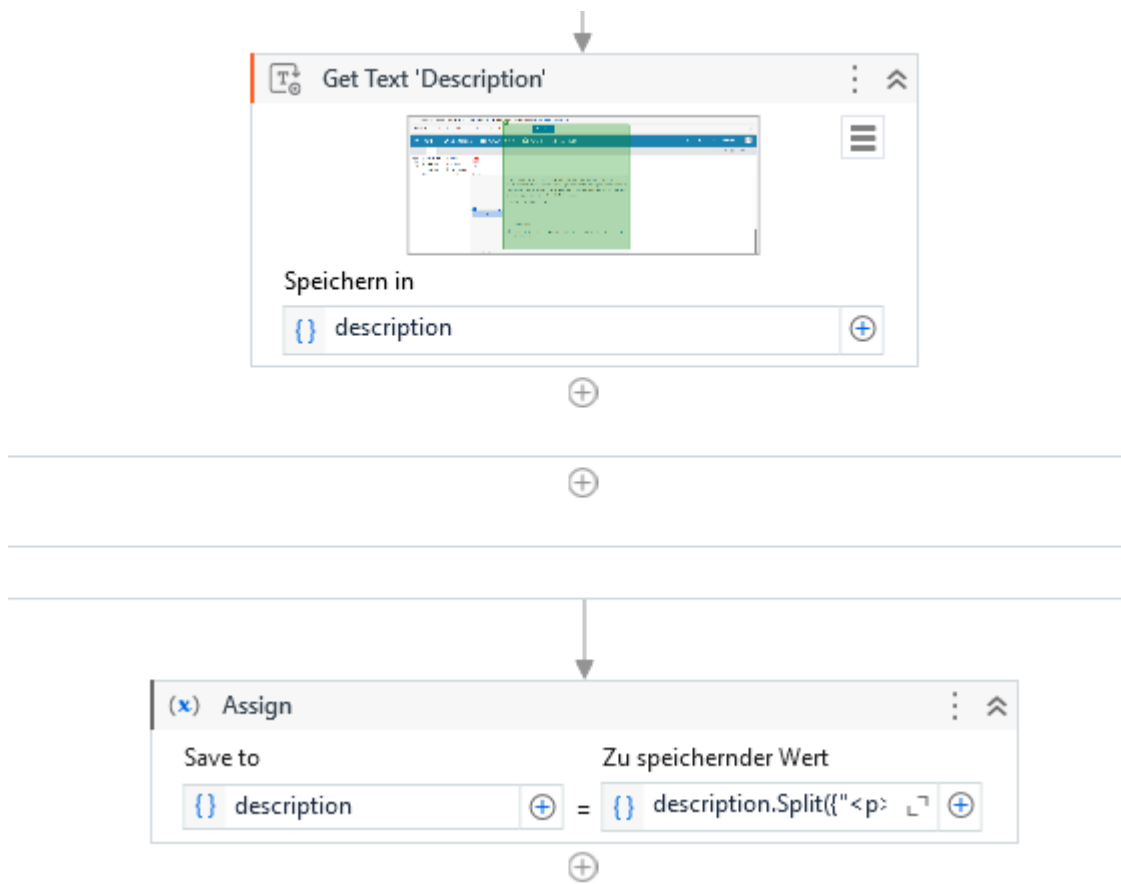


Abbildung 49: UiPath – SharePoint – Assign description

Wenn die Schleife durchgelaufen ist, wird das Element „Get Text“ ausgeführt, welche die Description die in dem Arbeitspaket als Text aufgeschrieben ist, in eine Variable description speichert. Dabei ist der Text in mehreren Zeilen geschrieben worden, weswegen der Gesamte Text mit dem Wert „description.Split({„<p>“}, StringSplitOptions.None)(0)“ gespeichert werden muss, um somit den gesamten Text zu bekommen.

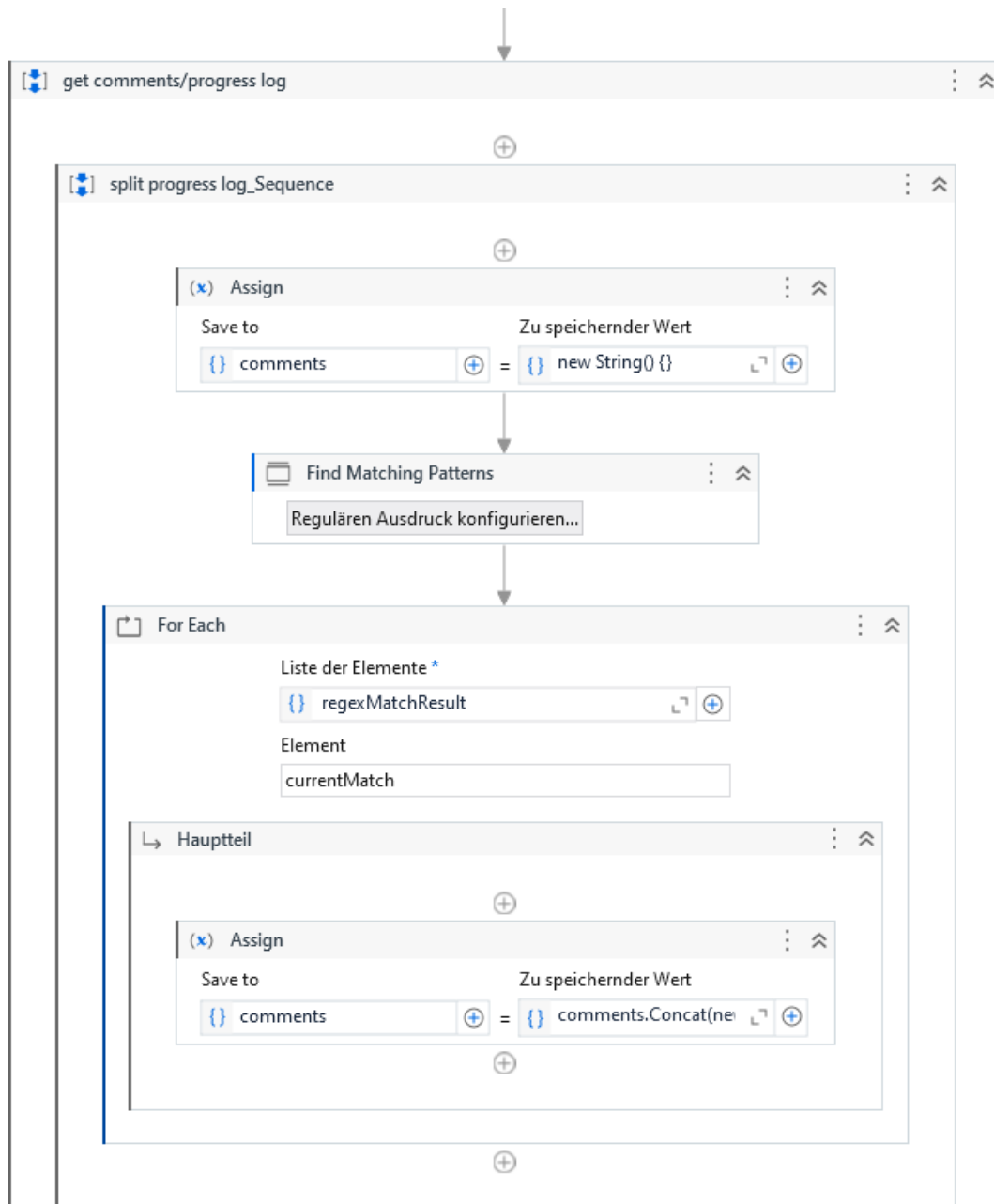


Abbildung 50: UiPath – SharePoint – comments array

Hierbei werden alle *Strings*, welche in der Comments-Sektion vorhanden sind, durch eine *for Each-schleife* in ein String Array „comments“ gespeichert. Durch die Funktion *Concat* wird jeder einzelne String hineingespeichert.

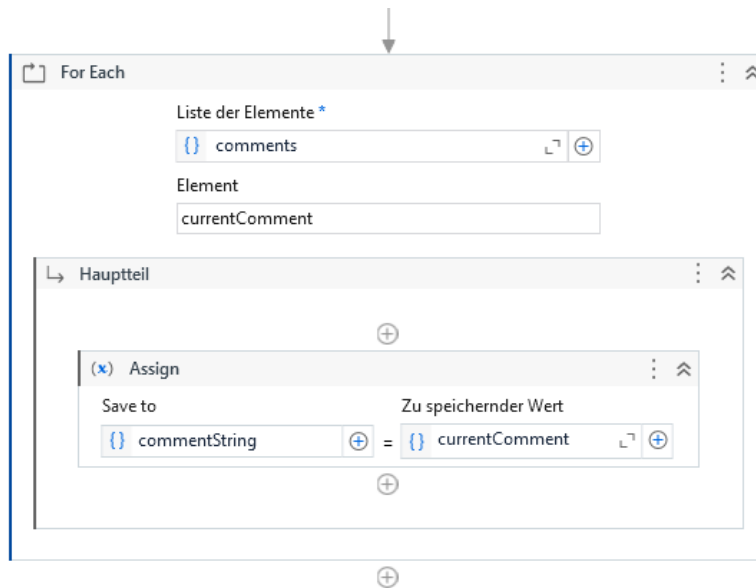


Abbildung 51: UiPath – SharePoint - commentString

Der volle Kommentar welcher in der vorherigen *For Each*-Schleife erzeugt wird, wird in der nächsten in eine Variable „commentString“ gespeichert.

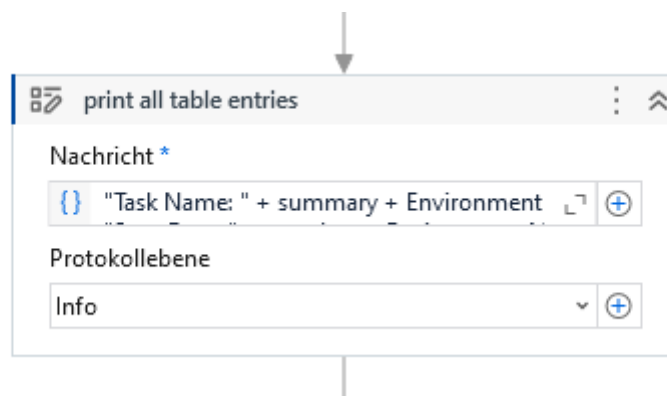


Abbildung 52:UiPath – SharePoint – print all table entries

Zum Testen ob alle Variablen erfolgreich erzeugt und befüllt worden sind, wird eine Ausgabe aller Variablen mit einem „print all table entries“ Element, durchgeführt.

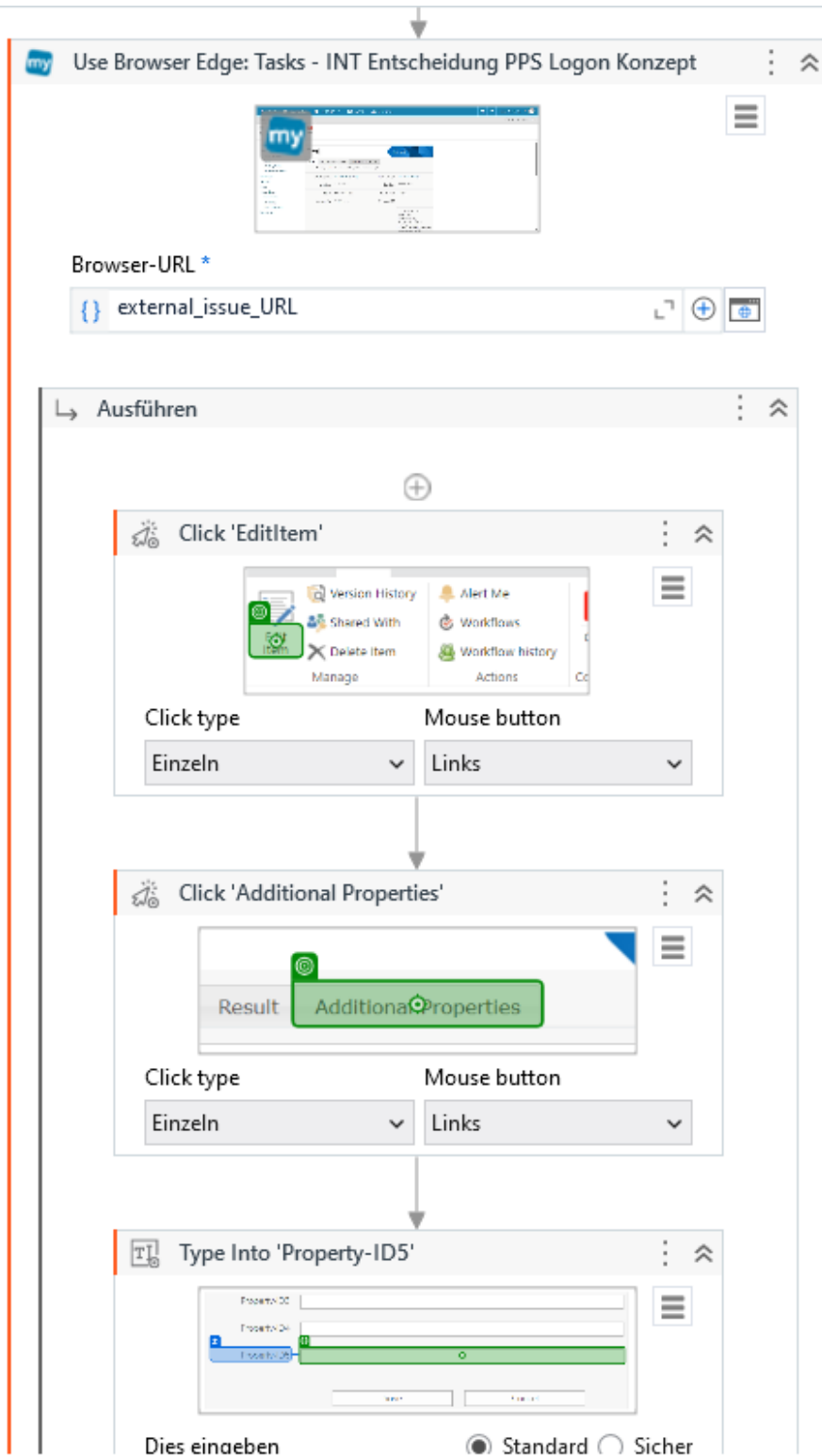


Abbildung 53: UiPath - SharePoint - JiraTaskUrl

Dieser Schritt ist erst nach dem Jira Teil der Automatisierung zu erledigen. Dabei wird durch ein „Use Browser“ Element, der Browser mit der „external_issue_url“, also dem Link zu dem Task, geöffnet. Danach werden mit „Click“ Elementen, jeweils auf „EditItem“ und „Additional Properties“ geklickt, um somit den Task bearbeiten zu können. Dies funktioniert wiederum mit einem „Type Into“ Element, welches den Link des Jira-Tasks in die Leiste speichert.

Jira

Jira wird in der Main.xaml hauptsächlich mit den „Invoke Workfile“-Aktivitäten aufgerufen. Die hierbei verwendeten „.xaml“-Dateien heißen:

- CheckJiraExistence.xaml
- GetAssigneeAccountID.xaml
- DefaultJiraChanges.xaml
- CreateNewComment.xaml

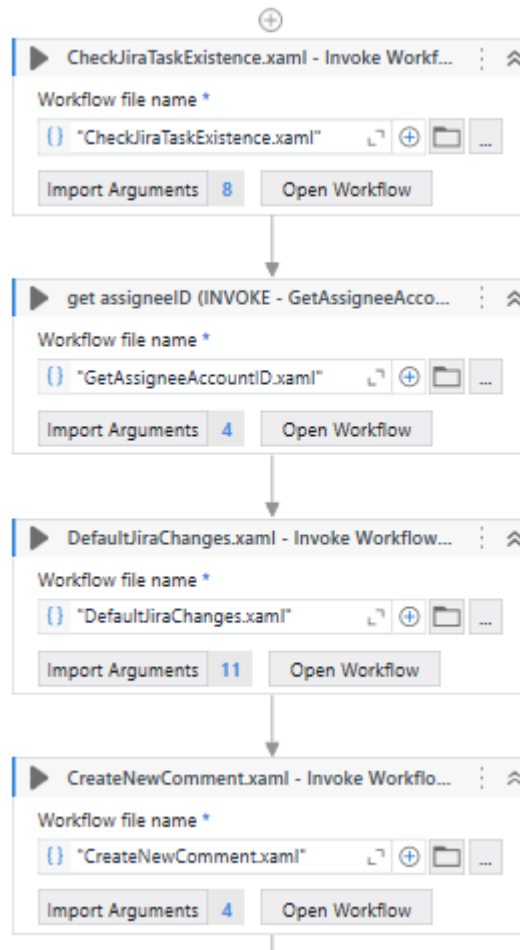


Abbildung 54: UiPath Jira (Invoke Workfiles)

CheckJiraTaskExistence.xaml

In diesem Workflow wird überprüft, ob der Task in Jira bereits existiert. Der erste HTTP Request ist nur ein GET Request, der im Link die Task ID von SharePoint mitgibt, um den Task zu finden. Bei diesem Request holen wir uns mit der Variable *existenceStatusCode* den Request Code. Dieser Status Code wird mit der if-Verzweigung wie folgt überprüft:

- Falls der Request Code 200 ist, existiert der Task und es muss nichts getan werden.
- Falls dies nicht der Fall ist, existiert der gesuchte Task nicht in Jira und ein dementsprechender Task im Jira erstellt werden.

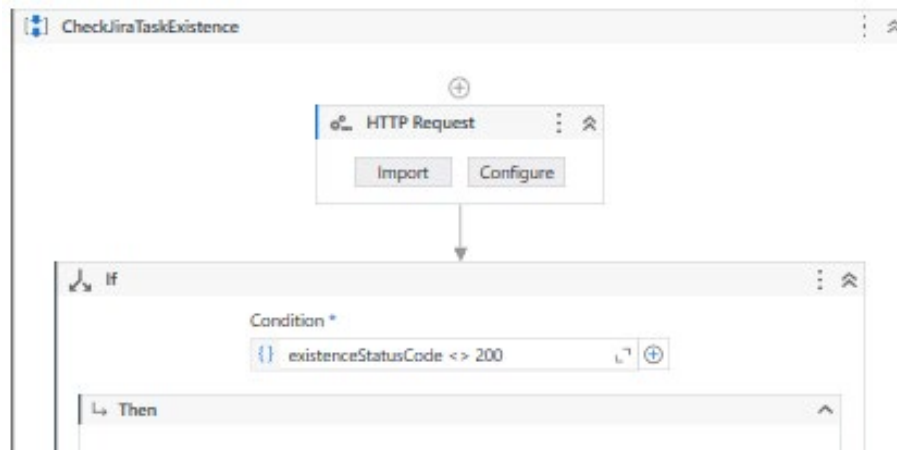
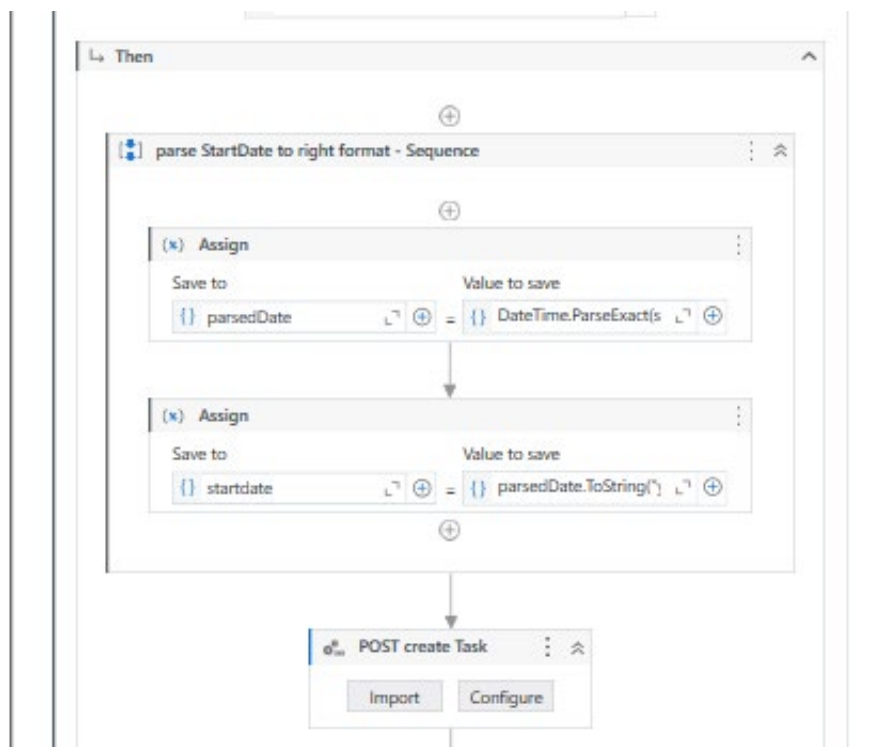


Abbildung 55: UiPath - Jira - CheckJiraTaskExistence

Wenn der Task nicht existiert, parsen wir zuerst das Datum des Taskstarts zum richtigen Format und führen anschließend einen weiteren HTTP Requests aus. Dieser Request ist natürlich der jeweilige POST Request, um einen Jira Task zu erstellen.



GetAssigneeAccountID.xaml

Dieser Workflow ist dafür zuständig, dass man die korrekte Assignee ID herausfindet. Also die ID des Zuständigen, so wie es in Jira angegeben ist. Dies macht man indem man zuerst alle User, die im Jira der Voestalpine vorhanden sind, mithilfe des jeweiligen GET Requests bekommt. Die zurückbekommene JSON-Array kann man nun mithilfe einer For Each-Schleife durchlaufen, um den gesuchten Assignee zu bekommen.

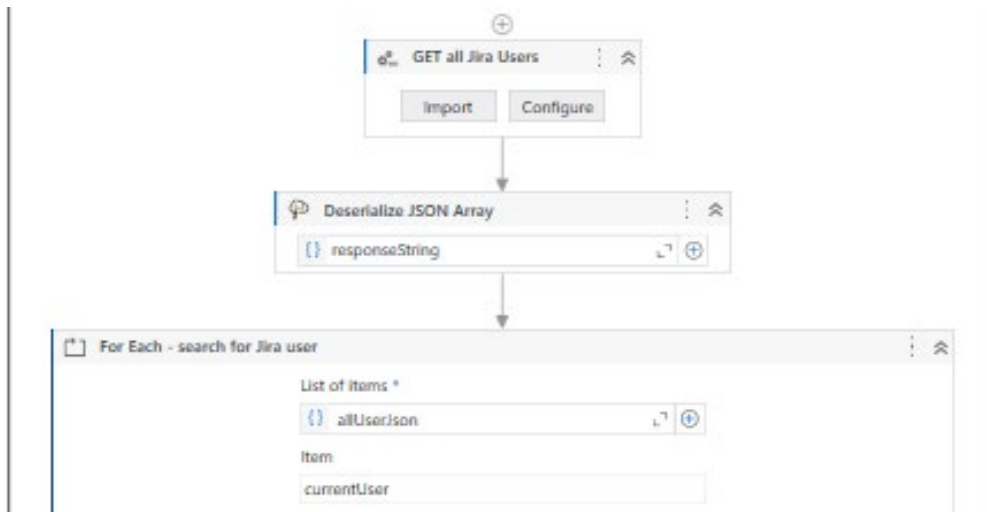


Abbildung 57: UiPath - Jira - GetAssigneeAccountID

Um nun herauszufinden, ob es nun der korrekte User gefunden wurde, deserialisiert man zuerst das momentane JSON-Objekt und speichert dieses Objekt in die *userJsonObj*-Variable.

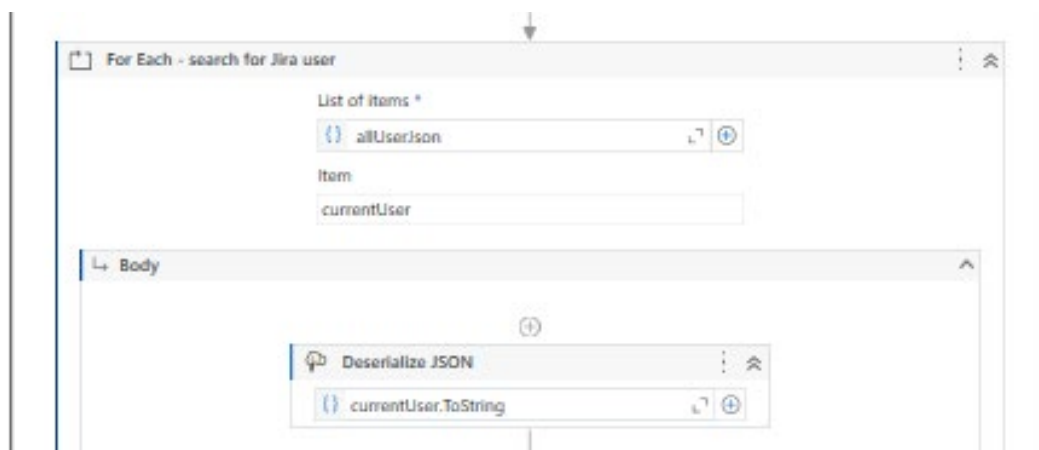


Abbildung 58: UiPath - Jira - GetAssigneeAccountID

Bevor man nun überprüft, ob man das JSON-Objekt mit dem jeweilig richtigen Zuständigen hat, muss zuerst überprüft werden, ob das Objekt überhaupt eine E-Mail-Adresse hat, die man mit der E-Mail-Adresse von SharePoint vergleichen kann. Dies tut man, da nicht nur relevante Nutzer in dem User Array existiert, sondern auch angelegte Accounts für Testuser oder Andere, die eben keine E-Mail-Adresse haben und somit keine realen Menschen sind, die in der

Voestalpine arbeiten. Ob nun eine E-Mail-Adresse vorhanden ist überprüft man mit der ersten if-Verzweigung mit dem folgendem Befehl:

```
userJsonObj.Item("emailAddress") IsNot Nothing
```

Falls eine E-Mail-Adresse im JSON-Objekt existiert, wird nun mit der zweiten if-Verzweigung überprüft, ob diese mit der gesuchten E-Mail vom SharePoint übereinstimmt. Dies macht man mit folgendem Befehl:

```
userJsonObj.Item("emailAddress").ToString.Equals(assigneeMailAddress)
```

(Die Variable *assigneeMailAddress* mit der man hier vergleicht, beinhaltet die E-Mail-Adresse, die man vom SharePoint extrahiert hat.)

Wenn die E-Mail-Adresse stimmt, wird diese in die Variable *assignee* geschrieben. Diese wird später auch in dem Workflow *DefaultJiraChanges.xaml* verwendet, um dem Task unter anderem den richtigen Assignee (Zuständigen) zu zuweisen.

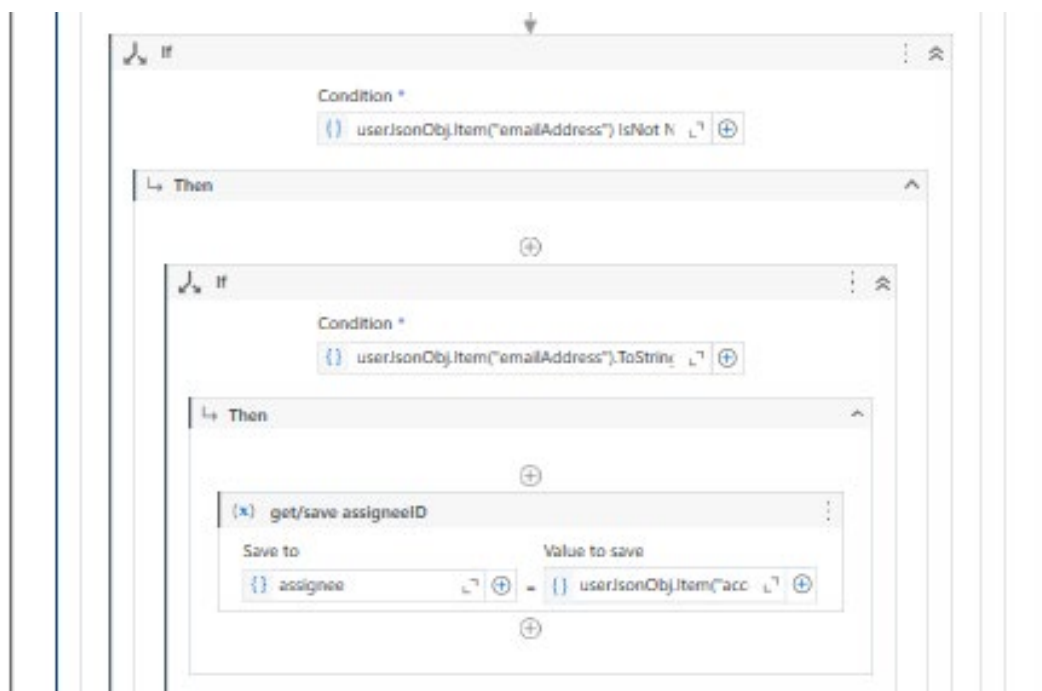


Abbildung 59: UiPath - Jira - GetAssigneeAccountID

DefaultJiraChanges.xml

Für diesen Workflow werden folgende Dinge benötigt und in Jira immer geändert:

- *summary* (Name des Tasks)
- *duedate* (Datum an dem der Task fällig ist)
- *assignee* (die ID des Zuständigen)
- *description* (Beschreibung des Tasks)

Diese Dinge müssen deshalb immer geändert werden, da diese von SharePoint aus Priorität haben (Anforderung unseres Auftraggebers).

Bevor man den HTTP PUT Request durchführt, wird der Request Body vorbereitet und der mit den jeweiligen Variablen korrekt befüllt.

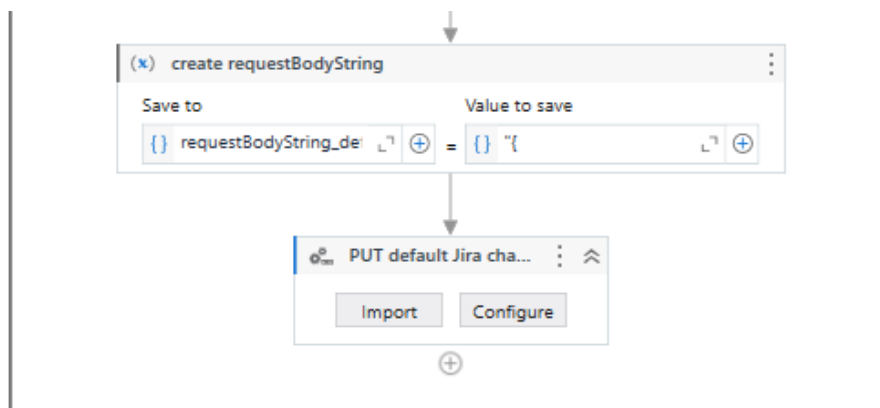


Abbildung 60: UiPath - Jira - DefaultJiraChanges

CreateNewComment.xaml

Dieser Workflow ist dazu zuständig, die Kommentare in einen String umzuwandeln, mit dem man den POST Request ausführen kann, der dafür zuständig ist neue Kommentare hinzuzufügen.

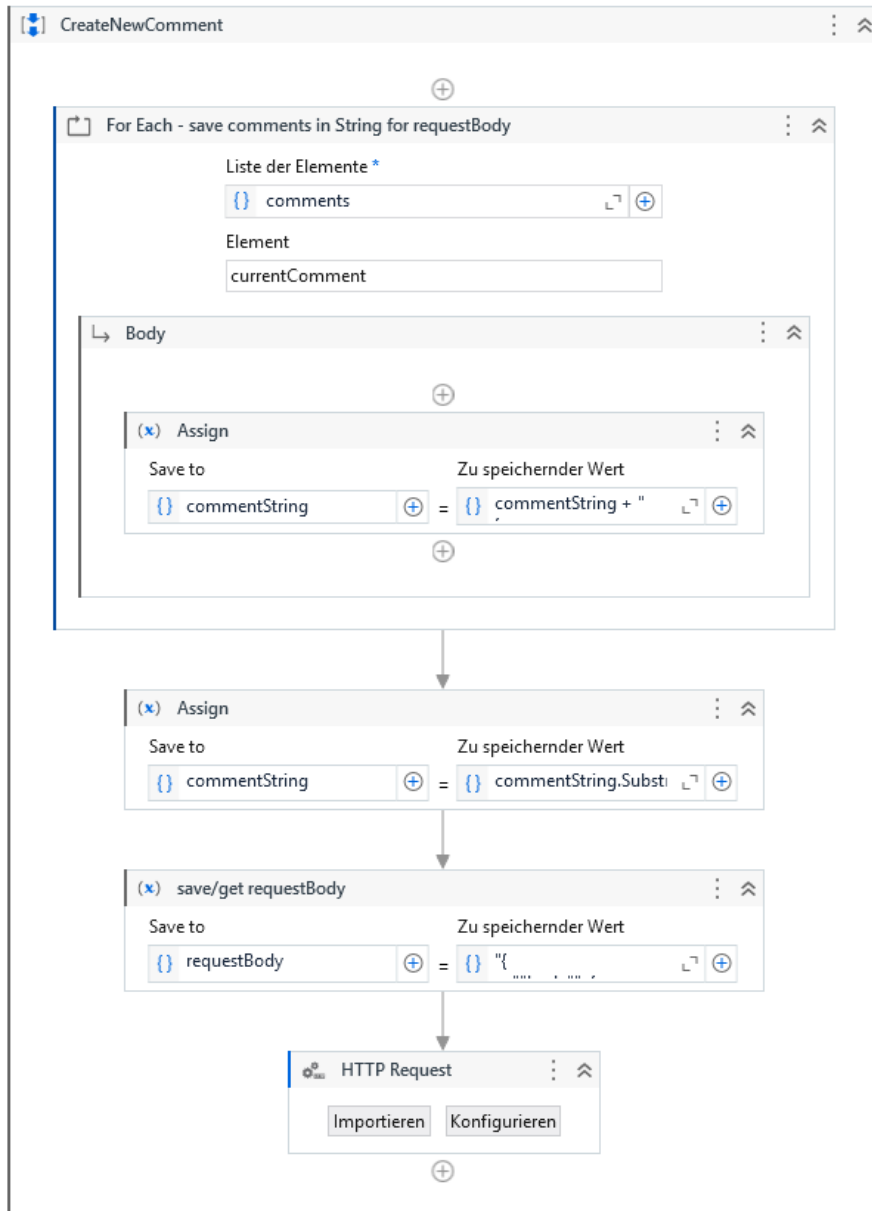


Abbildung 61: UiPath - Jira - CreateNewComment

ChangeStatus

Um den Status im Jira auf den richtigen umzuändern, wird in der Main.xaml noch eine Sequence ausgeführt, die sich mittels eines GET Requests alle PPS-Tasks herausholt. Die Response des Requests wird in ein JSON Object gespeichert, um mit dem Schlüsselwort „issues“ die Tasks dann endgültig in ein Array zu speichern.

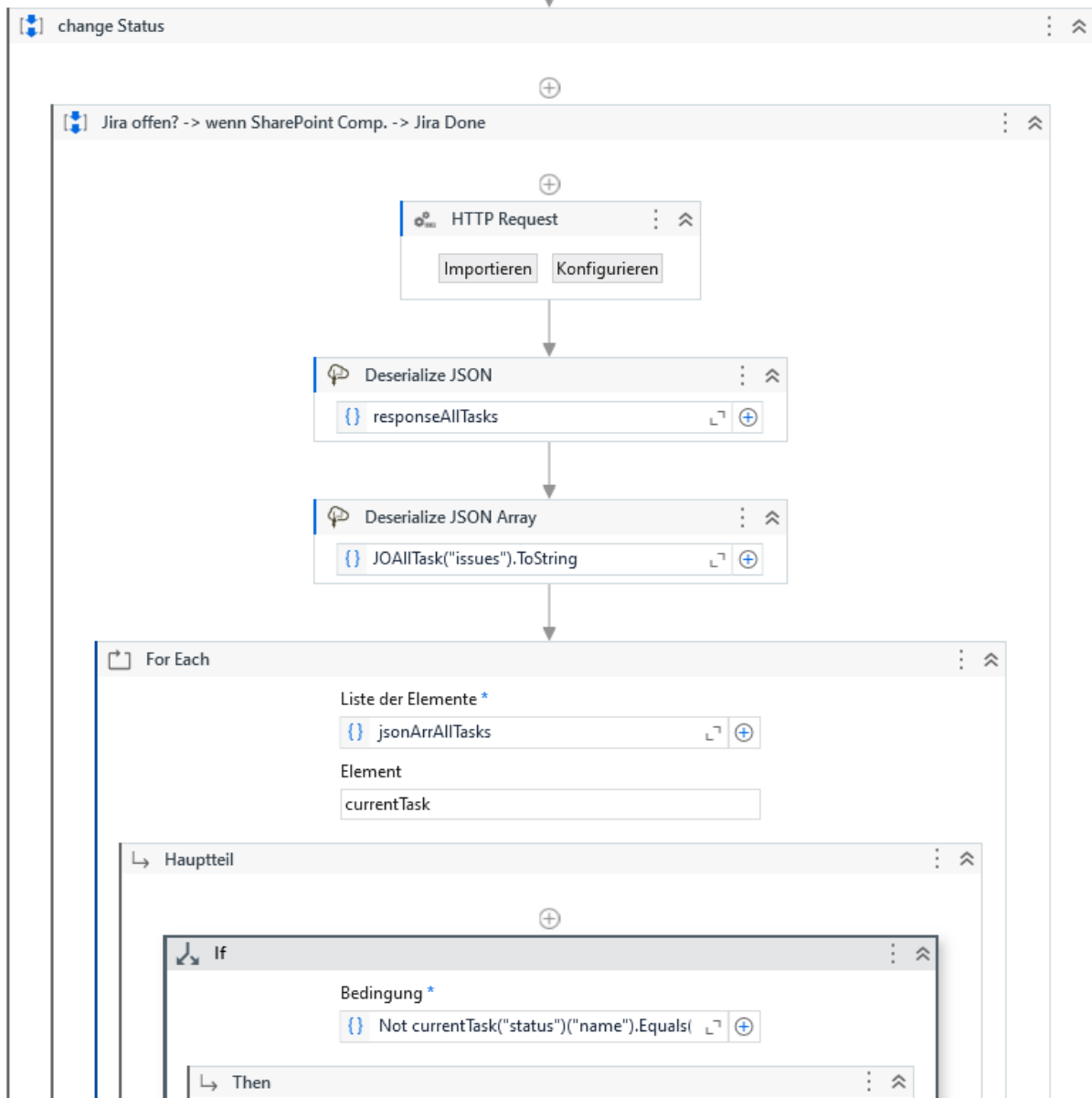


Abbildung 62: UiPath - Jira - ChangeStatus (Sequence)

Nun gehen wir die Tasks mit einem Array durch und überprüfen, ob der Task nicht auf „Done“ gesetzt wurde. Dies tun wir mit dem folgenden Befehl:

```
Not currentTask("status")("name").Equals("Done")
```

Wenn der Task also nicht auf Done gesetzt ist, dann wird der Status vom SharePoint geholt, um zu überprüfen, ob dieser auf Completed gesetzt wurde.

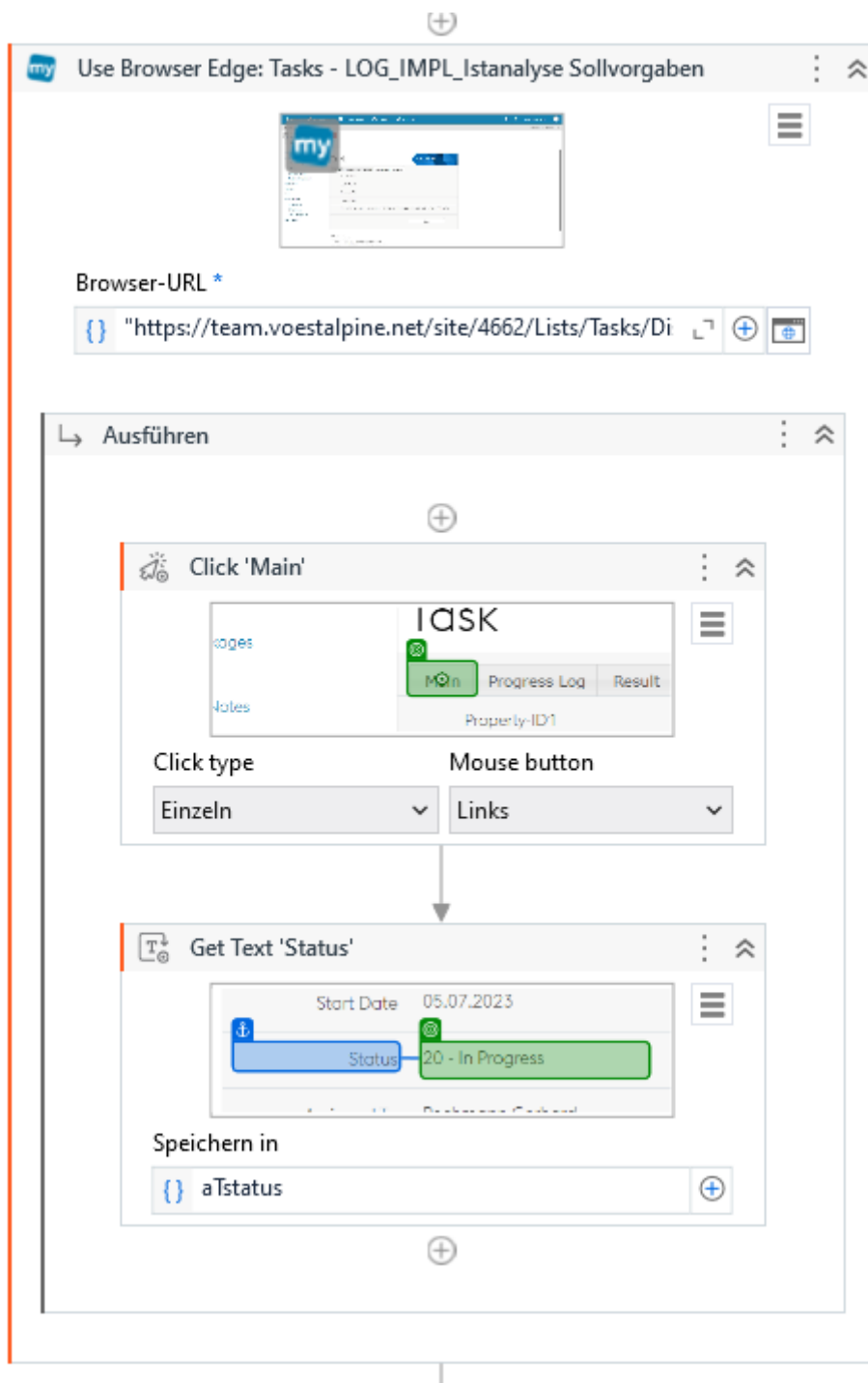


Abbildung 63: UiPath - Jira - ChangeStatus (Sequence)

Der Status vom SharePoint wird in die Variable aTstatus gespeichert und dann in der if-Bedingung wie folgt überprüft, ob der Status in SharePoint auf Completed gesetzt wurde:

```
aTstatus.Equals("90 - Completed")
```

Wenn dies der Fall ist und der Status ist im SharePoint auf Completed gesetzt worden, soll der Workfile ChangeStatus.xaml aufgerufen werden.

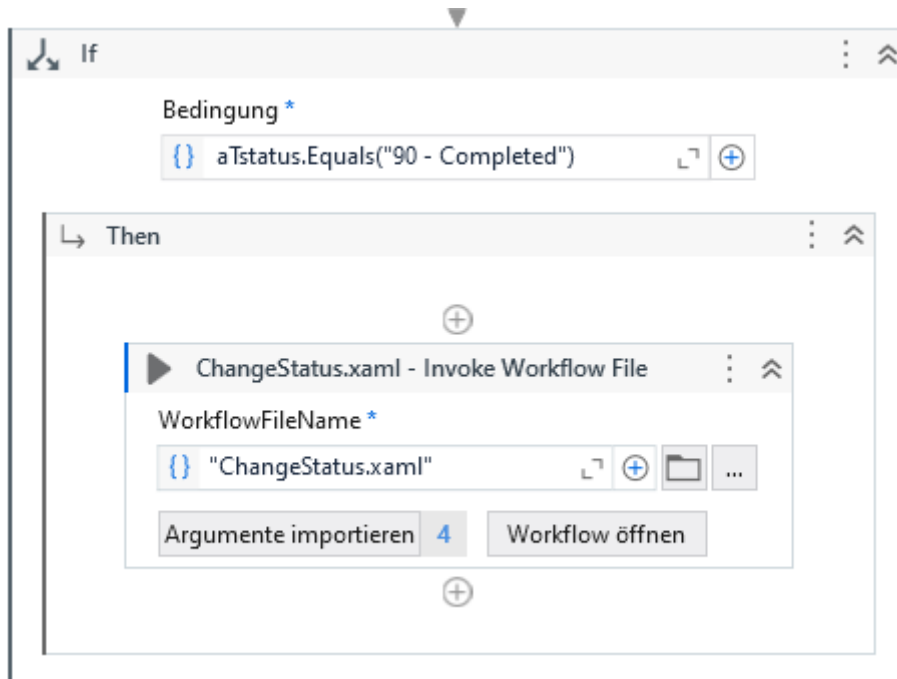


Abbildung 64: UiPath - Jira - ChangeStatus (Sequence)

Im Workfile „ChangeStatus.xaml“ wird zunächst der Status auf den Wert umgeschrieben, wie er in Jira heißt. (SharePoint: Completed; Jira: Done)

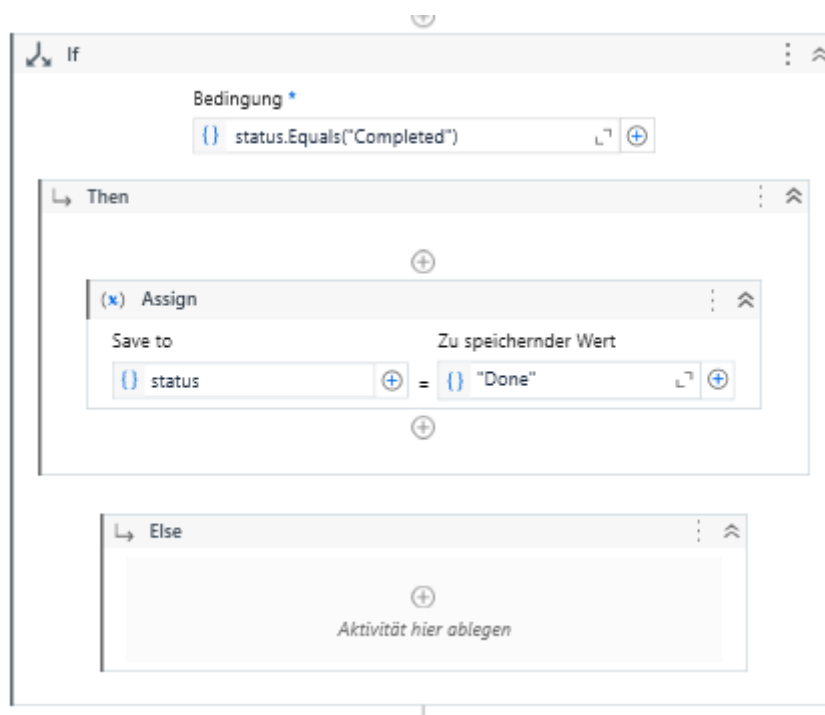


Abbildung 65: UiPath - Jira - ChangeStatus (Workfile)

Danach werden alle möglichen Transitions für diesen Task mithilfe des dementsprechenden GET Requests in ein JSON-Array gespeichert und mittels einer Schleife alle Transitions durchgegangen.

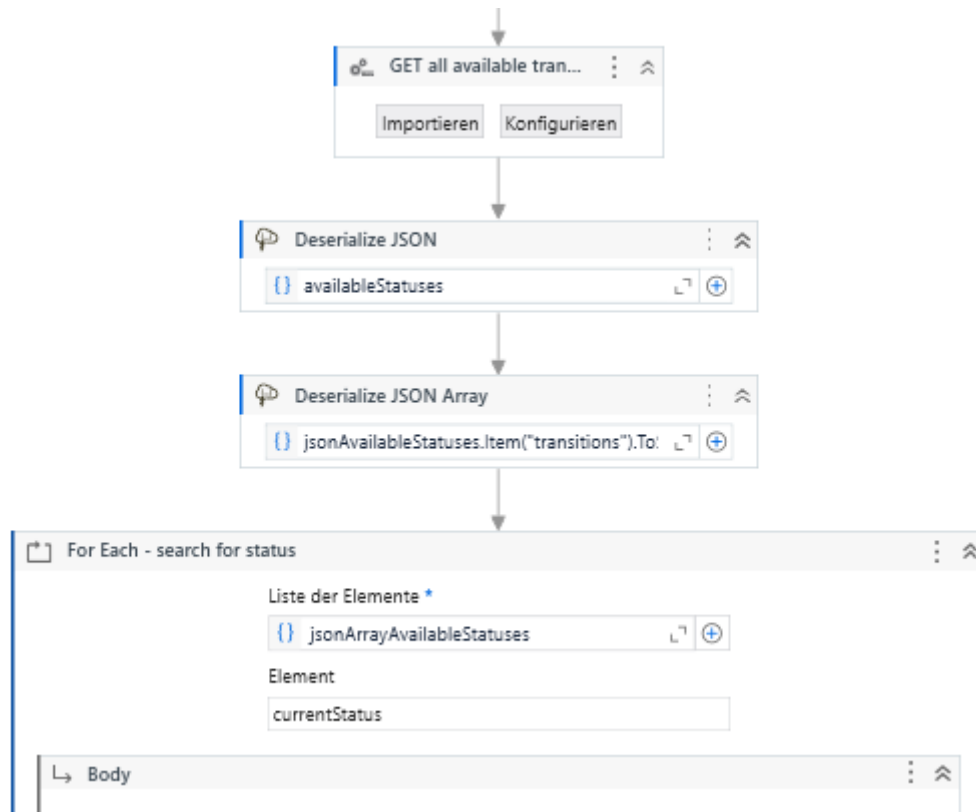


Abbildung 66: UiPath - Jira - ChangeStatus (Workfile)

In dieser Schleife wird dann mithilfe einer if-Bedingung überprüft, ob man die richtige Transition gefunden hat. Falls dies der Fall ist, holt man sich den Transition ID heraus.

if-Bedingung:

```
currentStatus.Item("name").ToString.Equals(status)
```

Transition ID:

```
currentStatus.Item("id").ToString
```

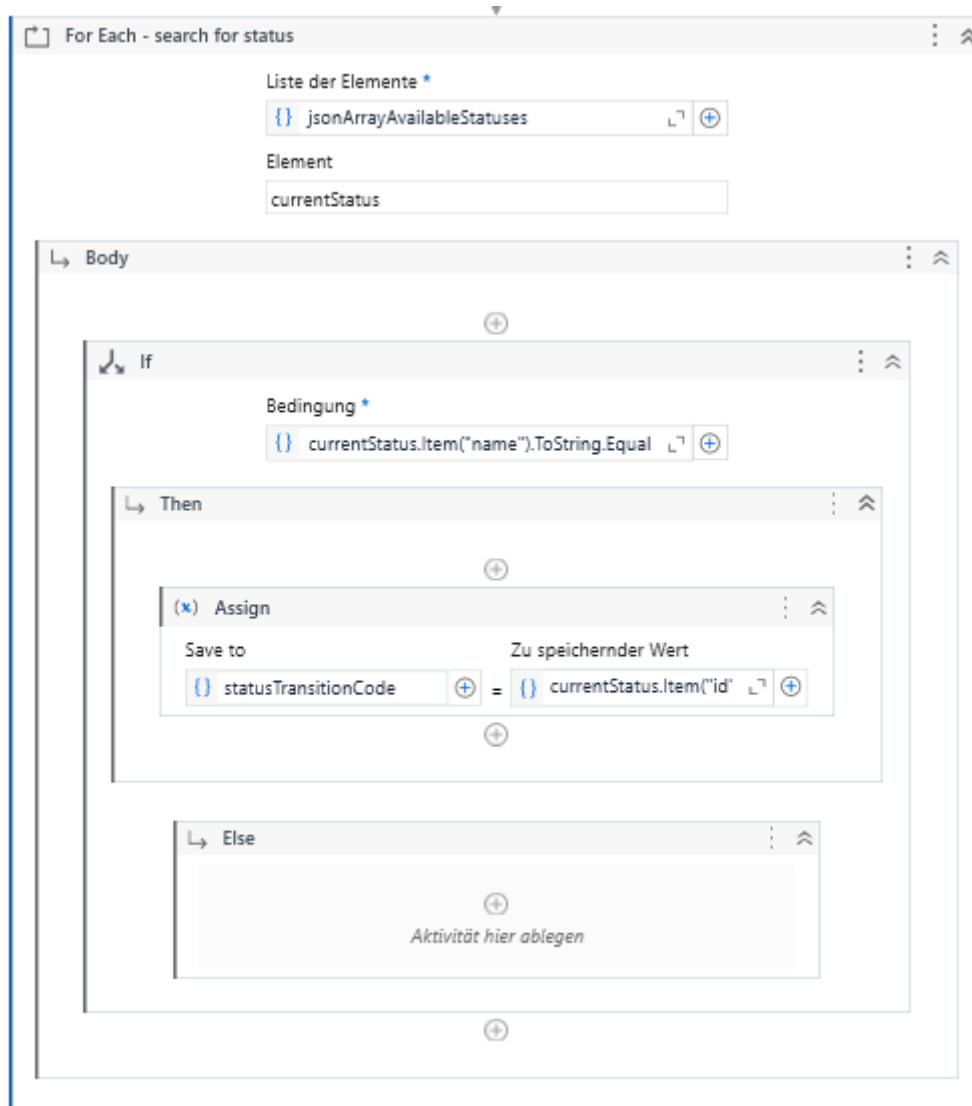


Abbildung 67: UiPath - Jira - ChangeStatus (Workfile)

Nachdem man die ID der jeweiligen Transition gefunden hat, wird ein dementsprechender Body für den POST Request, um den Status des Tasks umzuändern, angefertigt.

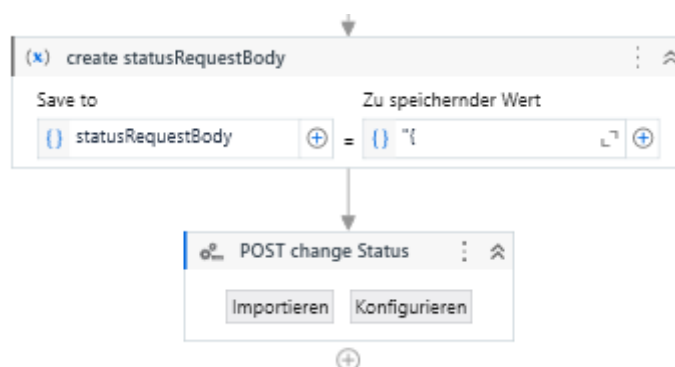


Abbildung 68: UiPath - Jira - ChangeStatus (Workfile)

11.2 Simple Query

11.2.1 Technologieauswahl

Wie bei der Technologieauswahl von der Synchronisation von SharePoint und Jira, wurde dem Projektteam auch bei Simple Query genügend Freiraum gegeben. Da die Mitglieder des Projekts bereits Kenntnisse in Java hinsichtlich ähnlicher Problemstellungen hatten, und diese Technologie auch für den Auftraggeber unproblematisch ist, fiel die Wahl auf diese Programmiersprache.



Abbildung 69: Java Logo

31



Abbildung 70: Spring Logo

³² Das Framework *Spring* wurde in Java eingebunden, um das Erstellen einer Website zu erleichtern, wobei dieses auch noch viele andere Anwendungsmöglichkeiten bietet.

Die Wahl von Spring als Framework anstelle von JSP (Java Server Pages) wurde durch diese Benutzerfreundlichkeit die Spring bietet getroffen.

In diesem Projekt wurde jedoch nicht nur Spring verwendet, sondern Spring Boot. Spring Boot ist eine Erweiterung von Spring, welche die umfangreiche Konfiguration, die für Spring benötigt wird, zu vereinfachen.

³¹ Java Logo: <https://icon-icons.com/de/symbol/java-logo/169577>

³² Spring Logo: <https://github.com/Rom1kz>

11.2.2 Spring Installation

Spring kann ganz einfach als Maven Dependency hinzugefügt werden. Wobei auch Gradle was auch eine Java Library ist welche Spring unterstützt. In diesem Fall wurde Maven verwendet:

```
<dependency>
  <groupId>org.springframework.boot</groupId>
  <artifactId>spring-boot-starter-webflux</artifactId>
  <version>3.1.2</version>
</dependency>
```

Wenn diese Maven dependency hinzugefügt worden ist, kann man Spring Boot verwenden. Um den WebClient nun zu starten muss man in einer Java Klasse folgendes schreiben:

```
@SpringBootApplication
public class StartWebclient {
    public static void main(String[] args) {
        SpringApplication.run(StartWebclient.class, args);
    }
}
```

Mithilfe des gestarteten WebClients kann man nun die verschiedensten Webseiten hosten. Dies macht man indem man eine Klasse wie folgt konfiguriert:

```
@RestController
public class SpringClass {
}
```

Der Annotation *@RestController* wird geschrieben um die Klasse zugreifbar für das Spring-Boot Framework zu machen.

```
@GetMapping("/link")
@ResponseBody
public String springmethod(@RequestParam("name") String name) {
    return "<html><body><h1>Hallo, " + name + "!</h1></body></html>";
}
```

Die Methode *springmethod* wird in diesem Beispiel für das Framework mit der Annotation *@GetMapping* gekennzeichnet. Über die Annotation *@RequestParam* können Daten der Methode im http Request mitgeben. Mit der Annotation *@ResponseBody* wird dem Framework mitgeteilt, dass die Methode einen oder mehrere Wert zurückliefert. In diesem Beispiel werden nicht nur Werte, sondern eine einfache HTML-Seite zurückgesendet, bei der eine personalisierte Begrüßung durchgeführt wird.

11.2.3 SimpleQuery API

Die Simple Query API wurde von der Voest bereitgestellt. Simple Query steht für den Zusammenbau von Queries, die an die API gesendet werden. Von dieser API werden Versionsdaten zurückgesendet. Die Aufgabe des Projekts ist es, die Versionsdaten von zwei beliebigen Servern zu vergleichen. Dies hat den Nutzen, dass die Mitarbeiter der Voest nicht mehr manuell alle Daten durchgehen müssen, um veraltete Versionen von Software auf den verschiedenen Servern zu identifizieren, um sie dann aktualisieren zu können.

Die Voest hatte bereits ein grafisches Formular, mit welchem man eine Query zusammenstellen kann, jedoch gibt es auf dieser Website keinerlei Möglichkeit Daten von zwei verschiedenen Servern zu vergleichen. Anfangs war das Ziel die Website nachzubauen, damit im neuen Programm individuelle Queries zusammengestellt werden können.

11.2.1 Programmaufbau

Der Programmaufbau wird in *ressourcen* wie dateien welche verwendet werden, in *model* und *webclient*, sowie einige Klassen zur Datenverarbeitung und zum Verbindungsaufbau mit der API der Voest, unterteilt.

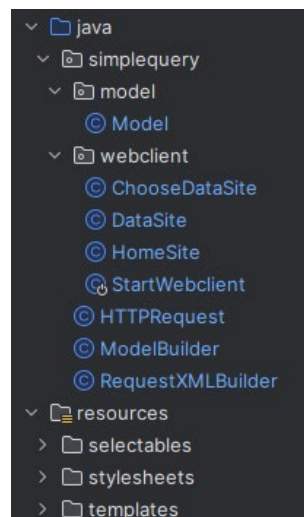


Abbildung 71: Simple Query Programmaufbau

11.2.2 Query Zusammenstellung

Als aller erstes muss ein Server ausgewählt werden, von welchen man die Daten haben will. Dies wird im Modell der Voest über die URL der Website gemacht. Ein Beispiel hierfür wäre:

<http://vadvexscs.voestalpine.root.local:57600/rep/support/SimpleQuery>

Dies kann im Projekt über ein Fenster, welches den Auswahl von ein bis zwei Servern erlaubt, erledigt werden:

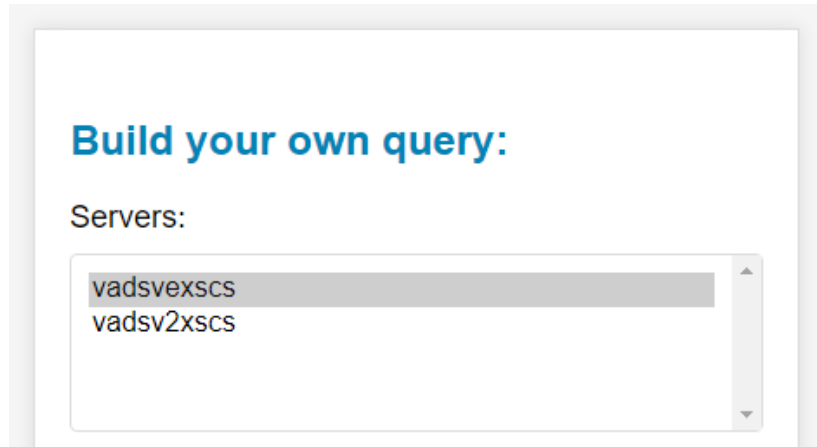


Abbildung 72: Simple Query Server

Danach kann eine Softwarekomponente ausgewählt werden, die benötigt wird.

Simple Query

Define the Software Component

- All non-local software components
 - All local software components
 - All software components
 - All newest software components
 - This one
- Local Software Components -> Local
- Consider underlying software components Return only modified objects

Abbildung 73: Simple Query Software components

Dies wurde auch im Projekt umgesetzt. Hier kann die Softwarekomponente über ein Drop-Down Fenster ausgewählt werden.

Software components:

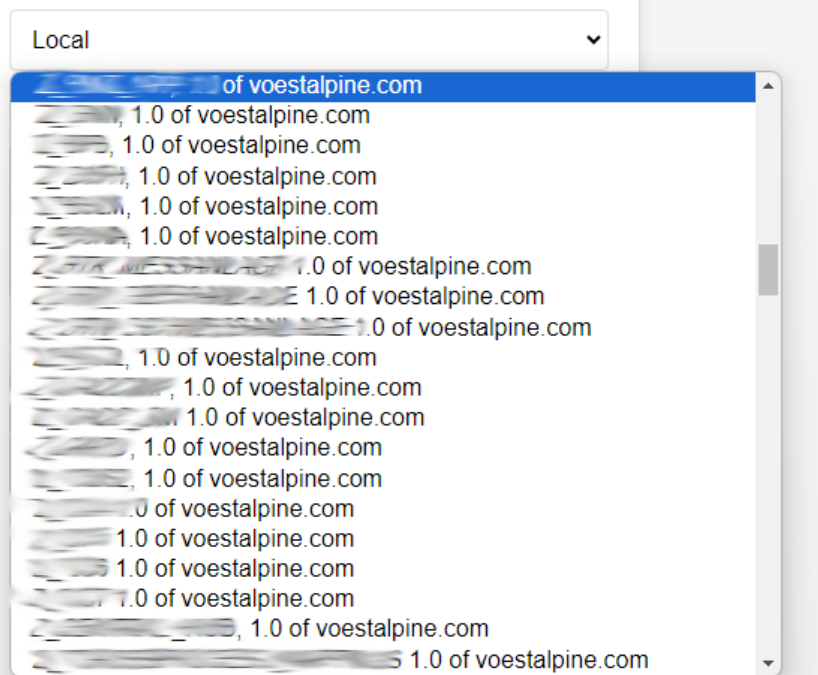


Abbildung 74: Simple Query Software components

Weiters kann der Objekt Typ in einer Software auf dem Server ausgewählt werden, welche auch im Projekt wieder umgesetzt wurde. Hierbei konnte auch wieder mehrere ausgewählt werden.

Define the Object type(s)

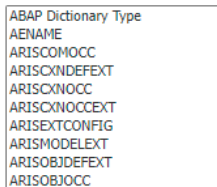


Abbildung 75: Voest Simple Query Object types

Object types:

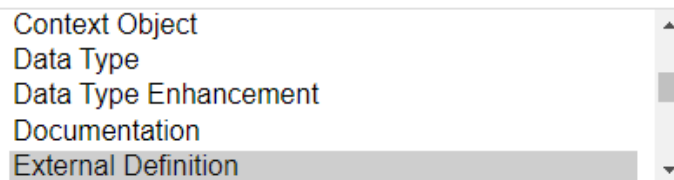


Abbildung 76: Simple Query Object types

Weiters kann noch nach Konditionen wie Namen, Versionsnummern, Zuständige Person, Datum der letzten Änderung oder vieles mehr Auswählen. Von diesen Suchoptionen können mehrere vorhanden sein. Weiters kann ausgewählt werden welche Attribute benötigt werden.

Define the Query Condition(s)

Value types: #BO#(Boolean), #DT#(Date), #GU#(Guid), #IN#(Integer), #XK#(XKey), #XR#(XRef), #AG#(Guid[])

<input checked="" type="checkbox"/>	QA_DOCU_LANGUAGE QA_DOCU_LANGUAGE_STRICT QA_KEY_RESTRICTION	EQ NOT_EQ LIKE NOT_LIKE GE LE GT	<input type="text"/>	Remove0
-------------------------------------	---	--	----------------------	---------

Define the Result Attributes

Link to object RA_CHECK_EXISTENCE_OF_TYPE RA_CHECK_EXISTENCE_OF_TYPEnot RA_LINK_LIST RA_LINK_LIST_ROLE RA_LINK_LIST_ROLE_POS

Abbildung 77: Voest Simple Query Condition & Result Attributes

Auch diese Optionen sind in der neue erstellen Version von Simple Query vorhanden, mit der Ausnahme von den EQ, NOT_EQ, LIKE, NOT_LIKE, GE, LE und GT Auswahlmöglichkeiten. Da diese fast nie verwendet werden, sagte der Auftraggeber, dass diese nicht mehr benötigt werden und es wird nun standardmäßig als LIKE gesucht also. Das bedeutet es werden die Konditionen die eingegebene Suchbedingung enthalten soll.

Conditions:

Namespace

ra

Result attributes:

Changed By
Changed On
Description
FOLDERREF
Link to object

Abbildung 78: Simple Query Conditon & Result Attributes

Auch ist es möglich die gesamte Query in einem XML-Format anzugeben. Dabei muss jedoch immer die gesamte Query eingefügt werden und ist nirgends gespeichert. Dieses Problem ist im neuen Simple Query gelöst und es können vorgefertigte Queries aufgeführt werden, welche in einem *templates* Ordner gespeichert sind.

Query XML

Release:

Show request XML

Show response XML

Use the following XML for the request:

```
<?xml version="1.0" encoding="UTF-8"?>
<generalQueryRequest>
  <types xmlns="">
    <type id="workspace"/>
  </types>
  <qc xmlns="" qcType="S" delMode="N"
  useSyncT="true" cCtxOnly="false">
    <cCtx consider="A" user="s270952"/>
  </qc>
</generalQueryRequest>
```

Abbildung 79: Voest Simple Query XML template

Choose from templates:

Abbildung 80: Simple Query templates

XML-Datei:

```
<?xml version="1.0" encoding="UTF-8"?>
<generalQueryRequest>
  <types xmlns="">
    <type id="XI_TRAFO"/>
  </types>
  <qc xmlns="" qcType="S" delMode="N" useSyncT="true" clCtxOnly="false">
    <clCtx consider="A"/>
    <swcListDef def="W"/>
  </qc>
  <condition xmlns="">
    <elementary>
      <single>
        <key>NAMESPACE</key>
        <val>
          <simple>
            <strg>*KDS-S005*</strg>
          </simple>
        </val>
        <op isCaseIns="false">LIKE</op>
      </single>
    </elementary>
    <elementary>
      <single>
        <key>FILENAME</key>
        <val>
          <simple>
            <strg>*KDS-S005*</strg>
          </simple>
        </val>
        <op isCaseIns="false">LIKE</op>
      </single>
    </elementary>
  </condition>
  <result xmlns="">
    <attrib>NAME</attrib>
    <attrib>NAMESPACE</attrib>
    <attrib>OBJECTID</attrib>
    <attrib>RA_XILINK</attrib>
  </result>
</generalQueryRequest>
```

Abbildung 81: Simple Query XML Datei

Diese XML-Datei beinhaltet ein Template, welches ausgeführt werden kann. Dabei wird als Kondition *Namespace* mit dem Suchbegriff **KDS-S005**, sowie *Filename* mit **KDS-S005** verwendet. Im `<types>` am Anfang der Query werden die Objekt types ausgewählt. In diesem Fall wäre das *XI_TRAFO*, was für Message Mapping steht. Als *Result* sollen *Name*, *Namespace*, *ObjectId* und *RA_XILINK* ausgegeben werden.

Model:

Das Model, in dem die Daten zur Verarbeitung zwischengespeichert werden, ist ein *Map* Objekt, das eine Liste von *Strings* speichert. Ein Map-Objekt ist ein Key-Value-Store, bei dem man den Inhalt zurückbekommt, wenn man den Schlüssel dafür weiß.

```
public class Model {
    private Map<String,List<String>> infos;

    public Model() {
        this.infos = new HashMap<>();
    }

    public Map<String, List<String>> getInfos() {
        return infos;
    }

    public void setInfos(Map<String, List<String>> infos) {
        this.infos = infos;
    }
}
```

Abbildung 82: Simple Query Model

API-Zugriff:

```
import okhttp3.*;
```

Der API-Zugriffe werden in der Klasse *HttpRequest* durchgeführt. In dieser Klasse gibt es die Methode *pullFromDatasource* welche mit dem *OkHttpClient* den Rest Request durchführt. Der Methode wird ein zuvor erstellter XML-String übergeben, welcher dann in den Request eingebaut wird. In diesem XML-String ist die gesamte Query gespeichert.

```
public Model pullFromDatasource(String requestXML, String url) {
    this.url = url;
    OkHttpClient client = new OkHttpClient().newBuilder().build();
    MediaType mediaType = MediaType.parse( $this$toMediaTypeOrNull: "application/x-www-form-urlencoded");
    RequestBody body = RequestBody.create(mediaType,
        content: "qc=All software components&underL=true&useXMLL=true&responseXMLL=true&queryRequestXMLL="
            + requestXML + "&action=Start query");
    Request request = new Request.Builder().url(url).method("POST", body)
        .addHeader(name: "Content-Type", value: "application/x-www-form-urlencoded")
        .addHeader(name: "Authorization", Credentials.basic(username: "username", password: "password")).build();
    Response response = null;
    try {
        response = client.newCall(request).execute();
        return getModelFromResponse(response);
    } catch (IOException e) {
        throw new RuntimeException(e);
    }
}
```

Abbildung 83: Simple Query Api Zugriff

```
import org.jsoup.Jsoup;
import org.jsoup.nodes.Document;
```

Die Antwort der API wird mit der Methode `getModelFromResponse` weiterverarbeitet. Zuerst wird mit Jsoup der XML-String welchen man als Antwort bekommt in ein Dokument von Jsoup umgewandelt, und der Klasse `ModelBuilder` übergeben.

```
public Model getModelFromResponse(Response response) throws IOException {
    assert response.body() != null;
    Document doc = Jsoup.parse(response.body().string());
    ModelBuilder modelBuilder = new ModelBuilder();
    return modelBuilder.buildModel(doc);
}
```

Abbildung 84: Simple Query ModelFromResponse

Model erstellen:

Der `ModelBuilder` wird verwendet um das übergebene `Jsoup Document` in ein eigenes Model Objekt zu verwandeln, dabei werden zuerst die Überschriften welche im `Map` der `Model` als `Key` dienen. Als `Value` der `Map` werden Listen erstellt, welche danach mit den einzelnen Daten befüllt werden.

```
public Model buildModel(Document doc) {
    Model model = new Model();

    // Combine all textarea content and parse it once
    Elements textareaElements = doc.select(cssQuery: "textarea");
    StringBuilder textareaContent = new StringBuilder();
    for (Element textareaElement : textareaElements) {
        textareaContent.append(textareaElement.text());
    }
    doc = Jsoup.parse(textareaContent.toString());

    // Get all elements and header info
    Elements elements = doc.select(cssQuery: "matrix r");
    Elements headerElements = doc.select(cssQuery: "headerInfo colDef def");

    // Get headers and initialize map entries
    List<String> headers = new ArrayList<>();
    for (Element headerElement : headerElements) {
        String header = headerElement.attr(attributeKey: "type");
        map.put(header, new ArrayList<>());
        headers.add(header);
    }

    // Get all objects
    int headerIndex = 0;
    for (Element element : elements) {
        List<String> list = recursiveGetStringFromElement(element);
        for (String value : list) {
            String header = headers.get(headerIndex);
            map.get(header).add(value);
            headerIndex = (headerIndex + 1) % headers.size();
        }
    }
    model.setInfos(map);
    return model;
}
```

Abbildung 85: Simple Query ModelBuilder

Um die Listen zu befüllen, wurde eine Rekursive Methode erstellt, welche den ganzen Baum durchgeht, um alle Objekte herauszubekommen.

```
public List<String> recursiveGetStringFromElement(Element element) {
    List<String> list = new ArrayList<>();
    if (!element.children().isEmpty() && !element.children().get(0).children().isEmpty()) {
        if (element.tagName().equals("simple")) {
            for (Element child : element.children()) {
                list.add(child.text());
            }
        } else {
            StringBuilder combinedText = new StringBuilder();
            for (Element child : element.children()) {
                if (child.hasText()) {
                    combinedText.append(child.text()).append(" | ");
                }
            }
            if (combinedText.length() > 3) {
                list.add(combinedText.substring(0, combinedText.length() - 3));
            }
        }
    } else {
        for (Element child : element.children()) {
            list.addAll(recursiveGetStringFromElement(child));
            addAttributesFromElement(child);
        }
    }
    return list;
}
```

Abbildung 86: Simple Query ModelBuilder Rekursive Methode

Request XML-String Aufbauen:

```
<?xml version="1.0" encoding="UTF-8"?>
<generalQueryRequest>
  <types xmlns="">
    <idHere</types>
  <qc xmlns="" qcType="S" delMode="N" useSyncT="true" clCtxOnly="false">
    <clCtx consider="A"/>
    <swcListDef def="G">
      <swcInfoList>
        <swc id="uIdHere" sp="-1" underL="true" onlyMod="false"/>
      </swcInfoList>
    </swcListDef>
  </qc>
  <condition xmlns="">
    <complex>
      <!--conditionHere-->
    </complex>
  </condition>
  <result xmlns="">
    <attributeHere</result>
</generalQueryRequest>
```

Abbildung 87: Simple Query Standard.xml

Der Request wird aufgebaut, indem eine Standard.xml-Datei verändert wird, in die die benötigten Daten, die von der Website stammen, an den markierten Stellen eingefügt werden.

```
public String createRequestXML(String component, List<String> objectType,
    List<String> condition, List<String> searchField, List<String> resultAttributes) {
    String xmlFilePath = "src/main/resources/standard.xml";
    String requestXML = getStringFromFile(xmlFilePath);

    StringBuilder objectTypeString = new StringBuilder();
    for (String objectType : objectType) {
        objectTypeString.append("<type id=\"" + objectType + "\"/>");
    }
    StringBuilder resultAttributeString = new StringBuilder();
    for (String resultAttribute : resultAttributes) {
        resultAttributeString.append("<attrib>" + resultAttribute + "</attrib>");
    }
    StringBuilder conditionString = new StringBuilder();
    if (condition != null && !condition.isEmpty()) {
        for (int i = 0; i < condition.size(); i++) {
            conditionString.append("<elementary>\n<single>\n<key>" + condition.get(i)
                + "\n</key>\n<val>\n<simple>\n<strg>"
                + searchField.get(i)
                + "\n</strg>\n</simple>\n</val>\n"
                + "<op isCaseIns=\"" + false + "\">LIKE</op>\n</single>\n</elementary>\n");
        }
    }

    requestXML = requestXML.replace(target: "idHere", objectTypeString.toString());
    requestXML = requestXML.replace(target: "uIdHere", component);
    requestXML = requestXML.replace(target: "<!--conditionHere-->", conditionString.toString());
    requestXML = requestXML.replace(target: "attributeHere", resultAttributeString.toString());
    return requestXML;
}
```

Abbildung 88: Simple Query RequestXMLBuilder create Request

Home Seite:

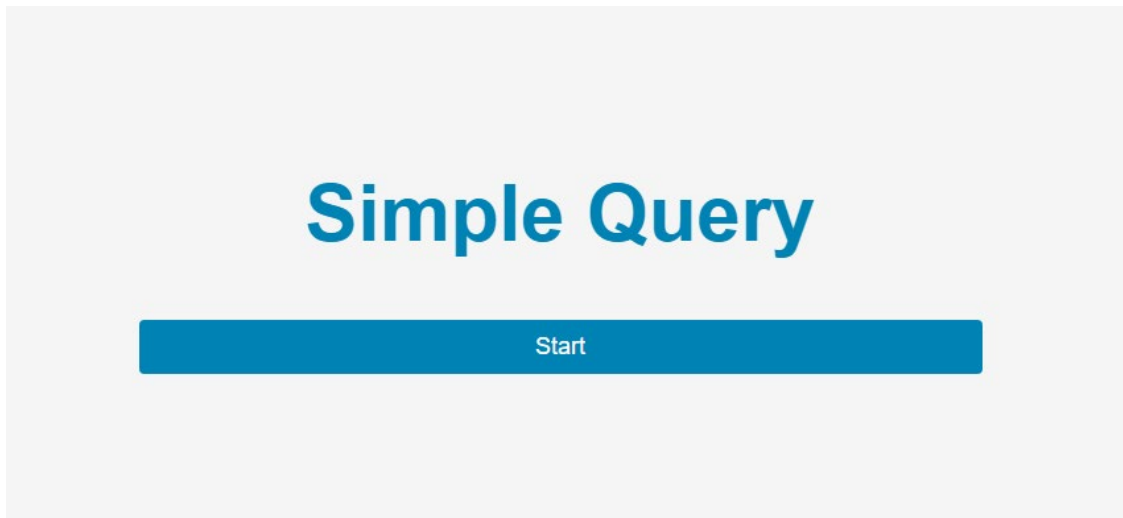


Abbildung 89: Simple Query Home Site

Als erstes wird man von einer einfachen Startseite begrüßt. Diese wird über den link <http://localhost:8080> erreicht.

```
@RestController
public class HomeSite {
    RequestXMLBuilder requestXMLBuilder = new RequestXMLBuilder();

    @GetMapping("/")
    public String homeScreen() {
        StringBuilder html = new StringBuilder("<!DOCTYPE html><html><head><style>" +
            requestXMLBuilder.getStringFromFile("src/main/resources/stylesheets/Home.css")
            + "</style></head><body>");
        html.append("<h1>Simple Query</h1><br><br>");
        html.append("<input type='submit' value='Start' " +
            "onclick='window.location.href='/chooseData?conditionAmount=0';>");
        html.append("</body></html>");
        return html.toString();
    }
}
```

Abbildung 90: Simple Query HomeSite

Nach dieser einfachen Site wird man auf Knopfdruck zur nächsten Seite weitergeleitet, auf der die Zusammenstellung eigener Queries möglich ist.

Query Builder Seite:

Wie bereits erwähnt werden auf dieser Seite die Query zusammengestellt. Je nachdem wie viele Konditionen man haben will wird die Seite mit unterschiedlich Konditionsfeldern geladen. Danach werden die Templates aus dem *templates* Ordner aus gelesen und angezeigt.

```
@GetMapping("/chooseData")
@ResponseBody
public String choosingField(@RequestParam("conditionAmount") int conditionAmount) {
    StringBuilder html = new StringBuilder("<!DOCTYPE html><html><head><style>" +
        requestXMLBuilder.getStringFromFile(filepath: "src/main/resources/stylesheets/Select.css") +
        "</style></head><body>");
    html.append("<h1>Simple Query</h1>");

    // XML File Drop down
    html.append("<form action=\"/data\" id=\"form\">\n" + " <h2>Choose from templates:</h2>\n" +
        " <select id=\"xmlFilePath\" name=\"xmlFilePath\">\n");
    File[] listOfFiles = new File("src/main/resources/templates").listFiles();
    for (File file : listOfFiles) {
        if (file.isFile()) {
            html.append("<option value=\"" + file.getPath() + "\"> " + file.getName() + "</option>\n");
        }
    }
    html.append("</select><input type=\"submit\" value=\"Submit\"></form>");
    html.append("<br><br>");

    html.append(addAllSelections(conditionAmount, formLink: "dataCustom", headline: "Build your own query:"));
    html.append("</body></html>");
    return html.toString();
}
```

Abbildung 91: Simple Query choose Data Site

Von dieser Methode aus wird auch noch eine weitere Methode mit dem Namen *addAllSelections* aufgerufen welche für die Eingaben des Nutzers zuständig ist.

In dieser Methode wird zuerst ein Drop-Down Menu für die Softwarekomponenten erstellt, und ein weiteres für die Objekt Typen. Danach noch eines für die Konditionen und Schlussendlich eines für die Ergebnis Attribute.

```

private String addAllSelections(int conditionAmount, String formLink, String headline) {
    StringBuilder html = new StringBuilder();
    // form
    String xmlFilePath = null;
    html.append("<form action=\"/" + formLink + "\" id=\"" + formLink + "\">\n");
    html.append("<h2>" + headline + "</h2>\n");

    //
    html.append(addSelection(xmlFilePath: "src/main/resources/selectables/Urls.xml",
        name: "url", text: "Servers:", multiple: true));

    //Drop down for Software components
    html.append(addSelection(xmlFilePath: "src/main/resources/selectables/Software components.xml",
        name: "component", text: "Software components:", multiple: false));
    // Drop down for object types
    html.append(addSelection(xmlFilePath: "src/main/resources/selectables/Object types.xml",
        name: "objectType", text: "Object types:", multiple: true));

    //Drop down for conditions
    xmlFilePath = "src/main/resources/selectables/Conditions.xml";
    html.append("<label for=\"condition\">Conditions:</label>\n");
    for (int i = 0; i < conditionAmount; i++) {
        html.append(addCondititon(dropdownOptions(xmlFilePath)));
    }
    html.append("<div class=\"input-buttons\">");
    html.append("<input type=\"button\" value=\"Add\" " +
        "onclick=\"window.location.href='/chooseData?conditionAmount="
        + (conditionAmount + 1) + "';\">");
    html.append("<input type=\"button\" value=\"Delete\" " +
        "onclick=\"window.location.href='/chooseData?conditionAmount="
        + (conditionAmount - 1) + "';\">");
    html.append("</div>");
    html.append("<div id=\"dynamicElements\"></div>");
    html.append("<br><br>");

    // Drop down for result attributes
    html.append(addSelection(xmlFilePath: "src/main/resources/selectables/Result attributes.xml",
        name: "resultAttribute", text: "Result attributes:", multiple: true));

    // submit button & end form
    html.append("<input type=\"submit\" value=\"Submit\"></form>");

    return html.toString();
}

```

Abbildung 92: Simple Query addAllSelections

Da die Anzahl an Konditionen und Ergebnis Attributen variable ist, gibt es jeweils eine Methode, welche für das hinzufügen dieser Komponenten ist. Diese Methoden können beliebig oft aufgerufen werden.

```

public String addSelection(String xmlFilePath, String name, String text, boolean multiple) {
    StringBuilder html = new StringBuilder();
    html.append("<label for=\"" + name + "\">" + text + "</label>\n");
    if (multiple) html.append("<select id=\"" + name + "\" name=\"" + name + "\" multiple>\n");
    else html.append("<select id=\"" + name + "\" name=\"" + name + "\">\n");
    html.append(dropdownOptions(xmlFilePath));
    html.append("</select><br><br>");
    return html.toString();
}

public String addCondition(String dropdownOptions) {
    StringBuilder html = new StringBuilder();
    // Drop down for conditions
    html.append("<div>");
    html.append("<select id=\"condition\" name=\"condition\">\n");
    html.append(dropdownOptions);
    html.append("</select>");
    // Search field
    html.append("<input type=\"text\" id=\"searchField\" name=\"searchField\" " +
        "placeholder=\"Enter search term\">\n");
    html.append("</div>");
    return html.toString();
}

```

Abbildung 93: Simple Query Komponenten hinzufügen

Weiters gibt es eine Methode die für das hinzufügen der Drop-Down Optionen ist. Diese wird sowohl für die *Selections* als auch für die *Conditions* verwendet.

```

public String dropdownOptions(String xmlFilePath) {
    StringBuilder html = new StringBuilder();
    try {
        File xmlFile = new File(xmlFilePath);
        Document doc = Jsoup.parse(xmlFile, charsetName: "UTF-8");
        Elements optionElements = doc.select(cssQuery: "option");
        for (Element optionElement : optionElements) {
            html.append(optionElement);
        }
    } catch (IOException e) {
        e.printStackTrace();
    }
    return html.toString();
}

```

Abbildung 94: Simple Query dropdown Optionen

Damit ist diese Seite auch schon komplett und die Query kann ausgeführt werden, was einen API-Aufruf ausführt.

Simple Query

Choose from templates:

ids.xml ▼

Submit

Build your own query:

Servers:

vadsvexscs
vadsv2xscs

Software components:

Local ▼

Object types:

Context Object
Data Type
Data Type Enhancement
Documentation
External Definition

Conditions:

Add **Delete**

Result attributes:

Changed By
Changed On
Description
FOLDERREF
Link to object

Submit

Abbildung 95: Simple Query ChooseDataSite

Daten Seite:

Auf der *DataSite* werden entweder die Daten eines Server, oder Daten von zwei Servern angezeigt.

Wenn eine Template verwendet wird, wird die Methode mit der URL */data* aufgerufen:

```
@GetMapping(⊕"/data")
@ResponseBody
public String showXMLData(@RequestParam("xmlFilePath") String xmlFilePath) {
    String url = "http://vadsvexscs.voestalpine.root.local:57600/rep/support/SimpleQuery";
    HTTPRequest httpRequest = new HTTPRequest();
    Model model = httpRequest.pullFromDatasource(requestXMLBuilder.getStringFromFile(xmlFilePath), url);
    models.add(model);
    return showDataByModels();
}
```

Abbildung 96: Simple Query template Data

Sollte eine eigene Query erstellt werden, wird die Methode mit der URL */dataCustom*.

```
@GetMapping(⊕"/dataCustom")
@ResponseBody
public String showXMLDataCustom(@RequestParam("urls") List<String> urls, @RequestParam("component") String component,
    @RequestParam("objectType") List<String> objectType,
    @RequestParam(value = "condition", required = false) List<String> condition,
    @RequestParam(value = "searchField", required = false) List<String> searchField,
    @RequestParam("resultAttribute") List<String> resultAttributes) {
    HTTPRequest httpRequest = new HTTPRequest();
    for (String url : urls) {
        models.add(httpRequest.pullFromDatasource(requestXMLBuilder.createRequestXML(component, objectType, condition, searchField, resultAttributes), url));
    }
    return showDataByModels();
}
```

Abbildung 97: Simple Query Custom Data

Am Ende beider Methoden wird die Methode *showDataByModels* aufgerufen. Diese Methode baut den HTML-Code für die Daten in den Models zusammen und zeigt diese an.

```

public String showDataByModels() {
    StringBuilder html = new StringBuilder("<!DOCTYPE html><html><head><style>" +
        requestXMLBuilder.getStringFromFile("filepath: " + "src/main/resources/stylesheets/TableView.css")
        + "</style></head><body><table id='dataTableStyle'>");
    //Button to compare data
    html.append("<input type='submit' value='Compare' onclick='window.location.href='/compareData';>");

    for (Model model : models) {
        Map<String, List<String>> infos = model.getInfos();
        List<String> headerList = new ArrayList<>(infos.keySet());
        int rows = infos.get(headerList.get(0)).size();

        html.append("<tr>");
        for (String header : headerList) {
            html.append("<th>").append(header).append("</th>");
        }
        html.append("</tr>");

        for (int columnIndex = 0; columnIndex < rows; columnIndex++) {
            html.append("<tr>");
            for (String header : headerList) {
                List<String> columnValues = infos.get(header);
                String columnValue = columnIndex < columnValues.size() ? columnValues.get(columnIndex) : "";
                html.append("<td>").append(columnValue).append("</td>");
            }
            html.append("</tr>");
        }
    }
    html.append("</table></body></html>");
    return html.toString();
}

```

Abbildung 98: Simple Query display Data

Compare				
Print	Raw	WSDL	Software Component Version	Key
	R		Z DKI, 1.0 of voestalpine.com	External Definition ED_RolSachnummern_DKI I
Print	Raw	WSDL	Software Component Version	Key
	R		Z DKI, 1.0 of voestalpine.com	External Definition ED_RolSachnummern_DKI I

Abbildung 99: Simple Query DataSite

Die Daten der Query werden angezeigt und können verglichen werden. In diesem Beispiel liefert die Query nur ein Ergebnis.

Vergleich Seite:

Es wird eine Liste mit den doppelten Infos erstellt und diese wird von der originalen Liste abgezogen, so bleiben nur noch die übrig welche nicht gleich sind. Und das können dann nur die Elemente mit unterschiedlicher Version oder welche die am anderen Server nicht vorhanden sind.

```

@GetMapping("/compareData")
@ResponseBody
public String showComparedData() {
    Map<String, List<String>> matchingInfos = new HashMap<>();

    for (Model model : models) {
        Map<String, List<String>> infos = model.getInfos();
        List<String> headerList = new ArrayList<>(infos.keySet());
        int rows = infos.get(headerList.get(0)).size();

        for (int columnIndex = 0; columnIndex < rows; columnIndex++) {
            for (String header : headerList) {
                List<String> columnValues = infos.get(header);
                String columnValue = columnIndex < columnValues.size() ? columnValues.get(columnIndex) : "";
                if (matchingInfos.containsKey(header)) {
                    matchingInfos.get(header).add(columnValue);
                } else {
                    List<String> list = new ArrayList<>();
                    list.add(columnValue);
                    matchingInfos.put(header, list);
                }
            }
        }
    }
}

```

Abbildung 100: Simple Query matching Infos

Auf der Vergleichsseite werden diejenigen Daten angezeigt, die nicht doppelt vorkommen. Das heißt in diesem Beispiel, dass überhaupt keine Daten angezeigt werden, da sie doppelt waren.

Print	Raw	WSDL	Software Component Version	Key

Abbildung 101: Simple Query Compared Site

12 RISIKOMANAGEMENT

Um Risiken, welche das Projekt gefährden könnten, zu minimieren, sollte Risikomanagement genutzt werden. Mit Hilfe des Risikomanagements können potenzielle Hindernisse frühzeitig identifiziert werden und entsprechende Maßnahmen frühzeitig ergriffen werden. Es steigert die Erfolgchancen des Projekts in den unerwünschten Konsequenzen so gut wie möglich vermieden werden. Zudem ermöglicht es eine strukturierte Herangehensweise an Probleme, wodurch Entscheidungen, bezüglich der Probleme, leichter getroffen werden können. Auch die Kommunikation im Team und mit Stakeholdern wird durch die Ausarbeitung der Risiken gefördert. Insgesamt trägt das Risikomanagement dazu bei, dass Projekte mit einer erhöhten Wahrscheinlichkeit erfolgreich abgeschlossen werden können.

12.1 Risiken

Projektumfang ungenau definiert

Projektphase	Initialisierung
Eintrittswahrscheinlichkeit	Mittelmäßig
Auswirkungen	<ul style="list-style-type: none">• Unklare Ziele• Führt zu Missverständnissen• Anforderungen nicht erfüllt
Maßnahmen	<ul style="list-style-type: none">• Regelmäßige Kommunikation mit Stakeholdern• Detaillierte Planung

Tabelle 9: Risiko Projektumfang

Erfahrung der Projektmitglieder ist unzureichend

Projektphase	Initialisierung, Implementierung
Eintrittswahrscheinlichkeit	Mittelmäßig
Auswirkungen	<ul style="list-style-type: none">• Qualitätsprobleme• Verzögerungen
Maßnahmen	<ul style="list-style-type: none">• Klare Rollen und Verantwortlichkeiten• Realistische Projektplanung

Tabelle 10: Risiko Projektmitglieder Erfahrung

Fehler im Zeitmanagement

Projektphase	Initialisierung
Eintrittswahrscheinlichkeit	Niedrig
Auswirkungen	<ul style="list-style-type: none">• Qualitätsverlust• Mangelnde Flexibilität
Maßnahmen	<ul style="list-style-type: none">• Pufferzeiten einplanen

Tabelle 11: Risiko Zeitmanagement

Mangelnde Teamkoordination

Projektphase	Implementierung
Eintrittswahrscheinlichkeit	Niedrig
Auswirkungen	<ul style="list-style-type: none">• Ineffizienz
Maßnahmen	<ul style="list-style-type: none">• Verantwortlichkeitsklärung• Klare Kommunikationswege• Aufgaben klar verteilen

Tabelle 12: Risiko Teamkoordination

Ausfall eines Projektmitgliedes

Projektphase	Alle
Eintrittswahrscheinlichkeit	Niedrig
Auswirkungen	<ul style="list-style-type: none">• Verzögerungen• Erhöhter Arbeitsdruck
Maßnahmen	<ul style="list-style-type: none">• Flexibilität im Zeitplan• Backup-Plan

Tabelle 13: Risiko Ausfall Projektmitglied

Ausfall des Auftraggebers

Projektphase	Alle
Eintrittswahrscheinlichkeit	Niedrig
Auswirkungen	<ul style="list-style-type: none">• Kommunikationsprobleme• Verzögerungen• Entscheidungsunsicherheit
Maßnahmen	<ul style="list-style-type: none">• Ersatztermine planen• Ersatzkontakte der Firma

Tabelle 14: Risiko Ausfall Auftraggeber

Auftraggeber benötigt das Projekt nicht mehr

Projektphase	Alle
Eintrittswahrscheinlichkeit	Niedrig
Auswirkungen	<ul style="list-style-type: none">• Kein Abnehmer
Maßnahmen	<ul style="list-style-type: none">• Verhältnis zu Product Owner verbessern

Tabelle 15: Risiko Auftraggeber springt ab

Fehlender Zugang zu benötigten Ressourcen

Projektphase	Implementierung
Eintrittswahrscheinlichkeit	Hoch
Auswirkungen	<ul style="list-style-type: none">• Verzögerungen• Unvollständige Umsetzung• Erhöhte Belastung des Projektteams
Maßnahmen	<ul style="list-style-type: none">• Kommunikation der benötigten Ressourcen an den Product Owner

Tabelle 16: Risiko fehlende Ressourcen

12.2 Eingetretene Risiken

Kein Projekt ist fehlerfrei, und so auch dieses nicht. Folgende Risiken sind während der Arbeit am Projekt eingetreten und haben den zeitlichen Verlauf der Diplomarbeit beeinflusst.

Fehlender Zugang zu benötigten Ressourcen

Beim Erstellen von Zugängen, um auf die benötigten Ressourcen zugreifen zu können entstanden vorerst einige Probleme. Aufgrund der in der Voest vorhandenen Richtlinien zum Schutz der Sicherheit im Unternehmen entstanden einige Probleme. Wir mussten einige Zeit warten, um Zugriff auf die benötigten Softwareprodukte zu bekommen, wodurch der Zeitdruck in der Implementierungsphase größer wurde als zunächst angenommen.

Ausfall des Auftraggebers

Ein weiteres eingetretenes Risiko ist der Ausfall eines Auftraggebers, oder besser gesagt, von einem davon. Denn wir hatten zwei verschiedene Projekte zum selben Thema geplant. Jedoch konnte eines dieser Teilprojekte gar nicht umgesetzt werden, da die Person, die es in Auftrag geben wollte, keine Zeit für die Erklärung des Problems hatte. Daraufhin hat unsere Ansprechperson in der Voest ein alternatives Projekt für uns organisiert, welches dieses ursprünglich geplante Projekt ersetzte.

12.3 Fazit aus den Problemen

Zusammenfassend lässt sich sagen, dass selbst bei sorgfältig geplanten Projekten nicht immer alles rund läuft. Der fehlende Zugang zu den benötigten Ressourcen und der Ausfall eines Auftraggebers waren Herausforderungen, die zwar den zeitlichen Verlauf beeinflussten, jedoch das Projekt immer noch bewältigbar machten. Diese Herausforderung zeigte uns, wie wichtig die Flexibilität in einem Projekt ist, und dass man immer auf unvorhergesehene Schwierigkeiten vorbereitet sein sollte. Abschließend sollte noch gesagt werden, dass wir aus den eingetretenen Risiken gelernt haben und diese Erfahrung in zukünftige Projekte mitnehmen, um ähnliche Probleme zu vermeiden.

13 WEITERE BEISPIELPROGRAMME

13.1 E-Mail Bot

13.1.1 Benutzung

Das Beispielprogramm E-Mail Bot dient dazu, Anfragen von Weinbroschüren entgegenzunehmen, die an unsere festgelegte E-Mail-Adresse gesendet werden. Nachdem einige dieser Anfragen gesammelt wurden, kann jemand einen Prozess per Knopfdruck starten und die E-Mails gesammelt aussenden. (Um das Programm benutzen zu können, wird eine E-Mail benötigt, mit der man die E-Mails versendet.)

Die E-Mail, die der Anfragende bekommt, sieht am Ende wie folgt aus:

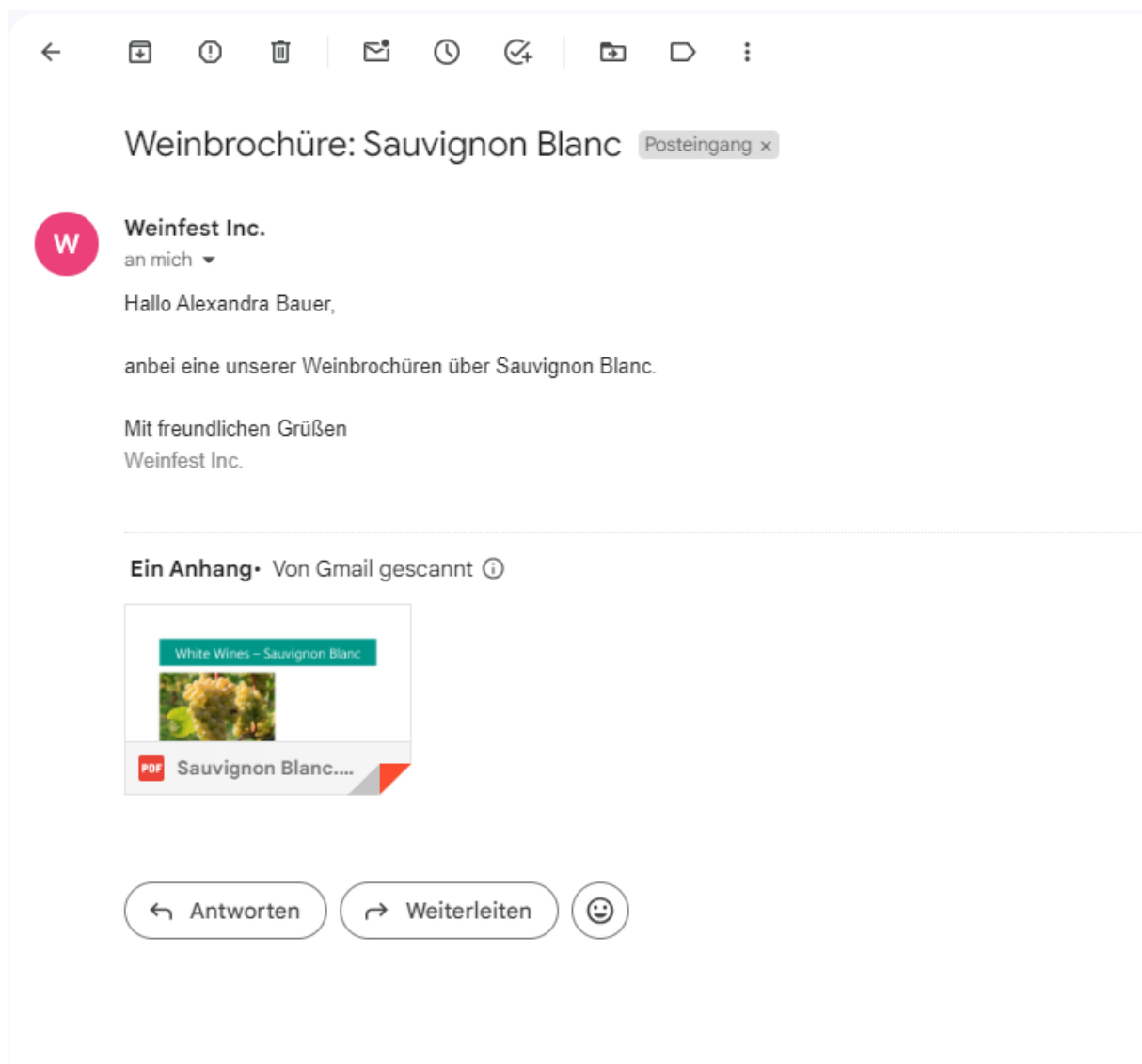


Abbildung 102: E-Mail vom E-Mail Bot

13.1.2 Projektstruktur

Der E-Mail-Bot besteht prinzipiell aus zwei verschiedenen UiPath Studio Prozessen:

- E-Mail Bot Request
- E-Mail Bot Response

Zusätzlich gibt es noch zwei verschiedene Excel Dateien:

- E-Mail Adressen.xlsx
- Verfügbare_Weinbroschüren.xlsx

Des Weiteren gibt es noch zwei Weinbroschüren als Beispiel:

- Chardonnay.pdf
- Sauvignon Blanc.pdf

Die Weinbroschüren werden mithilfe des Response Prozesses als Anhang bei der E-Mail versendet. Für das Versenden der E-Mails wurde zudem eine Test-E-Mail erstellt:

- weinfestinc@gmail.com

Der Request Prozess kümmert sich bei einer Weinanfrage, dass er die anfragende E-Mail-Adresse inklusive des Namens die bei der E-Mail hinterlegt ist und natürlich die gewollte Broschüre in eine Excel Datei („E-Mail Adressen.xlsx“) überträgt.

Wenn sich mehrere Anfragen in der Excel Datei befinden, kann ein User nun den Response Prozess starten. Dieser liest die Excel Datei mit den jeweiligen Anfragen aus und versendet dementsprechend eine E-Mail an den Anfragenden. Um die E-Mails nicht doppelt zu versenden, werden alle Anfragen gelöscht, nachdem der Prozess abgeschlossen wurde und die Datei kann erneut befüllt werden.

13.1.3 Implementierung

E-Mail Bot Request

Als erstes muss man die E-Mail definieren, die man verwenden möchte, um die Weinbroschüren zu erhalten. (In unserem Fall ist die verwendete E-Mail-Adresse *weinfestinc@gmail.com*.) Da wir immer auf die neueste E-Mail schauen wollen, die gerade eingetroffen ist, können wir die bereits vordefinierte Aktivität „Get Newest Email“ verwenden. Zusätzlich darf man nicht vergessen, den gewollten Ordner auszuwählen. Das wir die E-Mail bearbeiten wollen, die neu in unseren Posteingang gekommen ist, verwendet man für dieses Feld „Inbox“.

(Die E-Mail wird in die Variable *email* gespeichert.)

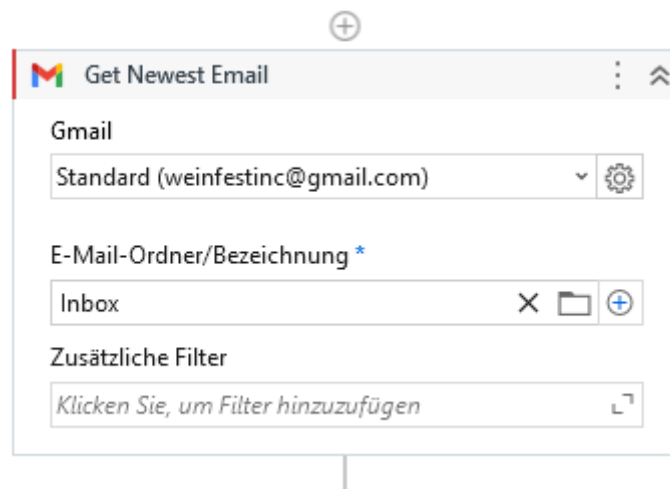


Abbildung 103: E-Mail Bot Request: Get Newest Email

Als nächstes greifen wir auf die Excel Datei zu, in die das Programm die E-Mail-Adressen speichern soll („E-Mail Adressen.xlsx“). Zudem geben wir für die Datei eine Referenzierung an, damit UiPath Studio weiß, um welche Excel Datei es sich handelt und man nicht überall den Dateipfad angeben muss.

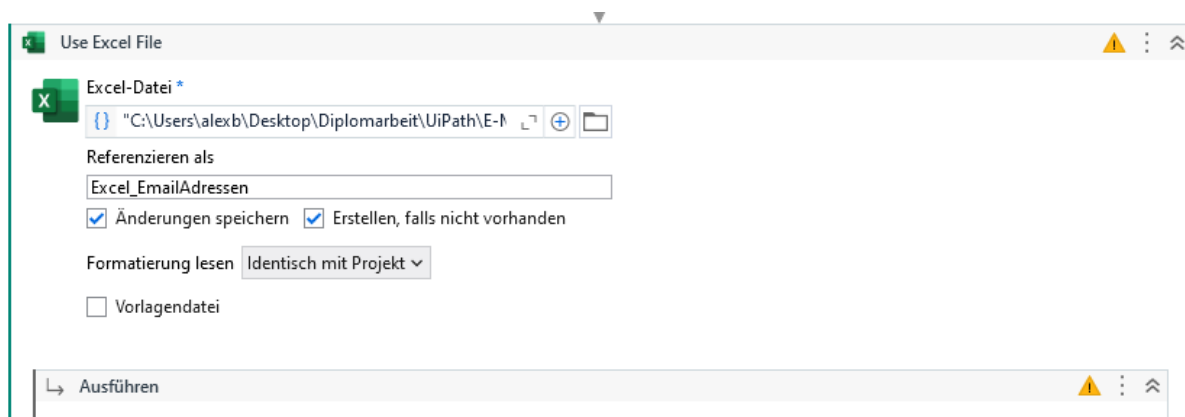


Abbildung 104: E-Mail Bot Request: Zugriff auf "E-Mail Adressen.xlsx"

Um nun eine neue Zeile in der Excel Datei einfügen zu können, erstellt man eine Data Table mithilfe der Aktivität „Build Data Table“. (Die hierbei kreierte Data Table wird der Variable *datatable* zugewiesen.) Man muss hierbei beachten, dass die Überschriften gleich benannt werden, wie in der Excel Datei und bei diesem Schritt noch keine Zeilen eingefügt werden.

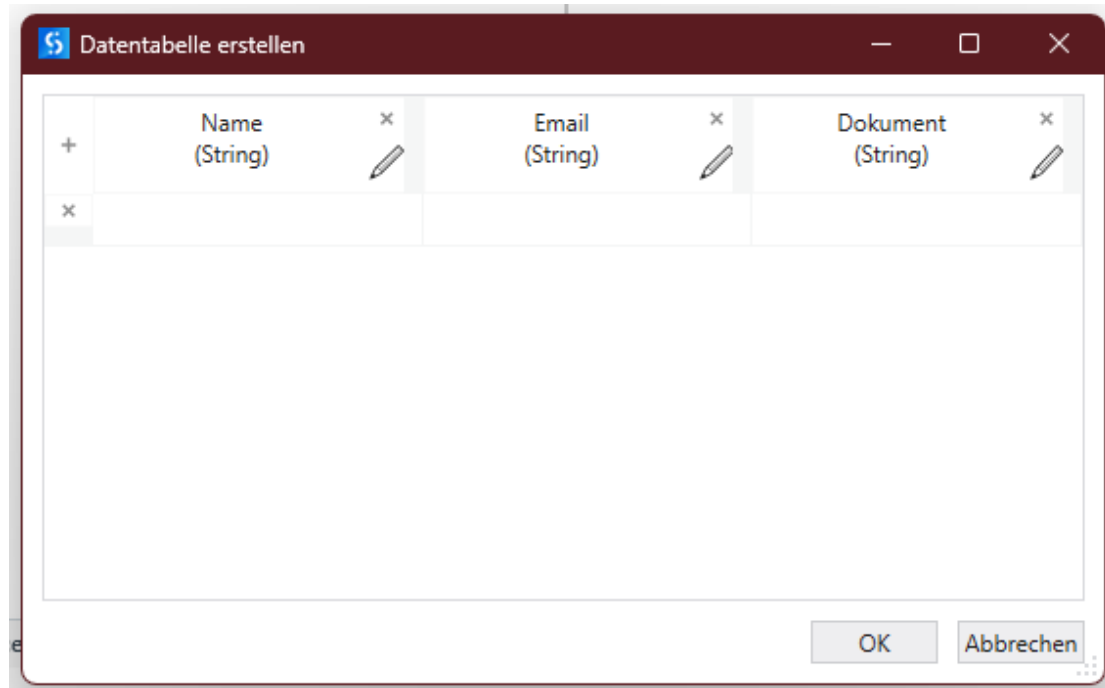


Abbildung 105: E-Mail Bot Request: Build Data Table Fenster

A	B	C
Name	Email	Dokument
Alexander Bense	alex.bense4205@gmail.com	Chadessen

Abbildung 106: E-Mail Bot Request: Excel Tabellen Überschriften

Als nächstes kreiert man eine neue Variable. Diese Variable heißt in unserem Fall *newRow*. Man definiert *newRow* als DataRow und weist ihr folgenden Wert zu:

```
datatable.NewRow
```

Man weist *newRow* deshalb diesen Wert zu, damit sie weiß welche Spalten ihr zur Verfügung stehen.

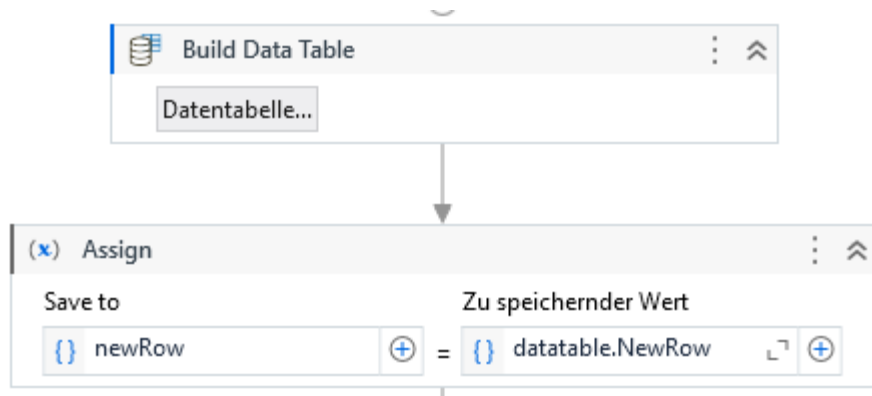


Abbildung 107: E-Mail Bot Request: newRow

Nun können wir bereits die ersten beiden Felder zuordnen. Diese Felder sind der Name und die E-Mail-Adresse desjenigen, der uns die Anfrage geschickt hat.

Um der Variable *newRow* zu sagen, zu welcher Spalte es zugeordnet gehört, schreibt man wie folgt: (Statt *<Spalte>* wird natürlich der richtige Spaltenname eingefügt. In diesem Fall ist dies zuerst „Name“ und dann „Email“.)

```
newRow ("Name")
```

Dem Namen weist man dann Folgenden Wert zu (Dies ist der Name, der bei der E-Mail vom E-Mail-Besitzer hinterlegt wurde.):

```
email.From.DisplayName
```

Die E-Mail weist man mit Folgendem Wert zu:

```
email.SenderEmailAddress
```

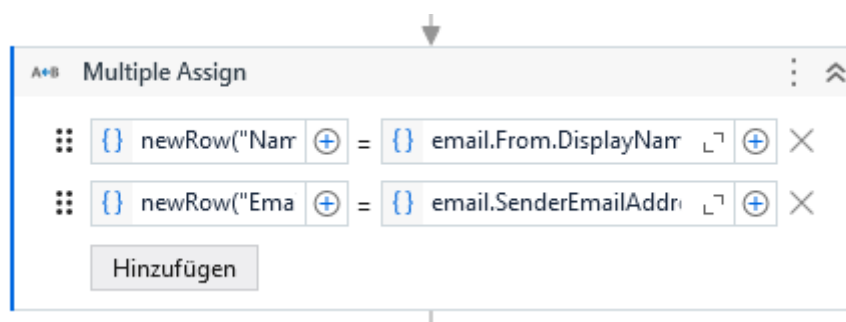


Abbildung 108: E-Mail Bot Request: Multiple Assign für "newRow"

Um nun festzustellen, welche Weinbroschüre der Anfragende möchte, überprüft man die E-Mail, darauf, ob die verlangte E-Mail einen Wein beinhaltet, von dem wir auch eine Broschüre haben. Dies tut man, indem man zuerst in die jeweilige Excel Datei „verfügbare_Weinbroschüren.xlsx“ geht (Referenz: *Excel_Broschüren*) und dort Reihe für Reihe überprüft, ob der Wein enthalten ist.

Um Reihe für Reihe durchzugehen, nimmt man zuerst die Aktivität „For Each Excel Row“. Diese Aktivität agiert, wie der Name bereits verrätet, als For Each-Schleife für die einzelnen Reihen.

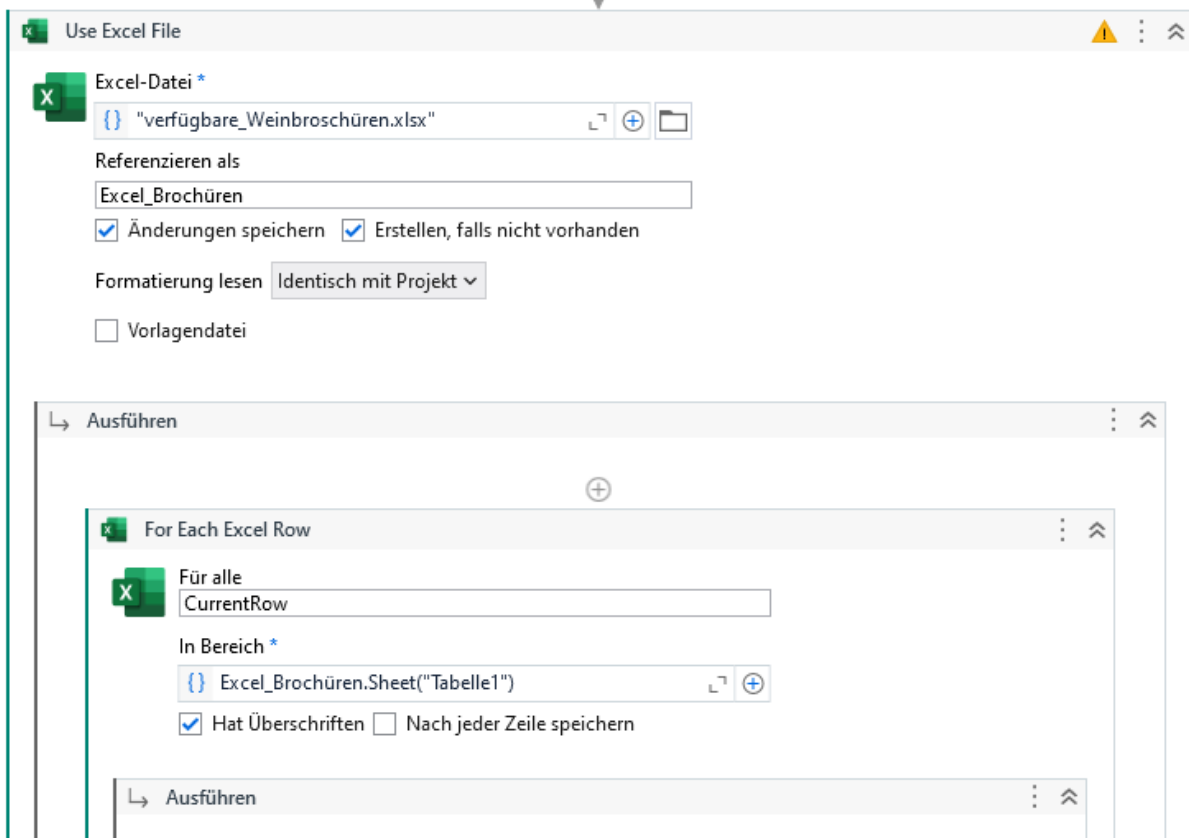
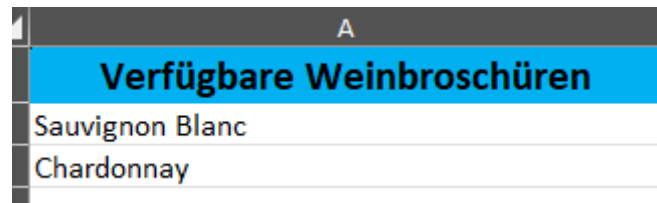


Abbildung 109: E-Mail Bot Request: For Each Excel Row ("verfügbare_Weinbroschüren")

In der For Each-Schleife gibt es nun eine if-Verzweigung, die überprüft, ob der die E-Mail einen Wein enthält, für den eine Broschüre existiert. Dies tut die if-Verzweigung mit folgender Bedingung:

```
email.Body.Contains(CurrentRow.Item("Verfügbare  
Weinbroschüren"))
```

Excel Datei „verfügbare_Weinbroschüren.xlsx“:



Verfügbare Weinbroschüren
Sauvignon Blanc
Chardonnay

Abbildung 110: E-Mail Bot Request: Excel Datei "verfügbare_Weinbroschüren.xlsx"

Wenn man den Wein im Excel nun gefunden hat, wird der Variable newRow für die Spalte „Dokument“, der Name der Weinbroschüre mithilfe einer Assign-Aktivität wie folgt zugewiesen:

```
CurrentRow.Item("Verfügbare Weinbroschüren")
```

Da wir nun alle Daten beisammenhaben, wird die neue Reihe der Data Table hinzugefügt. Dies tut man mit der „Add Data Row“-Aktivität. Dies sieht dann wie folgt aus:

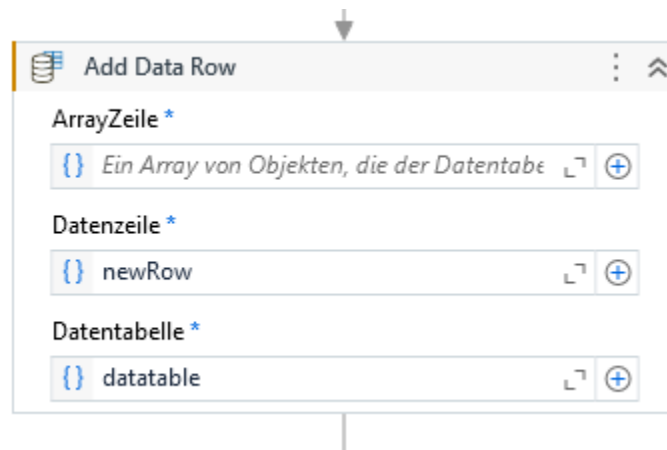


Abbildung 111: E-Mail Bot Request: Add Data Row

Danach werden die gesammelten Daten mithilfe der Data Table der Excel Datei hinzugefügt und die Datei dann gespeichert. Dies sieht wie folgt aus:

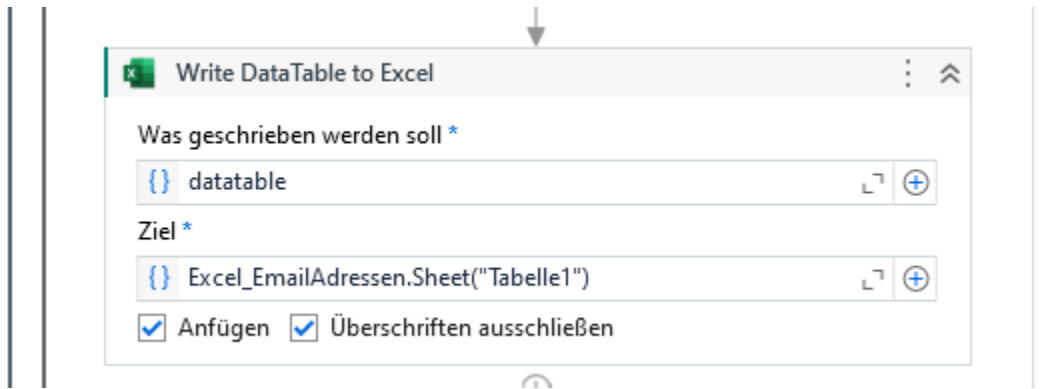


Abbildung 112: E-Mail Bot Request: Write DataTable to Excel

Wenn man nun schon einige Anfragen per E-Mail bekam, sieht die Excel Datei „E-Mail Adressen.xlsx“ möglicherweise wie folgt aus:

	A	B	C
	Name	Email	Dokument
	Alexandra Bauer	alex.bauer4205@gmail.com	Chardonnay
	Alexandra Bauer	alex.bauer4205@gmail.com	Sauvignon Blanc
	Hildegard Musterhaufen	suberbi27@gmail.com	Chardonnay

Abbildung 113: E-Mail Bot Request: mögliche Excel Datei mit Anfragen ("E-Mail Adressen.xlsx")

E-Mail Bot Response

Wenn man bereits einige Anfragen für Weinbroschüren in der Excel Datei hat, kann man mithilfe vom Response Prozess, nun eine Antwort senden.

Man fängt hierbei natürlich damit an, dass man auf die Datei mit den Anfragen („E-Mail Adressen.xlsx“) zugreift (Referenz: Excel). Als nächstes geht man die Excel Datei nun mithilfe einer For Each-Schleife für die Reihen, Reihe für Reihe die Datei durch. Mithilfe einer if-Verzweigung überprüft man in der Schleife nun, dass die Felder „Email“ und „Dokument“ in der Reihe nicht leer ist.

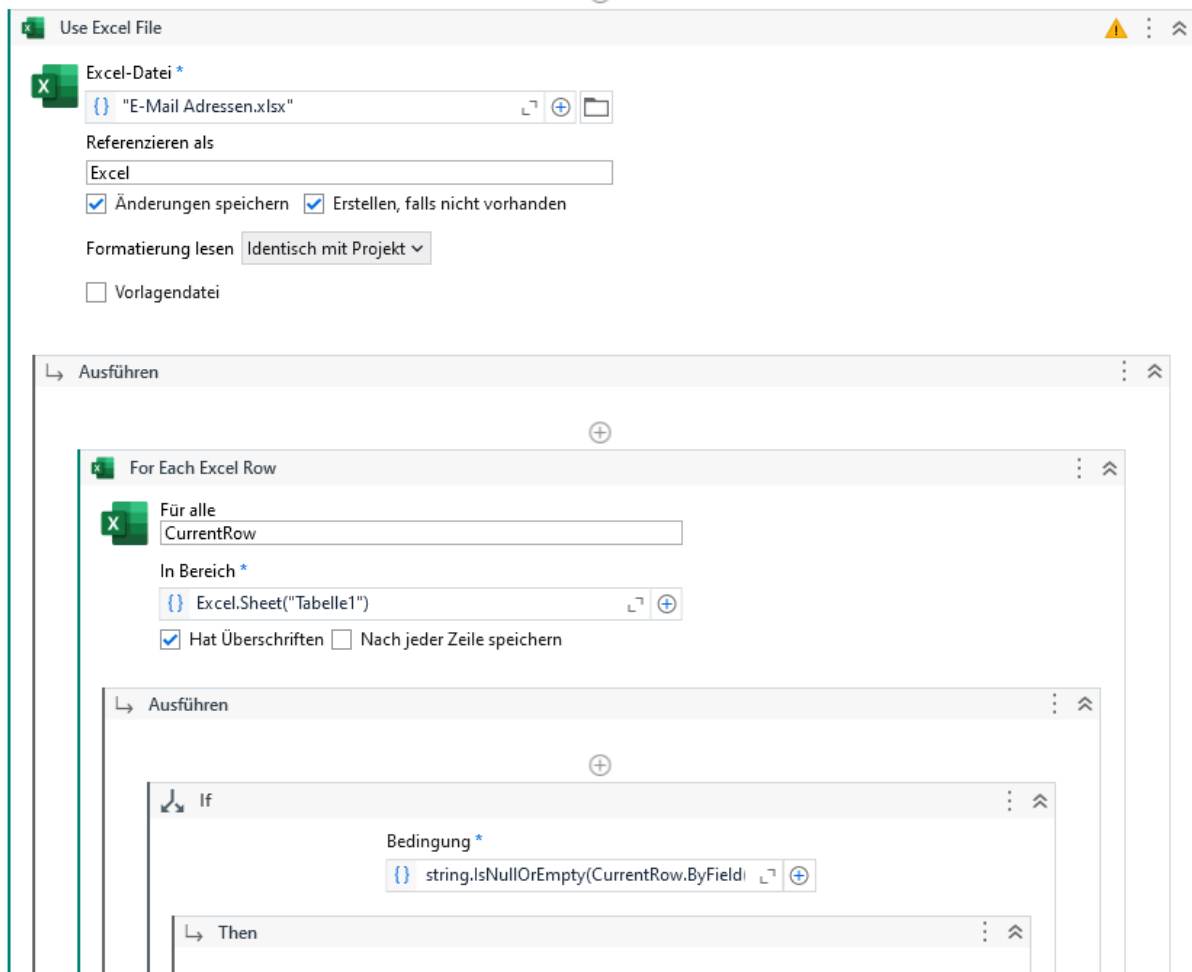


Abbildung 114: E-Mail Bot Response: Excel Datei Eintritt inkl. Schleife von "E-Mail Adressen.xlsx"

Die Bedingungen der if-Verzweigung werden mithilfe des Bedingungsgenerator von UiPath Studio generiert, der wie folgt aussieht:

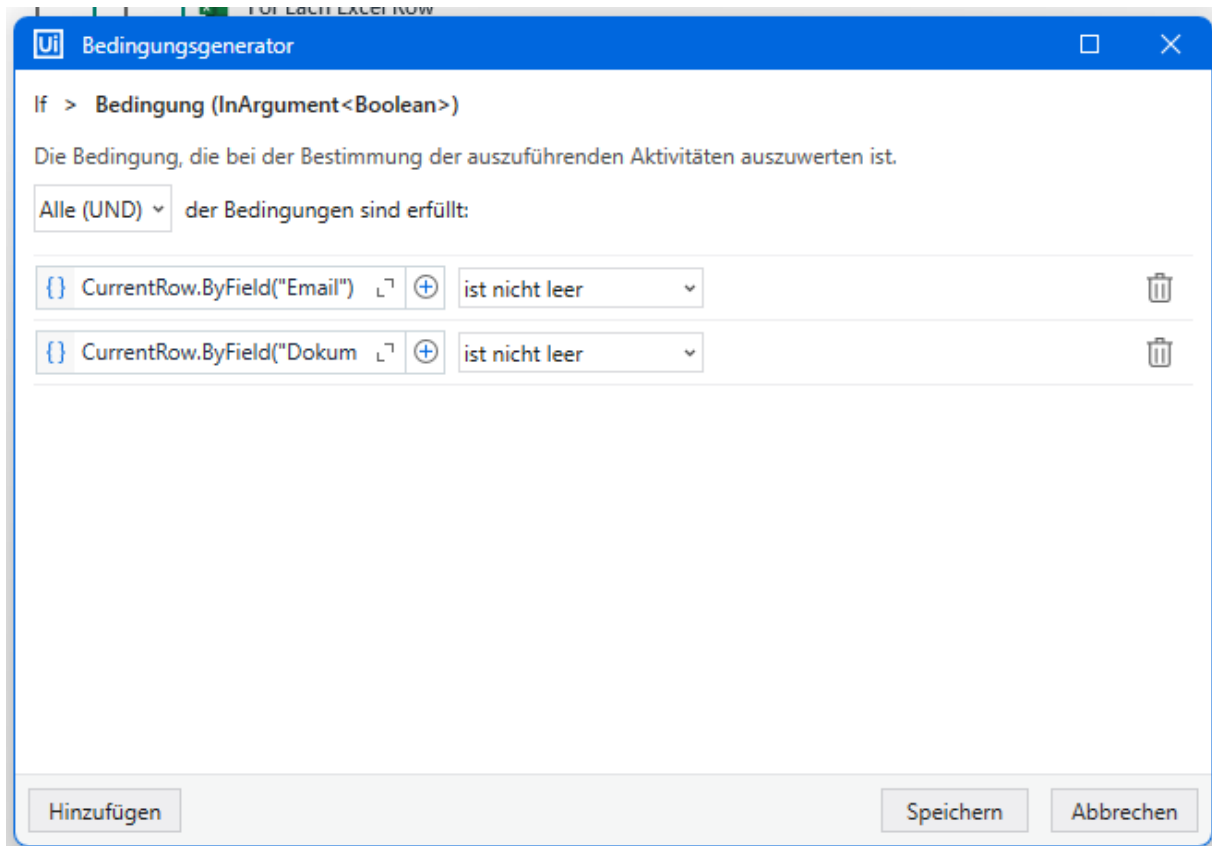


Abbildung 115: E-Mail Bot Response: Bedingungsgenerator

Wenn die if-Verzweigung zutrifft, wird eine E-Mail mit den Informationen der momentanen Reihe versendet. Dies tut man mithilfe der „Use Gmail“-Aktivität, wo man festlegt welche E-Mail man verwenden möchte und kann auf diese referenzieren. (verwendete E-Mail im Prozess: weinfestinc@gmail.com; Referenz: Gmail)

Als nächstes kann man mit der „Send Email“-Aktivität die E-Mails endgültig zusammenbauen und versenden. Das Konto, mit dem wir die E-Mail versenden ist logischerweise, das Konto, das wir bei der „Use Gmail“-Aktivität angegeben haben. Den Empfänger der E-Mail kann man mithilfe von folgendem Befehl aus der Excel Tabelle auslesen (E-Mail steht an zweiter Position und man zählt bei den Indexen immer von 0 weg. Aus diesem Grund wird hier 1 für den Index verwendet.)

```
CurrentRow.ByIndex(1)
```

Als nächstes definieren wir den Betreff. Der Betreff soll am Ende „Weinbroschüre: <Dokument>“ und man schreibt in UiPath Studio in Betreff den folgernden Befehl:

```
"Weinbrochüre: " + CurrentRow.ByIndex(2)
```

Des Weiteren soll die E-Mail natürlich einen Body/Hauptteil haben. Diesen kann man, wenn man auf HTML einstellt und auf den Button „Editor öffnen“ drückt, beliebig definieren. Der Body wurde momentan wie folgt konfiguriert:

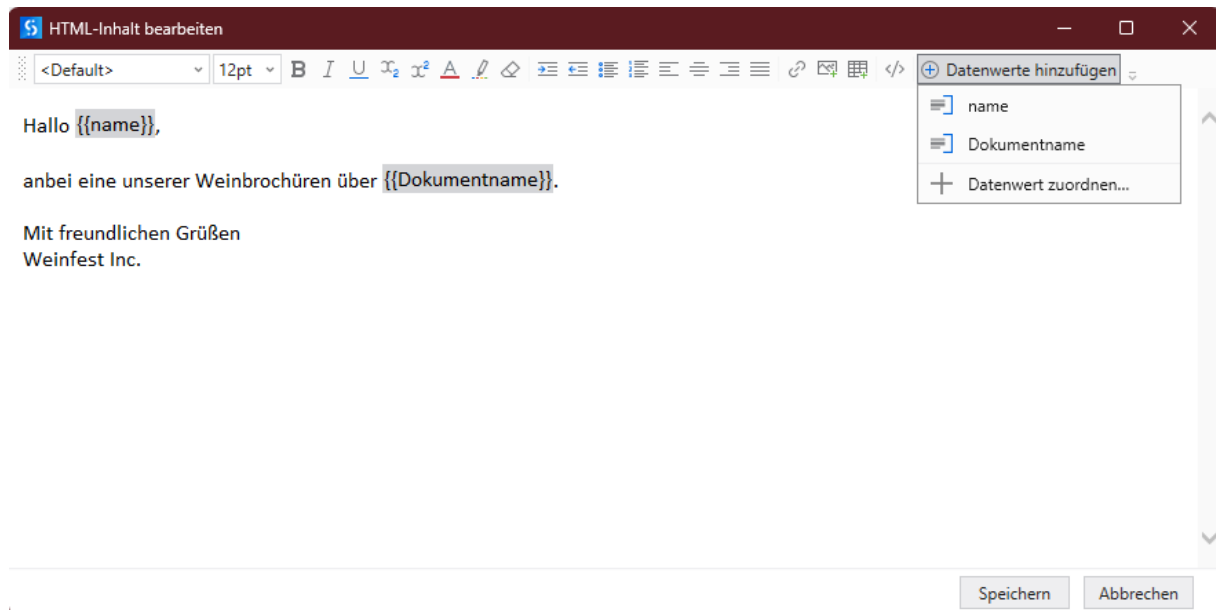


Abbildung 116: E-Mail Bot Response: E-Mail Body

Wie man im Screenshot erkennen kann, sind der Name des Empfängers und der Dokumentenname grau hinterlegt und in zwei geschwungenen Klammern eingeschlossen. Dies ist der Fall, da der Name des Empfängers und das verlangte Dokument natürlich variiert und somit dynamisch zugewiesen werden muss. Dies kann man mithilfe sogenannter Datenwerte. Der Datenwerteditor sieht in UiPath Studio wie folgt aus (Hierbei wurden bereits die Variablen *name* und *Dokumentenname* mit den jeweiligen Werten aus der Excel Datei zugewiesen.).

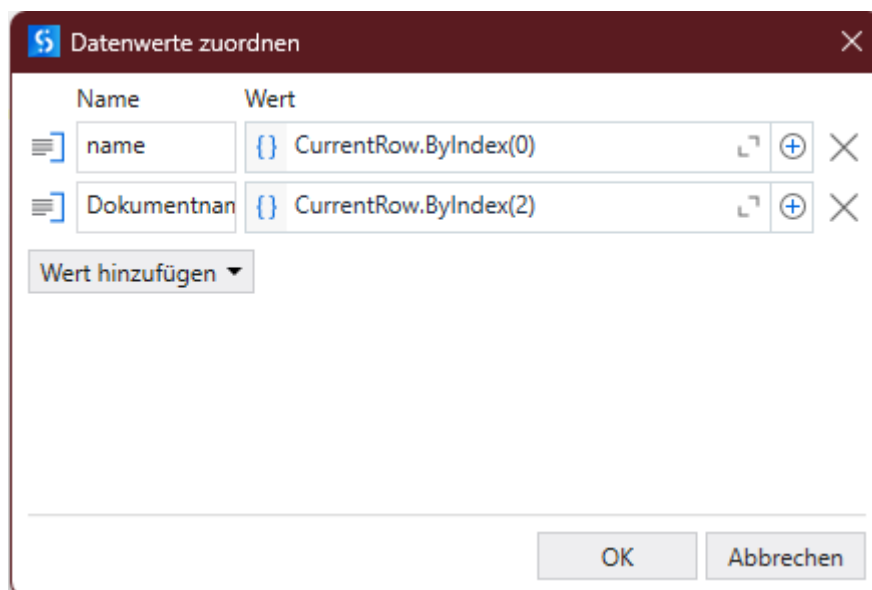


Abbildung 117: E-Mail Bot Response: Datenwerteditor (E-Mail Body)

Zuletzt wird noch der Anhang definiert. Die beiden Beispielbroschüren heißen wie folgt:

- Chardonnay.pdf
- Sauvignon Blanc.pdf

Somit ist alles was man machen muss bei den Anhängen den Namen des benötigten Dokuments von der Excel Datei auslesen und „.pdf“ dran hängen. Dies macht man mit folgendem Befehl:

```
CurrentRow.ByIndex(2) + ".pdf"
```

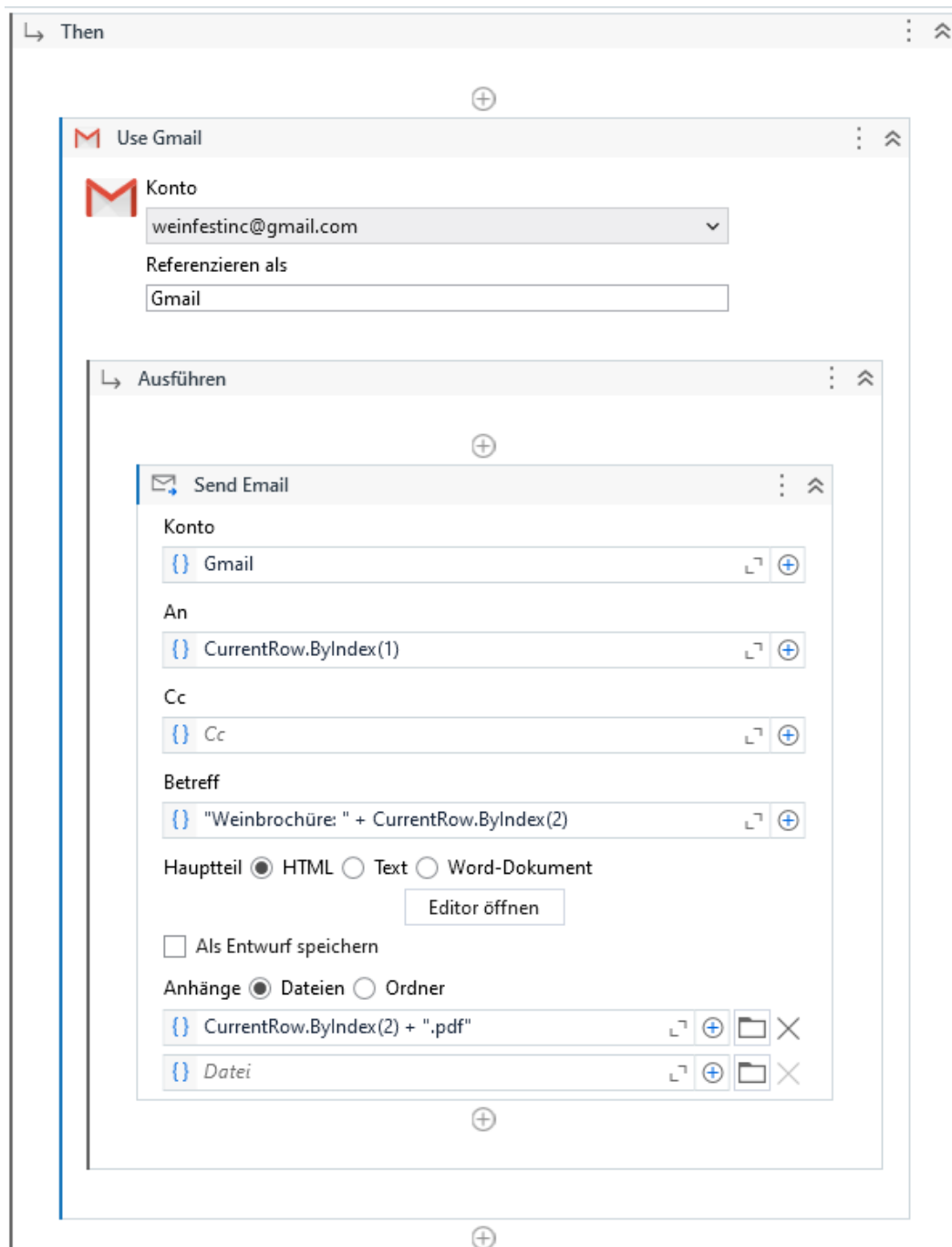


Abbildung 118: E-Mail Bot Response: "Use Gmail" und "Send Email" Aktivitäten

13.2 Datei-Organisation

13.2.1 Prinzip

Als Beispielprogramm zur Demonstration möglicher Automatisierungen eignet sich besonders eine Dateiorganisationsautomatisierung. Dabei gibt es in einem Projektordner verschiedene Dateien, die in die richtigen Ordner verschoben werden sollen. In diesem Fall wurde Python als Programmiersprache für die Automatisierung verwendet. Python verfügt über eine umfangreiche Standardbibliothek, die den Zugriff auf Dateien in verschiedenen Ordnern ermöglichen kann. Python ist auch eine sehr einfache und verständliche Programmiersprache, weshalb sie sich für den Automatisierungsprozess hervorragend eignet.

13.2.2 Benutzung



Abbildung 119: Datei-Organisation (Datei)

Die Automatisierung ist als Python Programm, gespeichert.

Diese kann mittels Benutzung von der Eingabeaufforderung (CMD), welche in Windows immer vorhanden ist, aufgerufen werden.

```
CA: Eingabeaufforderung
Microsoft Windows [Version 10.0.19045.4170]
(c) Microsoft Corporation. Alle Rechte vorbehalten.

C:\Users\mertj>cd Desktop

C:\Users\mertj\Desktop>python organisation.py_
```

Abbildung 120: Datei-Organisation (Eingabeaufforderung)

Durch das Eingeben von „cd Desktop“ weiß die Eingabeaufforderung, dass die nächsten Befehle im Desktop-Verzeichnis ausgeführt werden sollen. Die Ausführung des eigentlichen Programms erfolgt dann mit dem Befehl „python organisation.py“. Dabei steht „organisation.py“ für den Dateinamen.

Jedoch bevor das Programm ausgeführt wird, muss im Code der Dateipfad, für den Quell-Ordner angegeben werden.

```
projekt_ordner = 'C:/Users/mertj/Desktop/Projekt'
organize(projekt_ordner)
```

Abbildung 121: Datei-Organisation (Datei Pfad)

13.2.3 Projektstruktur

Name	Änderungsdatum	Typ	Größe
Backend_beispielAPI	27.03.2024 21:09	Dateiordner	
Code	27.03.2024 21:11	Dateiordner	
Dokumentation	27.03.2024 21:25	Dateiordner	
Frontend_beispielWebsite	27.03.2024 21:09	Dateiordner	
Projekt_Präsentation	27.03.2024 21:25	Dateiordner	
Projekt_Dokumentation.docx	27.03.2024 20:58	Microsoft Word-D...	0 KB
Präsentation.pptx	27.03.2024 20:58	Microsoft PowerP...	0 KB
ProjektAnleitung.txt	27.03.2024 20:57	Textdokument	0 KB

Abbildung 122: Datei-Organisation Struktur

Die Automatisierung findet in diesem Verzeichnis „Projekt“ statt. Dabei hat der Ordner „Projekt“ verschiedene Unterordner. „Projekt_Präsentationen“, „Dokumentation“ und „Code“ sind untergeordnete Verzeichnisse des Projektordners, wobei im „Code“-Verzeichnis noch zwei weitere Ordner namens „Frontend“ und „Backend“ vorhanden sind.

Name	Änderungsdatum	Typ	Größe
Backend	27.03.2024 21:25	Dateiordner	
Frontend	27.03.2024 21:25	Dateiordner	

Abbildung 123: Datei-Organisation Backend/Frontend

Die Aufgabe besteht darin, das Verschieben der jeweiligen Dateien in die Richtigen Ordner zu Automatisieren.

Ablauf:

Datei/Projekt	Ordner
Projektanleitung.txt	Dokumentation
Projekt_Dokumentation.docx	Dokumentation
Präsentation.pptx	Projekt_Präsentationen
Backend_beispielAPI	Code→Backend
Frontend_beispielWebsite	Code→Frontend

Tabelle 17: Datei/Projekt | Ordner

13.2.4 Implementierung

Zunächst werden „os“ und „shutil“ importiert. Diese Module ermöglichen das Arbeiten mit Verzeichnissen in Windows, dabei können Dateien und Verzeichnisse gelöscht, verschoben, erstellt und kopiert werden

```
1 import os
2 import shutil
```

Abbildung 124: Datei-Organisation Imports

```
def organize(projekt_ordner):
    for element in os.listdir(projekt_ordner):
        element_pfad = os.path.join(projekt_ordner, element)
        if os.path.isfile(element_pfad):
            verschiebeDatei(element_pfad, projekt_ordner)
        elif os.path.isdir(element_pfad):
            verschiebeCode(element_pfad, projekt_ordner)
```

Abbildung 125: Datei-Organisation organize Methode

Die organize-Methode ist die Hauptmethode, die andere Methoden aufruft. Dabei wird der organize-Methode der Pfad des Projektordners mit „projekt_ordner“ übergeben. Dieser wird, wie bereits bei der Benutzung beschrieben, festgelegt. Als nächstes werden alle Elemente des angegebenen Projekt-Pfades mit einer foreach-Schleife durchlaufen. Dann wird mittels *if-Abfragen* geprüft, ob das Element eine Datei oder ein weiterer Ordner ist, um dann die jeweiligen Methoden zum Verschieben der Datei aufzurufen.

```
def verschiebe(source, destination):
    if not os.path.exists(destination):
        os.makedirs(destination)
    shutil.move(source, destination)
```

Abbildung 126: Datei-Organisation verschiebe Methode

Durch diese Methode wird ein Verschieben der Datei oder des Ordners anhand der Quell-Datei/Ordner, also der Quelle und des Zielorts, ermöglicht. Diese Methode wird durch die zwei verschiedenen Methoden „verschiebeDatei“ und „verschiebeCode“ verwendet.

```
def verschiebeDatei(file_path, projekt_ordner):
    file_extension = file_path.split('.')[-1].lower()
    if file_extension == 'pptx':
        verschiebe(file_path, os.path.join(projekt_ordner, "Projekt_Präsentation"))
    elif file_extension == 'docx' or file_extension == 'txt':
        verschiebe(file_path, os.path.join(projekt_ordner, "Dokumentation"))
```

Abbildung 127: Datei-Organisation verschiebeDatei Methode

Diese Methode ist speziell für das Verschieben von Dateien gedacht, also Dateien wie PowerPoint-Präsentationen, Word-Dateien oder Textdateien. Dabei wird durch den „file_path“ die Dateiendung mit einem Split herausgefunden. Mit *if-Abfragen* wird dann wiederum entschieden, in welchen Ordner die jeweilige Datei verschoben werden soll. Wenn es eine Präsentation ist, dann wird diese in den Ordner „Projekt_Präsentationen“ verschoben. Falls die Datei eine docx- oder eine txt-Datei ist, wird diese automatisch in den „Dokumentation“-Ordner verschoben. Die „verschiebe“-Methode wird in dem Fall mit dem Dateipfad und den Zielordner mithilfe von „os.path.join“ verschoben.

```
def verschiebeCode(folder_path, projekt_ordner):
    folder_name = os.path.basename(folder_path)
    if "Frontend" in folder_name:
        verschiebe(folder_path, os.path.join(projekt_ordner, "Code", "Frontend"))
    elif "Backend" in folder_name:
        verschiebe(folder_path, os.path.join(projekt_ordner, "Code", "Backend"))
```

Abbildung 128: Datei-Organisation verschiebeCode Methode

Mit "verschiebeCode" ist das Verschieben von Code-Projekten gemeint, also beispielsweise ein Java-Projekt. In dem Fall wird geprüft, ob im Namen des Ordners „Frontend“ oder „Backend“ vorhanden ist. Dies wird durch die basename herausgefunden. Wenn „Frontend“ im Namen vorhanden ist, dann wird der Ordner in das Verzeichnis „Projekt -> Code -> Frontend“ verschoben. Jedoch wenn „Backend“ im Namen vorhanden ist, wird der Ordner in das Verzeichnis „Projekt -> Code -> Backend“ verschoben.

```
projekt_ordner = 'C:/Users/merti/Desktop/Projekt'
organize(projekt_ordner)
```

Abbildung 129: Datei-Organisation Methodenaufruf organize

Als letztes wird die Methode „organize“ mit dem Pfad zum Projekt-Ordner aufgerufen.

13.3 Windows Virtualisierung

13.3.1 Problem

Der PC wird automatisch ohne Hyper-V³³ gestartet um bessere Performance in einer VMware Workstation³⁴ zu liefern. Jedoch wird in manchen Fällen die Virtualisierung eines Linux-Systems benötigt, wie zum Beispiel bei der Nutzung von Docker³⁵. Wenn jetzt umgeschaltet werden soll muss der PC heruntergefahren werden und danach das richtige Startprofil ausgewählt werden.

Um dieses Problem zu lösen, wurde nun ein Batch-Skript entwickelt, welches das jeweilig andere Profil auswählt und den Computer neu startet, um diesen Prozess zu Automatisieren.

13.3.2 Benutzung

Als aller erstes müssen die benötigten Daten einmalig für einen PC eingegeben werden. Es wird die Beschreibung eines mit Hyper-V aktivierten Profils eingegeben werden. Außerdem wird der eindeutige Kennungscode für die Profile benötigt. Diese Daten können ausgelesen werden, indem in ein Command-Line Interface, als Administrator gestartet, der Befehl „bcdedit“ eingegeben wird.

```
::Insert your HyperV profile description
set "description=Win11"
::Insert your HyperV profile identifier
set "idHV=00000000-0000-0000-0000-000000000000"
::Insert your No HyperV profile identifier
set "idNHV=default"
```

Danach muss in den Einstellungen des Skripts einmalig „Als Administrator ausführen“ eingestellt werden, da das Skript auch den Command „bcdedit“ benötigt.

Um nun das Windows Profil zu wechseln, muss nur noch die Datei mit einem Doppelklick gestartet werden.

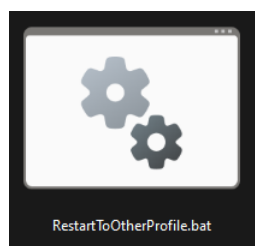


Abbildung 130: Windows Batch-Datei

13.3.3 Implementierung

Nachdem die Informationen zum eigenen System eingegeben worden sind, prüft das Skript erst einmal, ob es als Administrator gestartet worden ist, da ohne diese Berechtigungen dieses nicht funktioniert. Sollten keine Administrator Berechtigungen vorhanden sein, bricht das Skript ab mit der Fehlermeldung: „Script is not running with administrative privileges. Please run the script as administrator.“

³³ Hyper-V: <https://de.wikipedia.org/wiki/Hyper-V>

³⁴ VMware Workstation: https://de.wikipedia.org/wiki/VMware_Workstation

³⁵ Docker: [https://de.wikipedia.org/wiki/Docker_\(Software\)](https://de.wikipedia.org/wiki/Docker_(Software))

Da nun sichergestellt ist, dass alle benötigten Berechtigungen vorhanden sind, wir geprüft welches der beiden Systeme gerade verwendet wird. Dabei wird die Ausgabe vom Befehl „bcdedit“ ausgelesen und verarbeitet.

```
echo Checking the currently active Windows profile...
set "run=true"
set "identifierOutput="
set "descriptionOutput="

for /f "delims=" %%i in ('bcdedit ^| findstr /r /v "^$ "') do (
    set "line=%%i"
    echo !line! | find /i "identifier" > nul && (
        for /f "tokens=" %%a in ("!line:identifier=!") do (
            set "identifierOutput=%%a"
            if "!identifierOutput!" equ "{current}" (
                set "run=false"
            )
        )
    )
    echo !line! | find /i "description" > nul && (
        for /f "tokens=" %%a in ("!line:description=!") do (
            set "descriptionOutput=%%a"
            if !run! equ false (
                goto :breakloop
            )
        )
    )
)
)
```

Abbildung 131: RestartToOtherProfile.bat aktuelles Profil

Wenn nun das aktuelle Profil bekannt ist, wird das jeweils andere für den nächsten Start festgelegt.

```
if "%descriptionOutput%" == "%description%" (
    echo Changing to the 'No HyperV' profile...
    bcdedit /bootsequence {!idNHV!} /addfirst
) else (
    echo Changing to the 'HyperV' profile...
    bcdedit /bootsequence {!idHV!} /addfirst
)
```

Abbildung 132: RestartToOtherProfile.bat Start Profil festlegen

Nun wird nur noch der Computer automatisch neu gestartet und das Profil wurde gewechselt, so dass die benötigten Dienste vorhanden sind.

14 RESÜMEE

14.1 Alexandra Bauer

Durch die Implementierung von SimpleQuery und der Synchronisation von SharePoint und Jira habe ich vor allem wertvolle Erfahrungen in den verschiedensten technischen Bereichen gesammelt. Dies bezieht sich vor allem auf die Automatisierungstechnologie UiPath Studio sowie Spring.

Da ich bereits mit den Tools von UiPath und Spring vertraut war und uns die Technologie frei auswählen konnten, haben wir uns für diese entschieden. Da meine anderen beiden Teammitglieder noch keine Erfahrung mit diesen Technologien hatten, habe ich den ihnen beim Einarbeiten und Verwenden der Tools geholfen.

Wegen meiner bereits vorhandenen Kenntnisse war das Ausarbeiten des UiPath Studio Programms keine große Herausforderung für mich. Auch das Implementieren von Spring war nicht sonderlich schwer, da ich nicht nur in einem vorherigen Praktikum damit gearbeitet habe, sondern auch eine Hilfestellung von unserem Auftraggeber erhalten haben. Somit konnte ich meine bereits vorhandenen Kenntnisse noch einmal deutlich vertiefen. Zusätzlich konnte ich vor allem im Bereich des Requirement Engineerings einige wertvolle Erfahrung sammeln, da keine klare Aufgabenstellung in Bezug auf Ansatz, Weg und zu verwendeten Technologie erhalten haben. Dies stand uns vollkommen frei und wir diesen auch selber wählen durften.

14.2 Mert Ibisler

Durch die Implementierung von SimpleQuery und der Synchronisation von SharePoint und Jira konnte ich wertvolle Erfahrungen in verschiedenen technischen Bereichen sammeln. Besonders herausragend waren dabei meine Erkenntnisse in der Automatisierungstechnologie UiPath Studio. Darüber hinaus konnte ich hinsichtlich der Zusammenarbeit mit meinen Diplomarbeitern auch Erfahrungen und Einblicke in Spring gewinnen, wobei mein Schwerpunkt deutlich auf der Automatisierung mit UiPath Studio lag.

Da Alexandra Bauer bereits Kenntnisse mit der Prozessautomatisierung besaß, konnte sie uns von Anfang an für das Thema interessieren und uns beim Erlernen des Automatisierens Beihilfe leisten.

Im Wesentlichen haben wir uns durch eine enge Zusammenarbeit sowie eigenes Engagement und Lernen sämtlicher Tools, die Techniken und Prinzipien der Prozessautomatisierung angeeignet, wobei dies einen Großteil unserer Zeit während des Praktikums bei der Voestalpine beanspruchte.

Eine weitere wichtige Erfahrung war für mich das Zusammenarbeiten mit einem großen Unternehmen und die eigenständige Arbeitsleistung. Zudem stellte mich die Möglichkeit, die Vorgehensweise und Tools selbst auszuwählen, vor lehrreiche Herausforderungen.

14.3 Simon Praher

Durch die Implementierung von SimpleQuery und der Synchronisation von SharePoint und Jira konnte ich vor allem wertvolle Erfahrungen in den verschiedensten technischen Bereichen gesammelt. Dies bezieht sich vor allem auf die Automatisierungstechnologie UiPath Studio sowie Spring. Da sich Alexandra bereits mit dem Tool UiPath beschäftigt hatte, führte sie mich durch die Installation des Programms und half mir bei den ersten Schritten damit.

Weiterhin war die Zusammenarbeit mit einem großen Unternehmen wie der Voest eine wichtige Erfahrung, die mir in meinem späteren Berufsleben sicherlich helfen wird. Vor allem die selbstständige Arbeitsweise sowie die Möglichkeit, Vorgehensweisen und Tools eigenständig auszuwählen, waren neu, da in der Schule oder auch in anderen Unternehmen die Mittel immer vorgegeben wurden.

Zudem wurden auch meine Kenntnisse im Bereich Robotic Process Automation gestärkt und ich habe viel dazugelernt. Auch der Umgang mit Rückschlägen und die Fähigkeit, flexibel auf Änderungen zu reagieren, wurden gestärkt. Die Erfahrungen aus dieser Praxisphase werden mir bei meinem weiteren beruflichen Werdegang sicherlich von großem Nutzen sein. Insgesamt war es eine sehr lehrreiche und prägende Zeit bei der Voestalpine.

15 LITERATURVERZEICHNIS

- Aich, R. (1. 5 2023). *LinkedIn*. Abgerufen am 21. 02 2024 von LinkedIn:
<https://www.linkedin.com/pulse/power-risks-robotic-process-automation-rpa-rajarshi-aich#:~:text=If%20the%20RPA%20software%20is,financial%20losses%20and%20reputational%20damage>
- Anywhere, A. (1. 1 2024). *Automation Anywhere*. Abgerufen am 21. 02 2024 von Automation Anywhere: [https://www.automationanywhere.com/de/legal/gdpr-compliance#:~:text=Die%20Datenschutz%2DGrundverordnung%20\(DSGVO\),den%20Schutz%20personenbezogener%20Daten%20regeln.](https://www.automationanywhere.com/de/legal/gdpr-compliance#:~:text=Die%20Datenschutz%2DGrundverordnung%20(DSGVO),den%20Schutz%20personenbezogener%20Daten%20regeln.)
- Automation Anywhere. (1. August 2023). *Robotergesteuerte Prozessautomatisierung (RPA)*. Von Automation Anywhere: [https://www.automationanywhere.com/de/rpa/robotic-process-automation#:~:text=Robotergesteuerte%20Prozessautomatisierung%20\(RPA\)%20ist%20eine,anschlie%3%9Fend%20regelgest%3%BCtzte%20Gesch%3%A4ftsprozesse%20ausf%3%BChren%20k%3%B6nnen](https://www.automationanywhere.com/de/rpa/robotic-process-automation#:~:text=Robotergesteuerte%20Prozessautomatisierung%20(RPA)%20ist%20eine,anschlie%3%9Fend%20regelgest%3%BCtzte%20Gesch%3%A4ftsprozesse%20ausf%3%BChren%20k%3%B6nnen) abgerufen
- AutomationNEXT. (18. 5 2017). *Die Geschichte der Robotertechnik begann 1954*. Abgerufen am 12. 01 2024 von AutomationNEXT: <https://www.automation-next.com/kollegroboter/grundlagen/die-geschichte-der-robotertechnik-begann-1954-108.html>
- Baitech Data. (2023). *Was ist Automation Anywhere*. Abgerufen am 8. Januar 2024 von Baitech Data: <https://baitechdata.de/was-ist-automation-anywhere/>
- Baitech Data. (2023). *Was ist Blue Prism*. Abgerufen am 8. Januar 2024 von Baitech Data: <https://baitechdata.de/was-ist-blue-prism/>
- Baitech Data. (2024). *Was ist Microsoft Power Automate*. Von Baitech Data: <https://baitechdata.de/was-ist-power-automate/> abgerufen
- Bremmer, M. (02. 09 2021). *Computerwoche*. Abgerufen am 23. 02 2024 von Computerwoche: <https://www.computerwoche.de/a/in-4-schritten-zu-mehr-rpa-security,3551131>
- Camunda. (1. 1 2023). *Kontrolle über Ihre RPA-Bots übernehmen*. Abgerufen am 10. 1 2024 von Camunda: <https://camunda.com/de/solutions/take-control-of-your-rpa-bots/>
- Cyberark. (1. April 2023). *Robotergesteuerte Prozessautomatisierung (RPA)*. Von Cyberark: <https://www.cyberark.com/de/what-is/robotic-process-automation/> abgerufen
- databricks. (21. 02 2024). *databricks*. Von Datenautomatisierung. abgerufen
- Draese, N. (20. 05 2023). *Automatisierung in der Industrie: Welche Vor- und Nachteile – welche Potenziale?* Abgerufen am 12. 01 2024 von Ingenieur.de: <https://www.ingenieur.de/technik/fachbereiche/automation/automatisierung-in-der-industrie-welche-vor-und-nachteile-welche-potenziale/>
- GBTEC. (2023). *No-Code und Low-Code Automation*. Abgerufen am 10. Januar 2024 von GBTEC: <https://www.gbtec.com/de/ressourcen/no-code-und-low-code-automation/>

- Gurd, T. (17. 01 2022). *20 RPA-Anwendungsbeispiele aus der Fertigung*. Abgerufen am 12. 01 2024 von ss&c blueprism: <https://www.blueprism.com/de/resources/blog/20-effective-rpa-use-cases-in-manufacturing/>
- Hamberger, D. (22. 08 2022). *Robotik 2030: So sieht die Zukunft aus*. Abgerufen am 12. 01 2024 von Industrie Magazin: <https://industriemagazin.at/artikel/robotik-2030-so-sieht-die-zukunft-aus/>
- IT-P GmbH. (4. Januar 2024). *IT-P*. Von IT-P: [https://www.it-p.de/blog/was-ist-intelligent-process-automation/#:~:text=Intelligent%20Process%20Automation%20\(IPA\)%20ist,menschliche%20Entscheidungen%20und%20Urteile%20erfordern](https://www.it-p.de/blog/was-ist-intelligent-process-automation/#:~:text=Intelligent%20Process%20Automation%20(IPA)%20ist,menschliche%20Entscheidungen%20und%20Urteile%20erfordern) abgerufen
- Kosmopoulos, C. (27. 01 2021). *Blueprint*. Abgerufen am 23. 02 2024 von Blueprint: <https://www.blueprintsys.com/blog/rpa/7-hidden-risks-automation-design>
- Künzli, P. (25. 08 2019). *Die Geschichte der Automatisierung*. Abgerufen am 12. 01 2024 von PR Automation: <https://prautomation.ch/2019/08/25/die-geschichte-der-automatisierung/>
- Lanuschy, S. (21. September 2022). *Prozessautomatisierung*. Abgerufen am 12. 01 2024 von Papershift: <https://www.papershift.com/lexikon/prozessautomatisierung#prozessautomatisierung-in-unternehmen>
- Marko. (21. Juli 2020). *IT Automation Architect: IT-Job mit Zukunft*. Abgerufen am 1. 9 2024 von computerWeekly: <https://www.computerweekly.com/de/feature/IT-Automation-Architect-IT-Job-mit-Zukunft>
- Naeem, T. (05. 01 2024). *Astera*. Abgerufen am 21. 02 2024 von Der 8-stufige Datenautomatisierungs-Spielplan – Astera Centerprise: <https://www.astera.com/de/Typ/Blog/Datenautomatisierung/>
- Pega. (2024). *Was ist Pega?* Abgerufen am 10. Januar 2024 von Pega: <https://www.pegacom.com/de/about>
- Persiono. (01. 01 2024). *Persiono*. Abgerufen am 26. 02 2024 von Persiono: <https://www.personio.de/hr-lexikon/prozessautomatisierung/>
- Roboyo. (14. 01 2019). *Roboyo*. Abgerufen am 23. 02 2024 von Roboyo: <https://www.cio.com/article/190962/6-hidden-risks-of-it-automation.html>
- Ryte GmbH. (1. April 2023). *Jira*. Von Ryte Wiki: <https://de.ryte.com/wiki/Jira> abgerufen
- SAFAR, M. (1. August 2023). *RPA Security – So vermeiden Sie Sicherheitsprobleme*. Von Weissenberg: <https://weissenberg-group.de/rpa-security-so-vermeiden-sie-sicherheitsprobleme-beim-bot-betrieb/> abgerufen
- SAFAR, M. (1. August 2023). *SAP S/4HANA-Migration mit Robotic Process Automation (RPA)*. Von Weissenberg: <https://weissenberg-group.de/sap-s4hana-migration-mit-rpa/> abgerufen
- SAFAR, M. (1. August 2023). *Was ist Robotic Process Automation (RPA)?* Von Weissenberg: <https://weissenberg-group.de/was-ist-robotic-process-automation/> abgerufen

- Safar, M. (1. 1 2024). Abgerufen am 9. 1 2024 von <https://weissenberg-group.de/robotic-process-automation-im-kundenservice/>
- Safar, M. (1. 1 2024). *Weissenberg*. Abgerufen am 9. 1 2024 von Was macht ein RPA Developer: <https://weissenberg-group.de/berufsbild-rpa-developer/>
- Safar, M. (1. 1 2024). *Weissenberg*. Abgerufen am 10. 1 1 von RPA Security – So vermeiden Sie Sicherheitsprobleme: <https://weissenberg-group.de/rpa-security-so-vermeiden-sie-sicherheitsprobleme-beim-bot-betrieb/>
- SAFAR, M. (01. 01 2024). *Weissenberg*. Abgerufen am 23. 02 2024 von Weissenberg: <https://weissenberg-group.de/rpa-security-so-vermeiden-sie-sicherheitsprobleme-beim-bot-betrieb/>
- Savanovich, A. (29. 9 2023). *Open Access Government*. Abgerufen am 21. 02 2024 von Open Access Government: <https://www.openaccessgovernment.org/robotic-process-automation-security-risks-and-ways-to-eliminate-them/167309/#:~:text=Nevertheless%2C%20RPA%20technology%20poses%20some,attractive%20gain%20to%20cyber%20criminals>
- Schulz, T. S. (1. April 2023). *RPA mit UiPath – der etablierte Alleskönner für Robotic Process Automation*. Von Palladio-Consulting: <https://www.palladio-consulting.de/rpa-mit-uipath/> abgerufen
- Talentra. (22. 05 2023). *Die Zukunft des Arbeitsplatzes: Automatisierung und KI revolutionieren die Arbeitswelt*. Abgerufen am 12. 01 2024 von Talentra Talents at work: <https://talentra.at/zukunft-des-arbeitsplatzes/>
- Thiele, D.-I. G. (2024). *Trends in der Robotik 2024*. Abgerufen am 12. 01 2024 von Fraunhofer IPK: <https://www.ipk.fraunhofer.de/de/kompetenzen-und-loesungen/automatisierung/trends-der-robotik-2024.html>
- UiPath. (1. April 2023). *Studio User Guide - Introduction*. Von UiPath Documentation: <https://docs.uipath.com/studio/standalone/2022.10/user-guide/introduction> abgerufen
- UiPath. (1. April 2023). *Studio User Guide - Workflow Design*. Von UiPath Documentation: <https://docs.uipath.com/studio/standalone/2022.10/user-guide/workflow-design> abgerufen
- UNIORG Gruppe. (1. August 2023). *Was ist SAP?* Von Uniorg: <https://www.uniorg.de/lexikon/was-ist-sap/#definition> abgerufen
- Violino, B. (1. 12 2020). *CIO*. Abgerufen am 23. 02 2024 von CIO: <https://www.cio.com/article/190962/6-hidden-risks-of-it-automation.html>
- Visualmakers. (01. 01 2023). *Visualmakers*. Abgerufen am 27. 02 2024 von Visualmakers: <https://www.visualmakers.de/blog/wann-sollte-ich-automatisieren>
- Voestalpine. (2024). *Unser Konzern im Überblick*. Abgerufen am 11. Januar 2024 von Voestalpine: <https://www.voestalpine.com/group/de/konzern/ueberblick/>
- Wiefel, A. (13. April 2013). *OpenText*. Abgerufen am 9. Januar 2024 von OpenText: <https://mind-forms.de/knowhow/opentext/>

Wikipedia. (23. 11 2023). *OpenAI*. Abgerufen am 22. 03 2024 von Wikipedia:
https://de.wikipedia.org/wiki/OpenAI#Gr%C3%BCndungsphase_und_Mission

Wikipedia. (23. 11 2023). *Wikipedia*. Abgerufen am 22. 03 2024 von OpenAI:
https://de.wikipedia.org/wiki/OpenAI#Gr%C3%BCndungsphase_und_Mission

Wikipedia. (17. 03 2024). *ChatGPT*. Abgerufen am 22. 03 2024 von Wikipedia:
<https://de.wikipedia.org/wiki/ChatGPT>

16 ABBILDUNGSVERZEICHNIS

Abbildung 1: Voestalpine	8
Abbildung 2: Projektteam	9
Abbildung 3: geeignete Prozesse für Automatisierung	19
Abbildung 4: UiPath Logo	26
Abbildung 5: Jira Logo	28
Abbildung 6: Automata 13, Heron von Alexandria	31
Abbildung 7: VisiCalc	32
Abbildung 8: DSGVO Logo	43
Abbildung 9: Automation Anywhere Logo	48
Abbildung 10: Blue Prism Logo	49
Abbildung 11: OpenText LiquidOffice Logo	50
Abbildung 12: Pega Logo	51
Abbildung 13: Künstliche Intelligenz, Machine Learning und Deep Learning	53
Abbildung 14: künstlich neuronale Netze (links: einfaches KNN; rechts: KNN mit Deep Learning kombiniert)	54
Abbildung 15: Machine Learning	55
Abbildung 16: OpenAI Logo	57
Abbildung 17: Chatbot Vergleich Text	58
Abbildung 18: Chatbot Vergleich Bild	59
Abbildung 19: UiPath - Datei Ausführen	60
Abbildung 20: UiPath Orchestrator	60
Abbildung 21: Benutzung Simple Query Start Site	61
Abbildung 22: Benutzung Simple Query HomeSite	61
Abbildung 23: Benutzung Simple Query DataSite	61
Abbildung 24: Benutzung Simple Query Compare Data Site	61
Abbildung 25: Benutzung Simple Query Choose Data Site	61
Abbildung 26: SharePoint (wegen Datenschutzrichtlinien geschwärzt)	62
Abbildung 27: Jira (wegen Datenschutzrichtlinien geschwärzt)	62
Abbildung 28: UiPath Studio Download	65
Abbildung 29: UiPath Setup	66
Abbildung 30: UiPath Studio & StudioX	67
Abbildung 31: UiPath Extensions	68
Abbildung 32: UiPath Extension in Edge	68
Abbildung 33: UiPath – SharePoint – User Browser	74
Abbildung 34: UiPath – SharePoint - UiElementExists	75
Abbildung 35: UiPath – SharePoint - Element Exists Eigenschaften	75
Abbildung 36: UiPath – SharePoint – Nächste Seite	76
Abbildung 37: UiPath – SharePoint – Type Into (PPS)	76
Abbildung 38: UiPath – SharePoint – Filter Status	77
Abbildung 39: UiPath – SharePoint – Extract Structured Data Table	78
Abbildung 40: UiPath – SharePoint – next-Button	78
Abbildung 41: UiPath – SharePoint – Output Data Table	79
Abbildung 42: UiPath – SharePoint – Log Message	79
Abbildung 43: UiPath – SharePoint – Multiple Assign	80
Abbildung 44: UiPath – SharePoint – taskID & taskCat	81
Abbildung 45: UiPath – SharePoint – Bedingung taskCat	82

Abbildung 46: UiPath – SharePoint – external_issue_ID & URL.....	83
Abbildung 47: UiPath – SharePoint – Arbeitspaket aufruf external_issue_URL	83
Abbildung 48: UiPath – SharePoint – Element Exists description	84
Abbildung 49: UiPath – SharePoint – Assign description	85
Abbildung 50: UiPath – SharePoint – comments array	86
Abbildung 51: UiPath – SharePoint - commentString	87
Abbildung 52:UiPath – SharePoint – print all table entries	87
Abbildung 53: UiPath - SharePoint - JiraTaskUrl.....	88
Abbildung 54: UiPath Jira (Invoke Workfiles)	89
Abbildung 55: UiPath - Jira - CheckJiraTaskExistence	90
Abbildung 56: UiPath - Jira - CheckJiraTaskExistence	90
Abbildung 57: UiPath - Jira - GetAssigneeAccountID.....	91
Abbildung 58: UiPath - Jira - GetAssigneeAccountID	91
Abbildung 59: UiPath - Jira - GetAssigneeAccountID	92
Abbildung 60: UiPath - Jira - DefaultJiraChanges.....	93
Abbildung 61: UiPath - Jira - CreateNewComment	94
Abbildung 62: UiPath - Jira - ChangeStatus (Sequence)	95
Abbildung 63: UiPath - Jira - ChangeStatus (Sequence)	96
Abbildung 64: UiPath - Jira - ChangeStatus (Sequence)	97
Abbildung 65: UiPath - Jira - ChangeStatus (Workfile)	97
Abbildung 66: UiPath - Jira - ChangeStatus (Workfile)	98
Abbildung 67: UiPath - Jira - ChangeStatus (Workfile)	99
Abbildung 68: UiPath - Jira - ChangeStatus (Workfile)	99
Abbildung 69: Java Logo	100
Abbildung 70: Spring Logo.....	100
Abbildung 71: Simple Query Programmaufbau.....	102
Abbildung 72: Simple Query Server	103
Abbildung 73: Simple Query Software components.....	103
Abbildung 74: Simple Query Software components.....	104
Abbildung 75: Voest Simple Query Object types.....	104
Abbildung 76: Simple Query Objet types	104
Abbildung 77: Voest Simple Query Condition & Result Attributes.....	105
Abbildung 78: Simple Query Condtiton & Result Attributes	105
Abbildung 79: Voest Simple Query XML template	106
Abbildung 80: Simple Query templates	106
Abbildung 81: Simple Query XML Datei	107
Abbildung 82: Simple Query Model.....	108
Abbildung 83: Simple Query Api Zugriff	108
Abbildung 84: Simple Query ModelFromResponse	109
Abbildung 85: Simple Query ModelBuilder	109
Abbildung 86: Simple Query ModelBuilder Rekursive Methode	110
Abbildung 87: Simple Query Standard.xml.....	111
Abbildung 88: Simple Query RequestXMLBuilder create Request	111
Abbildung 89: Simple Query Home Site	112
Abbildung 90: Simple Query HomeSite	112
Abbildung 91: Simple Query choose Data Site	113
Abbildung 92: Simple Query addAllSelcetions	114
Abbildung 93: Simple Query Komponenten hinzufügen.....	115

Abbildung 94: Simple Query dropdown Optionen	115
Abbildung 95: Simple Query ChooseDataSite.....	116
Abbildung 96: Simple Query teamplate Data	117
Abbildung 97: Simple Query Custom Data	117
Abbildung 98: Simple Query display Data	118
Abbildung 99: Simple Query DataSite.....	118
Abbildung 100: Simple Query matching Infos	119
Abbildung 101: Simple Query Compared Site	119
Abbildung 102: E-Mail vom E-Mail Bot	124
Abbildung 103: E-Mail Bot Request: Get Newest Email	126
Abbildung 104: E-Mail Bot Request: Zugriff auf "E-Mail Adressen.xlsx"	126
Abbildung 105: E-Mail Bot Request: Build Data Table Fenster.....	127
Abbildung 106: E-Mail Bot Request: Excel Tabellen Überschriften.....	127
Abbildung 107: E-Mail Bot Request: newRow	128
Abbildung 108: E-Mail Bot Request: Multiple Assign für "newRow"	128
Abbildung 109: E-Mail Bot Request: For Each Excel Row ("verfügbare_Weinbroschüren")	129
Abbildung 110: E-Mail Bot Request: Excel Datei "verfügbare_Weinbroschüren.xlsx"	130
Abbildung 111: E-Mail Bot Request: Add Data Row	130
Abbildung 112: E-Mail Bot Request: Write DataTable to Excel	131
Abbildung 113: E-Mail Bot Request: mögliche Excel Datei mit Anfragen ("E-Mail Adressen.xlsx")	131
Abbildung 114: E-Mail Bot Response: Excel Datei Eintritt inkl. Schleife von "E-Mail Adressen.xlsx"	132
Abbildung 115: E-Mail Bot Response: Bedingungsgenerator	133
Abbildung 116: E-Mail Bot Response: E-Mail Body	134
Abbildung 117: E-Mail Bot Response: Datenwerteditor (E-Mail Body).....	134
Abbildung 118: E-Mail Bot Response: "Use Gmail" und "Send Email" Aktivitäten	135
Abbildung 119: Datei-Organisation (Datei).....	136
Abbildung 120: Datei-Organisation (Eingabeaufforderung).....	136
Abbildung 121: Datei-Organisation (Datei Pfad)	136
Abbildung 122: Datei-Organisation Struktur	137
Abbildung 123: Datei-Organisation Backend/Frontend.....	137
Abbildung 124: Datei-Organisation Imports.....	138
Abbildung 125: Datei-Organisation organize Methode	138
Abbildung 126: Datei-Organisation verschiebe Methode.....	138
Abbildung 127: Datei-Organisation verschiebeDatei Methode	139
Abbildung 128: Datei-Organisation verschiebeCode Methode	139
Abbildung 129: Datei-Organisation Methodenaufruf organize	139
Abbildung 130: Windows Batch-Datei	140
Abbildung 131: RestartToOtherProfile.bat aktuelles Profil	141
Abbildung 132: RestartToOtherProfile.bat Start Profil festlegen	141

17 TABELLENVERZEICHNIS

Tabelle 1: Organisation.....	12
Tabelle 2: Kommunikation.....	12
Tabelle 3: Applikation	12
Tabelle 4: Technologie	13
Tabelle 5: Daten.....	13
Tabelle 6: Location.....	13
Tabelle 7: Sicherheit.....	13
Tabelle 8: Feldernamen Sharepoint und Jira	63
Tabelle 9: Risiko Projektumfang.....	120
Tabelle 10: Risiko Projektmitglieder Erfahrung	120
Tabelle 11: Risiko Zeitmanagement.....	121
Tabelle 12: Risiko Teamkoordination	121
Tabelle 13: Risiko Ausfall Projektmitglied	121
Tabelle 14: Risiko Ausfall Auftraggeber	122
Tabelle 15: Risiko Auftraggeber springt ab.....	122
Tabelle 16: Risiko fehlende Ressourcen	122
Tabelle 17: Datei/Projekt Ordner.....	137