

A Novel Tool for Immersive Authoring of Experiential Learning in Virtual Reality

Fernando Cassola*
INESC TEC,
Faculdade de Engenharia,
Universidade do Porto

Manuel Pinto†
INESC TEC

Daniel Mendes‡
INESC TEC,
Faculdade de Engenharia,
Universidade do Porto

Leonel Morgado§
INESC TEC,
Universidade Aberta

António Coelho¶
INESC TEC,
Faculdade de Engenharia,
Universidade do Porto

Hugo Paredes||
INESC TEC,
Universidade de
Trás-os-Montes e Alto Douro

ABSTRACT

Training in VR can reduce risks and costs while allowing frequent and diversified experiential learning activities. We present a novel VR immersive authoring tool for experiential learning courses with industrial machinery. A trainer can create a course from scratch, defining all its components (structure, models, tools, and settings). The actions which trainees should perform can be specified by demonstration. After completing the course, trainees' actions will be matched against the trainer's.

Index Terms: Human-centered computing—Visualization—Visualization techniques—Treemaps; Human-centered computing—Visualization—Visualization design and evaluation methods

1 INTRODUCTION

In manufacturing and industry, adapting to requests with a high degree of customization and complexity implies longer cycle times. Immersive virtual reality (VR) for simulated training is a promising approach [1], helping operators minimize errors and increase safety. Also, VR is used to perform operational tasks in industrial settings [2], by mirroring the status of a physical environment analog ("digital twins"), supporting remote expert intervention. Both approaches (simulation and digital twins) enable experiential and situated learning, in areas such as work safety [3], sports [4], mechanical maintenance [5,6] and mining [7].

However, the creation of VR training experiences is not widespread, requiring expensive and time-consuming development processes (e.g. using Unity [8] or Unreal Engine [9]) involving software development experts [10]. To mitigate this, authoring tools have been proposed for the development of VR applications for training and certification (e.g. EON Reality [11] and XVR Simulation [12]). Their current limitations include lack of integration with existing data (such tools do not allow the creation of VR scenarios from pre-existing data in organizations, such as engineering CAD files) and desktop-oriented. Also, non-immersive authoring processes detach the immersive experience from its creation (users specify training procedures without experiencing the trainee task performance perspective).

*e-mail: fjc@inesctec.pt

†e-mail: manuel.a.alves@inesctec.pt

‡e-mail: danielmendes@fe.up.pt

§e-mail: leonel.morgado@inesctec.pt

¶e-mail: antonio.coelho@inesctec.pt

||e-mail: hugo.paredes@inesctec.pt

In this work, we tackle these limitations with a novel VR authoring tool, enabling immersive authoring of immersive training. Within a situated VR environment, authentic interactive digital representations are generated from pre-existing CAD data of Vestas Wind Systems Company [13] for wind turbine maintenance training. The trainer specifies intended interactions by demonstration immersed in the VR environment. The interactions are then abstracted from computational concepts such as meshes or coordinates and stored as virtual choreographies for the training domain's semantic domain. This enables matching trainees' actions, similarly interpreted as virtual choreographies, to be made at the same semantic level. This frees trainers to generate - and trainees to experience - varied and rich experiential and situated learning scenarios for manufacturing and other industrial applications, including training certification based on analytic analysis and review of the trainees' actions, independent from software developers.

2 RELATED WORK

This work builds upon two main research areas: immersive learning in virtual reality, and virtual choreographies

2.1 Immersive VR Training

The use of immersive learning environments occurs in six major clusters: for complementing, simulating, exploring, engaging, experiencing, and accessing [14]. The simulation uses include, besides imitating physical spaces and processes, logistic aspects, such as reducing costs, risks, time, or having more flexibility of equipment and personnel (ibid.).

Pedagogical rationales including promoting experiential and situated learning, ensuring replicability of trainee actions, self-regulated learning [15, 16], and triggering emotions furthering the interest and engagement of learners [17]. For example, Garcia et al. [3] created a virtual reality training system for maintenance of high-voltage overhead power lines, with three operation modes: learning, practice, and evaluation. It has shown to be a cost-effective tool for transferring skills and knowledge to new workers while reducing time and money associated with the training procedure. Miles et al. [4] combined a VR headset and user body tracking to train athletes without travelling to a gym. Im et al. [6] developed a VR mechanic engine for simulating disassembly and assembly procedures, demonstrating "high interest, immersion, satisfaction, and perceived learning effectiveness".

Simulations have also been used to practice aircraft engine assembly team procedures and their coordination [5] or for miners to develop correct and safe behaviors [7].

2.2 Virtual Choreographies

Transforming human actions data into virtual choreographies allows users' behaviours to be matched according to purpose and context, rather than raw time and space coordinates. In this work we use

them to generate interaction models for trainees and track relevant trainee's actions for certification.

Virtual choreographies are sets of actions performed by human or non-human elements of a scene, either prescribed or recorded, including their contextual goals and eventual restrictions [18]. Several contexts need to use virtual systems based on multi-user behaviours incorporating choreographed scenarios where users interact with the environment collaboratively (e.g., mechanical maintenance of F-16 [5], factory assembly simulation [19], disaster training [20]).

In addition to these multi-user online training systems, virtual choreographies are also used in particular scientific areas, including scientific experiments themselves [21]. These representations allow the testing of hypotheses that would otherwise be difficult to verify and validate empirically, significantly improving the predictive capacity of phenomena [22, 23].

The matching of virtual choreographies according to purpose and context requires to be specified in a platform-independent way, considering the knowledge domain's semantics rather than those of the technological platform [18]. Thus independent, choreographies associated with a platform become a concern only for their visual reproduction (ibid.) or for their identification from platform actions. This abstraction from technical deployment potentially enables the application to choreographies of bodies of knowledge from other fields, such as process management [21].

But there are some difficulties in developing such choreographies [18]: rendering expectations may bind choreography aspects to a particular environment; involved subject-matter experts (domain specialists and researchers, choreographers) have terminological differences, which may result in conflicting interpretation of the knowledge domain.

The use of virtual choreographies has pursued Model-Driven Engineering principles ([18], [21]), by automated transformation of their semantic representation into platform-specific representations, either via ontology mapping [18], formal models [21], transformation languages [24], partial order planning [25] or hierarchical state machines [5].

3 IMMERSIVE AUTHORING TOOL IN VR

We developed an immersive authoring tool in VR for trainers to create immersive training courses. The trainee's actions in the virtual environment are demonstrated by the trainer directly, and contextual information is appended automatically. The actions are structured as procedures, as described in the next subsection. The tool was designed as a generic approach for demonstrating training procedures on engineering components and is currently being applied to the training of maintenance procedures on wind turbines.

3.1 Architecture

Figure 1 shows the architecture of the VR Training environment, which can be presented as follows. The information regarding course structure, environments, tools, models, and placements are stored locally, where our tool is executed.

The actions for creating the training procedure, training and even the training carried out within the immersive environment, described

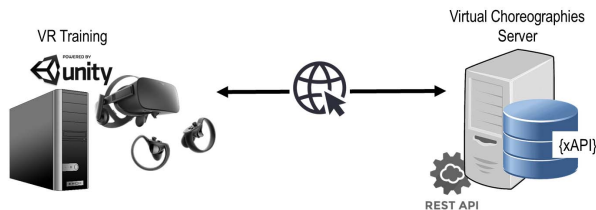


Figure 1: VR Training environment.

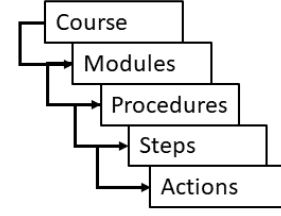


Figure 2: Course structure.

in the following points, are converted into virtual choreographies. When the user is performing a specific procedure, actions are generated within the virtual environment. Each action and the corresponding object are sent, through a restful API, to the virtual choreography server that keeps each one. The server engine seeks to find in the behaviour dictionary (in the present case, it is understood that a behaviour can correspond to a sequence of actions) a sequence of actions that is coincident. Each duly validated behaviour is added to the choreography that is no more than a set of sequential and associated behaviours for a given procedure, that uses xAPI syntax [24].

3.2 Defining the Course Structure

The user interface provides different types of information in the VR environment, adapting the available menu options to the current context (immersive space, content, and training or demonstration phase and status). Common aspects include a messaging system for alerts and immediate feedback of non-visible events; textual input using a virtual keyboard; traditional windowed dialogue metaphor; and side-by-side information on a lateral panel, to support head-rotation viewing (Figure 3).

With this interface, the trainer creates and edits courses and their structure: training modules, their procedures, and the steps of each procedure (Figure 2). Each of these structural elements has a name and description (Figure 3). Each step's actions are not specified nor edited in this dialogue mode but by immersive demonstration. We envision recreating this window-based interaction in a more immersive and natural manner.



Figure 3: Module creation panels.

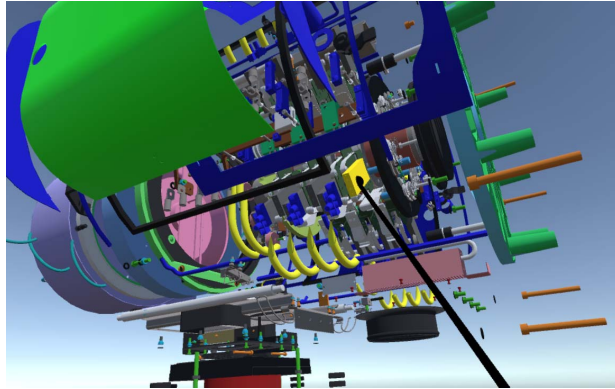


Figure 4: Defining the interactive parts of the model.

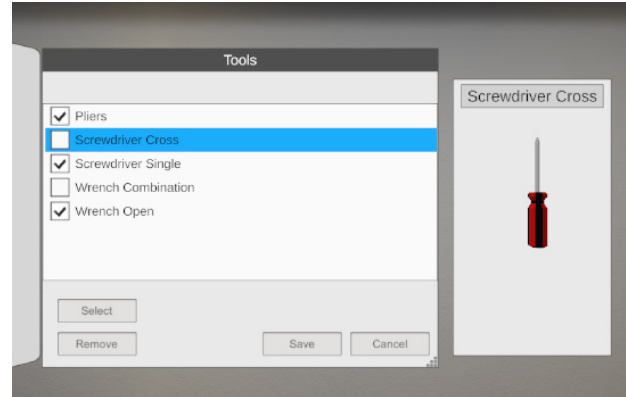


Figure 6: Choosing tools.

3.3 Setting Up the Environment

The tool allows the trainer to set up the training environment, specifying each of the different components, starting with selecting the machinery's 3D model. The trainer selects one of the available models obtained through the import of CAD-based data for accuracy. To support this selection, the trainer is shown a preview of each available model on the virtual world.

With the selected 3D model visible in the immersive space, the trainer can select a section of the model as the focus for the training to limit the trainees' available interaction options. This process is made by refining the model until the desired section remains, i.e., selecting and exploding model components as necessary until the section is exposed and picked (Figure 4). This restricts any interactions for action demonstration by the trainer, or subsequent action execution by the trainee, to that section. The trainer can further restrict interaction within the focus section by identifying a subset of the exposed parts. This identification is made directly within the immersive environment by selecting those model parts. While conducting this operation, the trainer can further explode any subsection of the model, to identify ever-more-specific parts, if necessary.

Another setup aspect of the training environment that the trainer can perform is selecting any desired technical documentation that should be made available to the trainee (Figure 5). This will be provided during the training as a floating documentation pane. Further,

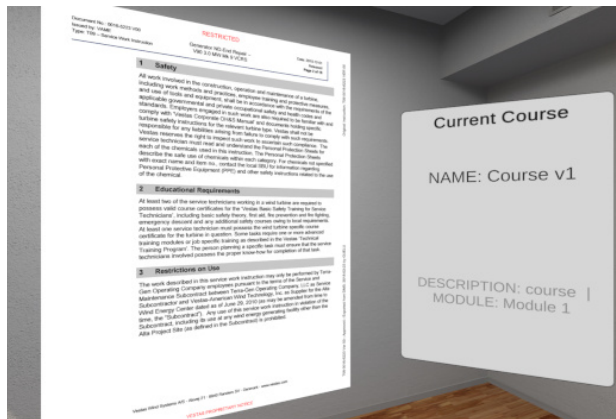


Figure 5: Instructions' document.

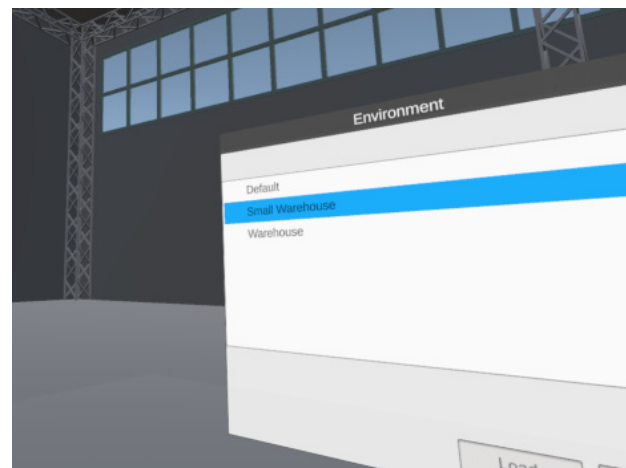


Figure 7: Choosing an environment.

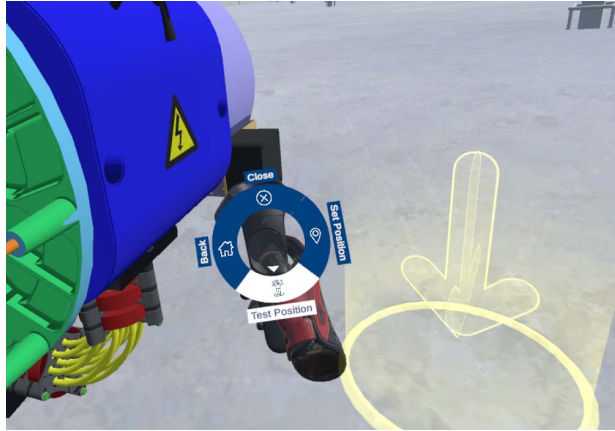


Figure 8: Defining the initial position for the trainee.

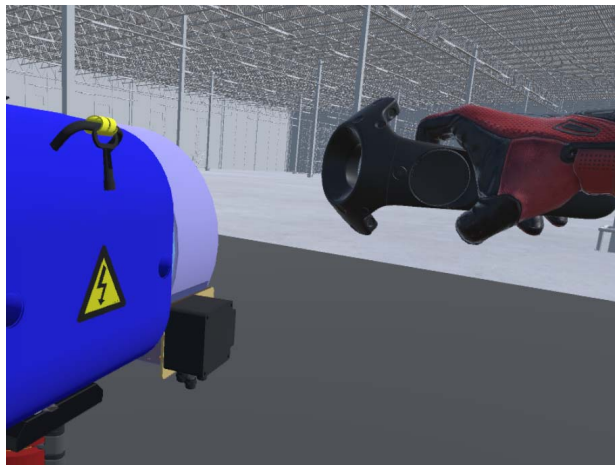


Figure 9: Adjusting the height of the training.

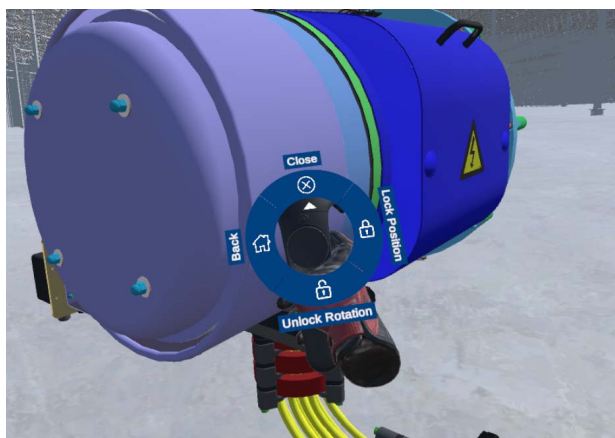


Figure 10: Positioning the target training component model on the environment.

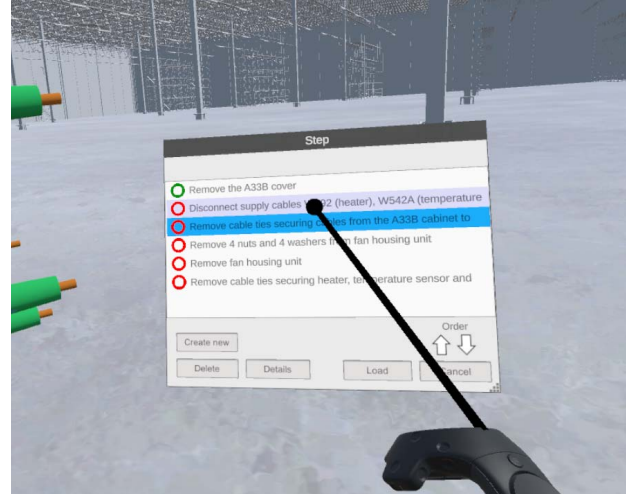


Figure 11: Choosing step to record.

3.4 Procedure Recording

As put forward in section 3.2 (Figure 2), a procedure is composed of named steps, which in turn are composed of actions. The specification and naming of the steps occur while defining course structure, but the recording of the actions is performed by demonstration. To demonstrate the actions of a step, the trainer selects a specific step in the dialogue interface. In it, alongside each step, an icon indicates whether that step has already been recorded or not (Figure 11).

Before the actual recording of actions for a step, the trainer can define a part as a drop zone, enabling it as a dockable destination for other parts. This has two purposes: firstly, it simplifies interaction, by ensuring that placement is triggered easily, thus avoiding awkward attempts to position virtual parts in space with extreme precision; secondly, it minimizes undesirable situations akin to those common in real life when small parts of machinery become lost or obscured among more bulky ones. The tools selected at setup are available in order to be included as part of the recorded actions (Figure 12).

While the trainer performs the desired actions directly on the 3D model, demonstrating them, the system will record them as a virtual

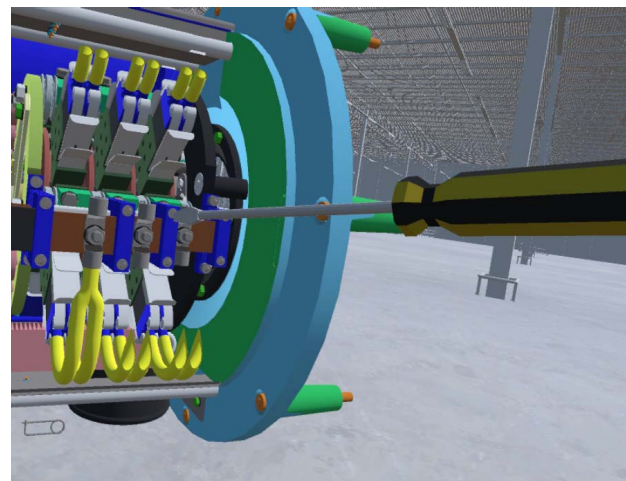


Figure 12: Using a tool while training.



Figure 13: Recording actions by demonstration.

choreography. This choreography of actions becomes associated with that step, and each recorded action is represented by a sphere displayed alongside the hand controller, for visual tracking and feedback of the recording status (Figure 13). The content of the spheres indicates the type of that action and its order number within the step.

The virtual choreography contains the actions at a semantic level associated with the training intent, not specific coordinates, meshes, or the Unity platform's commands. Using the terminology of section 2.2, the virtual choreography is stored in the knowledge domain, not the platform domain. For this, our implementation used a syntax based on xAPI, serialized as JSON, per Lacet et al.'s method [24]. For instance, removing a machine cover is stored as the verb "remove" and the object "cover", rather than the Unity-dependent sequence of cursor hovering a polygon, picking up the polygon, moving it outside the main model, and releasing it while away from an attachment point.

3.5 Training Execution

Once trainer records all actions, of all steps, of all procedures, of all course modules, a trainee can execute that training, by attempting to perform the procedures on the 3D model, following the specified technical documentation. If the documentation was included in the course's setup, it would be available as a floating panel via a menu option (Figure 14). Trainees can also request to watch a replay of the trainer demonstration of a procedure's next action.

All previously-mentioned settings are considered during training execution. For instance, only the parts set up for interaction can be interacted upon; actions requiring specific tools require them; only tools set up to be part of the course will be made available, and so forth.

While executing the training, every action the trainee takes is recorded by the system, also as a virtual choreography at a high semantic level, as explained in the previous section. This choreography can be analysed by the trainer subsequently, either as a full replay, or from a learning analytics perspective: the platform-independence of the choreography, anchored in the knowledge domain enables a wide range of learning analytics, visualization, and learning management approaches.

This choreography-based approach also enables automated matching of the trainee's actions with the trainer's intent, since irrelevant actions can be disregarded: the trainee is provided feedback about the correctly-performed actions.

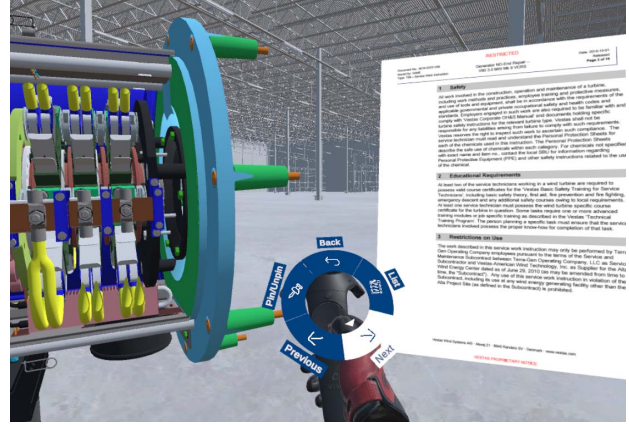


Figure 14: Accessing the instructions while training.

4 CONCLUSION

The use of Virtual Reality in industrial training brings several advantages, being one of them enabling more frequent and safe experiential and situated learning activities, an approach with proven results. The dependence of software development for course creation restricts trainers' ability to develop, maintain, and update immersive virtual courses, thus hampering these technologies' dissemination.

We developed a novel immersive authoring tool for experiential and situated learning in virtual reality-based training. This tool serves as a proof-of-concept to demonstrate the feasibility of immersive authoring of immersive training, enabling trainers to create, maintain, and update immersive courses without software development resources.

The industrial machinery we employed (wind turbines) is particularly detailed, which originated various challenges, such as heavy model loading, and massive numbers of parts to interact with. This prompted various interaction solutions, such as the already mentioned dockable parts specifications and various other interaction adjustments and tweaks. Thus, the system is an exemplar of interaction solutions that can be employed in further systems, enabling subsequent research inquiries and avenues to be explored in the future.

As for our work's next steps, we will run an extensive validation of our tool with the several stakeholders to ascertain our approach's feasibility with end-users

ACKNOWLEDGMENTS

This work is 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 POCI-01-0247-FEDER-038524.

REFERENCES

- [1] Å. Fast-Berglund, T. Fässberg, F. Hellman, A. Davidsson, and J. Stahre, "Relations between complexity, quality and cognitive automation in mixed-model assembly," *Journal of Manufacturing Systems*, vol. 32, no. 3, 2013.
- [2] F. Tao, J. Cheng, Q. Qi, M. Zhang, H. Zhang, and F. Sui, "Digital twin-driven product design, manufacturing and service with big data," *The International Journal of Advanced Manufacturing Technology*, vol. 94, no. 9, pp. 3563–3576, 2018.
- [3] A. A. García, I. G. Bobadilla, G. A. Figueroa, M. P. Ramírez, and J. M. Román, "Virtual reality training system for maintenance and operation

- of high-voltage overhead power lines,” *Virtual Reality*, vol. 20, no. 1, 2016.
- [4] H. C. Miles, S. R. Pop, S. J. Watt, G. P. Lawrence, and N. W. John, “A review of virtual environments for training in ball sports,” *Computers & Graphics*, vol. 36, no. 6, pp. 714–726, 2012.
 - [5] A. Pinheiro, P. Fernandes, A. Maia, G. Cruz, D. Pedrosa, B. Fonseca, H. Paredes, P. Martins, L. Morgado, and J. Rafael, “Development of a mechanical maintenance training simulator in opensimulator for f-16 aircraft engines,” *Entertainment Computing*, vol. 5, no. 4, 2014.
 - [6] T. Im, D. An, O.-Y. Kwon, and S.-Y. Kim, “A virtual reality based engine training system - a prototype development & evaluation,” in *Proceedings of the 9th International Conference on Computer Supported Education - Volume 1: CSEDU*, pp. 262–267, INSTICC, SciTePress, 2017.
 - [7] A. Grabowski and J. Jankowski, “Virtual reality-based pilot training for underground coal miners,” *Safety science*, vol. 72, 2015.
 - [8] “Unity.” <https://unity.com/>, last visited on January 10th 2021.
 - [9] “Unreal engine.” <https://www.unrealengine.com/>, last visited on January 10th 2021.
 - [10] H. Gaspar, L. Morgado, H. Mamede, T. Oliveira, B. Manjón, and C. Gütl, “Research priorities in immersive learning technology: the perspectives of the ilrn community,” *Virtual Reality*, 2019.
 - [11] “Eon reality.” <https://eonreality.com/>, last visited on January 10th 2021.
 - [12] “Xvr simulation.” <https://www.xvrsim.com/>, last visited on January 10th 2021.
 - [13] “Vestas wind systems a/s.” <https://www.vestas.com/>, last visited on January 11th 2021.
 - [14] D. Beck, L. Morgado, and P. O’Shea, “Finding the gaps about uses of immersive learning environments: a survey of surveys,” *Journal of Universal Computer Science*, vol. 26, pp. 1043–1073, 2020.
 - [15] H. De Visser, M. O. Watson, O. Salvado, and J. D. Passenger, “Progress in virtual reality simulators for surgical training and certification,” *Medical journal of Australia*, vol. 194, 2011.
 - [16] A. Mossel, A. Peer, J. Göllner, and H. Kaufmann, “Requirements analysis on a virtual reality training system for cbrr crisis preparedness,” in *Proceedings of the 59th Annual Meeting of the ISSS-2015 Berlin, Germany*, vol. 1, 2015.
 - [17] L. M. Alves Fernandes, G. Cruz Matos, D. Azevedo, R. Rodrigues Nunes, H. Paredes, L. Morgado, L. F. Barbosa, P. Martins, B. Fonseca, P. Cristóvão, *et al.*, “Exploring educational immersive videogames: an empirical study with a 3d multimodal interaction prototype,” *Behaviour & Information Technology*, vol. 35, no. 11, 2016.
 - [18] E. Silva, N. Silva, and L. Morgado, “Model-driven generation of multi-user and multi-domain choreographies for staging in multiple virtual world platforms,” in *International Conference on Model and Data Engineering*, pp. 77–91, Springer, 2014.
 - [19] A. Nakasone, H. Prendinger, M. Miska, M. Lindner, R. Horiguchi, J. Ibarra, K. Gajananan, R. Mendes, M. Madruga, and M. Kuwahara, “OpenEnergySim: A 3D Internet Based Experimental Framework for Integrating Traffic Simulation and Multi-User Immersive Driving,” 2011.
 - [20] J. Ribeiro and J. Almeida, “Using serious games to train evacuation behaviour,” *2012 7th Iberian Conference on Information Systems and Technologies (CISTI)*, pp. 1–6, 2012.
 - [21] A. Weiss, V. Andrikopoulos, M. Hahn, and D. Karastoyanova, “Model-as-you-go for choreographies: rewinding and repeating scientific choreographies,” *IEEE Transactions on Services Computing*, 2017.
 - [22] R. Young and M. Riedl, “An architecture for integrating plan-based behavior generation with interactive game environments,” *Journal of Game Development*, 2004.
 - [23] S. Deterding, M. Sicart, L. Nacke, K. O’Hara, and D. Dixon, “Gamification. using game-design elements in non-gaming contexts,” in *Proceedings of the 2011 annual conference extended abstracts on Human factors in computing systems - CHI EA ’11*, 2011.
 - [24] D. Lacet, F. Penicheiro, L. Morgado, and A. Coelho, “Preserving story choreographies across multiple platforms: An approach to platform-independent reuse of characters’ behaviors for games, simulations, animations and interactive videos,” in *Proceedings of the 9th International Conference on Digital and Interactive Arts*, 2019.
 - [25] A. Vilela, A. Marques, H. Costa, J. Rafael, R. F. F. Prada, and L. Morgado, “Aplicação de avatares autónomos para desempenhar o papel de membros na execução de trabalhos em equipa,” 2012.