

Development of an application for teaching PLC and robot programming based on OPC UA

C. Zieringer¹, Prof. Dr.-Ing. C. Wittenberg², F. Schneider³

Abstract

This paper describes the development of a proof-of-concept application, which can be used to train people in the field of programmable logic controller (PLC) and robot programming. This concept leads to a better understanding, what kind of infrastructure and software packages can be combined, to not only have a visual representation of an industrial like scenario, but also the possibility to use commonly established programming environments such as TIA Portal, to set up a PLC program. In addition, it provides insights on how OPC UA can be used and is suitable for controlling a typical six-axis industrial robot with the help of a PLC.

Keywords

Unity3D, TIA Portal, S7-PLCSim Advanced, OPC UA, Industry 4.0, PLC, gamification, serious games

1 Introduction

The industry is undergoing a significant transformation regarding the way we produce products thanks to the digitalization of manufacturing. This fourth revolution is so compelling, that it is being called Industry 4.0 (i4.0). This revolution is based on the connection of computers to communicate with one another. A combination of cyber physical systems (CPS), the Internet of Things (IoT) and the Internet of Systems (IoS), makes this transition become possible and the smart factory a reality.[1] The characteristics of a smart factory, which are visibility, connectivity and autonomy, are achieved through an extensive use of IoT sensors and devices, to connect the plant internally and with other entities in the digital supply network [2]. So called digital twins, which act as a mirror of the real world, become a central concept in those smart factories. By using such digital representations, together with intelligent algorithms, organizations can achieve data driven operation monitoring, simulation, predicting and optimization of physical manufacturing systems and processes.[3] With the emergence of such technologies, there is also responsibility for the extent to which this affects or adversely affects the supply-chain industries. Some experts point to a risk of inequality developing in society - those providing purely labour will become dispensable, while those who provide intellectual and physical capital are the biggest beneficiaries.[4] A possible help could be new platforms for independent or guided learning of the core concepts of this revolution. With the emerge of i4.0 and the enhanced connectivity of machines and devices, there was a new educational paradigm created, called Education 4.0 (e4.0), which builds on learning by doing and intends to address the needs and potentialities of the fourth industrial revolution. E4.0 occurs in complex virtual learning environments (VLEs), where there is a greater need to have interactive and collaborative educational components. A key point in the addition of VLEs is the technological dimension. In this sense, the use of emerging technologies

¹ TriCAT GmbH

² Hochschule Heilbronn

³ TriCAT GmbH

such as robotics and virtual reality (VR) emerge as educational practices in pioneering projects.[5] Using these technologies, a specialized application to train people from around the globe in the field of i4.0 is possible to develop. Therefore, this research contribution, which arose as part of a master's thesis at Heilbronn University in cooperation with TriCAT GmbH, deals with one possible method for developing such an application.

2 Concept of the Application

The concept of the application can easily be split into two larger parts. The information or data layer, that contains logic and communication, and the operating layer through which the application can be coordinated and controlled. The data layer determines the structure of the infrastructure and will be seen and named as such in the following. The operating layer corresponds to a user interface with graphic elements.

2.1 Infrastructure

The basic concept was developed in such a way, that it can meet the requirements of professional PLC training. It should therefore be possible to use software that is also widely used in industry. Also, the simulation should only differ slightly from reality. So, the choice fell on the PLC programming environment TIA Portal V15.1 [6] in interaction with the PLC emulation S7-PLCSim Advanced v2.0 SP1 [7]. The emulation provides a software PLC on which the programs created in TIA Portal can be loaded and executed. In addition to that, there is also the possibility to use the API [8] to interconnect with any co-simulation. This allows sending data, like in- and outputs, or commands, like start and stop, back and forth from the PLC to the simulation environment. As a central framework or in that manner a co-simulation, Unity3D [9] was used. Its openness makes it possible to combine different software packages with the typical characteristics of a game engine. Within the game engine, the user interface, different devices like sensors, robots, grippers and conveyer belts were realized. Those devices can be placed freely in the scene to let trainees create scenarios on their own. The sensors serve as inputs to the PLC, whereas the actuators such as the motors of the conveyer belts serve as outputs. With that alone, it's already possible testing simple PLC programs. To go one step further, as already mentioned, robots are integrated which send and receive data and commands to or from the PLC via OPC UA. For this purpose, the .NET OPC UA SDK of Unified Automation [10] is used, which creates a Namespace for each robot and each gripper placed in the scene, on one common server running directly in the Unity3D framework. The software PLC establishes a client connection to this OPC UA server. With the help of ready to use data blocks [11] within TIA Portal, the robots and grippers can be logically included into the PLC program. The resulting concept is shown using the following simplified diagram.

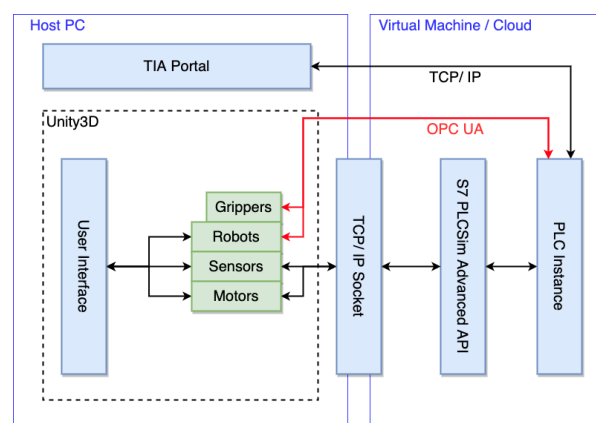


Figure 1 : Infrastructure of the application

2.2 User Interface

The application should be based on widely used interface interactions, to easily navigate and operate. So, there is a Drag & Drop menu created to drag out the different modules like conveyer belts, sensors and robots with different grippers attached. Additionally, there is a menu to start and stop the other components used, like the PLC and the OPC UA server. The user can configure the server and the PLC emulation by typing in an IP address or give the PLC instance a name, under which it can be found again in the network. Furthermore, a feedback interface was integrated to give the user all kinds of information like the status of the server, or the started PLC and the messages sent to the virtual machine. There is also a message box on the top of the screen which informs the user about certain errors or warnings and provides support by showing tooltips. To program the built industry example, there should also be a list where all the in- and outputs of the sensors and motors, which communicate with the PLC, are listed in a tag menu with their respective value. There may also be the option to set these values manually to true or false, to quickly test the built system. To prevent the necessity of rebuilding the scene and arrange every object from scratch, a saved scene can be loaded again. The operator should also be able to decide whether the menu is expanded or hidden, to have a wider range of view. To have objects for interaction, there is a short menu to spawn objects, like boxes or plates, on a designated spot the user can decide.

3 Method

First, the planned modules were created and prepared for import into Unity3D. If a skeleton is required to implement inverse kinematics (IK) as is necessary for the robot, this can be done using Blender or a comparable 3D modeling environment. As soon as the models are imported, a gameobject can be generated, to which new properties or so-called components can be added. These components are then responsible for how they behave within the scene. In the case of the sensors, they are responsible for exchanging data. The data is sent via TCP/IP connection to the virtual machine if the sensor detects an object. This data is transferred to the memory of the PLC instance via the S7-PLCSim Advanced API. The program created by the user then decides how the data will be processed. The sensor can be logically connected to a motor on a conveyer belt. The conveyer belt will receive the corresponding command in the next update cycle and then behave accordingly. This easy method can be transferred to other modules, depending on whether the module is to be defined as an input or an actuator. This relationship is illustrated in the following diagram.

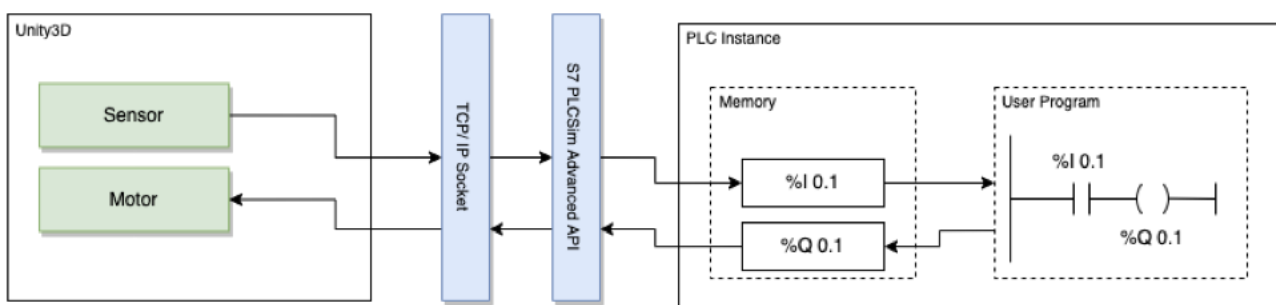


Figure 2 : Basic communication of the modules and the PLC instance

As already mentioned above, the robot needs IK for a more convenient operation. Later, the movement can only be defined via the tool center point (TCP), which defines the end point of the kinematic chain. The individual members will then align themselves accordingly. For solving the IK, a Unity Asset called FinalIK[12] was used as the IK solver. The grippers were animated, because they only have very simple

clamping movements. In addition to use these modules in the application, the robots and grippers receive components that automatically create an OPC UA server, which contains the corresponding information models. The information model the robot can be taken from the specifications of the OPC UA Foundation [13]. The namespace of common grippers was still in development phase up to this point, but an actual state of the current development provided insights about the final structure. A newer source is available under [14]. A state machine forms the core concept of the operation and control of the gripper as well as the robot. A short extraction from the specification describes this in more detail.

Table 1 : Description of the state machine

Transition	Name	Cause
1	HaltedToReady	Reset Method
2	ReadyToRunning	Start Method
3	RunningToHalted	Halt Method or Internal Error
4	RunningToReady	Internal
5	RunningToSuspended	Suspend Method
6	SuspendedToRunning	Resume Method
7	SuspendedToHalted	Halt Method
8	SuspendedToReady	Internal
9	ReadyToHalted	Halt Method

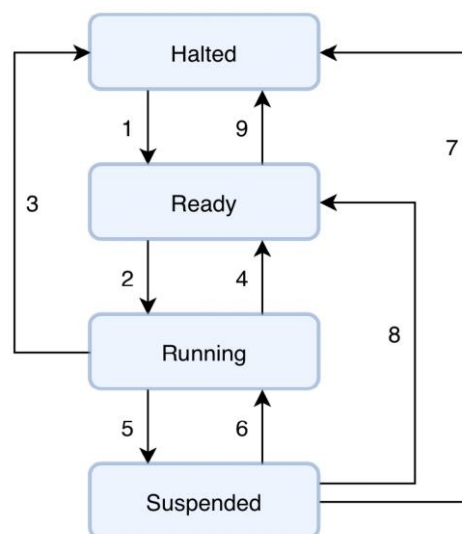


Figure 3 : The state machine as described in the specification [15]

As with the idea of a state machine, there is already a scientific paper [16] on how this state machine can be used to control robots from different manufacturers by a skill-based approach. According to Profanter et al., one major requirement for having generic hardware independent robot, tool and device skills is to use an adaptable underlying skill model. In their view, a skill is a specific piece of functionality which is provided by a hardware or software component. The proposed Information Model of a SkillType that fulfill all the requirements Profanter et al. thought of, can be seen in the extracted diagram.

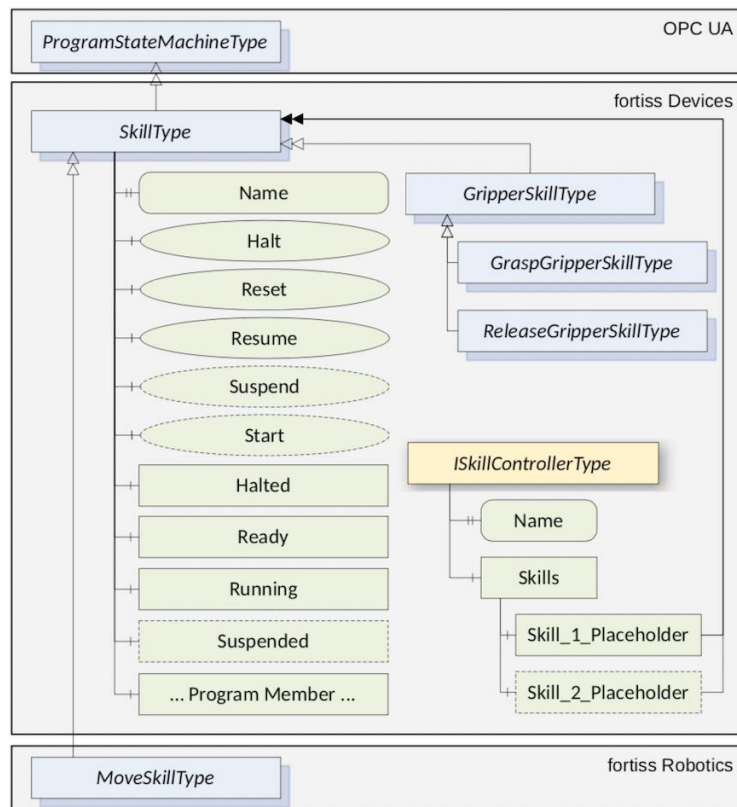


Figure 4 : Description of a SkillType [16]

This concept is used to create routines for different devices. For example, defining PTP movements for the robot or close and open operations for the gripper. The state machine can then be controlled using various methods such as start, stop or pause. Any parameters can also be transferred to this method, in the example of a PTP movement, these can be coordinates, the value of blending points or the speed. Different methods of different devices can also be linked together, through the precise information about the progress of the method. In addition, the idea here is to carry out entire operations such as pick & place tasks [16]. These methods can then be called by the PLC instance in the manner of the user program. To do this, the PLC instance connects to any server via a client and gives instructions to start or stop these methods. Ready-to-use data blocks can be inserted into a network of TIA Portal. The data blocks then are provided with the necessary information. The following diagram shows how the robot and the gripper were integrated into the application.

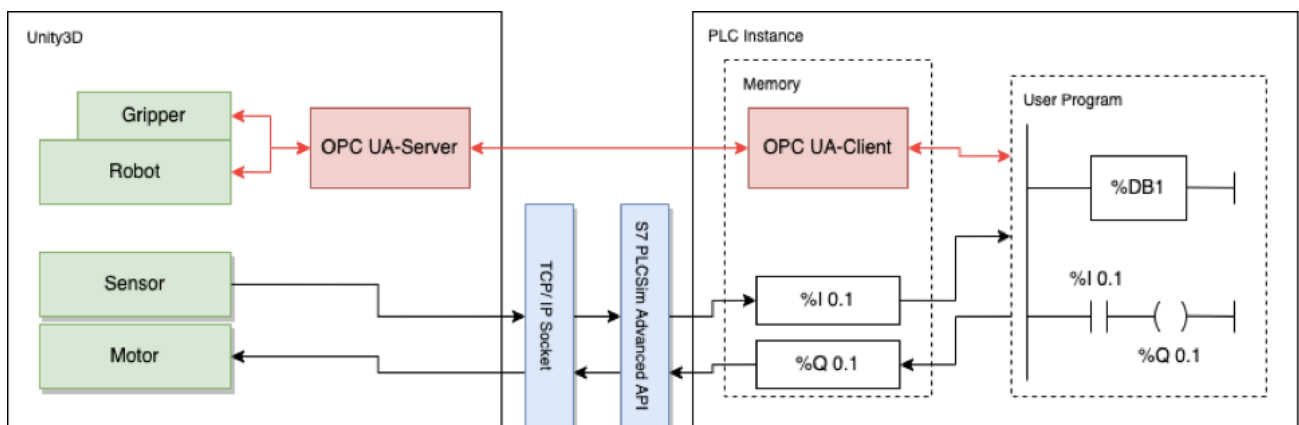


Figure 5 : Integration of the OPC UA components

4 Outcome

The steps outlined in the previous chapters can be implemented in different ways regarding programming or design. They should just be seen as an overview of the different components that are necessary to create such an application with the specified interfaces. The interface of the application developed in this project is shown in the following picture. The numbering is used for description and is briefly summarized in the table below. The use of the software will be discussed afterwards

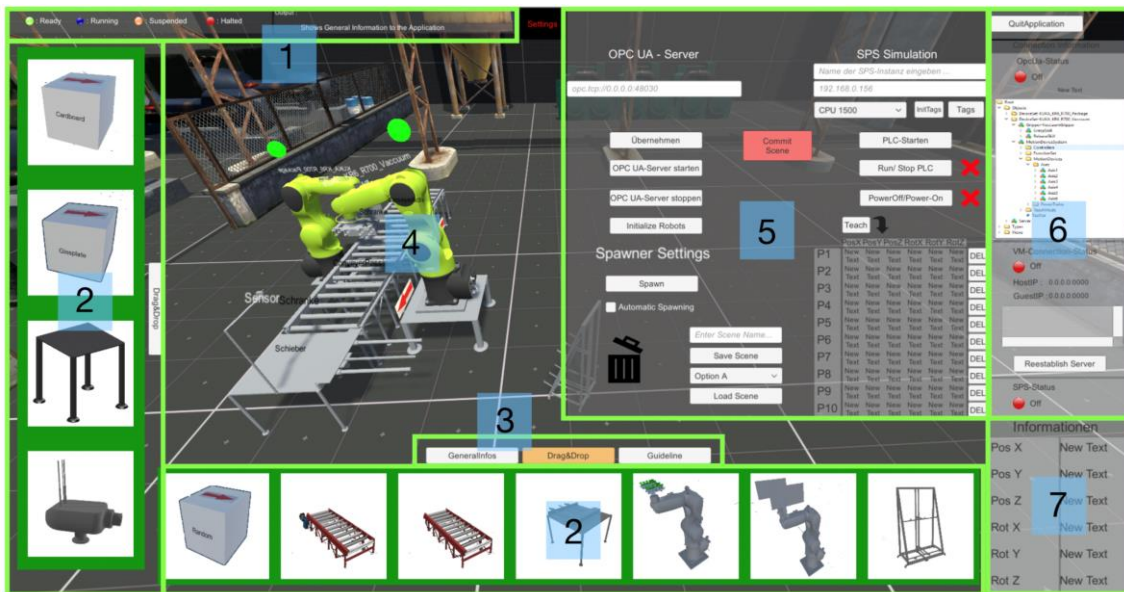


Figure 6 : Overview of the developed application

Table 2 : Explanation of the numbering

Nr.	Description
1	Meaning of the status colors, they indicate the respective status of the robot, plus additional message box.
2	The drag & drop menu, from here the individual modules can be dragged into the scene and can then be aligned.
3	Different tabs for further information or a guideline.
4	Field of view, the menu can be folded in and out to enhance the view.
5	Settings for the PLC instance, OPC UA server, save, load, delete, spawn objects or view the taught coordinates of the selected robot.
6	Feedback system to check the connection to the virtual machine or the OPC UA server.
7	Position of the selected object

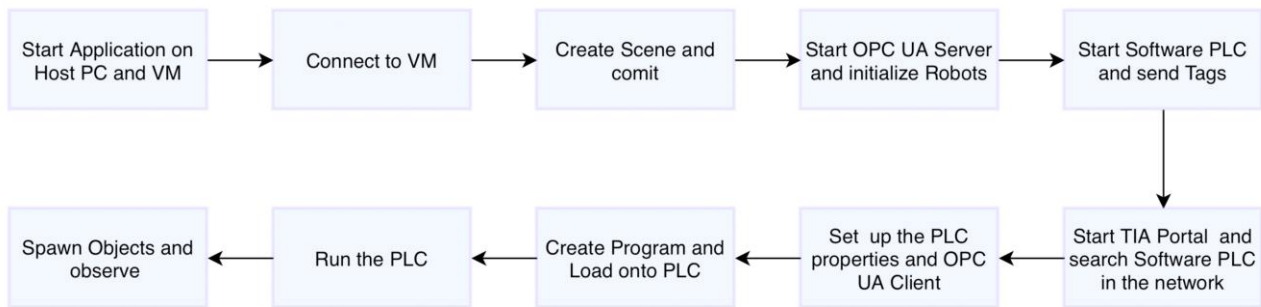


Figure 7 : General usage of the application

In the first steps, the entire infrastructure is started. This can also be prepared by the trainer, if one is available. Then the scene is created and all modules are prepared for data transmission. After that TIA Portal is connected to the created PLC instance and the OPC UA client is set up, it is best to use the wizard for this. The program can now be created, which is later loaded onto the PLC and run through. In order that the system can react to objects, these can be created by the user and transported by the conveyer belt. In the last step, the behavior of the system can be observed.

5 Evaluation

At the end of the work, there was also an evaluation performed with several probands. Due to unforeseeable external circumstances, no large-scale evaluation could take place. Therefore, it must be noted here, that the results of the subsequent survey may be biased, only probands from the immediate vicinity were considered, so the probability is high that the results are shifted to the positive. However, these people did not have any knowledge in the field of PLC or robot programming. The survey showed that the opinion about the application is rather mixed. This is due to the many small errors and malfunctions, but also because of the opacity of the communication. The answers show that the operators need even more support. For example, the infrastructure could be significantly simplified. The first steps to set it up, could also be automated in order to avoid additional sources of error. The results also show that an extended introduction can be very helpful for many beginners. Particularly regarding communication, there should be a detailed explanation on how and why things work, so it may be useful to teach the basics first and then move on to building and programming. There could also be a mission or level system that adapts to the corresponding skills of the trainee. This allows the students to learn independently and at their own pace. According to that a beginner or expert mode would also make sense. When it comes to programming with TIA Portal and the OPC UA data blocks, many find it not entirely plausible. As TIA Portal is a real expert tool for industrial use, beginner has a really hard time and almost no chance to understand the necessary steps right away. But controlling the robot with the help of a client with graphical interface, was considered very easy. This shows that at least a certain requirement must be met when dealing with TIA Portal in order to be able to use this application at all. Merely executing the OPC UA data blocks in a suitable sequence is then a rather minor hurdle. Nevertheless, many of the probands recognized the application as a benefit to traditional teaching and considered that this type of programming would play a greater role in the future. The opinion about the overall application is rather in the middle field. This is where the difficulty of operation, small errors in the program flow and errors in communication due to inadequately secured code play a role. However, it can be said that this is still a success. In order to get such an application ready for the market, many more steps must be taken, and things need to be simplified.

6 Conclusion

The project shows that the idea of a PLC trainings application included robots programmed via OPC UA is possible. With the help of the different used software packages and their APIs, a very complex and featurerich application can be produced. This application fulfils all the requirements to have a very industrial like PLC- and OPC UA training with only software solutions. With the ability integrating and combining the APIs with Unity3D, there can be a very extensive simulation carried out, to visualize the results of the PLC program. The application, as it is in the current state, leaves the user with many options as to what he or she wants to train. The scene can be created individually and all modules are independent of each other. In addition, the manual mode allows the user to try out different approaches and experiment a lot. Additional modules may be added during further development. Any robot can also be easily added since the programming is completely independent of the geometry and manufacturer of the robot. These things already make the application ready for first, smaller but meaningful tests, some of which have already been carried out. The knowledge gained from the tests can be used to continuously improve the application or, if necessary, to set it up again based on this work and the associated test results. These test results show that such an application requires more time and further considerations. The solution with the virtual machine has many sources of error and should be avoided if possible. Therefore, the structure would have to be completely redesigned and put more emphasis on more robust, synchronized communication. Other communication interfaces could also be tested, there one will surely come across other possibilities to expand the application usefully.

7 Future work

There are also other very interesting software packages from Siemens that can be integrated into this application. One would be TIA Portal Openess [17], with its API it is possible to create a whole TIA Portal project from code. It's so to say possible to support the user even further and to integrate different modes like the desired beginner mode. Another attempt would be to connect SIMIT [18] also from Siemens to the application. With SIMIT it is possible to carry out complex simulations. In addition, this software package can be coupled via a shared memory gateway [19] to other software such as Unity3D or using the software PLC instance established by S7- PLCSIM Advanced. In a further advanced, the idea of gamification could have further influence. For example, it is conceivable to create certain scenarios that are based on a field of activity or a task within the industry. After achieving this goal, a kind of reward could be activated, or access to even more complex modules. Furthermore, statistics can be implemented that record the progress of learning or the success of learning. Another algorithm could then suggest tasks to compensate the weaknesses and increase learning success. Another option would be to learn online with several people who can work on their ideas together on the same system and solve much more difficult tasks together. If the online access is then also enabled for experts, they can assess the result or provide assistance even from an exposed position.

8 References

- [1] Bernard Marr : “What is industry 4.0? Here’s a super easy explanation for anyone”. In: *Forbes Magazine* 2 (2018).
- [2] Margaret Rouse : “Definition : smart factory”. 2018. URL: <https://searcherp.techtarget.com/definition/smart-factory> (visited 13/12/2021)
- [3] Yuqian Lu, Chao Liu, Kevin I-Kai Wang, Huiyue Huang and Xun Xu : “Digital Twin-driven smart manufacturing : Connotation, reference model, applications and research issues”. In : *Robotics and Computer-Integrated Manufacturing* 61 (2020), p. 101837.
- [4] Trent Munro : “What you need to know about Industry 4.0”. 2016. URL: <https://blog.matthews.com.au/what-you-need-to-know-about-industry-4-0/> (visited 13/12/2021)
- [5] Fernando Almeida and Jorge Simoes : “The role of serious games, gamification and Industry 4.0 tools in the Education 4.0 paradigm”. In: *Contemporary Educational Technology* 10.2 (2019), pp. 120-136.
- [6] Siemens AG : “TIA Portal”. URL: <https://new.siemens.com/global/en/products/automation/industry-software/automation-software/tia-portal.html> (visited 13/12/2021)
- [7] Siemens AG : “TRIAL Download SIMATIC S7-PLCSIM Advanced V2.0 SP1”. URL: <https://support.industry.siemens.com/cs/document/109758848/trial-download-simatic-s7-plcsim-advanced-v2-0-sp1?dti=0&lc=en-AE> (visited 13/12/2021)
- [8] Siemens AG : “SIMATIC S7-PLCSIM Advanced : Co-Simulation via API”. URL: <https://support.industry.siemens.com/cs/document/109739660/simatic-s7%E2%80%91plcsim-advanced-co%E2%80%91simulation-via-api?dti=0&lc=en-AE> (visited 13/12/2021)
- [9] Unity Technologies : “Unlock your creativity” URL: <https://unity.com/download> (visited 13/12/2021)
- [10] Unified Automation : “.NET Based OPC UA Client and Server SDK (Bundle)”. URL: <https://www.unified-automation.com/products/server-sdk/net-ua-server-sdk.html> (visited 13/12/2021)
- [11] Siemens AG : “S7 user block for the OPC UA client of a SIMATIC s7-1500”. URL: <https://support.industry.siemens.com/cs/document/109762770/s7-user-block-for-the-opc-ua-client-of-a-simatic-s7-1500?dti=0&lc=en-AE> (visited 13/12/2021)
- [12] Unity Technologies : “Final IK”. URL: <https://assetstore.unity.com/packages/tools/animation/final-ik-14290> (visited 13/12/2021)
- [13] OPC UA Foundation : “OPC 40010-1 – UA for Robotics, Part 1 : Vertical Integration”. URL: <https://reference.opcfoundation.org/Robotics/docs/> (visited 13/12/2021)
- [14] Dr. Dirk Osswald, Patrick Zimmermann, Etienne Axmann : “OPC UA IAS Initiative”. In: *VDMA WebSeminar Series 21.07.2020*. URL: https://opcua.vdma.org/documents/26605244/49873132/OPC%20UA%20IAS_Web%20Seminar_21.07.20_1595411749536.pdf/d3588f54-a1cc-7898-142e-fb2dd88df775 (visited 13/12/2021)
- [15] OPC UA Foundation : “OPC Unified Architecture Part 10 : Programs”. URL: <https://reference.opcfoundation.org/v104/Core/docs/Part10/> (visited 13/12/2021)
- [16] Stefan Profanter, Ari Breitzkreuz, Markus Rickert and Alois Knoll : “A hardware-agnostic OPC UA skill model for robot manipulators and tools”. In: *2019 24th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*. IEEE. 2019, pp. 1061-1068
- [17] Siemens AG : “TIA Portal Openness : Introduction and Demo Application”. URL: <https://support.industry.siemens.com/cs/document/108716692/tia-portal-openness-introduction-and-demo-application?dti=0&lc=en-WW> (visited 13/12/2021)
- [18] Siemens AG : “Siemens SIMATIC SIMIT Simulation Platform (v10.0) Operating Manual”. URL: <https://www.automation.siemens.com/sce-static/learning-training-documents/tia-portal/example-processes/sce-020-120-documentation-en.pdf> (visited 13/12/2021)
- [19] Siemens AG : “Siemens SIMIT 7 Shared Memory Gateway User Manual”. URL: https://cache.industry.siemens.com/dl/files/351/73136351/att_111459/v1/SIMIT_SHM_Gateway_e.pdf (visited 13/12/2021)