Authoring tool for creating immersive virtual experiences expeditiously for training

Rui Machado
UTAD
Vila Real, Portugal
rui-12@hotmail.com

Ricardo Rodrigues
UTAD
Vila Real, Portugal
rrodrigues1999.rr@gmail.com

Hugo Coelho
UTAD and INESC TEC
Vila Real and Porto, Portugal
hugo.r.mendes@inesctec.pt

Miguel Melo
INESC TEC
Porto, Portugal
mcmelo@inesctec.pt

Luís Barbosa
UTAD and INESC TEC
Vila Real and Porto, Portugal
lfb@utad.pt

Maximino Bessa
UTAD and INESC TEC
Vila Real and Porto, Portugal
maxbessa@utad.pt

2022 International Conference on Graphics and Interaction (ICGI) | 979-8-3503-4573-5/22/$31.00 ©2022 IEEE | DOI: 10.1109/ICGI57174.2022.9990392

***Abstract*—Virtual reality (VR) is still a field that is in constant development, and people are trying to use it to have a close representation of reality by creating immersive environments. However, despite the existence of some tools that have been adapted to work with VR, they require some experience to work with, and there is a considerable amount of resources that need to be spent to create and maintain the VR experiences, which prevents the adoption and use of all the benefits that VR can bring. This work proposes an architecture for an authoring tool that allows users to create their own virtual experiences without the need for an extensive understanding of it and use them to create a virtual training exercise. This paper uses a case study built upon a real training context scenario applied to the agroforestry field.**

To validate this proposal, a prototype was built and subject to usability and satisfaction tests that demonstrated the ease of understanding and learning of the interfaces and all the functionalities implemented.

***Index Terms*—immersive environments, virtual reality, authoring tool, immersive training**

I. Introduction

Over the years, Virtual Reality (VR) has been increasing in popularity. Few people are aware that VR first appeared in 1960, when the equipment was expensive and the computational power did not allow interactive experiences which made VR popularity at the time not so great but, with the emergence of lower prices for gaming and entertainment, VR experienced a rise in interest. As a consequence, the VR market has been rising as powerful companies want to revolutionize the gaming and entertainment industry [1].

Alongside VR technology, there have been multiple VR definitions one of which is Kirner et al. [2], which states that VR is an interface that allows users to access applications being executed on a computer. This interface provides real-time visualization, movement, and interaction with the generated 3D environments.

A considerable number of fields are using VR, including the creation of video games [3], education [4], and the training of professionals, which can involve military training and doctors. Many 3D resources have been developed for this technology to train users on how to behave in real-life situations. Using these 3D resources increases the interest in learning and can be used to represent real-life situations [5]. However, VR must constantly be updated to provide a good solution for each scenario. The tools used to create VR experiences are designed to be used by people with enough knowledge. Companies usually hire professionals with the knowledge to create the necessary experiences to solve this issue. In addition, these experiences will probably need some maintenance and updates throughout the time so they can work properly and keep providing a good experience to their users. Ultimately, the creation of VR experiences can be expensive and, because of that, there is a need to create a tool that allows the creation of such experiences without spending too many resources.

Training in VR brings several advantages, like specific situations can be simulated inside a controlled environment using detailed training conditions at any time; they can repeat the training the number of times necessary to understand the procedure; a large number of people can participate without spending many resources. However, as mentioned before, creating and maintaining these virtual training environments (VR experiences) can be expensive. Therefore it is necessary to create an authoring tool to overcome such obstacles.

This work proposes an authoring tool that creates VR experiences for training, developed with a real case study, a partnership with the company VICORT which fits within an ongoing project called SMARTCUT. The authoring tool allows users to create VR experiences that can be saved and opened as many times as needed to perform specific training. After its implementation, the authoring tool is properly evaluated and validated.

II. Background characterization

VR is a computing field that allows users to emerge and interact with a virtual world while using specific devices to simulate the environment and stimulate the user by providing feedback. This makes the virtual experience as close as possible to reality. One needs to address two main concepts about VR: immersion and perception. Immersion is how much a system can abstract its user from reality. Perception is the user's awareness of the surroundings through their senses [6].

A few different types of systems allow people to experience VR. The context of the experience causes the differences between these systems. Three types of systems can be defined by their immersion and perception: immersive, semi-immersive, and non-immersive systems [6], [7]. Non-immersive systems interact with the virtual environment presented to him on a computer screen using a mouse and a keyboard [8]. Semi-immersive systems allow users to interact with the virtual world while performing activities in the real world [9]. Fully immersive systems use a Head Mounted Display (HMD) to present the virtual environment to the user and where they can interact with it using specific controllers [10].

Over the years, VR has been evolving, and its involvement in different fields is increasing. Fields like video games, education, medicine, military, and psychology, among others, are being studied to understand what is the best approach to using VR in these fields. In medicine, VR can solve the interrelation between anatomic structures in the 3D space; a trainee can explore and interact with these structures anytime [11]. In the military, virtual training is used to improve individual and group abilities. A high-resolution image and advanced controls are used to be as close as possible to reality. Without a geographical limitation, many individuals can be involved in simulated confrontations based on real situations and changes in the virtual environment. The train's durability and the resources used to perform this train will decrease, and new operation ideas can appear [12]. Various applications exist, in the field of education, to help students learn about a specific matter. In NewtonWorld, students can learn about the laws of motion by interacting with objects' various states of non-gravity [13]. In MaxwellWorld, they can learn about electrostatic fields and forces with different polarities and intensities [14]. In PaulingWorld, the student can interact with a structure of atoms and see the consequences when one or more atoms are replaced [15].

A. Authoring tools for creating VR experiences

Even though tools like Unity [16], Unreal Engine [17] or Cry Engine [18]] allow the creation of virtual environments, they were not developed with such purpose. Instead, they were developed to create games and adapted to these situations over the years. One thing that did not change was the knowledge one has to have to use them. To create these environments, there are a few important characteristics to be taken into account: if there is an objective or not, the visual feedback, the light and sounds from the environment, and interfaces to help navigate through the experience [19].

Unity, being a game engine, gives the user several tools that he can use to create 3D virtual environments. The inspector interface gives information like the position, rotation, and scale of an object, and the number of assets (objects, sound, textures, etc.) the project browser has available to use. The hierarchy interface gives the objects the user uses in the virtual environment. All this does not allow the user to create a virtual environment, so there are plugins like SteamVr [20] that help content creators create the virtual environment. This tool brings new assets that can be used and new interactions, such as teleport, so the user can move around the virtual world without moving in the real world or grabbing and throwing objects. New interactions can also be created, and the user can change them anytime. When the virtual environment is completed, he can run the application, use the necessary equipment, and interact with the environment. In Unity, any user can use a vast selection of assets (some free, others need to be paid for) to implement in their experience. Still, when there is the need to create some logic with those assets, they will need some programmatic knowledge.

Despite how good tools like Unity are, they are not ideal for a person that does not have the essential knowledge to use them. One tool was proposed by Coelho et al. [21], which allows users to create a 3D virtual environment for multisensory experiences, and is based on 360º videos in two ways: with a desktop and with an immersive interface to test which one is the best. This tool allows users to select any frame from the video timeline and add/remove a stimulus. At any time, the user can visualize the experience on the desktop or with the help of HMD. The only difference is the interface presented to the user and how he interacts with it. Some tests were made, and the team concluded that both interfaces were acceptable and preferred the immersive interface to the desktop interface. This tool is simple, and everyone can use it, but it is limited to using a 360º video to create a virtual environment.

VRIA is another tool for creating a virtual environment, and it is a web-based framework for creating 2D or 3D environments. This framework allows users to create and test prototypes without leaving the web application, making this tool easier, quick, and more accessible to a larger audience [22]. The graphic can be constructed by importing a file in JSON or CSV or creating one. Even though it is impossible to watch these creations with the HMD in the web application, the user can use a mouse and a keyboard to navigate the 3D environment. There are available some examples that we can see and even change. To visualize the created environment in a virtual world, using NodeJs and VR equipment is necessary. This tool is simple, and people who know about JSON can benefit from it.

Both these tools that were developed to facilitate the creation of VR experiences have limitations. The authoring tool proposed by Coelho et al. needs 360º videos to create a virtual environment, and the VRIA tool will need to have some data to create a graphic. Even further, these tools were developed to create a more visualization environment than an interactive one. The tool presented next does not need information from the outside to create a virtual environment, and the interaction levels are higher.

III. Proposal for an authoring tool for immersive experiences

The development of this authoring tool considers SMART-CUT project, which is connected to company VICORT. This company is dedicated to manufacturing, commercializing, and assisting agricultural, forestry, and environmental equipment. They produce forest machines and harvesting heads in the forest area that will be used in specific work situations. All this equipment needs maintenance to verify if it will work correctly to prevent possible accidents. Trained professionals do the maintenance; hiring or training them can be expensive. And so, a discussion was held with VICORT, and it retrieved their expectations, which the authoring tool should address when creating immersive experiences intended for training. With their feedback, functional and non-functional requirements were defined, representing the functionalities the authoring tool should allow users to perform and are presented by the following requirements:

- Create environments - The authoring tool should allow the creation of an environment where it will be executed the different tasks by the operators during the training they are performing.
- Save environments - The authoring tool should allow storage of the environments created to be loaded whenever needed.
- Load environments - The authoring tool should allow loading environments previously created and saved to be edited or executed in a training.
- Define the terrain deformations - The authoring tool should allow the creation of the terrain, being plane, tilted, or on terraces.
- Define the type of trees - The authoring tool should allow the introduction, in the terrain, of different trees being straight, irregular, single-foot trees or multiple-foot trees.
- Distribution of objects - The authoring tool should allow the position and movement of every object available in the environment.
- Create training - The authoring tool should allow the creation of training where it will be defined the tasks that the operator needs to perform.
- Save training - The authoring tool should allow storing the created training to be loaded whenever needed
- Load training - The authoring tool should allow loading training previously created and saved to be edited or executed.
- Define the schedule - The authoring tool should allow the definition of the time of the day when the train will be executed.
- Choose a harvesting head - The authoring tool should choose which harvesting head will be used in the train that will be executed.

This authoring tool aims to create specific real-life situations in the deforestation area that can be used at any time and as many times as needed, so the users can use these scenarios to train procedures used in the day-by day job that this area requires. To do this, there are a few requirements that need to be accounted for: the creation of an environment that represents the real one, where in this part, the definition of the terrain is what is most important to have when it comes to representing the reality; the creation of training that represents a real day of work. Thus, this authoring tool allows any user to create training scenarios where they can choose to deform the terrain and the life forms it contains. Furthermore, users can also choose which harvesting head they want to use in the environment that they create because every harvesting head has its purpose for a specific situation. Before loading the created training, the user can choose between a non-immersive or an immersive training environment.

A. *Authoring tool architecture*

The authoring tool should allow the user to create an experience and, with that experience, create a training environment. The authoring tool comprises two editors: an environment editor and a training editor. The environments created will be used on the training editor because training cannot be executed without an environment. Several pieces of training can occur using the same environment, allowing reuse instead of creating new ones. Whenever the user wants to access the existing environments or training, a communication is made to retrieve some data (if there is any). The user can then use that information to find an environment or training to be executed or edited. The authoring tool architecture is presented in Fig. 1.

The environments will be created by a user in the environment editor where, after he executes all the interactions available and uses the available components (assets repository), this will suffer a serialization in a specific format (JSON format) and be transformed into a string. All this information will then be saved into a database, and the list that contains the available environments will be updated. If the user wants to edit an environment or execute training, the serialized components need to be deserialized to build the environment correctly. After the environments are all updated, the user can use them to choose one for his training.

The training scenarios are created in the training editor by a user. This editor has an interface where the user configures the scenario, assigning it a name and description, defining the time of day, the environment previously created in the environment editor, and the harvesting head. The environment and the harvesting head are mandatory parameters so the training can be executed later. In the end, the training will be saved into the database, and the list that contains the training environments will be updated. Every training has a scenario identifier so it can be loaded when the training is executed or if the user wants to edit it. This information can be consulted with the same editor if the user wants to change some training parameters.

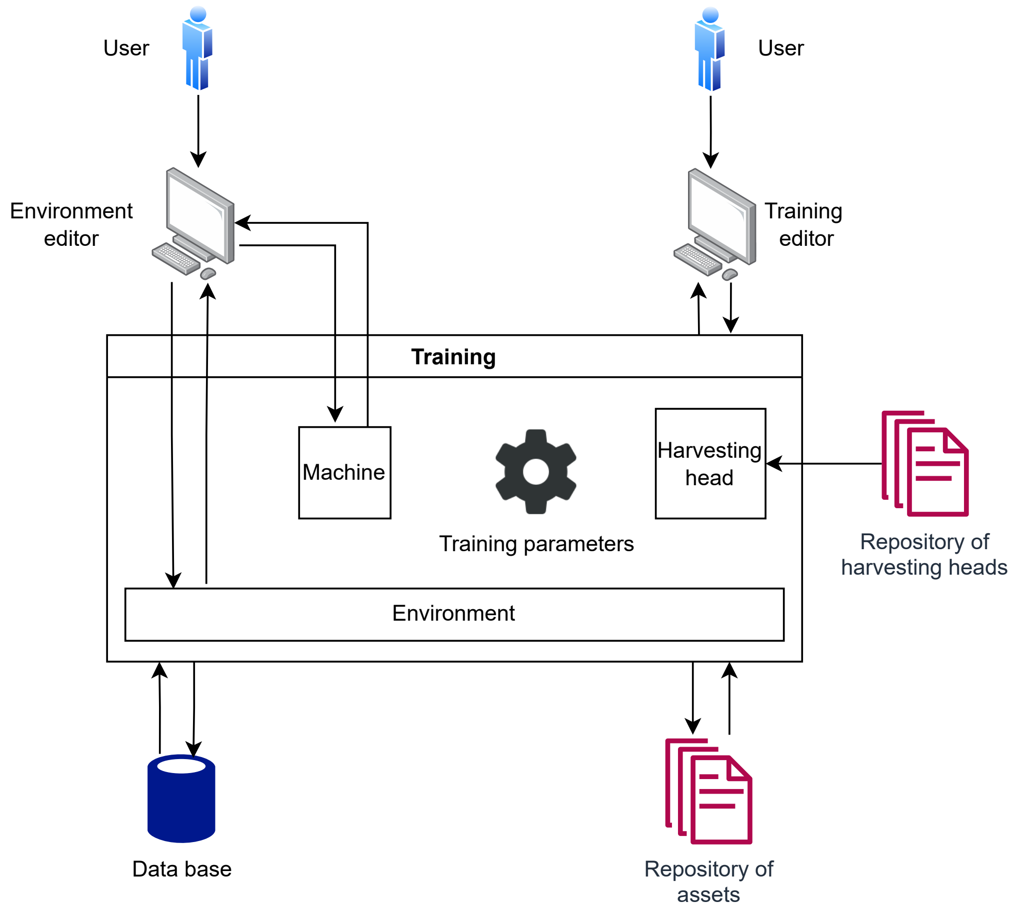

Fig. 1. Authoring tool architecture

B. Authoring tool functionalities

The objective of this work was to create an authoring tool that allows users to create their experiences, immersive or non-immersive. This authoring tool is not immersive but allows any user to create a training where, before being executed, they can choose if they want to be in a non-immersive or an immersive environment. Because of this, this tool was divided into two editors: an environment editor and a training editor. In the environment editor, the user can create or edit several environments with some available objects and with the functionalities implemented. In the training editor, users can create or edit several courses where they can choose in which environment they want to execute the training. A database was implemented to store every necessary information to have a more accessible way to access and save the experiences.

In the environment editor, there are three main functionalities. First, the user can deform the terrain (Fig. 2) in several ways depending on the chosen mode. Despite the mode, he also can choose the area of effectiveness, represented by a white brush following the mouse movement, and the strength that will be used to deform the terrain. After that, he can raise or lower the terrain, flatten, which allows the user to choose the target height he wants to apply to deform the terrain, and smooth, which allows the user to soften some parts of the terrain. The authoring tool also allows for generating a random terrain, with all deformations in just one button. When it comes to the objects, the user can choose three different modes: selection, generate one and generate in the area (Fig. 3). In the selection mode, the user can select the objects already positioned in the terrain (the selected ones will change the colour to green). He can select one, select one by one, create a group, or select several simultaneously. After that, he can change the position and the rotation of the selected objects or even delete them. In the generate one mode, the user can select one object, and the object will follow the mouse movement until it is placed in the terrain. In the generate in area mode, the user can see a white brush that will follow the mouse. The brush size can be chosen, and the objects will be placed inside. The last functionality available is textures paint (Fig. 4), where the user can choose between vegetation or details to paint the terrain. Just like the deform terrain functionality, the user can choose an area and a strength to paint the terrain, or we can generate a detail all over the terrain with the random generation option. Ultimately, the environment can be saved to be used later on.

The training editor (Fig. 5) is more straightforward than the environment editor. In this one, the user only needs to fulfill a few fields and choose which harvesting head he wants to use in his training and in what environment he wants to execute it. One important functionality is the possibility to edit the environment associated with the training being created/edited. In this way, the user does not need to look for that specific environment if he wants to edit it. To interact with the authoring tool, the user needs to use a mouse and a keyboard, and the interfaces will be presented on a monitor.

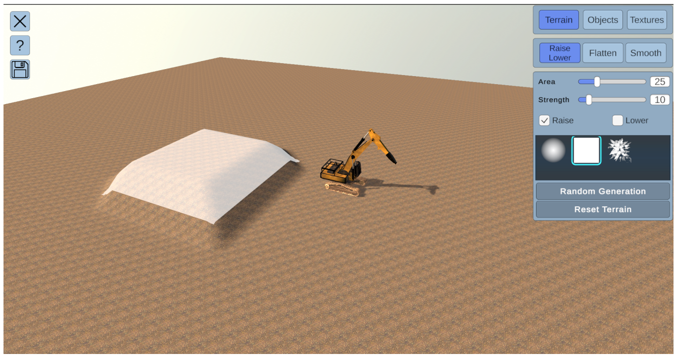

Fig. 2. Interface for editing the terrain in the environment editor.

Fig. 3. Interface for editing the objects in the environment editor.

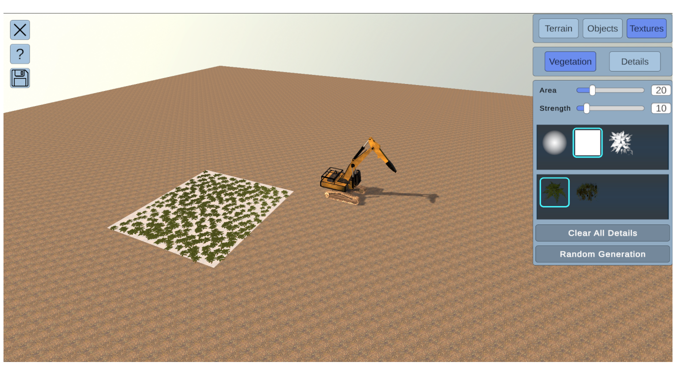

Fig. 4. Interface for editing the textures in the environment editor.

IV. Usability evaluation

A user usability study was conducted to evaluate and validate the authoring tool. Namely, usability and satisfaction evaluations were conducted for the authoring tool, as described next.

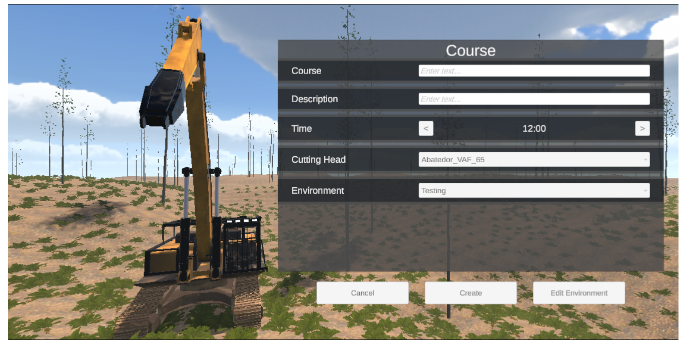

Fig. 5. Interface for the training editor.

A. Sample

Twenty participants (17 males and 3 females) tested all the authoring tool's features. The participants were between 20 and 30 years old (M = 21.3, SD = 2.20), and everyone reported normal or corrected-to-normal vision. According to the socio-demographic questionnaire introduced to the participants before they started the protocol, they all use computers in their everyday lives.

B. Instruments

To test the authoring tool and evaluate the usability of the interfaces and its functionalities, a protocol was defined with a total of 33 tasks involving authoring a virtual environment and a training scenario (28 tasks for the environment editor and five tasks for the training editor). In addition, a socio-demographic questionnaire was applied to gather information about them. As this was a study to evaluate the usability of the authoring tool, a System Usability Scale (SUS) for the overall system usability and the After Scenario Questionnaire (ASQ) for the overall satisfaction were used. The SUS scale is composed of 10 questions that evaluate the usability of the authoring tool. The ASQ scale is composed of 3 questions that evaluate the participant's satisfaction with the experience.

C. Materials

All the experiences that were made ran on a desktop computer with an Intel I7-8700K CPU, NVIDIA Geforce 2080 TI, 32GB of RAM, and Windows 10. As an interface display, the monitor used was a DELL S2340T display with 60HZ. With the desktop, a keyboard and a mouse were used to interact with the interfaces of the authoring tool.

D. Variables

The variables defined in this study were: system usability (SUS scores), experience satisfaction (ASQ scores), the number of keyboard clicks, the number of mouse clicks, the total number of clicks combined, the number of help requests, and the number of errors.

E. Procedure

The studies were conducted in a research laboratory in which the research team controlled the environment. The room has acoustic treatment and there was no noise from outside sources, so the participants could not be distracted. The first process in the experience was to receive the participants and give them a small explanation about the experience they were about to perform. It also explained the purpose of the authoring tool, why it was built and what it would be used for. After that, it was asked to the participant to read a consent paper and sign to express his agreement to participate in the study. The participant was asked to fill out a concise socio-demographic questionnaire after the signing. Next, it was given to the participant a small tutorial about one of the parts of the authoring tool (environment editor), so he would get used to the navigation of the interface and the different functionalities implemented. Following that the test protocol was given to the participant, and he was informed that if he needed any help, he could request it if he had any doubt about the functionalities of the tasks defined during the experience. During the experience, the researcher registers the number of help requests from each participant.

After completing the test protocol, it was asked to the participant to fulfil the SUS and the ASQ questionnaire. Finally, it was given the opportunity to each participant to test the environment that he created using the HMD.

F. Results

When it comes to usability and satisfaction, the values that were obtained were the following: for usability, the minimum value was 72.5, the maximum value was 100, and the average value was 87.13 (Table I); for satisfaction, the minimum value was 5.8, the maximum value was seven and the average value was 6,68 (Table I).

Regarding the number of clicks that each participant had on the keyboard, mouse, and in total, the values are between: for the keyboard, the minimum value is 83 clicks, the maximum value is 204 clicks, and the average value is 104.5 clicks; for the mouse clicks the minimum value is 91 clicks, the maximum value is 216 clicks, and the average value is 162.35 clicks; for the total number of clicks, the minimum value is 192 clicks, the maximum value is 415 clicks, and the average value is 302.85 clicks (Table II). For the number of help requests and number of errors, the values are the following: the average value for the number of help requests is 0.4, and the total is 8; the average value for the number of errors is 0.15, and the total is 3 (Table III).

TABLE I
RESULT OF THE USABILITY AND SATISFACTION TESTS

Participants	SUS scores	ASQ scores
Participant 1	95	7
Participant 2	85	6,166666667
Participant 3	90	6,666666667
Participant 4	80	6,5
Participant 5	90	6,833333333
Participant 6	75	6,666666667
Participant 7	95	7
Participant 8	90	7
Participant 9	77,5	5,833333333
Participant 10	90	7
Participant 11	77,5	6,833333333
Participant 12	87,5	7
Participant 13	87,5	7
Participant 14	97,5	7
Participant 15	77,5	5,833333333
Participant 16	100	7
Participant 17	97,5	NR
Participant 18	92,5	6,5
Participant 19	72,5	6,166666667
Participant 20	85	7
Mean	87.13	6.68

TABLE II
NUMBER OF CLICKS OF THE KEYBOARD, MOUSE AND TOTAL

	M	SD
Keyboard	140.5	34.45
Mouse	162.35	36.99
Total	302.85	60.47

V. DISCUSSION

The values obtained from the experiences were used to see if the interaction with the authoring tool was easy to learn, understand, and interact with every functionality implemented. The results showed that all the functionalities and interfaces were easy to learn, interact with, and understand. As one can see by the usability number, this result can be found in the A percentile (if the result is bigger than 85, it can be classified as A, which equals excellent), and it tells us that the authoring tool is easy to interact with, and the participants will most likely recommend it to other users. As for the satisfaction results, the average number is superior to 6, which can be classified as positive (on a scale of 1 to 7). This can explain the low average number of help requests and errors, where the participants finished the protocol without any difficulties. The information provided to them to complete the protocol and learn the interactions with the authoring tool was helpful.

For the number of keyboard clicks, mouse clicks, and total clicks, mouse clicks are higher than the keyboard clicks. However, this should not be the case. The research team expected the number of keyboard clicks to be higher than the number of mouse clicks. The number obtained is also too high then was expected. The keyboard clicks should have been between 100 - 110 clicks, the mouse clicks should have been between

TABLE III
RESULTS OF THE NUMBER OF HELP REQUESTS AND NUMBER OF ERRORS

	M	SD
Requests	0.4	0.6
Errors	0.15	0.67

80 - 90 clicks, which will make the total between 180 - 200 clicks. These are the values that were expected if the protocol was complete almost without mistakes. During the protocol participants made mistakes and realizing what they had done, they redo some tasks which can explain why the number of clicks is higher than expected. Not all the participants had these types of results. In some situations, the keyboard clicks were higher than the mouse clicks. Some participants used the keyboard to move around the environment so they could have a full view of the terrain. Others just left the camera in the initial position. Despite almost every interaction with the authoring tool being made with the mouse, the number of keyboard clicks should still be higher than the number of mouse clicks. It was noticed, during the experiences, that some participants had some difficulties in selecting the objects (which use the mouse) that were presented in the environment. So, they redo that specific task multiple times, which explains why the number of mouse clicks is higher than the number of keyboard clicks.

One of the problems many participants refer to was the lack of objects available to position in the environment. Still, as this authoring tool is inside project SMARTCUT it was developed only to create environments that could be used to train people in the deforestation area. This makes the tool created restricted to only one area of work, so the tool will not be helpful for any creation that does not have this purpose. Besides that, the tool could be improved by adding new objects later on and new functionalities so that it could be used in other areas and would not be restricted to only one.

VI. Conclusion

The purpose of this authoring tool is to find a way of creating VR experiences without having too much trouble maintaining them functional and spending too many resources creating them. With the tools available to create these experiences, the users need to have considerable knowledge to perform these creations and make them as close as possible to reality. This tool will help users create their own experiences without spending too many resources and without needing vast knowledge. These experiences can then be executed in a non-immersive or an immersive environment (the user will have the possibility to choose that before he executes the experience).

Based on the project's requirements and the defined architecture, a prototype was built to be tested and to understand if such a tool was easy to understand and interact with every functionality. The study revealed that usability and satisfaction had a higher score, meaning the authoring tool interfaces were easy to learn and interact with. Furthermore, despite the authoring tool being restricted to the project SMARTCUT, and, as a consequence, restricted to one application area, the code that was used to create some parts of the authoring tool (where the user can create, save and edit an environment and training) can be reused and adapt to other working areas and so, the objective is to increment its functionalities and expand the application area to other areas of work where VR training can be a benefit.

To complement the authoring tool, a functionality that can improve it would be the addition of newer 3D models (assets) that can be used to populate the environment that is being created. This will make the authoring tool more expeditiously and allow the user to have more options when it comes to populate the environment and not be restricted to the ones that he already has. When a user is performing training, in the end, he should be able to see how good he was and what he needs to do to improve so. Implementing Key Performance Indicators could help the creator of the training to establish some objectives so the trainee can learn and improve in all kinds of situations.

Acknowledgment

This work is co-financed by the ERDF – European Regional Development Fund through the Operational Programme for Competitiveness and Internationalisation - COMPETE 2020 under the PORTUGAL 2020 Partnership Agreement, and through the Portuguese National Innovation Agency (ANI) as a part of project “SMARTCUT - Diagnóstico e Manutenção Remota e Simuladores para Formação de operação e manutenção de Máquinas Florestais: POCI-01-0247-FEDER-048183”.

Appendix

Protocol

- Access the environment menu.
- Choose the environment ”Experimental terrain” and press the option edit terrain.
- Create a terrain using the random generation with 8555 seed.
- Select the tool raise/lower in the terrain section.
- In the frontal position of the machine, raise the terrain using an area of 50 and a strength of 65.
- Lower the terrain around the machine.
- Select the flatten tool in the terrain section.
- In the right lower corner of the terrain (use the machine as a reference), flatten the terrain to a height of zero, with an area of 20 and a strength of 10.
- Smooth the deformation created in that same corner.
- Select the category environment in the option ”Objects”.
- Add the available object with 1000 elements and seed 23.
- Add a trunk with the size 100 in the right lower corner that was deformed.
- Add a tree with the size 100 by the side of the machine.
- Select the option ”Add multiple objects”.
- Add 10 trees by the side of the previous one, with the area 10 and seed zero.
- Select all trees and try to position them inside the machine.
- Undo the previous action.
- Select the option ”Textures” and the first detail.
- Add the detail to the terrain using the random generation and the strength value to the max.
- Delete the detail around the machine.
- Select the second texture.

- Add the texture underneath the machine with a strength of 100.
- Select the trunk that was added to the terrain.
- Rotate it using any axis available.
- Select two trees and delete them.
- Save the environment with the name ”Experimental terrain V.2”.
- Press the button ”Save as”.
- Close the environment terrain.
- Access the course menu and create a new course.
- Use the name ”Experimental train” with the description ”My first train”.
- Change the schedule to five PM and choose the harvesting head 340-HD.
- Choose the environment that you just create.
- Save the course that you just create by pressing the button ”Save”.

References

[1] I. Wohlgenannt, A. Simons, and S. Stieglitz, “Virtual Reality,” *Business and Information Systems Engineering*, vol. 62, no. 5, pp. 455–461, 2020.

[2] C. Kirner, *Realidade Aumentada : Conceitos , Projeto e Aplicações*, 2007, vol. 1.

[3] J. van den Heuvel, J. Cunliffe, and E. Parker, “The Right Foot in the Wrong Place,” pp. 1–2, 2021.

[4] H. Rodrigues Batista, P. Ricardo Batista Mesquita, and M. Antonio Gaspar, “Simulador de Realidade Virtual Aplicado à Educação Patrimonial para Experiências Imersivas Gamificadas,” *Renote*, vol. 16, no. 2, pp. 230–242, 2018.

[5] T. Monahan, G. McArdle, and M. Bertolotto, “Virtual reality for collaborative e-learning,” *Computers and Education*, vol. 50, no. 4, pp. 1339–1353, 2008.

[6] Y. A. G. V. Boas, “Overview of Virtual Reality Technologies,” *Interactive Multimedia Conference*, p. 4, 2013.

[7] H. García-López, E. Obrero-Gaitán, A. M. Castro-Sánchez, I. C. Lara-Palomo, F. A. Nieto-Escamez, and I. Cortés-Pérez, “Non-immersive virtual reality to improve balance and reduce risk of falls in people diagnosed with parkinson’s disease: A systematic review,” *Brain Sciences*, vol. 11, no. 11, 2021.

[8] R. Bevilacqua, E. Maranesi, G. R. Riccardi, V. Di Donna, P. Pelliccioni, R. Luzi, F. Lattanzio, and G. Pelliccioni, “Non-immersive virtual reality for rehabilitation of the older people: A systematic review into efficacy and effectiveness,” *Journal of Clinical Medicine*, vol. 8, no. 11, pp. 1–14, 2019.

[9] S. B. Moro, M. Carrieri, D. Avola, S. Brigadoi, S. Lancia, A. Petracca, M. Spezialetti, M. Ferrari, G. Placidi, and V. Quaresima, “A novel semi-immersive virtual reality visuo-motor task activates ventrolateral prefrontal cortex: A functional near-infrared spectroscopy study,” *Journal of Neural Engineering*, vol. 13, no. 3, 2016.

[10] T. S. Mujber, T. Szecsi, and M. S. Hashmi, “Virtual reality applications in manufacturing process simulation,” *Journal of Materials Processing Technology*, vol. 155-156, no. 1-3, pp. 1834–1838, 2004.

[11] A. Cardoso, E. Júnior, C. Kirner, and J. Kelner, “Tecnologias e Ferramentas para o Desenvolvimento de Sistemas de Realidade Virtual e Aumentada,” *UFPE, Recife*, pp. 1–19, 2007.

[12] X. Liu, J. Zhang, G. Hou, and Z. Wang, “Virtual Reality and Its Application in Military,” *IOP Conference Series: Earth and Environmental Science*, vol. 170, no. 3, pp. 1–8, 2018.

[13] E. A.-L. Lee and K. W. Wong, *A Review of Using Virtual Reality for Learning*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 231–241.

[14] C. Dede, M. C. Salzman, and R. B. Loftin, “MaxwellWorld: learning complex scientific concepts via immersion in virtual reality,” pp. 22–29, 1996.

[15] S. Su, R. Loftin, D. Chen, and Y. Fang, “Distributed collaborative virtual environment: Paulingworld,” *Proceedings of the 10th International Conference on Artificial Reality and Telexistence*, pp. 112–117, 2000.

[16] U. Technologies. (2022) Unity. [Online]. Available: https://unity.com/

[17] E. Games. (2022) Unreal engine 5. [Online]. Available: https://www.unrealengine.com/en-US/unreal-engine-5

[18] Crytek. (2022) Cry engine. [Online]. Available: https://www.cryengine.com/

[19] J. Novak, *Game Development Essentials Third Edition*, 2012.

[20] J. W. Murray, *Building Virtual Reality with Unity and SteamVR*. CRC Press, 02 2020.

[21] H. Coelho, M. Melo, J. Martins, and M. Bessa, “Collaborative immersive authoring tool for real-time creation of multisensory VR experiences,” *Multimedia Tools and Applications*, vol. 78, no. 14, pp. 19 473–19 493, 2019.

[22] P. W. Butcher, N. W. John, and P. D. Ritsos, “VRIA: A Web-Based Framework for Creating Immersive Analytics Experiences,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 27, no. 7, pp. 3213–3225, 2021.