jamk.fi

Robotic Process Automation in containerized environment

Marko Kauppinen

Bachelor's thesis
May 2020
School of Technology,
Bachelor of Engineering, Network Security and Industrial Internet

Jyväskylän ammattikorkeakoulu
JAMK University of Applied Sciences

jamk.fi

Description

Author(s)	Type of publication	Date
Marko Kauppinen	Bachelor's thesis	May 2020
		Language of publication: English
	Number of pages: 34	Permission for web publication: x

Title
Robotic Process Automation in containerized environment

Degree programme
Bachelor's degree programme of Engineering in Industrial internet and network security

Supervisor(s)
Mieskolainen, Matti
Huotari, Jouni

Assigned by

Abstract

RPA has become one of the methods to automate computer-based processes. The use of RPA targets to implement software process automation into existing manual or semi-manual processes. Understanding the difference between automation methods is important to get the best possible results from automation. RPA itself does not necessarily solve all the necessary IT automation needs, and the final automation solution requires separate components to support the desired automation solutions more comprehensively. A comprehensive IT system automation solution consists of many automation tools where RPA can be at the centre of control and monitor the implemented robot automation.

The purpose of this thesis was to deal mainly with RPA and partly with other alternative automation methods. Additionally, the objective was to cover application-based automation methods and describe the implementation of RPA in a containerized environment using open tools.

Visual Code was used as development tool to establish and develop Docker containers and its system component functionalities in one interface.

As a result, much was learned using and establishing Visual Code development environment on Windows Linux subsystem. Its functionalities in containerized environment were observed.

Keywords/tags (subjects)
RPA

jamk.fi

Kuvailulehti

Tekijä(t)	Julkaisun laji	Päivämäärä
Kauppinen, Marko	Opinnäytetyö, AMK	Toukokuu 2020
	Sivumäärä 34	Julkaisun kieli Suomi
		Verkkojulkaisulupa myönnetty:

Työn nimi
Ohjelmistorobotiikka kontitetussa ympäristössä

Tutkinto-ohjelma
Insinööri (AMK), Teollinen internet ja tietoverkkojen tietoturva

Työn ohjaaja(t)
Mieskolainen, Matti
Huotari, Jouni

Toimeksiantaja(t)

Tiivistelmä

Ohjelmistorobotiikka eli RPA on noussut yhdeksi menetelmäksi automatisoida tietokonepohjaisia prosesseja. RPA:n käytön tavoitteena on toteuttaa ohjelmistoautomatisointi olemassa oleviin manuaalisiin tai puolimanuaalisiin prosesseihin. Erilaisten automaatiomenetelmien välisten toiminnallisuuksien ymmärtäminen on tärkeää, jotta saadaan paras mahdollinen tulos ohjelmistojen automatisoinnista. RPA itsessään ei välttämättä ratkaise kaikkia tarvittavia IT-automaatiotarpeita, ja lopullinen automaatioratkaisu edellyttää erillisiä komponentteja, jotka tukevat haluttuja automaatioratkaisuja kattavammin. Kattava IT-järjestelmän automaatioratkaisu koostuu monista automaatiotyökaluista, joissa RPA voi olla keskiössä ja ohjata ohjelmistoautomaatiota.

Tämän työn tarkoituksena on käsitellä pääasiassa RPA:ta ja osittain muita vaihtoehtoisia automaatiomenetelmiä. Lisäksi tavoitteena oli kuvata sovelluspohjaista automaatiosovellusta sekä tarkastella RPA:n käyttöönottoa kontitetussa ympäristössä avoimilla työkaluilla.

Visual Code -kehitystyökalua on käytetty Docker-konttien toiminnallisuuksien luomiseen ja kehittämiseen.

Tämän seurauksena opittiin paljon Visual code -sovelluskehitystyökalun käytöstä konttien kehityksessä sekä sen käytöstä Windows Linux -alijärjestelmän kanssa ja ympäristön sovellusten eri toiminnallisuuksista.

Avainsanat (asiasanat)
RPA

Figures

Tables

ACRONYMS

AI	Artificial Intelligence
API	Application Programming Interface
BPA	Business Process Automation
BPM	Business Process Management
CI/CD	Continuous integration / Continuous delivery
CRM	Customer Relationship Management
OS	Operating System
OSI	Open System Interconnection model
OCR	Optical Character Recognition
POC	Proof of Concept
RPA	Robotics Process Automation

ROI	Return on investment
UI	User interface
XML	Extensible Markup Language

GLOSSARY

Agent: "a component of software and/or hardware which is capable of acting exactingly in order to accomplish tasks on behalf of its user".

Automation: "the creation and application of technology to monitor and control the production and delivery of products and services".

Bit: The smallest unit of information on a machine. Shortening of binary digit.

Digitalization: Process to turn information to digital computer readable format in which information is represented into bits.

Participant: A process that may contain one or more sub-processes, activities, and tasks.

Process: "a series of actions or steps taken to achieve a particular end. Overall process has a start and an end."

Robot: Device or software that automatically performs complicated, often repetitive tasks. Robot refers to some actual robot.

Robotics: Conception of design, manufacture and operation of robots which replicate human actions. Technology dealing with the design, construction, and operation of robots in automation.

Robot Framework project: Robot framework project and its folders´ constructions with its functionalities. At RPA context one project could cover several functional robots.

Log: A file that records events that occur in an operating system or other software runs.

Microservices: Self-contained independent application unit.

1 Introduction

1.1 Robotic Process Automation

Automating business practices and IT systems started in the early beginning of computer age. Solving a business problem might need several applications to finish the task, and information is spread into a separate location in the computing system domain. Robotic Process Automation could be one of the solutions for integrating information into a continuous process flow and for having a control and monitoring for processing of the tasks.

RPA (**Robotics** Process Automation) is a rather new technology and RPA is also known as intelligent process automation (Wibbenmeyer 2018). RPA is basically an additional software to automate repetitive tasks and run allocated process in a specified order of tasks. RPA automates computer-aided tasks assisted by separate software robots on top of applications. RPA itself does not include additional functionalities e.g. databases or orchestration. Therefore, the approach in this thesis is to collect separate participant IT system functions using docker containers as unit of software containers and give easy portability and scalability when functionalities are configured into containers.

When operating in commercial RPA software the RPA process development is collected under one user interface. Usually when developing solution on open source tools this kind of collection of components in to one user interface development environment does not exist. During this thesis open source development environment was targeted on purpose.

One of the main components of RPA software is a software robot. Software robot is piece of software which executes allocated programmed tasks. Software robot is classed synonym of information agent in classification of software agents. (Nwana 1996, 6-29.) Software robots function application-independently on presentation

layer performing tasks in behalf of existing user software. The basics of RPA are adapted from desktop and business process automation. In practice, RPA chains tasks to sequence into process and provide automation with robots which manage the allocated tasks.

"Robotic Process Automation is one of the most advanced technologies in computers science, electronic and communications, mechanical engineering, and information technology" (Madakam, Holmukhe & Jaiswal 2019).

1.2 Automating processes

What is automation and what are the benefits that it provides? This brings another question: why do humans need to automate things and processes?

The main benefits of the automation systems are higher production rates, increased productivity, a more efficient use of resources, better product quality, improved safety and reduced manual errors (Dheeraj, Nimawat, Ashish & Shrivastava 2016).

1.3 Purpose of RPA

The main purpose of RPA is to imitate repetitive computer based human tasks. The purpose of implementing automation are e.g. cost reduction, improve productivity, availability, reliability, and improve quality.

With the help of a software robot, RPA automating processes to a flow of tasks to improve activities in business or in industrial environment. Within RPA it is possible to provide a continuous chain of allocated tasks for existing process. Overall typical office processes are functioning under several different independent software and RPA helps to interconnect and chain tasks from several independent sources to a continuous information flow. Typically, a computer system is a combination of several layers of computer systems and interconnected devices.

1.4 RPA presentation layer integration

OSI (Reference Model for Communication between Open Systems) is defined as one of the commonly known reference models of data communication structure. Each layer has a defined standardized purpose and defined protocol.

OSI model is a standard communication between hosts and applications (Reynders, Mackay, & Wright 2005, 2).

The ISO OSI reference model (Figure 1) helps to define abstraction of different layers of communication (ISO/IEC 7498-1:1994, 32-49.)

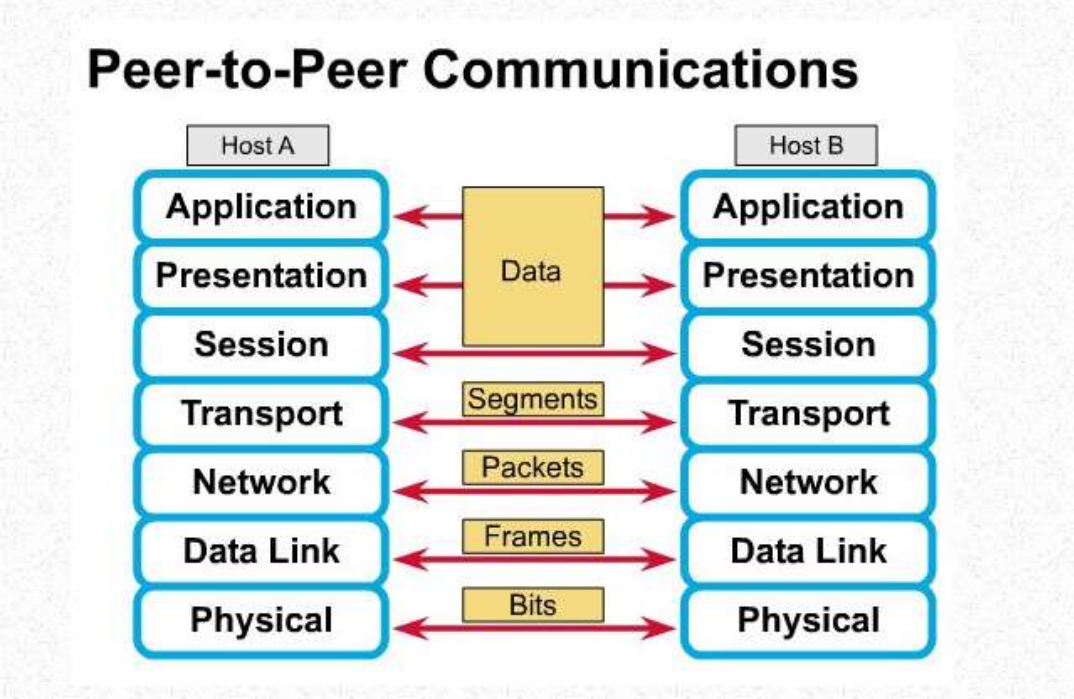

Figure 1. OSI reference model peer-to-peer communication

The presentation layer translates data to a format interpretable for humans, and it is there where the RPA functions.

RPA functions on the presentation layer (Willcocks, Lacity & Craig 2015a, 9). When making interoperability and connection between existing application and RPA software, information transferring is not standardized, and it is not an OSI model protocol-based communication implementation as described in Figure 1.

Presentation layer integration technique has been used for decades. For example, IBM WebSphere uses the so-called enterprise architecture integration which is working as a bridge between applications and has screen scraping functionalities. Figure 2 describes RPA software that accesses the application through the presentation layer.

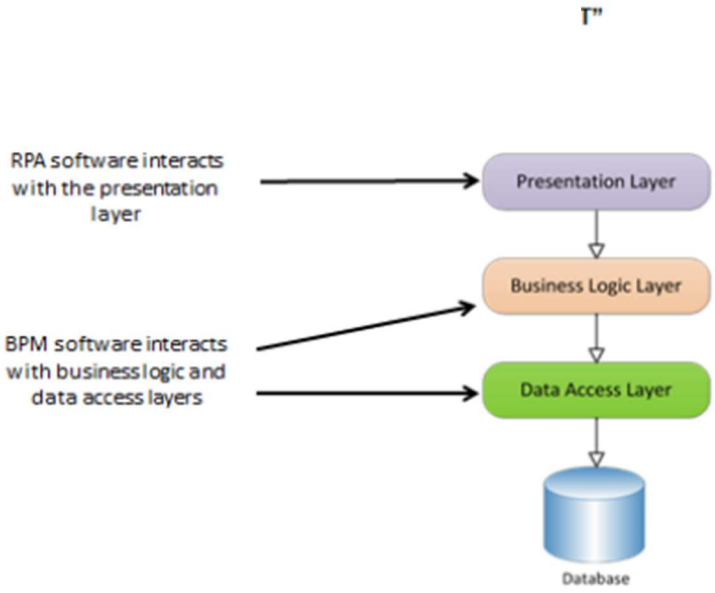

Figure 2. RPA interacts to underlaying applications through the presentation layer.

RPA software accesses other systems through the presentation layer — so underlying systems programming logic is not touched (Willcocks, Lacity & Craig2015a, 9).
RPA helps to prevent discontinuity within the computer-aided processes and needs of human intervention to “connect” activities to continuous information flow. RPA tools provide this ability to connect different kind of systems and combines spread data into information flow with less or completely without human intervention. Software robots are “workers” which enable and feed automated processes.

2 Automation

2.1 Automation

The essence of Industrial revolution was a replacement of animal muscles by machine (Cleator 1955). In the early history of pre-industrial revolution at 15th century knitting stocking frame, also known traditionally as framework knitting developed by William Lee, was made with the purpose to imitate the movements of hand knitters.

The impact of knitting machine improved fabrics' production time and quality. Within this time in the 15th century, there was no existing automation of the machine or additional auxiliary power. The knitting machine was fully operated and powered by humans. The knitting machine's role was to **help and automate** knitting phases with its constructive and standardized way. Knitting is standardized, repeatable work and the purpose of the knitting machine was to improve this workflow by automating reparative tasks.

The first industrial revolution starts in the 17th century when machines were equipped with auxiliary power, and the aim was to provide automation for the sake of increased efficiency. This mechanization is now being followed by a process of automation, particularly of a routine nature (Cleator 1955, 7).

2.2 Software automation

Automation as a term came from manufacturing industry and was first invented by Del Harder of the Ford Motor Company in 1947 (Grabbe 1957, 40). Software automation started at the early beginning of computing era. The goal was to create standard systems that are logical and reliable and do not break (Grabbe 1957, 34). These requirements are still valid today.

2.3 Waves of automation

Automation needs emerge from process management and improvement needs of making processes more efficient and at the same time improve product (output) quality. Figure 3 shows how different computer technologies have developed (Gazova Papulova, & Papula 2016, 199).

Table 1 Three waves of Process Management

Phase	Period	Focus on	Management of organization	Technology	Tools/ methods
Industrial age	1750–1960	Specialization Productivity performance Cost reduction	Functional Hierarchy Line production Orders/Controls	Mechanization Standardization Data Storage	Scientific management Financial Modeling
Information age					
The first wave - Process improvement	1970–1980	Quality Management Continuous flow Task efficiency	Diversification of companies Fusions and acquisitions	Automation Information Technology Management	TQM Statistical process control Process Improvement Methods
The second wave – Re-engineering	90´s	Process Innovation Best practices Better, Faster Business over the internet	Flat organizations Value Added for customers operational excellence	Enterprise Architecture ERP CRM Supply Chain Management	ABC Six Sigma Process redesign Methods of Reengineering
The third wave - Process Management	2000 +	Evaluation Adaptability Agility Continuous Change	Network-centric organizations Hyper-competition Market growth Process effectiveness before efficiency	Enterprise Application Integration Architecture oriented on services Performance Management Software BPM Systems	BSC BPM methods Outsourcing

Figure 3. Waves of automation in information age perspective

Typically, requirements to automate IT systems and processes start from the need of IT system development and business needs. Implemented automated software platform and system processes add value with the means of more accuracy in computing processes, more reliable information handling and computing task efficiency. In implemented automation the processes become less human-dependent. In RPA, robots are used for automating tasks and performing automation in the allocated processes.

3 What is RPA?

3.1 RPA definition

IEEE (Institute of Electrical and Electronics Engineers) Standard association describes **robotic process automation (RPA)** in the following way:

" A preconfigured software instance that uses business rules and predefined activity choreography to complete the autonomous execution of a combination of processes, activities, transactions, and tasks in one or more unrelated software systems to deliver a result or service with human exception management." (ISO 2755:2017, 11)

RPA integrates and chains a sequence of tasks to process and provides automation with robots. The program is a process inside of a machine.

RPA automates computer-aided tasks on top of applications. Software robots function application-independently performing tasks on behalf of existing user software. RPA integrates actions into a sequence of tasks to automate the whole process by using assisted or non-assisted robots and with RPA operating on top of the user interface. Robots can manage multiple type of tasks, e.g. gather data, generate reports, run applications, manage data, and communicate between hosts and processes. RPA is thus operating in the center of control and the monitoring platform in a separate software.

3.2 Defining RPA in detail

Robotics

Robotics is technology dealing with the design, construction, and operation in automation (Marriam-Webster 2019). RPA abbreviation is refering to robotics, not robots: robots are individual devices running the automation solution.

Process

In RPA, a part of the process is configured to a chain of tasks working as a process. A process has a start and an end. Process initiates and ends the sequence of tasks.

Automation

Automation implies a sequence beginning with input and proceeding with output (Grabbe 1957). Automation controls and monitors the allocated tasks in software domain including agent based routines. Process is an enabler for automating independent tasks to become a flow of tasks supported with robots to execute the process steps.

3.2 Why has RPA now increased its presence?

Computer-aided tasks have been automated since the early beginning of computer age. In the past, terms such as “workflow automation”, “robots and automation” were used, as the purpose of these descriptions is to automate computer and software processes. Dated back to first existence of RPA, Google search was utilized to find out the appearance of the RPA terms and the trend (Figure 4).

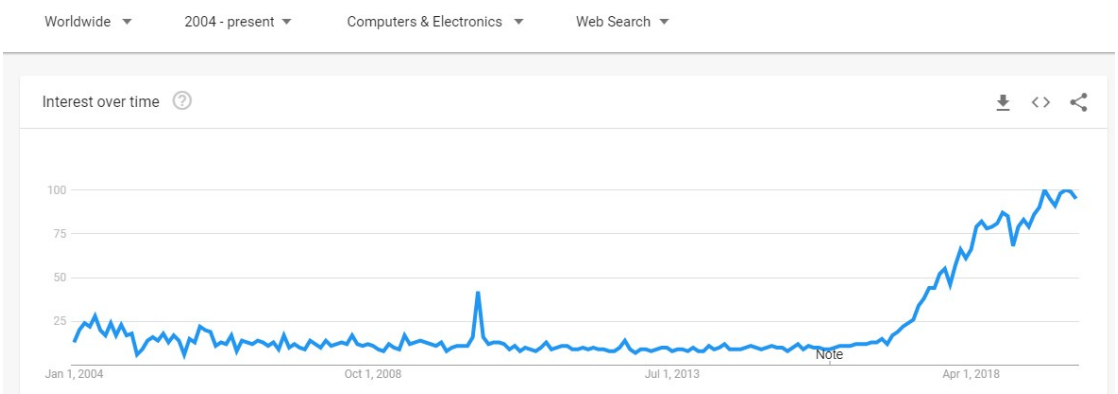

Figure 4. Google trends using search term RPA from Jan 2004 to Dec 2019

As can be seen, the popularity of the search term RPA has increased in the past two to three years. The trend with RPA started to increase significantly from beginning of 2017.

Big investments in RPA have been made in the past few years (Taulli 2020,1.) The need for automating application processes has been there for a long time, and in the past few years new software has come to markets. Marketing of RPA technology has also increased the number of new business opportunities for overall in automation approach.

3.3 RPA Automation

With some configuration, robots can be trained to read emails, open PDFs, identify information, enter data into ERP systems, and send email to specific supervisors when ambiguity or errors are encountered. All these actions can be monitored in real time by the user that designed the script, or by other software robots.

When the decision is made to start automating processes, it is preferred to start from processes which have a rule-based, high volume or based on the time spent in stable and existing environment. Low variation in selected automated processes with defined inputs have good ROI potential. The more stable the automated environment is, the more easy the robot implementation is.

3.4 Identifying RPA attributes

When implementing RPA for the first time, organizations should look first for easy wins high-volume, low-complexity processes. Hence, complex, and subjective tasks should be avoided.

Attributes to help identify RPA-appropriate tasks (Willcocks, Lacity & Craig 2015b, 9) are listed as follows. First, well-defined processes are automatable.

- Second, high volume, repeated tasks can benefit more from automation.

- Third, mature tasks should be targeted. They have more predictable outcomes and the costs are known. Automating these types of tasks is less risky.

3.5 When is RPA appropriate?

When starting to implement robot process automation, one should start from recognition of possible automation. (The Art of Service 2020.) The questions to consider are:

- Which processes should be targeted at first for automation?
- How can automated procedures be spread into small steps to suitable for automation?
- What process can result from automation?
- Is the data in a computer-readable format?
- Through process evaluation and feedback, can areas for process improvements be identified?

3.6 Combining RPA with additional technology

AI (Artificial intelligence) is the most promising new technology; RPA could be a feeder and a center of control in the use of AI in efficient way. AI is one extension in RPA process to improve additional robot funtionalities, add more sophisticated AI based data driven tools, and to run more complex task in a process than OCR, Data classification and other advanced algorithms.

3.7 Chain RPA with intelligent software

The purpose of RPA is to automate IT systems and software functionalities. Additional functionalities such as AI and OCR could be included into an RPA process to automate more complex computing tasks into an automated chain of tasks.

3.8 RPA process with AI extension

Artificial Intelligence has a central role in robotics if the connection is to be intelligent. Artificial Intelligence addresses the crucial questions of what

knowledge is required in any aspect of thinking; how that knowledge should be represented; and how that knowledge should be used (Madakam, Holmukhe, & Jaiswal 2019). Figure 5 describes the transformation from a process-driven to data-driven approach.

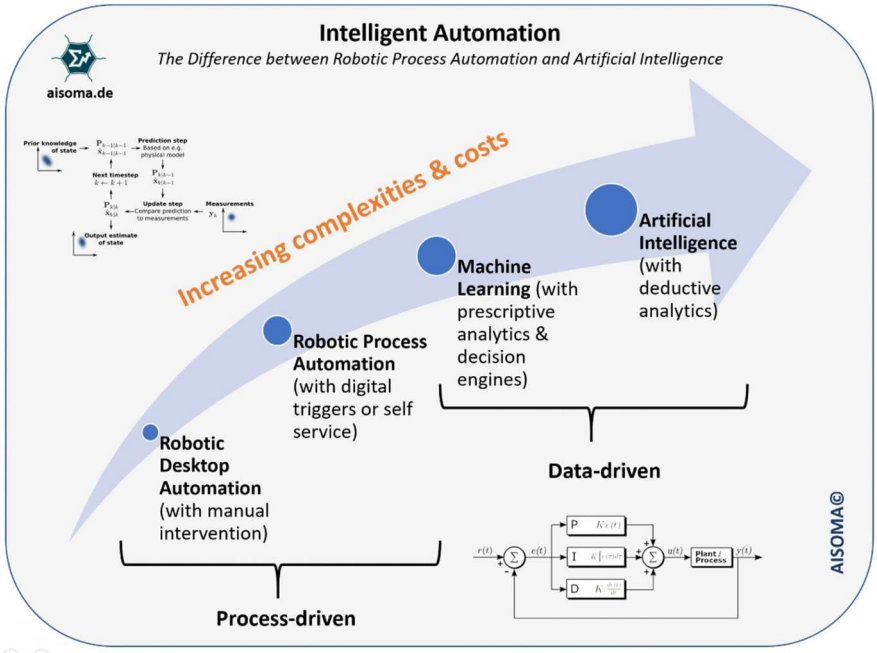

Figure 5. From Process-driven to Data-driven

3.9 Future trends

As has been described here, RPA and AI are new emerging technologies and they help to improve software process as AI is a data driven technology like decribed in Figure 5. Benefits of using RPA in business processes and top areas to use RPA are described in Figure 6 below.

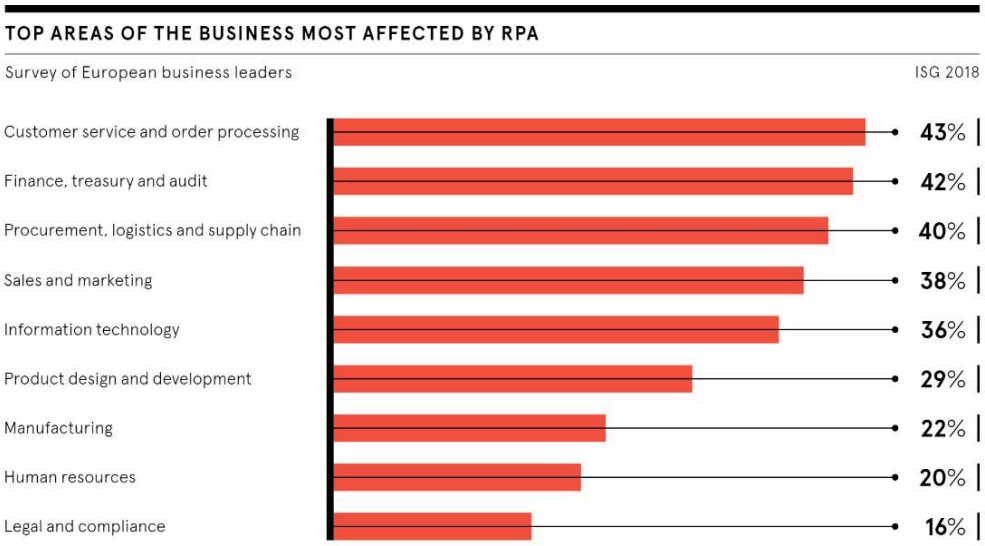

Figure 6. Top areas affected by RPA

3.10 Software automation tools and methods

There are plenty of automation tools and methods available. Some of these tools are with a more wider spectrum including more features and functionalities, and some of the tools have miner capabilities to automate only some parts. Suitable applications must be studied individually and in combination to evaluate their suitability for computer processing (Computers and automation 1955, 15). In the start of automating processes proof of consept (POC) for automated process between several RPA (Willcocks2017, 5) compared to an IT solution is to consider the most suitable solution which fullfills the required business requirements. When organizations consider proof of concept for RPA, they look at the business case and compare it to an IT solution.

3.11 Different automation methods

Automating IT systems is not a new invention and there are several methods to automate IT-based processes. With this thesis, no other method than RPA is studied and when making software automation, the solution is a combination of several automation methods. RPA itself does not solve all automation needs; other

automation tools could be more suitable to solve a problem. It is a business decision what software to use to automate processes. Table 1 shows automation tools which could be used to automate computing tasks.

Table 1. Different automation tools

Tools	Tool execution	Automation task example
Python file	Excecuting tasks by programming	for example, to write code that read complex file structure from various sources
Terminal scripts	Running simple process in command prompt	Running on command line interface. Open paths. Files programs . etc
RPA tools	Platform which is allocated to design and run a process	Handling overall automated processes
Macro	Rule or pattern how certain input sequence should be mapped to output	Sorting table structure in a specific software
Asible	Automated IT infrastructure of a server	Application deployment, Configuration management

The final automation solution could be a combination of several automation tools and software.

Terminal scripts

Terminal scripts are a shell or bash script file which executes a series of commands in sequence. The main purpose of terminal scripts is to batch series of operating system

commands into a single file using repeated needs. Terminal script files are operating system dependent. They are not so suitable for wider automation. Terminal scripts function under the operating system and need additional programming to create a sequence of commands. Separated terminal script files as a solution are not easy to maintain or service.

Macros

Macro stands for macro instructions. It is a programmable pattern to translate input/output. Macros can be used to instruct and repeat frequently used activities in a software, e.g. Excel.

Python scripts

Python is a programming language that provides an enormous range of capabilities to make software automation. Serviceability and understanding of independent Python scripts for non-programming person is limited because of written Python syntax.

Ansible

Ansible presents a completely other aspect of automation. Ansible is used in e.g. configuration management within application deployment.

3.12 Functionalities and usability between separate RPA software

Usability of RPA software could be evaluated from user, maintenance, and implementation perspective. A final implementation solution of RPA is expected to be rather same as RPA users might not even see the RPA function “behind” the IT systems. Usability of RPA from user perspective has the benefit that a functioning automated process might not even see the functioning software and results. Final requirements of functioning solution could be equal, whatever RPA software is chosen. The

usability evaluation of different RPA software from maintaining and implementation perspective is not included in this thesis. The usability of a final RPA from solution user perspective is expected to be the same in different software as users might not even see working RPA solution in IT systems. The final automated solution could be a solution consisting of several applications.

3.13 Business process automation technologies

Overall, there are several ways to automate computer processes and RPA is one of the new terms and methodologies. Suitable applications for automating processes can be found in a wide spectrum of application solutions.

BPM

Business process management is also a method for streamlining processes with integrating applications through application interfaces. BPM solutions are for processes which are requiring more IT expertise, and which are more topic specific IT systems like ERP and CRM. Those are also typically more fixed solutions.

Business process automation (BPM)

In business process automation, the method base of principles is the same as in RPA, i.e. to make an interconnection between separate software of its components to make a computer aided process more streamlined. The main difference is that BPA operates at software level and has a direct “connection” between separate applications (Figure 10) and RPA operates on top of software level and has a control function whereas BPA operates completely independently. In BPA, the changes on process control always need changes in the operating program as RPA chances could be done on the “center of operation” desk or dashboard.

3.14 BPM vs RPA

RPA does not replace BPM and rather complements BPM types of processes. There are two features that separate RPA from other BPM tools. (Willcocks, Lacity & Craig 2015b, 9).

RPA is easy to configure

In certain software, e.g. UiPath, Automation anywhere, or Blue Prism development environment users drag and drop activities to automate processes, and the code is generated automatically. The person responsible for business can read and understand the process flow and actions when the task is presented in readable format.

RPA is "lightweight" IT: it does not disturb underlying computer systems.

RPA is a software that supports processes and it is typically adopted outside of the IT department; nevertheless, it operates under the company's IT systems. RPA accesses the computer systems through presentation layer so no underlaying programming system is not touched and RPA does not store any data. In contrast, BPM solutions interact with business logic and data access layer.

RPA does not replace BPM but rather complements it (see Figure 4). RPA and BPM are both different methods to enable automation.

In principle, the difference between RPA and BPM is that RPA is working on top of UI layer and BPM infrastructure touches business logic and data access (Figure 1). From the operational point of view, BPM operates faster than an RPA application. In operation RPA has to open the application as it would take place in practice; it mimics human action whereas in BPM information flows without opening any software in practice. (Figure 7)

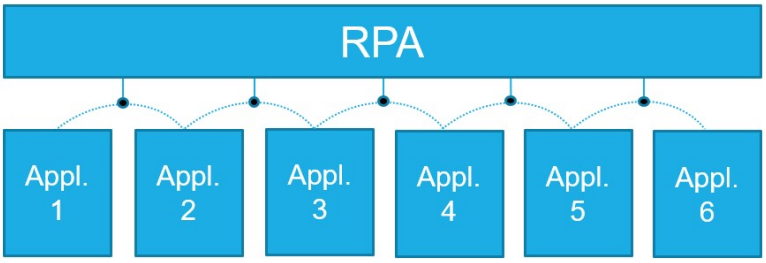

Figure 7. RPA connects information and runs tasks from separate applications

The implementation with RPA is more rapid compared to BPM. RPA provides a more flexible integration when RPA operates at UI level without any API or other program level interface, which might be costly to implement or buy in case of special software which does not provide interfaces as a standard. API could be one option to interconnect interfaces in RPA as well. (Figure 7)

BPM solutions requires more IT expertise and high-valued IT investments like e.g. ERP, Customer Relationship Management (CRM) systems. At BPM application integration is accessed through business logic layer(Figure 8).

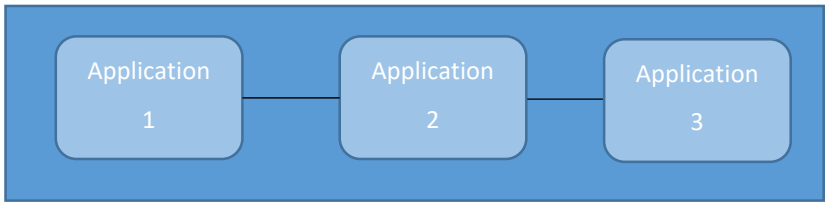

Figure 8. BPM application integration

BPM software is operating on business logic layer and RPA could complement separate BPM solutions. BPM is more fixed solution for automating processes as RPA could keep a temporary solution to provide software automation.

Table 1: BPM versus RPA Adapted from Forrester Research (2014)		
Attribute	**BPM**	**RPA**
Business Goal	Reengineer processes	Automate existing processes
Technical Outcome	Create a new application	Use existing applications
Integration Method	Access business logic layer	Access the presentation layer of existing applications
Developers	Software developers	Business operations
Testing Requirements	System Testing	Output verification

Table 3. BPM vs. RPA feature comparison table

As described in Table 3 (Willcocks, Lacity M & Craig 2015b, 9) both methods' final goal is to automate business processes and technical outcome for BPM is to create an application as in RPA it is to make integration by using existing application.

3.15 Mixed software automation solution and technologies

In practice, when considering an enterprise widely, the entirety of its system automation includes several software and IT system automation. In these implementations RPA could represent a major role in the center of control of several automation solution.

3.16 Mixed automation technologies

At an enterprise level automation, a possible solution is a mix of different BPM automation and RPA tools called mixed hybrid process automation solution.

A correct solution in the right function in the right place provides a more realistic solution and better ROI.

Table 2. Comparison table for two different automation methods

Automation method /Funtionality	RPA	BPA
Need service during operation	Yes	Randomly
Operation Speed	Slow	Fast
Operation layer	Application / User interface	Software
Flexibility	Good	Need detailed programming
Operation monitoring	Standard	Additional programming
Implementation time	Fast	Medium
Technology independent	Completely independent	Tight to existing technology

Implementation is a time-consuming process; however, it is rewarding when automated processes are implemented, documented, and planned in an organized form.

What the best tool for automation is, depends on the business requirements, the level of decided automation and overall automated problem.

4 Containers

4.1 What are containers?

Containers act like a virtual machine and in practise are self-contained processes. Container package isolates a computing unit environment along with its implemented dependences. Container is a standalone executable unit of software that

includes a complete environment to run everything that needs an application. Container becomes a container when a configured instruction file is built up or run by from command line.

Docker is one widely used and commonly known containers' hosting method and because of this, a de facto choice for packing microservices. Unlike virtual machines, Docker allows applications to the same Linux kernel (Figure 10).

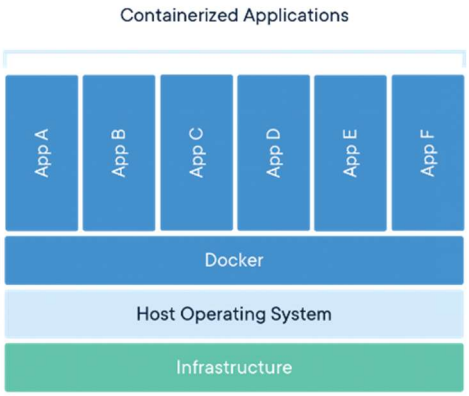

Figure 10. Containers operating top of host OS through Docker engine

4.2 Isolation

One of the main benefits of containers is their isolation capability. Docker allows to isolate applications from each other and from the host environment it also has restriction capabilities. (Introduction to Docker security 2016)

4.3 Portability

Regardless of the related operation system, containers could be run in every operation system, in which a Docker host software is available. Docker enables to separate an application from its infrastructure and allows to manage the containerized infrastructure in the same way as managing applications. Now Docker is available to most common operation systems giving the ability to run an application in any host system

within the same Docker configuration. This gives advantages for application “shipping”, testing, and deploying code more quickly.

5 Containerized Robotic Process Automation application

5.1 Introduction

Implemented RPA processes are mainly used to automate existing company business rules and routine computing work. This describes the selected tools and technology to establish one type of containerized RPA environment with its participants.

The tools are selected so that tools and architecture participants are open source. This is one approach for the running environment to combine additional functional components (participants) within RPA process to support the implemented RPA process.

Overall, Robot Framework functions in RPA context and as a tool in the centre of automating process. Other participants are used as data storage, orchestration, and monitoring functionalities supporting RPA tasks. All applications are “packed” into Docker containers to have better isolation from the surrounding host environment and for portability within different operation systems’ environment.

Process reports from automated process are generated and stored into MySQL database for further processing, and the task status is monitored with Grafana visualization.

5.2 Visual Code Development tool environment

For the development of environment, Visual Code was used with additional Docker, Python, Robot Framework intellisense VScode extensions. This gives more centralized visibility when developing and running the containers.

Production and production environment

Development tools are helpful to build up and develop a Docker application. When releasing a Docker application to production, the container environment is different than it was in the development phase. One of Docker's benefits is its portability between different operation systems. Figure 10 illustrates the differences between container development and production environment.

Development	Production
Use bind mounts to give your container access to your source code.	Use volumes to store container data.
Use Docker Desktop for Mac or Docker Desktop for Windows.	Use Docker Engine, if possible with userns mapping for greater isolation of Docker processes from host processes.
Don't worry about time drift.	Always run an NTP client on the Docker host and within each container process and sync them all to the same NTP server. If you use swarm services, also ensure that each Docker node syncs its clocks to the same time source as the containers.

Figure 10. Differences between development and production environment

5.3 RPA Process Identification

RPA projects start with the process identification of the automated processes. The project start focuses on implementing the actual RPA processes. Existing company processes are mainly implemented using company business rules and predefined actions in business processes. In case of RPA, these processes are "transformed" to automated processes with the help of the personnel.

When one starts to evaluate and establish RPA processes, the size of automated processes starts to grow rapidly in companies and controlling overall RPA project becomes a challenge. This chapter describes one solution of containerized RPA environment and its related participants of infrastructure.

5.4 Environment setup

An environment consists of following participants with following functions (Table 4).

Table 4. Tools of environment participants

TOOL	Function
Robot Framework	RPA
MySQL	Database
Grafana	Results visualisation
Jenkins	Robots Orchesteration

These components were selected and used in the whole architecture supporting the running of the RPA process. This environment with its support of open source functions fulfils and supports the overall RPA process.

5.5 Environment architecture

Architecture includes the components as described in Figure 11. Data storages of the containers are mounted outside from containers in a physical host with related functionalities.

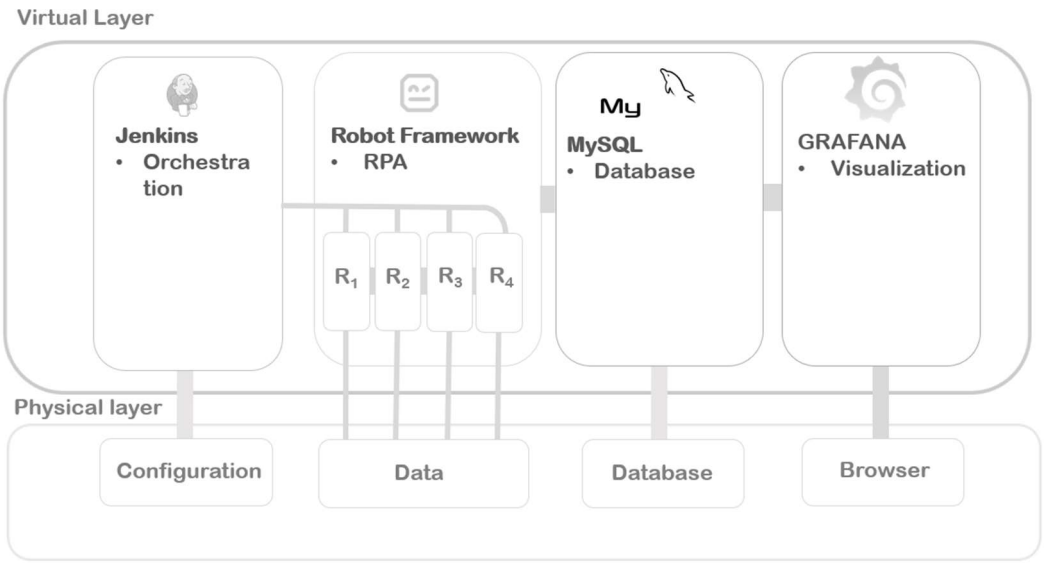

Figure 11. Architecture of RPA and its supportive participants

5.6 Containerized participants

The participants are configured to run in Docker containers, which brings benefits for easy portability and isolation from host system environment. When an environment is "packed" into containers, it has benefits for portability and scalability. When using containerized environment, the portability to transfer the implementation to different operation systems is easy and convenient. Implementing containerized environment also brings additional complexity when setting up containerized application setup.

Mounting volumes between containers and networks needs additional attention in an implementation and the selection of container base image with its relations. Creating docker-compose file is the description of all application in one file. Creating docker-compose file needs several rounds for testing and implementing to have the environment of whole functional components.

When all components are implemented and these communicate with each other, operating with this environment is convenient. Container service management was not

included into this thesis as Swarm or Kubernetes tools, which monitor the services and workloads of the containers.

5.7 Jenkins

Within Jenkins open source tools, the connection to run several robots could be made. With Jenkins also simultaneous robot execution could be performed.

Jenkins is used as an orchestrator of several robots in this environment. When RPA implementation gets wider and robot quantity increases, robot status and control need more focus. When orchestrated with Jenkins, distributed robot architecture gives the ability to use resources from an allocated container (worker nodes), in this case each robot represents allocated Jenkins worker node. The communication between master-worker architecture is managed by secured network protocol (SSH) communication.

Implementing Jenkins was straightforward. Allocation of robot execution from Jenkins connection to workers and instruction to slave “what to do when executing by master” must be described. This is done by implementing shell or bash instruction scripts to master. Jenkins master container controls and executes allocated robots through Jenkins workers in a centralized manner.

Additionally, Jenkins could also be configured to collect and view task reports. In a wider scale of implementation into multiple robot projects controlling of running robot becomes challenging. To enable to collect robot control in one place gives more visibility to control RPA environment. Jenkins was implemented to execute the robots by configuring the shell script.

5.8 Database

Database is used to collect information from an executed task in to one place for further visualisation or analysis as RPA itself does not provide this functionality. A

database could also be used to store and collect data processes when the process is running on additional packages. Robot reports are serialized from XML output files to tables with an additional library and pushed to MySQL database and tables.

5.9 Serializing Robot Framework output files

An additional Python script DbBot (DbBot 2020) allows to serialize XML output file into MySQL database tables which enable to have unified data storage.

Both the task data (name, content) and test statistics are stored by default. If database does not exist, it is created and when results exist and are inserted into existing database.

Separately, when API is needed, listener interface allows monitoring during each test execution while tests are running but this functionality was not used in this set up.

5.10 Visualization

Using Grafana provides easy to use visualization. Graphs could be easily customized. Robot status and visualization are implemented with open source visualisation tool Grafana. When robot reports are collected into one place, this gives more visibility for the executed processes and helps robots in monitoring and control. Grafana was chosen because it is easy to use and flexible for user made visualisation.

5.11 RPA tools

Robot framework is an open source tool which has a strong background in software development. Robot Framework was developed by Pekka Klärck in 2005 in his master's Thesis. Robot framework development continued at Nokia until 2013 the software was released.

Robot Framework roots are in software testing and since 2016 used in RPA context as well. The principles of automated software testing and automated processes are the same, and the target within these separate contexts is different.

The principle objectives are the same: to automate processes to improve reliability, quality and reduce human work. With test automation, the aim is to find and analyse test results whether the **software** fulfils the given requirements within its domain. (Dustin 2009, 3-21.) The aim of RPA is to fulfil the given requirements in its allocated **process.** RPA does not test the process itself. RPA executes the given tasks from the given process typically based on business requirements and their description. Robot framework is a framework used as a core task runner and produce reports.

5.12 Interfaces

Containers are built in network interfaces where communication is unrestricted between containers in that network. (Docker containers and network interfaces 2020). Each network is associated with the host with bridged network. An exception could be that applications run in a same network but are isolated by firewall from each other.

5.13 Basic security view

User privileges and control (accounts) and root access control

When not, an additionally specified container runs using root user as default. Having an application on the container running with user widens the attack surface and enables privilege escalation. A dedicated user and a dedicated group in the Docker application are created with at least privileges as possible. Best practice is to adhere principle of least privileges. (Docker security best practices)

Minimizing container usage

Limiting the intake of memory and CPU helps increase the efficiency of the environment and prevents an imbalance of resources across of containers and enhances an implemented environment. If a container becomes infected with malicious content, the container does not utilize a bigger amount of resources than configured.

Host machine security

When using volume mounting and operating with Docker demon, also host platform security should considered by encrypting host machine file in addition to access control to file locations.

Container scanning

Using CI/CD gives ability to scan container images for known vulnerabilities in pipeline that checks if vulnerabilities exist and reports possible findings.

4.1.1 Logs

It is important to collect logs. Docker logs are console outputs of a running container, and additional running containers generate logs from events. Additionally, especially robot logs are most important to collect and save for possible history views.

The logs contain the output and metadata of the actual messages. Logs from all containers are convenient to collect in a centralized way and manage with additional log management software for clearer visibility of the evens. Centralized event log management enables filtering the most significant events of data.

6 Conclusion

As history of automation has changed the way to work, and one of the emerging technologies is RPA to change the way of working today by automating computerized work. When automating processes, the roles will change more from use of computer as an additional “tool” to co-operation with computer systems as a center of operation. In this transformation an employee’s role switches from a front-end computer user more to a process operator role.

Within RPA technology working together with AI has massive capabilities in interaction with humans. RPA itself gives a platform to “parse” easily different several systems to interoperate, and artificial intelligence brings “intelligence” on top of RPA operations. This thesis scratched the capabilities of open source RPA and possibilities of interaction of another supporting participants. Developing open source software using visual code editor with additional extension give better visibility especially for container development and collection of several participant configuration into single interface. Additionally, using WSL2 on windows give simpler for example volume mounting and Linux system environment under Windows main operating system. Additional AI application interaction in the same platform was not included to see how to refine data to valuable information. AI kind of additional functionality could give a deeper data-analysis and give insight to unseen information with the help of RPA. RPA provides “mechanics” of data collection and control of an entire system.

In the future, there will be more and more RPA automation, and its impact and implementations are unavoidable. Finally, moving towards to automated processes is a strategic decision, not an operational, to stay and be on cutting edge in competition.

RPA is mainly used to automate business processes, and this explains the very basis of RPA in containerized environment. Containers give easy portability and fulfill the need easy to use when configured. Starting to develop and establish a certain containerized environment from scratch also need basic IT-system level understanding

e.g. network communication, volumes, images etc. When these IT system level functionalities are understood, developing applications into container is easier.

Why not use overall application and system automation starting from automating as much as possible of computing work? When automating as much as possible, RPA could be one of the methods in the center of overall automation with robots on top of automate processes. When the decision is made to use robotic process automation in processes, the final automation goal is to automate as much as possible.

One can use automation tactically for cost savings. But if you use RPA as a broader strategic tool, one gets much more out of it. (Willcocks 2017)

References

Docker development best practices. Accessed 2 May.2020. Retrieved from https://docs.docker.com/develop/dev-best-practices/

Cleator P. E. 1955. The Robot Era. New York: Thomas Y. Crowell.

Grabbe E. M. 1957. Automation in Business and Industry. New York: John Wiley & Sons.

Gazova A, Papulova Z, & Papula J. 2016. The application of concepts and methods based on process approach to increase business process efficiency. Accessed on 2 May 2020. Retrieved from https://www.researchgate.net/publication/305396460_The_Application_of_Concepts_and_Methods_Based_on_Process_Approach_to_Increase_Business_Process_Efficiency

Increasing Productivity through Automation
Dheeraj Nimawat and Ashish Shrivastava. Accessed on 27 December 2019. Retrieved from http://www.ejaet.com/PDF/3-2/EJAET-3-2-45-47.pdf.

ISO/IEC 7498-1:1994 Second ed. Switzerland. Accessed on 2 May 2020. Retrieved from https://www.iso.org/standard/20269.html

IEEE 2755:2017. Guide for Terms and Concepts in Intelligent Process Automation. Accessed on 1 May 2020. New York: IEEE. Retrieved from https://ieeexplore.ieee.org/document/8070671

Kevin C, Andrea M, Miklos A. Robotic Process Automation for Auditing. Accessed on 2 May 2020. Retrieved from https://meridian.allenpress.com/jeta/article/15/1/1/9413/Robotic-Process-Automation-for-AuditingNwana,

Network study content. ISO reference models. Accessed 20.4.2020. Retrieved from http://www.dbillings.com/networking/anotherosi.htm

Nwana, H. 1996. Robot and agent description classification Accessed on 2 February 2020. Retrieved from http://www.upv.es/sma/teoria/agentes/software%20agents-nwana.pdf

Robot. 2020. In *Merriam-Webster.com*. Accessed on 20 February 2020. Retrieved from https://www.merriam-webster.com/dictionary/robot

Robotics Process Automation A complete guide 2020 edition, the Art of Service 2020. Accessed on 1 May 2020. The Art of Service.

Taulli T. 2020. The Robotic Process Automation Handbook: A Guide to Implementing RPA Systems. Monrovia: Apress.

Willcocks L, Lacity M & Craig A. RPA access the presentation layer of existing application. 2015a. Accessed on 10 March 2020. Retrieved from https://eprints.lse.ac.uk/64519/1/OUWRPS_15_05_published.pdf

Willcocks L, Lacity M., & Craig A. Robotic Process Automation at Telefónica O2.2015b. Accessed on 20 December 2020. Retrieved from http://eprints.lse.ac.uk/64516/1/OUWRPS_15_02_published.pdf

Willcocks L. 2017 The value of robotic process automation: An interview with Professor Leslie Willcocks. Accessed on 10 April 2020. Retrieved from https://www.mckinsey.com/~/media/McKinsey/Industries/Financial%20Services/Our%20Insights/The%20value%20of%20robotic%20process%20automation/The-value-of-robotic-process-automation.ashx

10 Useful AI & MK Slides. AI and intelligent automation. Accessed 12.1.2020. Retrieved from https://www.aisoma.de/10-useful-ai-ml-slides/