

# Architekturmodell

## Komponenten und Module

Das Programm wird in 5 Module aufgeteilt:

- 00 Orchestrierung
- 01 Webscraping
- 02 Datenextraktion
- 03 DB schreiben
- 04 DB Query + Excel
- 05 E-Mail

### **00 Orchestrierung**

In diesem Modul wird täglich um 08:00 oder durch eine manuelle Benutzereingabe der Bot gestartet. Dies kann mit dem Orchestrator gemacht werden.

### **01 Webscraping**

In diesem Modul werden von der Website alle PDFs gefunden und heruntergeladen.

### **02 Datenextraktion**

In diesem Modul werden die geforderten Daten von den einzelnen Rechnungen ausgelesen und in einem JSON Format gespeichert (Dem Datenmodell zu entnehmen).

### **03 DB schreiben**

In diesem Modul werden die zuvor extrahierten Daten auf einer MongoDB abgelegt, falls sie nicht schon auf der Datenbank sind.

### **04 DB Query + Excel**

In diesem Modul werden alle relevanten Rechnungen von der MongoDB heruntergeladen, welche noch nicht in einem Mail versendet wurden und in ein Excel Dokument geschrieben.

### **05 E-Mail**

In diesem Modul wird das erstellte Excel per E-Mail an die zuständige Person geschickt.

Dem Ablaufdiagramm sind die feingranulareren Schritte zu entnehmen.

## Schnittstellen

Die einzelnen Module werden in unterschiedliche Workflows aufgeteilt, die jeweils in- und out- Argumente haben, um die nötigen Informationen auszutauschen. Die einzelnen Workflows werden jeweils von einem Main Workflow aufgerufen, bis der Bot fertig durchgelaufen ist.

Ausgelöst wird der Bot durch den Orchestrator. Entweder manuell durch den User oder jeden Tag um 08:00.

### **00 Orchestrierung**

Input: Userinput/Timetrigger

Output: keine

### **01 Webscraping**

Input: keine

Output: Liste der heruntergeladenen Rechnungen (Pfade)

## **02 Datenextraktion**

Input: PDF Datei (Pfad)

Output: Extrahierte Informationen von der Rechnung im JSON Format.

## **03 DB schreiben**

Input: Informationen der Rechnung im JSON Format.

Output: keiner

## **04 DB Query + Excel**

Input: keiner

Output: Pfad, wo das Excel abgelegt wurde.

## **05 E-Mail**

Input: Pfad, wo das erstellte Excel liegt

# **Datenfluss**

Im Webscraping werden die Dateien Lokal abgelegt, von diesen werden dann die Daten extrahiert und auf einer MongoDB abgelegt.

Von der MongoDB laden wir die Daten herunter und erstellen ein Excel Dokument, welches auch Lokal abgelegt wird. Das Excel wird schlussendlich noch mit einer Mail an die zuständige Person geschickt.

# **Technologien und Plattformen:**

Für die Durchführung des Projekts wird UiPath eingesetzt. UiPath bietet eine benutzerfreundliche Schnittstelle, die eine schnellere Einarbeitung ermöglicht und die Entwicklungszeit reduziert. Zudem bietet es eine breite Palette an vorgefertigten Automatisierungskomponenten, die die Implementierung beschleunigen und die Skalierbarkeit erleichtern. Schliesslich bietet UiPath eine starke Community-Unterstützung und eine umfangreiche Dokumentation, was eine effiziente Problemlösung gewährleistet.

Als Datenbanklösung wurde MongoDB ausgewählt. MongoDB bietet eine flexible und skalierbare NoSQL-Datenbanklösung, die eine nahtlose Handhabung von unstrukturierten Daten ermöglicht. Durch seine dokumentenorientierte Struktur und leistungsfähige Abfragefähigkeiten eignet es sich ideal für moderne Anwendungen mit sich ändernden Anforderungen. Zudem bietet MongoDB eine aktive Community und umfassende Unterstützung, die eine schnelle Integration und Problemlösung ermöglicht.

RPA-Tool: UiPath Robot

Workflow-Management-Tool: UiPath

Entwicklungsumgebung (DIE): UiPath Studio

Datenbank: MongoDB

Kollaborations-Tools:

- Versionierung, Sharing, Backup: GitHub
- Kommunikation: Slack, Discord

## Produktiv- & Testsystem:

- Produktivsystem: Laufender Bot
- Testsystem: Entwicklung, Validierung, Qualitätssicherung

## Architekturdiagramm / Programmablaufplan

