Teaching Industry 4.0 Related Topics During the COVID-19 Restrictions – An Experience Report

Marenice Melo de Carvalho
CETELI-PPGEE-UFAM
Federal University of Amazonas
Manaus – AM, Brazil
0000-0002-0309-7333

Isaias Valente de Bessa
CETELI-PPGEE-UFAM
Federal University of Amazonas
Manaus – AM, Brazil
0000-0002-7973-4239

Guido Soprano Machado
CETELI-PPGEE-UFAM
Federal University of Amazonas
Manaus – AM, Brazil
0000-0002-2365-9735

Wollace de Souza Picanço
PPGEE-UFAM and FMF Wyden
Federal University of Amazonas
Manaus – AM, Brazil
0000-0002-1455-6689

Renan Landau Paiva de Medeiros
CETELI-PPGEE-UFAM
Federal University of Amazonas
Manaus – AM, Brazil
0000-0002-1645-2736

Vicente Ferreira de Lucena Jr
CETELI-PPGEE-UFAM
Federal University of Amazonas
Manaus – AM, Brazil
0000-0001-9864-2850

***Abstract*— The COVID-19 pandemic has imposed numerous restrictions on face-to-face meetings, and one of the most impacted activities was engineering education. Most teaching professionals have adopted diverse solutions based on distance learning techniques. However, practical experiments that require laboratories could not be performed in most cases. One of the subjects studied in modern engineering that has been impacted is related to industrial automation systems. In April 2020, amidst tighter restrictions on social contact, we started cooperating with a local company to train our students on topics of interest to the corporation. Our research group was responsible for dealing with Industry 4.0 and its associated technologies. Over the past two years, we have conducted this interaction with the industry under the pandemic's constraints and obtained very interesting results. In this report, on an innovative experience in engineering education, we will describe our project's syllabus, how the relationship was between students and teachers through online tools, and mainly how we managed to ensure that students had access to experiments close to the industrial reality.**

***Keywords*— Remote learning, cloud robotics, embedded systems, design method, distant laboratories framework.**

I. INTRODUCTION

In LAST TWO YEARS, the COVID-19 pandemic has imposed numerous restrictions on face-to-face meetings in all professional and social activities. Engineering education is one of the most impacted activities [1, 2]. Distance learning alternatives, with synchronous and asynchronous classes, extensive use of digital platforms to control student access and attendance, and most importantly, to ensure peer-to-peer communication, have been the solution adopted by most teaching professionals worldwide.

These solutions have met the needs of those involved in most engineering education activities. However, practical experiments that require laboratories could not be performed in most cases. With practical activities being performed remotely, engineering teaching laboratories have existed and are used in many courses worldwide [3, 4]. However, their scope does not cover all curricular components of engineering courses, and the number of students served is limited to small classes.

In recent years, the Internet for teaching has increased rapidly. It was possible to perceive the development of new tools to help students learn. Among these tools, many remote laboratories with real and virtual industrial plants were proposed to contribute to the state of the practice [5, 6, 7]. The problem of developing simulation tools applied to industrial automation plants demands the study of many mathematical concepts [8]. Some of them are not taught in undergraduate courses.

In addition to the Internet, other technologies, such as digital twins, modern industrial controllers (PLCs and microcontrollers), and communications protocols, came to help the improved use of remote laboratories. Remote Laboratories (RLs) are software and hardware tools that allow students to remotely access a real or virtual experiment connected to the Internet [9, 10].

One of the subjects studied in modern engineering that has been impacted is related to industrial automation systems. These subjects are treated with classical control experiments and manufacturing or process automation in specific laboratories.

In April 2020, amidst tighter restrictions on social contact, we started cooperating with a local company to train our students on topics of interest to the corporation. Our research group was responsible for dealing with Industry 4.0 and its associated technologies. According to our plans, this training would be fundamentally practical, helping the participants obtain a realistic view of what happens in the industry. Due to the restrictions imposed by the pandemic, we needed to change our teaching approach. This paper will relate the cooperation principles and, most notably, the adopted procedures to give the students some practical insights into Industry 4.0 technologies.

II. STRUCTURE OF THE COOPERATION

The market in technology is currently facing a labor shortage. This has impacted the country's economic growth, awakening the need to train qualified human resources in the short term. Against this expectation, undergraduate science, technology, engineering, and mathematics (STEM) programs have shown high dropout rates. This problem is concentrated between the 1st and 4th semesters for various reasons: deficiency in primary education, delay in contact with activities related to the professional profile, socioeconomic issues, and little prospect of immediate application of knowledge in the labor market.

Such scarcity also strongly reflects the local industrial complex and the ecosystem linked to regional development: companies, science, technology institutes, startups, and universities. Moreover, other factors put pressure on the necessary actions to expand the quality and quantity of trained professionals in the technology areas.

The advent of Industry 4.0 and the field of information technology (IT) demand the training of multidisciplinary professionals, and the emergence of new areas of knowledge, such as the Internet of Things, big data, cloud computing, wearable computing, augmented and virtual reality, and software security, require updated training of professionals at the undergraduate level.

It is easy to conclude that it is necessary to carry out actions to train and develop human resources in STEM areas that consider both technical issues and new technologies [11, 12]. We need to overcome educational institutions today's difficulties in delivering more professionals from these specific areas to the labor market. This is how the SUPER Project was born: cooperation between a large multinational company installed in the city's name and the university, the region's most respected higher education institution.

The structure of the cooperation project has the structure and actors shown in Fig. 1. Professors coordinate the academic activities with extensive experience in areas of interest to the contracting company. Their master's and Ph.D. students support these professionals as teaching assistants and tutors in practical exercises. All selected students received a scholarship guaranteeing their permanence for the entire project. An important aspect is that this cooperation has made it possible to purchase current equipment with essential technologies for the industry.

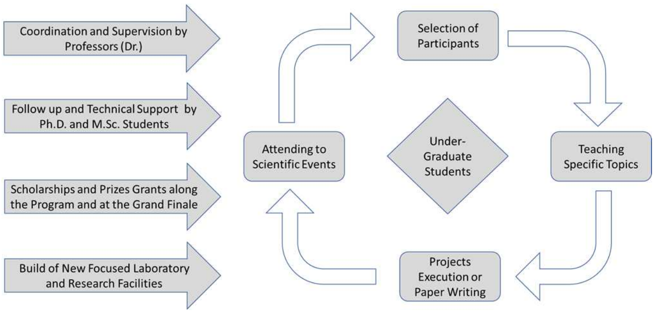

Figure 1: Structure of the cooperation.

The project is focused on undergraduate students in engineering (electrical, computer, and production) and computer science. A selection is made focusing on students who have finished the fifth semester. A public call is made for this, and the best candidates are chosen to participate. In the specific case of the Industry 4.0 training, 12 students were selected.

These students must attend theoretical classes (100 hours) on specific themes of the desired training. For Industry 4.0, the themes were Introduction to Cyber-Physical Systems (CPS) and Industry 4.0, Digital Control and Industrial Applications, Technical and Theoretical Concepts of Robotics, Industrial IoT, and Development and Management of Automation Projects.

At the end of the theoretical training, the students must perform a practical project with the development of research directly related to what the industry needs. This phase lasted six months and was concluded with a scientific event in which the results of all participants were presented to company representatives and the local community.

In times of pandemic, we faced several problems in executing the theoretical classes and even larger ones in realizing the practical activities. The theoretical activities were conducted through web-based communication media, and the main problem was adapting the physical classroom to a virtual classroom. The execution of the practical exercises at a distance was much more challenging, and we show three examples in the following sections. The software and hardware choices were based on our group's experience developing new teaching approaches with international institutions [13] and local industries [14]. When discussing the development methodology, we adopt agile development strategies [15] based on SCRUM [16] and, whenever possible, develop reusable components [17].

III. Industrial Real-Time Digital Twin System For Remote Teaching Using Node-RED

Teaching emerging technologies applied to industry in engineering courses has been challenging. Industry 4.0 combines physical and digital technologies that integrate the factory, suppliers, workers, and consumers [18, 19]. One of these technologies is digital twins (DTs), which provide a virtual representation of a physical system, such as a mirror, providing segmented and adequate information. The assembly line with digital twins offers instant information about production, consumption, and performance [20, 21]. A system is characterized as a digital twin when information flows between physical and virtual objects in both directions [22].

Implementing DT in manufacturing is complex due to executing several steps [23, 24]. Access to a controlled laboratory environment for understanding and learning this technology is essential. However, this access is not possible in emergencies such as the coronavirus pandemic. In [25], a series of technologies, such as DT, was taught at a distance in a course in the construction area, aiming to apply them in complete practices of virtual education.

Therefore, it is necessary to seek virtual ways to teach the concepts of these technologies to students. Our research group has developed industrial automation applications [26], home automation systems [27], and assistive technology applications [28, 29]. In this sense, a series of tools and technologies were used to teach the digital twin concept at a distance to undergraduate students in 2020.

Therefore, this practical exercise presents the steps to build the digital twin using Node-RED of a production line with different process models. The data from the virtualized plant are stored in a database, Mongo DB, through the CRUD system and are used in the plant replicated in Node-RED. It is possible to monitor the system in real time, perform data analysis, and send the analysis back to the actual virtualized plant, thus obtaining a two-way communication.

A. Node-RED

Node-RED is a programming tool that connects hardware devices, APIs, and online services quickly and easily. It is made using Node.js and provides a browser-based editor in which the user can visually connect streams using the wide range of nodes in the palette that can be implemented at runtime and are stored using JSON. Furthermore, it is possible to create JavaScript functions or reuse such processes, templates, or flows in the built-in library.

SVG (Scalable Vector Graphics) is a widely used, royalty XML-based graphics format developed and maintained by the W3C SVG Working Group that is widely used to describe two-dimensional graphics, images, and a set of related graphics script interfaces. The controlled SVG from the Node-RED Dashboard Template API was code made by Sharikov (2020) and allows simple animations using SVG images in a dashboard environment in Node-RED. Once the code has been copied, it is easy to use it in the template node in the dashboard, define the element you want to animate, and add an id. This id is, in HTML, an attribute used in objects to identify them.

The API defines usage for animations using code. An attribute to be animated must have the id in the form "unique_id@ + token_of the_animation_desired." A token is a way to identify, within the code, by a string, a concept. In the case of the API, these identifiers refer to animation types. Among the types of tokens existing in the API, cx_color specifies the object's color change; cx_hide determines the appearance/disappearance of an object, path, text, or group; cx_move defines the movement of a group, line, or text; and cx_status specifies to change an attribute of type text.

B. *Adopted Methodology*

The first step in teaching the concept of digital twins at a distance to undergraduate students due to the absence of laboratories is to obtain a virtual plant that will work as a physical plant. This system was constructed using the Factory I/O system. This virtualization, made in Factory I/O, allows integration with a programmable logic controller (PLC), so a ladder routine is written for the process using the TIA Portal program. Then, the data from the PLC sensors and actuators were connected to the Node-RED framework, allowing data transfers between the PLC and Node-RED. Finally, a dashboard was developed in the Node-RED to request products and trigger the system. Digital Twin uses SVG images that reflect everything that happens in the Factory I/O virtual plant. Fig. 2 summarizes the procedures using a block diagram.

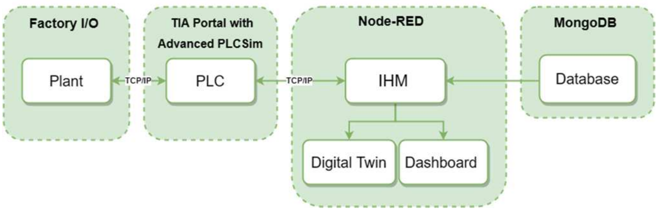

Figure 2: Diagram of project development.

1) *Virtual plant*

The construction of a digital twin consists of a mirror of a physical plant. To study this concept in a remote form, it is necessary to represent a physical plant virtually. Therefore, the virtual plant was inspired by an existing production line plant at the university and virtualized using the Factory I/O software. Fig. 3 presents a Staudinger plant base with Siemens PLCs connected by a Profinet industrial network and TCP/IP protocol. Fig. 4 shows the virtualized plant built to have the highest level of similarity in the number of sensors and actuators in the physical plant, thus making the distance from the laboratories as short as possible.

As in the Staudinger plant, the TIA Portal software from Siemens was used so that it was possible to program a desired operating routine and load it from the Factory I/O software. Then, it was possible to remotely emulate the production line and program it through a PLC simulator. The system has different products in its production line. In this work, five types of products were defined to produce a box containing (or not) a lid.

Figure 3: Industrial production line (Staudinger GmbH).

Figure 4: Virtual plant reproducing the physical plant.

The stacking box already has an identification tag (physical RFID tag) to be differentiated after starting the system in the physical plant. However, in the virtual plant, it was necessary to build a production step so that the coding could be carried out according to the definition of the products. There are two ways to store the products in stock. First, their tags are read by an RFID antenna, and if they are products A, D, or E, they go to the first path of the process and are taken directly to be stored in stock. Then, if the tags indicate products B or C, they go to the second process path to be capped and kept in stock. Fig. 5 summarizes the overall system process.

a) *Tracking order status*

In the TIA Portal, variables were created responsible for counting the number of products in stock. Then, given the development of this functionality, as the values of these variables are updated, Node-RED performs readings and displays them in the dashboard. The nodes read the quantity of their respective product type using the S7 communication protocol to access PLC information. At the output of each reading node, there is a type Gauge node responsible for displaying the data in the dashboard in graphics. They are inserted to facilitate the user's visualization and follow-up when requesting orders. The information on the empty quantity of products in stock is sent from the TIA Portal to the Node-RED locally, which receives these data and then sends it to the Node-RED.

a) *Digital Twin*

Knowing the objective of building and implementing the GD remotely for distance learning, the structure is based on the virtual plan shown in Fig. 4. The flow presented in Fig. 6 shows the steps taken to develop the GD. It began with the illustrations of the modules equivalent to the plant (belts, rotating belts, capper, spreader, and warehouse) made in SVG.

For this purpose, Inkscape was used as open-source software for vector illustration.

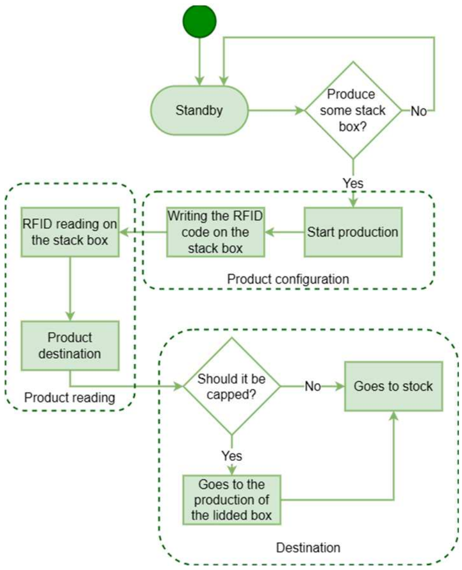

Figure 5: Diagram of the overall system process

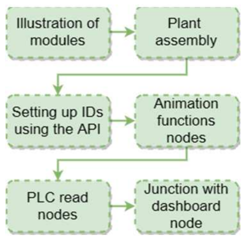

Figure 6: Diagram with steps for DT development.

These illustrations were used to compose a complete representation of the plant. Later, the images utilized by the animation were added. The identifications were configured using the API Controlled SVG from Node-RED Dashboard Template. This configuration allowed animations to be used with the function nodes, which specified how the animation would be used. For each animated item, a function node was created, and to receive a signal from the PLC, nodes of the plcindustry module were configured. A node was created for each PLC signal to be considered in the twin.

Finally, the flow was organized in Node-RED, connecting the nodes to the template node, which specifies the Twin window in Node-RED. Read nodes were connected to message nodes, such nodes were connected to functions, and function nodes were related to the template (window in Node-RED).

The animations consist of 2 basic types. The first one is the appearance/disappearance of the crates on the plant modules, suggesting the idea of their movement throughout the process. The second one is the color change of small circles on the modules (referring to the concept of an LED) that indicates whether the actuator of that module is activated or not. Sensors change color following the same principle.

The stacking box needed to be on the plan above the modules so that the appearance/disappearance animations could happen and suggest the intention of moving to be used in monitoring. Therefore, the final SVG consists of an illustration of the virtual plant. That contains several crates (of all colors) superimposed on the modules in the positions indicated by the sensors. When a box moves through a sensor in the virtual plant, the digital twin animates the SVG so that a crate (the color corresponding to the plant) appears over the module at the position pointed to by the sensor. Additionally, in SVG, both the sensor and the "LED" of the module change color.

The complete digital twin illustration is accessed by Node-RED using the system's local network. In general, the access to the digital twin is done as follows: the data from the sensors read in the virtual plant are sent from the PLC through PLCSim to Node-RED using the plcindustry module. These sensor data are sent to a Mongo DB database and sent to the Digital Twin built-in SVG. Then, the animations described above can be performed.

C. *Results and Discussions*

This section presents the dashboard of the product request system and the digital twin for process monitoring, developed in Node-RED software by electrical engineering undergraduate students. Given the objective of learning the concepts and tools needed to build the system, these results aim to provide a general understanding of the concept of digital twins.

Fig. 7 shows the result of the dashboard. The on/off switches and system status led are shown, and the selected product type and quantity items are selected. After being requested, a list with the history appears below for the user. There are five blocks with the current amount of each product type in stock. In addition, how many free spaces exist for each product, allowing the user to select only the quantity of items stored. Node-RED also provides a series of modifications that depend on the developer's creativity and skills with the tool's programming language.

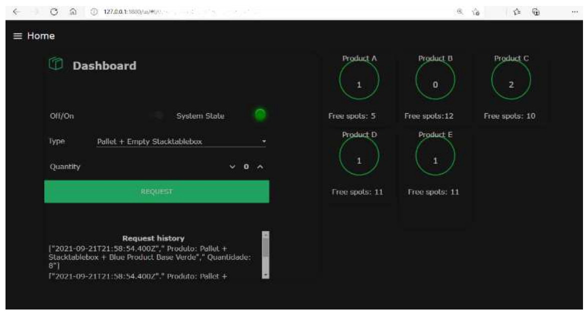

Figure 7: Request System on Dashboard in Node-Red

A digital twin was built for system plant monitoring using SVG images on Node-RED. Of the many ways to make a digital twin, the most accessible is through open-source software such as Node-RED. Fig. 8 shows the virtual floor plan at the top and the twin built by the students with all the animation settings in the Node-RED flow. Monitoring is done because data from PLC sensors and actuators are constantly being made, allowing the user to follow the process in real time. Vector graphics ensure image resolution versatility, allowing larger plants to be built and grouped on the same screen.

Addressing practical content to undergraduate students is a great challenge when one cannot access the controlled environments available at universities. In this context, this work presented a way to show the concept of digital twin technology, using an open-source framework called Node-RED to build a request panel and the digital twin through

SVG. The students had a big challenge with the developed project because they had little or no knowledge about the digital twin technology, the Node-RED tool, and the web programming ability necessary to progress in the solution within the Node-RED. Then, during the development of this work, the students had the opportunity to develop skills in various tools. They were also able to understand the concept of bidirectional communication of the DT, as Node-RED allowed the sending and receiving of plant data through communication with the PLC.

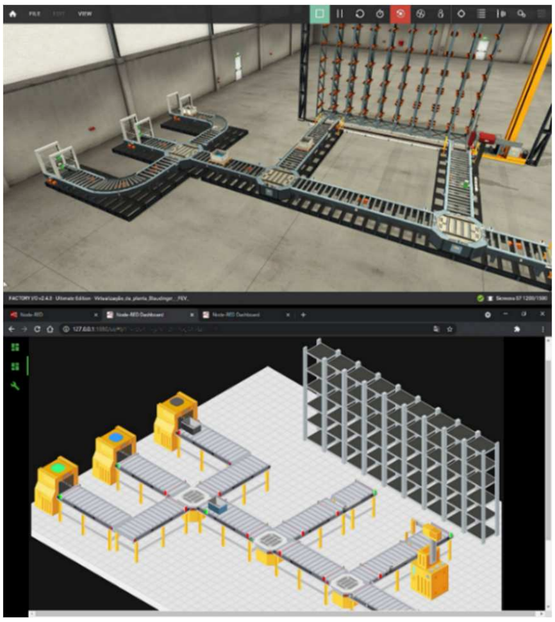

Figure 8. Virtual platform and Digital Twin with SVG images.

IV. Virtual Setup - Automatic Virtualization of Industrial Plants for Didactic Purposes

In this laboratory, a virtual setup system was developed to perform data acquisition from real or virtual industrial plants. Through communication protocols, these data are transformed into a virtualization platform capable of learning how the plant works and showing it in a graphic model. The central concept used in virtual setup is the digital twin (DT), defined as duplicating objects and existing systems into a virtual environment [24, 30, 31].

A. Problem Description

|In many institutions in our country, there is little equipment or a teacher to monitor the practices necessary for student learning [13]. Currently, remote laboratories can be used to reduce this problem. Nevertheless, the student only has real-time feedback from virtualization cameras or images, but it is impossible to know if the proposed activity is getting it right or wrong. Consider the following situation: the student is given the task of separating boxes by their respective sizes, but the action lacks information. The student does not know where to leave the boxes if the large box should go to ramp A, ramp B, or ramp C. How do you know the right ramp for the big box in this scenario? Which actuators must be activated for this process to take place? How long must the box stay on the height sensor to detect the correct size?

The problem addressed in this laboratory can be formalized with the following question: how to automatically correct the programming method the student developed for an industrial plant in his didactic task? This work proposes a system to solve the automatic correction of an industrial plant produced by students. This platform uses the program developed by the teacher to learn the correct functioning of the process and continue scanning the input and output data of the student code, generating a graphic model that aims to show whether the student did the activity correctly or not. This experiment will use the inputs and outputs of virtual industrial scenarios with the PLC with the Modbus communication protocol.

B. Proposed System

The proposed system was developed with an architecture containing the following components: control element, data acquisition module, and virtualization platform. Fig. 9 shows how the elements of this architecture are interconnected. This experiment implemented the control element in ladder language and was embedded in a PLC that controls the didactic plant. This program uses a Modbus protocol access logic output that is in series with the contacts of each input and output of the didactic plant to monitor the plant data and transfer this information to the DAM.

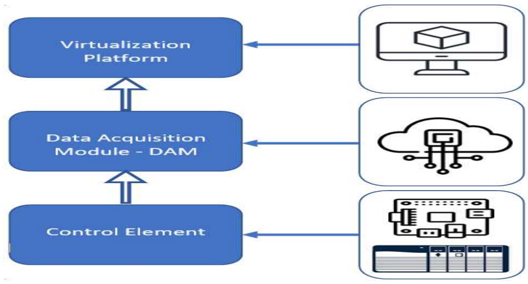

Figure 9 - Virtual setup system architecture.

The Data Acquisition Module (DAM) is responsible for collecting the Control Element data and sending it to the Virtualization Platform. Initially, DAM communicates with the Virtualization Platform to determine which didactic plants are registered and if they are available to the student at that moment. This module was implemented with the Node-RED tool, and based on the search for the registration carried out in the database, it will access the IPs of the didactic plants. It will scan every 250 ms to read the variables that monitor the PLC inputs and outputs.

The Virtualization Platform is a website that serves as an interface for student practices and allows remote access to its features. It was developed in PHP and JavaScript based on the Yes It Is (Yii) framework. This platform uses buttons on the main page to access the submodule registration, virtualization, supervision, learning, step view, and activity correction.

The registration module is responsible for storing the characteristics of the didact plant, such as plant name, controller IP, communication port, and the names of the inputs and outputs of the teaching process. These data are collected through a form and are saved in the system database. It was implemented using the Yii framework, which allows connecting with the database tables, automatically coding the code structure for the Model View Controller approach.

The virtualization module is the software tool that allows the student to graphically observe each input and output of the teaching process. In the visual, inputs are represented as squares and outputs as triangles. The red color indicates the off state, the purple color is the state of the inputs, light blue indicates the off state, and the yellow color shows the state of the outputs. This view was implemented via a JavaScript canvas in the Virtual Setup Platform.

The supervision module is the platform area that monitors the programs being executed by the student. It shows the states of inputs, outputs, and variables used by the PLC in the didactic plant through tables that show the updated values of the process. This monitoring happens when the student runs a program, and from the registered data, this module monitors the database to verify changes in the teaching process. The implemented table was generated with two columns, the first with the variable names and the second with these variables' states (0 – OFF and 1 - ON).

The learning module was developed to capture the operation steps of the didactic plant. If the teacher passes a programming activity to the student and already has the program to run this activity, then learning starts when you activate this module of the operating steps of the didactic task. This learning process operates from the storage of input and output states; the step name will be called when there is a variation in these states. Therefore, each didactic plan can be learned if its operation is correct at learning.

Fig. 10 shows how the learning algorithm of the didactic plant works. The first column of the table contains the names of the inputs and outputs of the didactic process. When the learning module is activated, it performs a first comparison with a standard step, where the states of this step are with all elements in logic level – 1 (on). When there is a different variation in this default step, the learning module algorithm compares the operation of the didactic plan in real time. The current Step data are stored when there is a variation between the previous and current steps. The red and green borders are examples of variation between the previous step and the current step. The algorithm ends its learning when it finds a sequence of repetitions between the steps. A path of steps is stored for the student to follow from that moment on when executing their activity. This algorithm was implemented in PHP with the help of JavaScript, performing page updates in a particular area.

List I/O	Standard Step	Step 1	Step 2	Step 3	Step 4
E0 – STOP	1	1	1	1	1
E1 – START	1	0	1	0	0
Q0 - Esteira 1	1	0	1	1	0
Q1 - Esteira 2	1	0	0	1	0

Figure 10 - How the learning module works.

Finally, the step view module and activity correction consist of visual monitoring of the programming performed by the student. The list generated with the learning algorithm is used to check which step the students are in at that moment and if they are on the right path. This module can only be activated if the learning of the didactic plant is carried out. Its implementation uses a database table to store the steps, where each row of this table represents a step learned by the learning algorithm. During the program's execution, each step already known is shown in the form of squares, and as the student program is executed correctly, a blue circle moves through the stages following the process. As the student does something step wrong, the ring turns red and stops at the last step hit by the student.

C. Experimental Procedures

A scenario was created to demonstrate the functioning of the virtualization and correction of a didactic process in virtual setup so that the student could check their developed program. For the learning and revision of the student's activity, a program for the correct functioning of the didactic plan was also created, simulating the program developed by the teacher. The following topics show the steps of this demonstration.

The didactic plant used in this experiment was developed in Factory IO software to simulate the student's didactic plant. The student's and the teacher's ladder programming were developed in the Somachine basic software. Fig. 11 shows the simulated didactic plan. The scenario of this plant has an industrial conveyor belt with a product diverter for a ramp and four optical sensors, a push button for START, and another for STOP.

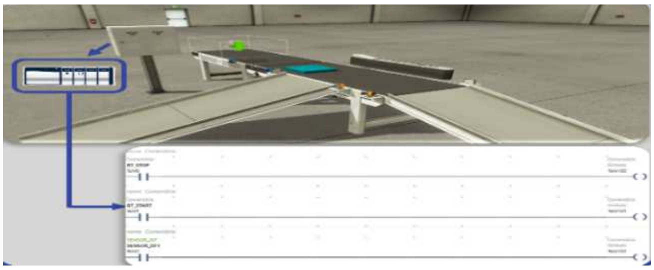

Figure 11. Simulated didactic plant and the control element.

The proposed activity for the student was to develop a program where the product would be placed on the conveyor belt when pressing the START button. Then, when passing through the optical sensor, trigger the diverter so that the product falls on the ramp. After the product falls, the diverter should be deactivated 5 seconds later.

The virtualization module generated the proposed graphic model so that the student could follow their activity, as shown in Fig. 12. It is also possible to observe that the graphical model was generated with the same number of inputs and outputs as the didactic plant.

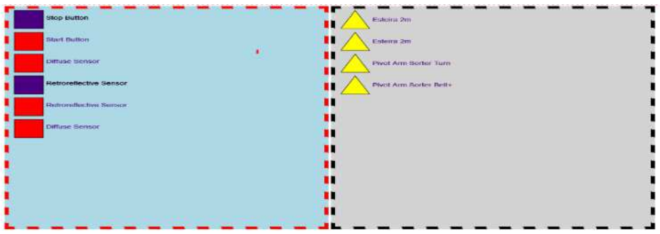

Figure 12 - Virtualization of the didactic plan.

After the learning algorithm had learned the didactic plant proposed for the student, a correct operation program was executed to verify if the blue color circle passed in each step according to the plant learning. Fig. 13 shows the steps created and the circle blue. Fig. 14 shows another program development with a malfunction inserted to check if the errors were detected by showing the red circle in the last correct step.

The objective of automatically correcting the ladder language programming activities performed by the student was achieved. The learning program performed correctly by running the developed algorithm. Automatic virtualization was obtained, always keeping the number of inputs and outputs of the didactic plan equal to the number of objects of the graphical model representing the process.

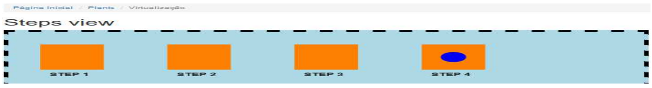

Figure 13 - Correct program operation.

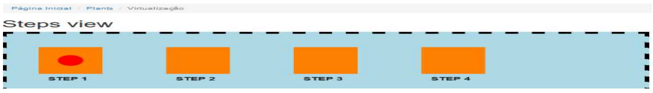

Figure 14 - Program with malfunction.

V. Didactic Tool for Teaching Automation And Control Based on An Industrial Tank System

The emergence of new simulation tools in conjunction with IoT and Industry 4.0 enables new teaching control and automation [1, 2, 32]. On the one hand, Magnus and colleagues detailed a case study for teaching the stability and control of synchronous machines based on hybrid project-based learning (h-PBL) [1]. On the other hand, the group from Krupnova evaluated the possibility of using virtual reality to teach environmental education [2]. Finally, Marin and coauthors developed a control simulation tool called linear control system design (LCSD) to teach students basic linear control concepts at the Pontifical Catholic University of Valparaiso in Chile [32].

This lab aims to present the integration of tools to teach basic modeling, identification, control, and automation concepts during the Task Industry 4.0 and Cyber-Physical System. The basic premise is to recreate industrial processes in the Factory IO software and perform process control via Node-RED, allowing the user (student) to monitor the proposed process based on the classic task of controlling the tank level. The primary tool for this work is Node-RED due to its ability to integrate various devices with various protocols. This is similar to the work developed by Ferrari's group [33], where the authors use Node-RED to measure the transmission delay and reception of messages using typical IoT protocols such as IPv4 and IPv6. We also used the experience of Chaczko and Braun [34], which described how to program a Node-RED application in conjunction with a Raspberry Pi step-by-step.

A. Technology Integration

Fig. 15 presents the flowchart of technologies used to carry out this work, and we give a simplified description of each part of the flowchart. Factory IO is a 3D industrial process simulator for teaching automation designed to be user-friendly. The software also allows the emulation of a programmer logic computer (PLC) to ensure the communication of the simulated plant's sensors and actuators.

SoMachine is a free graphical programming tool that supports the ladder language developed by Schneider Electric to allow communication with their PLCs. In this sense, the sensors and actuators of the simulated process in Factory IO communicate with SoMachine through the virtual PLC using the TCP/IP protocol in which SoMachine operates as a server and FactoryIO operates as a client. Node-RED is widely used in IoT applications [33, 34]. In our proposal, Node-RED has two fundamental roles. The first is constructing a graphical interface that allows the user to interact with the plant developed in Factory IO. The second is implementing the PI control routine that will allow the control and regulation of the liquid level in the tank.

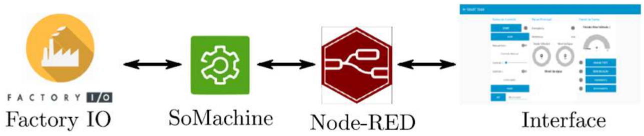

Figure 15. Flowchart of technologies used for the integration of Node-RED with Factory IO

The interface establishes communication between the user and the plant. The interface is programmed via Node-RED and allows the user to monitor and control the system's variables of interest. The essential operation is to ensure that the user can control and monitor the industrial process in Factory IO through the interface developed in Node-RED.

B. System Features

The plant used to teach control and automation is the process tank shown in Fig. 16. The plant consists of an input and output valve to allow liquid circulation. In addition, there is a level sensor and a flow sensor for monitoring the variables of interest. The tank sensors and actuators are arranged on the virtual PLC (Fig. 17).

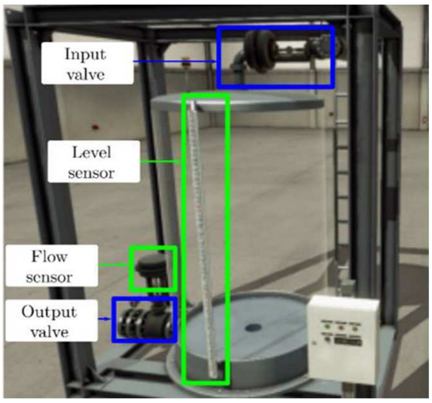

Figure 16. Process tank used for integration.

The opening of the input valve is controlled proportionally to the applied voltage, such that for being completely open, 5 V is used, and for being entirely closed, 0 V is applied. On the other hand, the output valve is configured to operate as on/off. 5 V is used to activate it; otherwise, 0 V is applied.

The system variable of interest is the height of the liquid, measured by the level sensor. The tank was configured to have a maximum size of 250 mm, which guarantees that the liquid level never exceeds 90% of this height for the safety of the devices related to the system. The tank model is mathematically described through the Bernoulli equation, allowing parameter identification strategies to estimate constructive aspects of the tank. With the plant correctly configured, SoMachine is used to perform ladder programming for process automation.

SoMachine acts as a connection path between Node-RED and FactoryIO. The sensor signals are sent from FactoryIO to the node. In contrast, the valve control signal goes the other way around, and SoMachine directs the message sent to its destination.

Fig. 18 shows the programming environment used to perform the logical flow in Node-RED. The flow created is responsible for monitoring the signal coming from the plant and providing decision-making to ensure the regulation of the tank level. Thus, the flow is also responsible for implementing the PI control routine to control the tank level. The graphical interface is configured to monitor the system in the flow. In this case, the interface is separated into panels with similar functions.

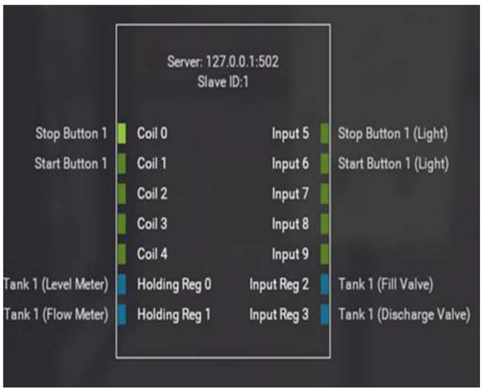

Figure 17. Factory IO virtual PLC showing input and output connections for sensors and actuators.

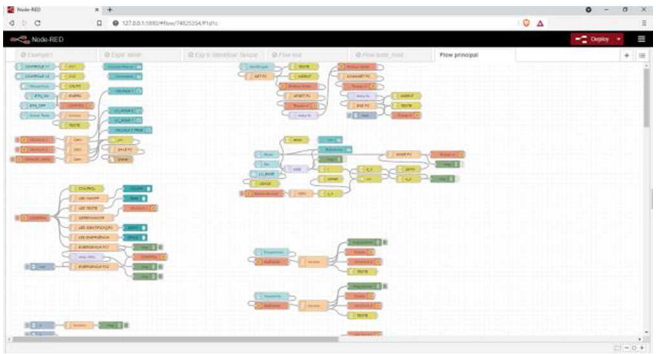

Figure 18. IDE for block programming in Node-RED.

The dashboard has a set of indicators that inform how the tank is performing in real time or based on saved data. You can also see that it has many elements to control. The dashboard is separated into three groups so that its parts are organized and related to the same objective: Control Panel, Main Panel, and Test Panel.

The control panel has the system control function, allowing the user to control the tank from a distance. There are two buttons called Start and Stop that turn the system on and off, and next to the power button, there is a light indicator to let you know that the system is on. The panel has just below a switch that switches between Manual and Automatic. Manual control allows the user complete control of the tank actuators. The controller represents automatic control, and the user can modify the operating point by choosing the height of the liquid in the desired tank.

The central panel indicates the main values of the system, allowing the user to view the working state. It has a light indicator called emergency, informing any system problems. It also has a label called Reference, which tells the operating point that the system is operating. The panel has two voltage gauges applied to the valves.

The test panel simulates events in the system and verifies if the operation is still correct. Allows the performance of leakage and clogging tests in the outlet valve and data recording for analysis in an external environment. Fig. 19 shows an example of the system operation where leakage and clogging faults in the output valve are implemented.

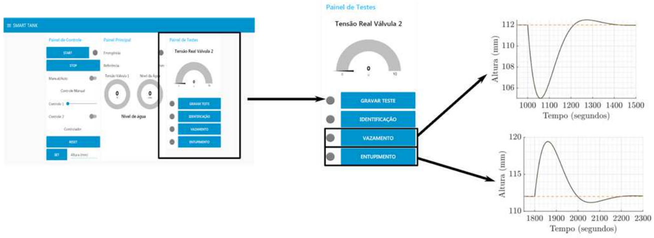

Figure 19. Example of implementation of the fault routine in the created interface.

In this example, the tank is set to the operating point under the action of a PI controller. In this way, faults can be emulated by artificially varying the voltage level applied to the output valve. In this example, it is possible to evaluate the controller's performance designed by the student and thus understand how the modification of gains alters the system's capacity in the event of fundamental problems such as clogging and leakage.

This laboratory aimed to present the integration of the 3D factory IO simulator with the Node-RED development tool for teaching control and automation to undergraduate students. In addition, process tank reconstruction was carried out, controlled by an interface that is easy to use by the user, integrating a controller for plant-level regulation even in the event of faults.

VI. Conclusion

Over the past two years, we have conducted teaching cooperation with the industry under the pandemic's constraints and obtained very interesting results. In this report, on an innovative experience in engineering education, we described our project's syllabus, how the relationship between students and teachers was through online tools, and mainly how we managed to ensure that students had access to experiments close to the industrial reality.

After completing two classes (2020 and 2021), we can state that the results were satisfactory. We acquire enough experience to make distance learning in a short time. Most importantly, we have know-how in conducting practical experiments on automation and other Industry 4.0-related topics.

We still have three years under the agreement in the subsequent years, and we plan to introduce a hybrid education approach. Starting this year (2022), the classes will contain a theoretical part ready for online teaching and a set of online practical experiments as a backup if we need to implement any lockdown. We will also introduce metrics to compare the students' performance and satisfaction in the two ways of teaching.

Acknowledgment

This research, carried out within the scope of the Samsung-UFAM Project for Education and Research (SUPER), according to Article 39 of Decree n° 10.521/2020 (SUFRAMA), was funded by Samsung Electronics of Amazonia Ltda., under the terms of Federal Law n° 8.387/1991, through agreement 001/2020, signed with Federal University of Amazonas and FAEPI, Brazil. We also would like to acknowledge the financial support from CAPES, CNPq, and FAPEAM.

REFERENCES

[1] D. d. M. Magnus, L. F. B. Carbonera, L. L. Pfitscher, F. A. Farret, D. P. Bernardon and A. A. Tavares, "An Educational Laboratory Approach for Hybrid Project-Based Learning of Synchronous Machine Stability and Control: A Case Study," in IEEE Transactions on Education, vol. 63, no. 1, pp. 48-55, Feb. 2020.

[2] T. Krupnova, O. Rakova, A. Lut, E. Yudina, E. Shefer and A. Bulanova, "Virtual Reality in Environmental Education for Manufacturing Sustainability in Industry 4.0," 2020 Global Smart Industry Conference (GloSIC), 2020, pp. 87-91.

[3] C. Karunianto, H. Saputro, "Design and Implementation Remote Laboratory based on Internet of Things." International Conference on Computer, Control, Informatics, and its Applications, No 17, 2017. DOI: 10.1109/IC3INA.2017.8251756

[4] L. Favario, "Remote Programming Environments: the Robotic Laboratory Case." IEEE International Conference on Teaching, Assessment, and Learning for Engineering (TALE), no 18, pp 820, December 2018. DOI: 10.1109/TALE.2018.8615243

[5] S. Balula, R. Henriques, J. Fortunato, T. Pereira, H. Borges, G. Amarante-Segundo and H. Fernandes, "Distributed e-lab setup based on the Raspberry Pi: the hydrostatic experiment case study." IEEE,2015. DOI: 10.1109/EXPAT.2015.7463280

[6] I. Angulo, J. García-Zubia, P. Orduña, and O. Dziabenko, "Addressing low-cost remote laboratories through federation protocols: fish tank remote laboratory," in Global Engineering Education Conference (EDUCON), 2013 IEEE. IEEE, 2013, pp. 757–762.

[7] M.L. Kalúz, R.V Cirka, and M. Fikar. "ArPi Lab: A Low-Cost Remote Laboratory for Control Education." IFAC Proc. Volumes 47 (2014): 9057-9062. https://doi.org/10.3182/20140824-6-ZA-1003.00963

[8] J.R.S. Dias, C.A Maia, and V.F Lucena, "A Computationally Efficient Method for Optimal Input-Flow Control of Timed-Event Graphs Ensuring a Given Production Rate." J Control Autom Electr Syst 26, 348–360 (2015). https://doi.org/10.1007/s40313-015-0181-7

[9] A. C. Caminero, S. Ros, R. Hernández, A. Robles-Gómez, L. Tobarra, and P. J. T. Granjo, "VirTUal remoTe labORatories Management System (TUTORES): Using Cloud Computing to Acquire University Practical Skills," in IEEE Transactions on Learning Technologies, vol. 9, no. 2, pp. 133-145, 1 April-June 2016.

[10] P. Orduña. et al. "An Extensible Architecture for the Integration of Remote and Virtual Laboratories in Public Learning Tools." IEEE Revista Iberoamericana de Tecnologias del Aprendizaje, Vol. 10, No. 4, November 2015. DOI: 10.1109/RITA.2015.2486338

[11] E. Yaprak et al., "National Science Foundation programs that support engineering education research," 2016 IEEE Frontiers in Education Conference (FIE), 2016, pp. 1-4, doi: 10.1109/FIE.2016.7757339.

[12] C. R. Brito, M. M. Ciampi, M. Feldgen, O. Clua, H. D. Santos and V. A. Barros, "Plenary: The Challenges of Education in Engineering, Computing, and Technology without exclusions: Innovation in the era of the Industrial Revolution 4.0," 2020 IEEE World Conference on Engineering Education (EDUNINE), 2020, pp. 1-3,

[13] N. Jazdi, P. Göhner, A. Brito, V.F. de Lucena, "A Germany-Brazil Experience Report on Teaching Software Engineering for Electrical Engineering Undergraduate Students," 19th Conference on Software Engineering Education & Training (CSEET'06), 2006, pp. 69-76, doi: 10.1109/CSEET.2006.6.

[14] V. Ferreira de Lucena, J. P. de Queiroz-Neto, I. B. Benchimol, A. Pereira Mendonca, V. Romao da Silva, and M. Ferreira Filho, "Teaching software engineering for embedded systems: an experience report from the Manaus research and development pole," 2007 37th Annual Frontiers In Education Conference - Global Engineering: Knowledge Without Borders, Opportunities Without Passports, 2007, pp. S4C-3-S4C-8, doi: 10.1109/FIE.2007.4417966.

[15] L. Cordeiro, R. Barreto, R. Barcelos, M. Oliveira, V. Lucena, and P. Maciel, "Agile Development Methodology for Embedded Systems: A Platform-Based Design Approach," 14th Annual IEEE International Conference and Workshops on the Engineering of Computer-Based Systems (ECBS'07), 2007, pp. 195-202, doi: 10.1109/ECBS.2007.16.

[16] L. Pinto et al., "On the use of SCRUM for the management of practical projects in graduate courses," 2009 39th IEEE Frontiers in Education Conference, 2009, pp. 1-6, doi: 10.1109/FIE.2009.5350404.

[17] R.E.V.S. Rosa and V.F. Lucena, "Smart composition of reusable software components in mobile application product lines," In Proceedings of the 2nd Int. Workshop on Product Line Approaches in Software Engineering (PLEASE '11). ACM, New York, NY, USA, 45–49. DOI:https://doi.org/10.1145/1985484.1985496

[18] M. Bahrin, F. Othman, N. Azli, and M. Talib (2016). "Industry 4.0: A review on industrial automation and robotic". Jurnal Teknologi, 78. doi:10.11113/jt.v78.9285.

[19] M. Cotteleer, and B. Sniderman, (2017). "Forces of change: Industry 4.0". URL https://www2.deloitte.com/us/en/insights/focus/industry-4-0/overview.html.

[20] M. Garetti, P. Rosa, and S. Terzi (2012). "Life cycle simulation for the design of product-service systems." Computers in Industry, 63, 361–369. doi:10.1016/j.compind. 2012.02.007.

[21] M. Grieves, (2015). "Digital Twin: Manufacturing Excellence through Virtual Factory Replication."

[22] W. Kritzinger, M. Karner, G. Traar, J. Henjes, and W. Sihn, (2018). "Digital twin in manufacturing: A categorical literature review and classification." IFAC-PapersOnLine, 51, 1016–1022. doi:10.1016/j.ifacol.2018.08.474.

[23] R.J. Rabelo, L.C. Magalhães, amd F.G. Cabral, (2020). "A Proposed Digital Twin Reference Architecture for CyberPhysical Systems in an Industry 4.0 Setting" In portuguese: "Uma proposta de arquitetura de referência de gêmeo digital para sistemas ciberfísicos em um cenário de indústria 4.0". Sociedade Brasileira de Automática, SBA,2. doi:10.48011/asba.v2i1.1382.

[24] V. Souza, R. Cruz, W. Silva, S. Lins, and V. Lucena, "A Digital Twin Architecture Based on the Industrial Internet of Things Technologies," 2019 IEEE Int. Conf. on Consumer Electronics (ICCE), 2019, pp. 1-2, doi: 10.1109/ICCE.2019.8662081.

[25] S.M. Sepasgozar, (2020). "Digital twin and web-based virtual gaming technologies for online education: A case of construction management and engineering." Applied Sciences, 10(13). doi:10.3390/app10134678.

[26] R. da Silva Mendonça, S. de Oliveira Lins, I.V. de Bessa, F.A. de Carvalho Ayres Jr., R.L.P. de Medeiros, V.F. de Lucena Jr. "Digital Twin Applications: A Survey of Recent Advances and Challenges." Processes 2022, 10, 744. https://doi.org/10.3390/pr10040744

[27] V. F. De Lucena, J. E. Chaves Filho, N. S. Viana, and O. B. Maia, "A home automation proposal built on the Ginga digital TV middleware and the OSGi framework," in IEEE Transactions on Consumer Electronics, vol. 55, no. 3, pp. 1254-1262, August 2009, doi: 10.1109/TCE.2009.5277985.

[28] R. Cruz, V. Souza, T. B. Filho, and V. Lucena, "Electric Powered Wheelchair Command by Information Fusion from Eye Tracking and BCI," 2019 IEEE International Conference on Consumer Electronics (ICCE), 2019, pp. 1-2, doi: 10.1109/ICCE.2019.8662090.

[29] M. Teófilo, V. F. Lucena, J. Nascimento, T. Miyagawa, and F. Maciel, "Evaluating accessibility features designed for virtual reality context," 2018 IEEE International Conference on Consumer Electronics (ICCE), 2018, pp. 1-6, doi: 10.1109/ICCE.2018.8326167.

[30] E. Negri, L. Fumagalli, and M. Macchi (2017), "A review of the roles of the digital twin in cps-based production systems." Procedia Manufacturing, 11, 939 – 948. 27th. International Conference on Flexible Automation and Intelligent Manufacturing, FAIM2017, 27-30 June 2017, Modena, Italy.

[31] J. David, A. Lobov, and M. Lanz, "Learning Experiences Involving Digital Twins," IECON 2018 - 44th Annual Conference of the IEEE Industrial Electronics Society, Washington, DC, 2018, pp. 3681-3686, doi: 10.1109/IECON.2018.8591460.

[32] L. Marin, H. Vargas, R. Heradio, L. de La Torre, J. M. Diaz, and S. Dormido, "Evidence-Based Control Engineering Education: Evaluating the LCSD Simulation Tool," in IEEE Access, vol. 8, pp. 170183-170194, 2020, doi: 10.1109/ACCESS.2020.3023910.

[33] P. Ferrari, A. Flammini, E. Sisinni, S. Rinaldi, D. Brandão and M. S. Rocha, "Delay Estimation of Industrial IoT Applications Based on Messaging Protocols," in IEEE Transactions on Instrumentation and Measurement, vol. 67, no. 9, pp. 2188-2199, Sept. 2018.

[34] Z. Chaczko and R. Braun, "Learning data engineering: Creating IoT apps using the Node-RED and the RPI technologies," 2017 16th International Conference on Information Technology Based Higher Education and Training (ITHET), 2017, pp. 1-8,