Reconfigurable Digital Twins for an Industrial Internet of Things Platform

Eliseu Pereira*†, Maria Arieiro*, Gil Gonçalves*†
†SYSTEC - Research Center for Systems and Tecnologies
*Faculty of Engineering, University of Porto
Rua Dr. Roberto Frias, 4200-465 Porto, Portugal
Email: {eliseu, up201606469, gil}@fe.up.pt

***Abstract*—Digital Twin (DT) emerged as an Industry 4.0 concept that reflects the behavior of physical entities and processes in the digital environment providing real-time information and insights. Typically, DTs are deployed on specific scenarios or components, virtualizing services and monitoring process variables. This extremely focused software implementation makes harder the adaptation and replication of DTs for new components with similar characteristics. This work's main goal was to develop a flexible platform that facilitates the implementation and modification of DTs for IEC-61499 compliant Cyber-Physical Systems (CPS), allowing their functionalities management and variables monitoring. The DT component was integrated into a web-based platform (Jurassic Park) that manages IEC-61499 compatible devices executing the runtime environment (DINASORE). The platform was validated in 2 different scenarios, 1) in a distributed system composed of different DTs and 2) controlling a gripper of a robotic arm.**

***Index Terms*—Cyber-Physical Systems, Digital Twin, IEC-61499, Industrial Internet of Things, Virtualization.**

I. Introduction

The changeover to Industry 4.0 was introduced in the shop-floor of new information and communication technologies such as the Industrial Internet of Things (IIoT), Cyber-Physical Systems (CPS), among others. The digitalization and virtualization of the physical devices provided to users and operators support for communicating with other devices, tracking errors, optimizing production and many other advantages [1][2]. Digital Twins (DTs) implement virtual models of physical entities to mirror the geometry and behaviour of that entities in the digital domain. With those digital models, real-time monitoring and control of the physical entities can be achieved [3]. Cyber-Physical Production Systems (CPPS) have a need to be easily reconfigurable, since these systems stand out by responding quickly to changes in production and being flexible enough to introduce new functionalities according to the different needs of the production lines. One of the existent standards for the CPPS reconfiguration is the International Electro-technical Commission (IEC) 61499 standard [4], which allows the encapsulation of different functionalities in software modules, the so-called Function Blocks (FBs).

Several DTs proposals have been made recently, however there is a lack of flexible solutions that are applicable in the production lines. Nowadays, the creation of a DT is a centralised process, oriented towards the device that is intended to be reflected in the digital world. This makes the DT not very flexible. In the complex case of a system with a diversity of physical entities fulfilling different functionalities, reusing a DT is a critical process and will require increased computational and human effort. This limitation could be solved through the usage of standards for the deployment of reconfigurable CPPS, such as the IEC-61499 standard. However, its integration with the DT concept is still limited and barely explored by the scientific community. Some of the most relevant limitations are the difficulty to manage complex systems with many software modules and the lower flexibility and adaptability of the systems to new tools, at the software and hardware level [5].

The main objective of this work was to develop a flexible DT application, integrated with a web-application (Jurassic Park [6]) and with the physical devices (executing the DINASORE [7] runtime environment). The DT solution will include 1) a monitoring module, 2) a control module and 3) a visualisation module. The monitoring module is intended to be the processing component of the information that arrives from the physical environment. The main purpose of this module is to understand the behaviour of the physical entities and store that information in the digital world. The DT control module aims to be the response feature of the DT solution. Through this module, the user can act on the physical entities, thus making the system not only an analysis component but also an acting component with the purpose of modifying the system. The general idea behind the DT visualization module is to allow the management of the DTs from the application system. The solution was validated in 2 different scenarios, 1) used to monitor and control a distributed CPS, and 2) to control a robotic arm gripper.

The remainder of the paper is structured as follows: in Section II, we present the literature review, Section III exposes the implementation carried out and Section IV presents the main tests and results made to validate the DT solution developed. The conclusions and future work are presented in Section V.

II. Literature Review

Digital Twins (DTs) can adopt different methodologies, concepts and technologies, providing capabilities of simulation, monitoring or optimization. For that, we initially present the main technologies used as basis for the implementation of the

DT platform, and after the related work presents a description of the more relevant DT implementations.

A. Background

There are several enabling technologies that allow the implementation of CPSs [8], for different purposes, like device management, data collection, workflow orchestration, among others. The DT solution stacks different technologies, having as basis an environment with essential capabilities, like reconfiguration of software, data interoperability or communication transparency [9].

The DINASORE [7] is a distributed platform which enables the pre-processing of data in edge devices, using Python based FBs. This platform allows the construction of a wide variety of FBs by the user, with different goals, like to control drivers, integrate sensors, apply processing techniques, among other functionalities. The DINASORE uses the 4DIAC-IDE as graphical user interface to orchestrate the FB pipelines, deployed after in the distributed CPS.

The 4DIAC-IDE [10] is a tool developed by Eclipse that allows the orchestration of a workflow of FBs, after being deployed in a distributed system. With this tool, it is possible to draw the FBs distributed systems through a graphical interface and then deploy it into the devices connected to the network.

The Jurassic Park [6] is a web-based application that acts as a centralised repository of FBs, allowing the creation and management of FBs. The Jurassic Park is composed of two individual components, the backend and the web application. This application is integrated with the DINASORE and the 4DIAC-IDE, allowing the full deployment and management of distributed systems through the IEC-61499 standard.

B. Related Work

As identified in [11], currently there is a huge need in providing on-demand manufacturing services through Industrial Internet of Things (IIoT) networks. These services are easily achieved with the implementation of cloud-based manufacturing solutions. The literature review presented in [12] shows a set of cloud solutions available to help in the implementation of Digital Twins (DTs), in particular the Microsoft Azure IoT solution and the Amazon Web Services (AWS) IoT solution. When large industrial companies began to realize that the implementation and consequent application of DTs could have a huge impact on the efficiency of their production lines, they begin their own initiative to develop their own tools [13][14].

In [15], the authors present DT use cases in the industrial environment, where Siemens presents two DT implementations, one for the power system and the other for the wastewater plant. General Electric has not only developed a DT for a wind farm, but has also been able to prove through this implementation that a wind farm should be operated, developed and maintained in a more efficient way. British Petroleum has developed a DT of oil and gas facilities located in more restricted areas. The air vehicle manufacturer Airbus has used DT-based solutions to monitor its production lines and to optimise its operations. Going into a more specific example of a DT implementation in the industry context, [16] presents a DT within a production line of hollow glass manufacturing. The authors of this implementation argue that this work has allowed them to realize that digital simulations have a great impact on how production lines can be optimized, since the simulations reflect their behavior in the real world.

In [17], the authors present a DT solution in the commercial greenhouse sector. The article highlights the need of the commercial greenhouse production systems to become more energy-efficient while maintaining a sustainable production. Therefore, the DT solution mentioned was proposed in order to provide a control and monitoring feature of the production flow of a greenhouse production system. The authors of this article also defend two essential ideas. One of them is that, with DT implementations, it is possible to turn the industrial environment more energy and climate friendly and that leads to a reduction of costs within the production lines. The other idea is that, nowadays, the industrial environment requires developed generic DT frameworks capable of controlling the processes accurately and to respond to changes in the orders of the production lines.

III. Implementation

The DT platform was conceptualized to structure multiple DTs, allowing their monitorization and control. This concept is composed of three main objects, 1) the digital twin, that is at the top of the architecture and is responsible for controlling the functionalities of the physical equipment, 2) the functionality, which describes the DT characteristics, mainly the monitoring system variables and the functions to trigger the execution of certain actions, and 3) the device, that reflects the physical entity in the digital world, like sensors, machines, etc. The DT platform developed is embedded in the Jurassic Park taking advantage of some of its features. Figure 1 shows the DT proposed solution composed of three components, the DT visualisation module, the DT monitoring module and the DT control module.

The DT visualisation component (presented in Section III-A) allows the interaction between the DT platform and the user. This component is incorporated into the web application of the Jurassic Park. The main functionalities of this visualisation component focus on the management of the DTs. Additionally, this component will be responsible for triggering the monitoring and control requests from the DTs. The DT monitoring module (presented in Section III-B) is intended to serve as a processing tool of the information that arrives from the physical entity to the digital one and serves the information to the user through the graphical interface. The component is incorporated into the web server of the Jurassic Park using the Open Platform Communications - Unified Architecture (OPC-UA) communication protocol as data source to collect the information from the physical world. The monitoring module collects data from the FBs variables and events that are running on the DINASOREs connected to the platform, allowing the user to filter the ones to monitor. The DT control component (presented in Section III-C) permits to trigger functionalities

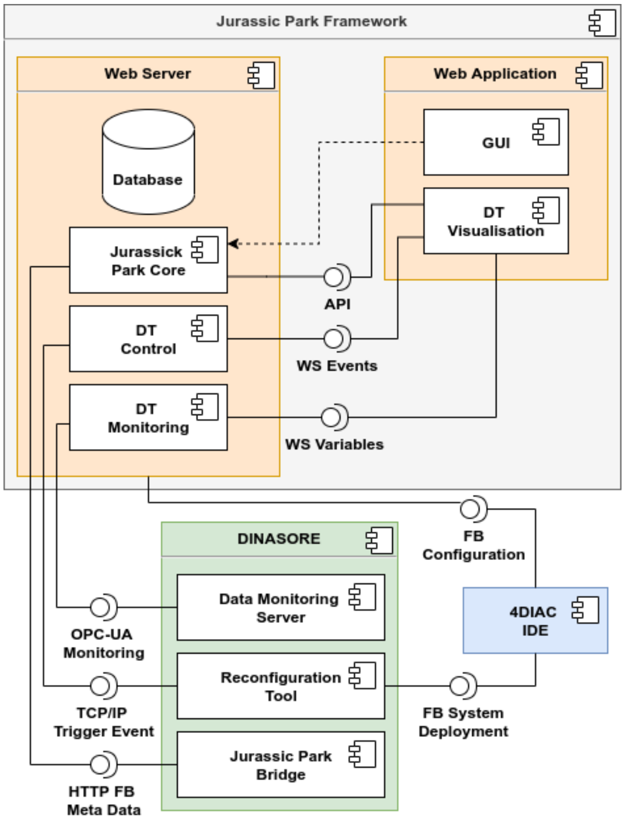


Fig. 1: DT architecture integrated with the Jurassic Park and DINASORE.

over the physical entities connected to the platform. The component uses the variables and events from the FBs that are running on the devices connected to the platform and assigns them a trigger feature. In this way, the user will have the opportunity to manipulate the events of the FBs. As in the case of the monitoring component, the control component is also incorporated into the web server.

The backend component (web server) is a complex module that uses several technologies to consolidate the integration of all the services it supports. This component is responsible for the HTTP API that manages the FBs and the real-time communication of the devices connected to the platform as well as the database. One of the most relevant technologies used in the backend is the Node.js framework, which integrates several components of the backend such as the API, the real-time communication with the web application and the real-time communication with the DINASOREs, through the OPC-UA communication protocol. The backend component uses as database the MySQL database. The backend includes also an File Transfer Protocol (FTP) server that makes available the download of FBs by the DINASOREs.

The frontend component (web application) is the graphical user interface of the platform, and it was developed using React [18]. This component is responsible for the page components (buttons, boxes, tables, among other User Interface (UI) elements), which are customized according to the existent resources in the CPS. The automatic update of the information is performed using the Socket IO library, which supports real-time communication with the backend, and avoids the need of refreshing the web page to update the information.

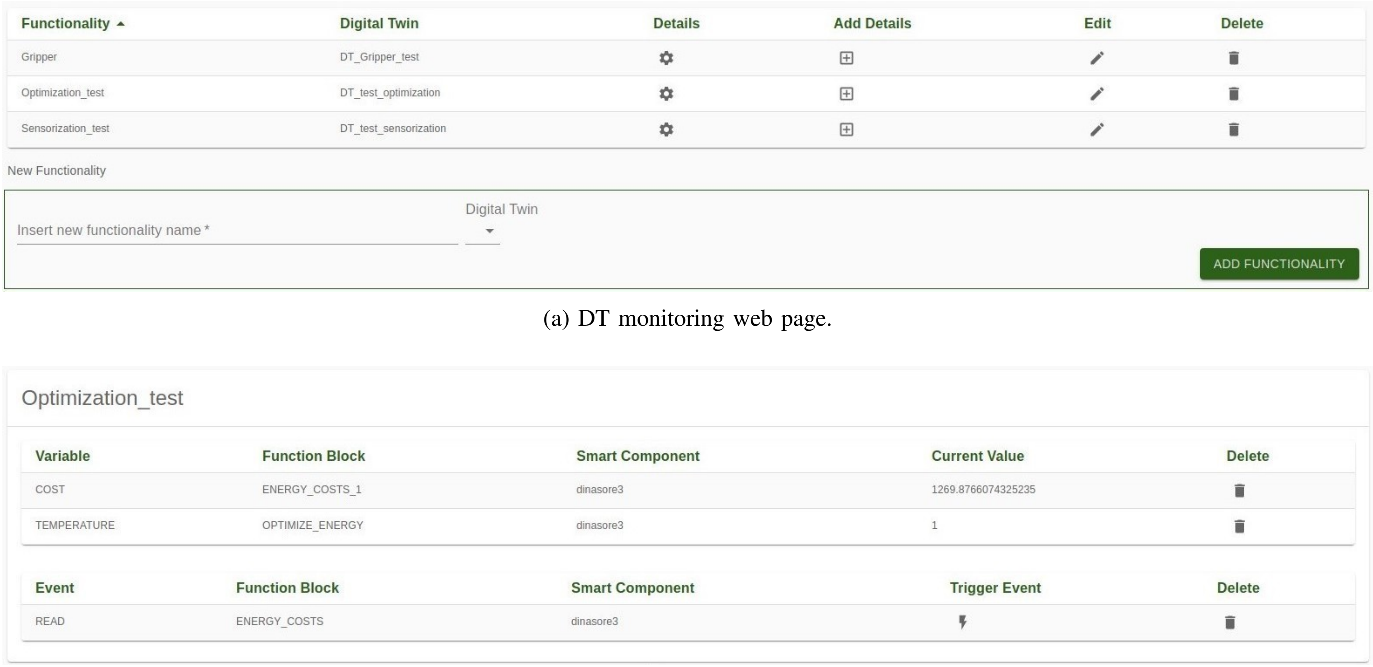
A. Digital Twin GUI

The DT visualisation component allows the user to create and manipulate DTs. The platform allows the user to group the devices they want to monitor and control into a category called DT and subsequently they can associate that DT with a general functionality. Therefore, in order to describe the page elements in a more user-focused way, we list below the different pages implemented to fulfil the requirements of the DT component.

The new DT page allows the user to add new DTs with a specific name and to associate them with respective DINASOREs. The page has a menu for creating a new DT, and in this menu the user has access to all the devices that are communicating with the Jurassic Park in real-time. Therefore, to create a new DT, the user only has to complete the field to insert the name of the new DT and open the list of available devices and selected those he wants to associate. It should be noted that the user can choose more than one DINASORE. Additionally, the user can only create the new DT if he has chosen at least one DINASORE.

The Digital Twin monitoring page, in Figure 2a, performs all the management of the DT. On this page, the user can observe the functionalities that are currently active with the respective DT. Besides that, it is also on this page that the user can create other functionalities and associate a DT capable of observing the variables and/or events of interest. Looking first at the feature of the page concerning the creation of a new functionality, the user can choose the name of the functionality they want to add to the monitoring platform. After choosing the name, the user will also have to choose one DT from the range of available DTs that were previously created in the New Digital Twin page, which they want to associate with the new functionality. Through this page, the user will have at his disposal a table, listing all the functionalities currently active and a set of features. The *Details* feature allows the user to view in detail the information that is being collected from the variables and events that are currently being monitored. The *Add Details* feature enables the user to choose the variables and events that they want to monitor on a given FB. With the *Edit* feature, the user can edit the name of the functionality. In addition to being able to edit and manipulate the variables and events that each functionality will monitor, the user can also delete the available functionalities they want with the *Delete* feature.

Whenever the user presses the functionality details button, the interface redirects to a new page. The functionality details page, in Figure 2b, has two different tables, one for monitoring variables and another for triggering events. Note that the page only displays the variables and events that have been previously selected on the DT monitoring page. Thus, the user has at their disposal information organised either by variables or by events. Regarding the variables monitoring,



Functionality ▲	Digital Twin	Details	Add Details	Edit	Delete
Gripper	DT_Gripper_test	⚙	⊞	✎	🗑
Optimization_test	DT_test_optimization	⚙	⊞	✎	🗑
Sensorization_test	DT_test_sensorization	⚙	⊞	✎	🗑

New Functionality

Insert new functionality name *

Digital Twin

(a) DT monitoring web page.

Optimization_test

Variable	Function Block	Smart Component	Current Value	Delete
COST	ENERGY_COSTS_1	dinasore3	1269.8766074325235	🗑
TEMPERATURE	OPTIMIZE_ENERGY	dinasore3	1	🗑

Event	Function Block	Smart Component	Trigger Event	Delete
READ	ENERGY_COSTS	dinasore3	⚡	🗑

(b) Functionality details web page.

Fig. 2: Digital Twin graphical interface, including the monitoring (a) and functionalities details (b) web pages.

the user can observe in the table some information of the variable such as the variable name, the FB where the variable is being monitored, the device that is allocating and also the current value of the variable. Additionally, the user can delete the variables that they no longer wish to monitor by clicking on the delete button present in each row of the table. The event monitoring table does not differ much from the variable monitoring table. It is also possible to see the name of each event selected by the user, the associated FB and the DINASORE where it is being mapped. However, unlike the variable monitoring table, the event table has a button that allows the user to trigger an action on the associated event. When the user presses the trigger event button, the Jurassic Park automatically sends a request to the server to execute the event on the associated FB. The user also has a button available on the event table that allows them to delete events that they no longer wish to observe.

B. Digital Twin Monitoring

The monitoring component of the DT platform is responsible for requesting and collecting information on each DT, which is then made available in the DT visualisation component. For that, the user should create the different DTs and functionalities, as well as define the monitored devices, variables and events. With the previous objects created, the next integration is to automatically receive real-time feedback of the values of the monitored variables. Initially, the request is sent via WebSockets from the web application, where a persistent listener waits for feedback from the monitored variables. On the server side, the communication with the DINASORES is performed using a OPC-UA client. When the monitored variables present a change, the OPC-UA client automatically identifies it, and then the data is sent back to the backend controller which ensures the notification of the new value to the web application, waiting for the information through the listener function. When the user no longer wants to observe the variables, an event to cancel the subscriptions is sent to the OPC-UA client.

C. Digital Twin Control

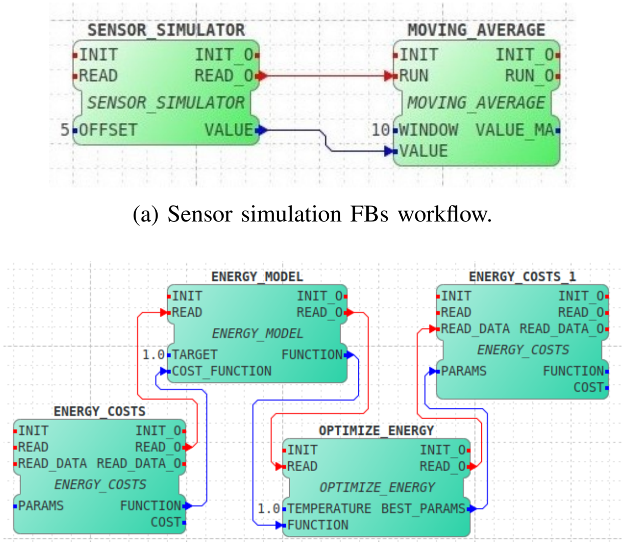
The DT control component allows the user to trigger events in certain FBs running on specific DINASOREs. After interacting with the UI element, a particular message, containing the information relating to the triggered event, is sent to the backend component. Having the information regarding the event in the backend, a function sends the action to the device in question, using Transmission Control Protocol/Internet Protocol (TCP/IP) sockets. For that, the function uses as input the event information, in particular the FB name, event name, device IP address and port. When the DINASORE receives the message, it pushes an event on the specified FB and executes the triggered functionality.

IV. Experiments and Results

The experiments perfomed to validate the DT component of the platform focused on two use cases that allow to test the different implemented features. The experimented use cases are 1) the monitorization of simulated sensors and the optimization of energy in a distributed CPS, and 2) the manipulation of a robotic arm gripper.

A. *Monitoring and optimizing a distributed CPS*

The first validation scenario is a distributed system with different devices connected to the platform. This experiment consisted of connecting a set of raspberry pis to the platform, which are responsible to execute two different FBs pipelines/workflows, one for sensing purposes and the other for energy optimisation. This set of FBs pipelines allows the monitoring of variables and trigger of events in a distributed system, allowing the validation of the previously explained monitoring and controlling features of the DT platform.



(a) Sensor simulation FBs workflow.

(b) Energy optimization FBs workflow.

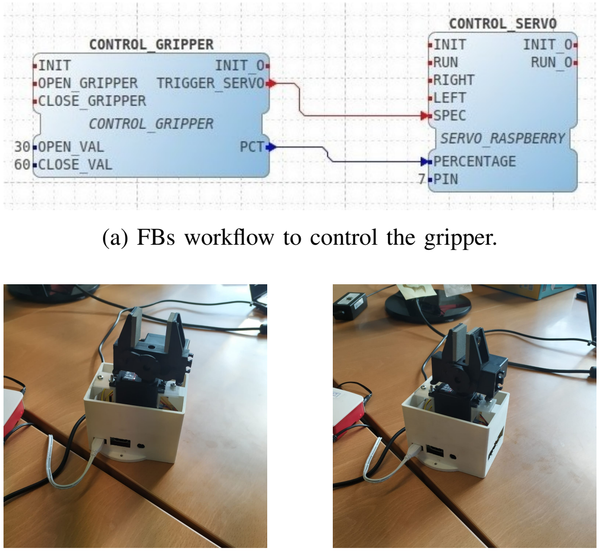
Fig. 3: Distributed CPS composition with a sensorization (a) and optimization (b) workflows.

The first raspberry pi was used to validate the variable monitoring component, using a small FBs workflow, that simulates a sensing system, present in Figure 3a. The FBs composing the workflow are 1) the SENSOR_SIMULATOR, responsible for generating a random value to simulate the value of a sensor, and 2) the MOVING_AVERAGE, which calculates the average of the last N values. After the deployment of the workflow, the next step was to create a DT for the sensing component, associated with the raspberry pi that executes the DINASORE. Then, we created a functionality that includes the following monitoring variables, 1) the variable VALUE (variable containing the simulated sensor data) from the SENSOR_SIMULATOR FB and, 2) the variable VALUE_MA (variable containing the moving average which is calculated) from the MOVING_AVERAGE FB.

The second raspberry pi hosts a set of FBs that optimize the energy consumption of a process, present in Figure 3b. These FB pipeline allows the validation of the event triggering process through the DT platform and consequently the monitoring of variables linked to that event. The FBs composing the workflow are 1) the ENERGY_COSTS, which specifies the function for energy costs regarding velocity and power, 2) the ENERGY_MODEL, which is the model that allocates the energy consumption, using as input the energy costs, and 3) the OPTIMIZE_ENERGY, which optimizes the given function, through the use of the Dual Annealing algorithm. The DT associated to the optimization workflow will include a functionality that contains the following variables and events: 1) the COST variable (optimised energy costs) present in the ENERGY_COSTS_1 FB, and 2) the READ event (triggering optimisation) of the ENERGY_COSTS FB. Thus, with this variable and event combination, we are able to trigger an event via the DT platform, sent to the raspberry pi, that executes the optimization.

B. *Creating a Digital Twin for a Gripper*

The second validation scenario focuses on the creation of a DT for a robotic arm gripper, validating the control and monitoring features in a physical component. The component used was a 3D printed gripper using a raspberry pi as controlling system for the servo motor. This experiment consisted in controlling the gripper through the developed DT platform by manipulating the event trigger to make the gripper arm open and close according to the event, as shown in Figure 4b and 4c. The FB workflow that allows the control of the gripper, present in Figure 4a, is composed of the FBs: 1) the CONTROL_GRIPPER, responsible for identifying whether there is a request to open or close the gripper, depending on which, the FB sends the corresponding percentage to the output (PCT), and 2) the CONTROL_SERVO, which receives as input the percentage and then updates the general-purpose input/output (GPIO) that controls the servo motor with the corresponding value.



(a) FBs workflow to control the gripper.

(b) Gripper open.

(c) Gripper closed.

Fig. 4: Results obtained with the gripper scenario, including the FBs workflow (a) and the physical gripper (b) and (c).

The DT created intends to monitor and control the gripper manipulation. For this use case, two distinct events and a variable (to evaluate whether the events are being properly triggered) have been added to the functionality associated to the DT. The selected events to control the gripper were the OPEN_GRIPPER and CLOSE_GRIPPER events. In order to check whether the control on the DT platform is functional, the gripper would have to open when the user pressed the release button. If the close button was pressed, the gripper would have to execute the closing movement. The execution of both movements were validated visually and also through the monitoring variable (percentage), which confirms if the gripper moves according with the monitored percentage.

V. Conclusion

In conclusion, the main objectives of this platform were to implement a flexible and reconfigurable DT solution capable of increasing the monitoring capacity and enable the remote control of a CPS. The components developed, mainly the DT visualisation component, the DT monitoring component and the DT control component permitted the CPS to accomplish the described attributes, like DT reconfiguration. The DT solution was integrated together with a mature stack of technologies including the Jurassic Park and the DINASORE. The experiments performed support the usability and flexibility of the web based platform. Finally, it is important to highlight that one of the greatest contributions of this project was to develop a platform that, given its flexibility and configurability, can be easily integrated in the industrial sector and supports the IEC-61499 standard.

As future work, one of the main goals will be the storage of data generated by the DT variables, taking into consideration constraints of data volume, in terms of storage and data flow/rate. On top of this unstructured database, it will be possible to implement predictive algorithms to forecast and optimize the behavior of DTs. Additionally, the entire platform (Jurassic Park with DT component and DINASOREs) will be integrated in an industrial scenario composed of different machines, including integration of sensors, simulation of processes, and optimization of resources.

Acknowledgment

INDTECH 4.0 - New technologies for intelligent manufacturing. Support on behalf of IS for Technological Research and Development (SI a Investigacao e Desenvolvimento Tecnologico). POCI-01-0247-FEDER-026653

References

[1] G. Gonçalves, J. Reis, R. Pinto, M. Alves, and J. Correia, “A step forward on intelligent factories: A smart sensor-oriented approach,” in *Proceedings of the 2014 IEEE Emerging Technology and Factory Automation (ETFA)*, 2014, pp. 1–8.

[2] L. Antão, R. Pinto, J. Reis, and G. Gonçalves, “Requirements for testing and validating the industrial internet of things,” in *2018 IEEE International Conference on Software Testing, Verification and Validation Workshops (ICSTW)*, 2018, pp. 110–115.

[3] K. Ding, F. T. S. Chan, X. Zhang, G. Zhou, and F. Zhang, “Defining a Digital Twin-based Cyber-Physical Production System for autonomous manufacturing in smart shop floors,” *International Journal of Production Research*, vol. 57, no. 20, pp. 6315–6334, 2019.

[4] K. Thramboulidis, “Iec 61499 in factory automation,” in *Advances in Computer, Information, and Systems Sciences, and Engineering*, K. Elleithy, T. Sobh, A. Mahmood, M. Iskander, and M. Karim, Eds. Dordrecht: Springer Netherlands, 2006, pp. 115–124.

[5] G. Lyu and R. W. Brennan, “Towards IEC 61499-Based Distributed Intelligent Automation: A Literature Review,” *IEEE Transactions on Industrial Informatics*, vol. 17, no. 4, pp. 2295–2306, 2021.

[6] J. Pedro Furriel, E. Pereira, J. Reis, and G. Gonçalves, “Jurassic park - a centralized software modules repository for iot devices,” in *2021 10th Mediterranean Conference on Embedded Computing (MECO)*, 2021, pp. 1–4.

[7] E. Pereira, J. Reis, and G. Gonçalves, “Dinasore: A dynamic intelligent reconfiguration tool for cyber-physical production systems,” in *Eclipse Conference on Security, Artificial Intelligence, and Modeling for the Next Generation Internet of Things (Eclipse SAM IoT)*, 2020, pp. 63–71.

[8] Q. Qi, F. Tao, T. Hu, N. Anwer, A. Liu, Y. Wei, L. Wang, and A. Nee, “Enabling technologies and tools for digital twin,” *Journal of Manufacturing Systems*, vol. 58, pp. 3–21, 2021, digital Twin towards Smart Manufacturing and Industry 4.0.

[9] E. Pereira., R. Pinto., J. Reis., and G. GonÇalves., “Mqtt-rd: A mqtt based resource discovery for machine to machine communication,” in *Proceedings of the 4th International Conference on Internet of Things, Big Data and Security - IoTBDS,*, INSTICC. SciTePress, 2019, pp. 115–124.

[10] A. Zoitl, T. Strasser, and A. Valentini, “Open source initiatives as basis for the establishment of new technologies in industrial automation: 4diac a case study,” in *2010 IEEE International Symposium on Industrial Electronics*, 2010, pp. 3817–3819.

[11] Y. Lu and X. Xu, “Cloud-based manufacturing equipment and big data analytics to enable on-demand manufacturing services,” *Robotics and Computer-Integrated Manufacturing*, vol. 57, pp. 92–102, 2019.

[12] T. Catarci, D. Firmani, F. Leotta, F. Mandreoli, M. Mecella, and F. Sapio, “A conceptual architecture and model for smart manufacturing relying on service-based digital twins,” in *Proceedings - 2019 IEEE International Conference on Web Services, ICWS 2019 - Part of the 2019 IEEE World Congress on Services*. IEEE, Piscataway, NJ, USA, 2019, pp. 229–236.

[13] A. Fuller, Z. Fan, C. Day, and C. Barlow, “Digital Twin: Enabling Technologies, Challenges and Open Research,” *IEEE Access*, vol. 8, pp. 108 952–108 971, 2020.

[14] A. Rasheed, O. San, and T. Kvamsdal, “Digital twin: Values, challenges and enablers from a modeling perspective,” *IEEE Access*, vol. 8, pp. 21 980–22 012, 2020.

[15] F. Tao, H. Zhang, A. Liu, and A. Y. Nee, “Digital Twin in Industry: State-of-the-Art,” *IEEE Transactions on Industrial Informatics*, vol. 15, no. 4, pp. 2405–2415, 2019.

[16] H. Zhang, Q. Liu, X. Chen, D. Zhang, and J. Leng, “A Digital Twin-Based Approach for Designing and Multi-Objective Optimization of Hollow Glass Production Line,” *IEEE Access*, vol. 5, pp. 26 901–26 911, 2017.

[17] D. Anthony Howard, Z. Ma, J. Mazanti Aaslyng, and B. Nørregaard Jørgensen, “Data architecture for digital twin of commercial greenhouse production,” in *2020 RIVF International Conference on Computing and Communication Technologies (RIVF)*, 2020, pp. 1–7.

[18] A. Fedosejev, *React. js essentials*. Packt Publishing Ltd, 2015.