



Developing a scenario-based video game generation framework for computer and virtual reality environments: a comparative usability study

Elif Surer¹ · Mustafa Erkayaoglu² · Zeynep Nur Öztürk³ · Furkan Yücel¹ · Emin Alp Bıyık¹ · Burak Altan¹ · Büşra Şenderin¹ · Zeliha Oğuz⁴ · Servet Güler² · H. Şebnem Düzgün⁵

Received: 27 May 2020 / Accepted: 9 October 2020 / Published online: 31 October 2020
© Springer Nature Switzerland AG 2020

Abstract

Serious games—games that have additional purposes rather than only entertainment—aim to educate people, solve, and plan several real-life tasks and circumstances in an interactive, efficient, and user-friendly way. Emergency training and planning provide structured curricula, rule-based action items, and interdisciplinary collaborative entities to imitate and teach real-life tasks. This rule-based structure enables the curricula to be transferred into other systematic learning platforms. Although emergency training includes these highly structured and repetitive action responses, a general framework to map the training scenarios' actions, roles, and collaborative structures to serious games' game mechanics and game dialogues, is still not available. To address this issue, in this study, a scenario-based game generator, which maps domain-oriented tasks to game rules and game mechanics, was developed. Also, two serious games (i.e., Hospital game and BioGarden game) addressing the training mechanisms of Chemical, Biological, Radiological, Nuclear, and Explosives (CBRNe) domain, were developed by both the game developers and the scenario-based game generator for comparative analysis. Finally, the outcomes of these games were mapped to the virtual reality environment to provide a thorough training program. To test the usability, immersion, presence, and technology acceptance aspects of the proposed game generator's outcomes, 15 game developer participants tested a complete set of games and answered the questionnaires of the corresponding phenomenon. The results show that although the game generator has higher CPU time and memory usage, it highly outperforms the game development pipeline performance of the game developers and provides usable and immersive games. Thus, this study provides a promising game generator which bridges the CBRNe practitioners and game developers to transform real-life training scenarios into video games efficiently and quickly.

Keywords Serious games · Video game generator · CBRNe · System usability scale · Technology acceptance model

1 Introduction

Serious gaming [1,2], the umbrella term describing the video and board games having additional goals rather than only entertainment, is widely used in several domains such as

This framework is fully supported by European Network Of CBRN Training Centers (eNOTICE) project funded under EU H2020 (Project ID: 740521).

Electronic supplementary material The online version of this article (<https://doi.org/10.1007/s12193-020-00348-6>) contains supplementary material, which is available to authorized users.

✉ Elif Surer
elifs@metu.edu.tr

Extended author information available on the last page of the article

health [3], defense [4,5], and education [6,7]. The most common focus of serious games has been the games' physiological and spatial effects on the players [4]. Several studies demonstrate that playing video games can improve spatial, cognitive, and motor skills [8,9]. Some studies show that these improvements could affect the players in the long term [10,11]. Game technologies and serious games have been used in different disciplines for the purposes of simulation and training, and these technologies have also been adapted to virtual reality (VR) environments. Milgram et al. [12] introduced the reality-virtuality continuum where the definitions of a real environment, reality, augmented reality, mixed reality, augmented virtuality, and virtual reality are defined through a spectrum between reality and virtuality. In that study, Milgram et al. defined VR as *Computer generated*

artificial simulations, generally recreation of the real environment. Using the game technologies in VR enhanced the research on spatial tasks while also enabling the research on immersion, presence, and usability.

There are some research domains where serious gaming and VR have recently been introduced to the scene, and CBRNe is one of them. CBRNe is an acronym for Chemical, Biological, Radiological, Nuclear, and Explosives, and recent research on this domain focuses on personnel training, emergency planning, and organizing of field, tabletop, simulation, and serious gaming exercises for preparedness [13]. In CBRNe exercises and training, repetitiveness, structure, scenarios, roles of the practitioners are well designed. Although these pieces of training have such well-defined structures, serious gaming and VR have not taken these pre-defined rules into consideration while creating well-defined games.

In this study, we propose a scenario-based game generator where the actors, scenarios, and the locations of the CBRNe training exercises are mapped to state diagrams, and simple games are composed. In the beginning, these “plain” games are generated with simple shapes, in order to test the correctness of the scenario only. Then, these simple shapes are easily replaced with real game assets using the tag information of the game objects. The results of this plain game generator and asset game generator were compared with two serious games in terms of performance, usability, and technology acceptance. Finally, the game generator’s and the game developers’ outcomes were mapped to VR, and additional presence, immersion, and usability tests were performed.

2 Literature survey

The idea behind the serious gaming term can be traced back to the Renaissance era, where the philosophers used the term “serio ludere,” which can be translated as “serious play (theatre)”—i.e., using humor in plays to emphasize serious matters [14]. The first time “Serious Game” was used—as closest to its current meaning—in Clark Abt’s *Serious Games* [15] book. In his book, Abt described serious games as follows:

Games may be played seriously or casually. We are concerned with serious games in the sense that these games have an explicit and carefully thought-out educational purpose and are not intended to be played primarily for amusement. This does not mean that serious games are not, or should not be, entertaining. [15]

Abt’s serious game approach was used in order to solve real-life educational, governmental, and industrial problems [16]. Early serious games were mostly named simulations, and their main focus was to simulate military scenarios—i.e.,

a training and testbed for dangerous-to-live and emergency scenarios [17].

One of the use cases of serious gaming in emergency planning is on firefighter training. In a study by Heldal [18], firefighter training was examined using serious games and tools. To do so, qualitative questionnaires and observations on two use cases (i.e., ship evacuation in the Baltic Sea and railway accident with cyanide leakage) were used to analyze the impacts of serious gaming on non-users. Results showed that serious games would be useful in emergency training situations, and in-depth training scenarios and evaluation methods were necessary. In another study, Lukosch et al. [19] performed the steps of the traditional design process with the contributions of the end-users. The primary purpose of the study was to check if the simulations could be used to train situational awareness skills, and the end-user participation simulation demonstrated the positive impact of using simulations. However, the study did not have a game-scenario-based approach, and future research would focus on this aspect while creating virtual agents.

The use of VR simulations was also a common topic in the literature. Ingrassia et al. [20] focused on testing and comparing performances of 56 medical students during mass casualty triage in the real-world and VR. The results showed that VR and live simulation were both useful in improving the accomplishments of the medical students. Ragazzoni et al. [21] also focused on VR training’s medical aspect, where the objective was to increase staff safety in life-or-death risks. Hybrid simulation for infection control and Ebola treatment was also successfully performed virtually, and the results demonstrated that the awareness of the health personnel increased.

While developing serious games for training purposes, scenario generation becomes an important task. In the literature, there have been several scenario generation studies, which are mainly based on natural language processing and ontology development. The main focuses of these studies have been the automatization and personalization of the scenarios, where smart object annotations and semantic labeling are widely used, that create varied and adaptive scenarios to the needs of the trainees [22]. Niehaus and Riedl [23] also worked on the automated scenario adaptation that was targeted to adjust available scenarios to the necessities of the trainees. Storyteller [24] has been a recent commercial tool that creates dialogue, stories, and character interactions. Another recent use of scenario generation has been on product design, where complex and dynamic user behaviors were simulated with 3D game engines [25]. Thus, although there have been several attempts in automated scenario generation, an end-to-end scenario-based game generator tool is not available. Such a generator will be especially useful in repetitive, structured, and rule-based training materials, such as CBRNe training.

Serious gaming in CBRNe has been a recent topic, and there are some misinterpretations on the definitions and core concepts, such as the misuse of the words ‘game’ or ‘simulation.’ To overcome these misinterpretations, a pre-development survey was developed [26] to be used before implementing the serious games of the European Network Of CBRN Training CEnters (eNOTICE) project [27]. In the pre-development survey, 24 questions were asked to the practitioners and experts of CBRNe under the following subgroups: (1) Participant’s video gaming background, (2) Participant’s knowledge of serious games, and (3) Participant’s expectations on eNOTICE serious games. Results from 14 CBRNe professionals showed that the majority of the participants were highly positive on using serious games in CBRNe and provided open concepts, suggestions, and guidelines to develop serious games for CBRNe domain [28].

In this study, a game generator that maps linear real-life scenarios to serious games with training objectives is generated as the scenario-based game generator. The objectives of this study are as follows: (1) Enabling the practitioners, game designers, and game developers to communicate effectively while developing scenarios for training purposes, (2) Creating an end-to-end automated tool that creates mini-games in a fast and efficient way so that the training institutions which do not have experts on game design can effectively produce serious games for training purposes, (3) Comparing the performance outcomes of the proposed tool with the games developed by the game developers, (4) Evaluating the resulting mini-games of the scenario-based game generator in terms of usability and technology acceptance, and (5) Exporting the generated mini-games to VR environments to make a comparative analysis with the PC-based version, focusing on presence and immersion.

The crucial part of this study is the CBRNe domain, where the roles, tasks, and goals of the actors are clearly defined. The fundamental objectives of the serious games are as follows: (1) Providing a new environment for further training, (2) Building synergy, and (3) Adopting distinctive territories and different concepts to the CBRNe community. Two games are implemented with both this scenario-based game generator and also by game developers, which are real scenarios of eNOTICE project’s joint activities in Nimes (France) and Brussels (Belgium), which were physically demonstrated in 2018 by the CBRNe practitioners. The performance analysis of the scenario-based game generator was performed based on their CPU usage, rendering, memory usage, and game development pipeline on the generator-based and developer-based games.

Evaluating developed tools, games, and related environments is necessary to check if the games realized their primary objectives. Among several evaluation methods, the main aspects that are evaluated in the literature are Presence, Immersion, Technology Acceptance, and Usability. Presence

can be defined as a person’s experience of being in a place while he or she is in another place. This definition was redefined for virtual environments as a person’s experience of being in a virtual, computer-generated world while he or she is in the real world [29]. To evaluate presence in our VR-based tools and games, we used Witmer’s and Singer’s work [30] on the presence questionnaire for virtual environments. They stated that two psychological states play an important role in experiencing presence—*involvement* and *immersion*. *Involvement* refers to a psychological state as a result of focusing one’s energy on a particular activity while the *immersion* is defined as one’s perception of being included during an activity. These psychological states determine the user’s involvement degree to the virtual environment experience, i.e., when the involvement is higher, the degree of immersion and presence increases. Another measure, Technology Acceptance Model (TAM), was introduced by Davis [31], and TAM became a widely accepted model for studying user’s acceptance of the technology. TAM focuses on perceived usefulness and perceived ease-of-use and was finalized by Vanketash and Davis [32]. Evaluating usability is another important aspect of virtual environment research, and Brooke’s [33] proposed the questionnaire method, System Usability Scale (SUS), which has been used widely to evaluate usability on websites, applications, and virtual environments. Two phases of the software development were tested by 15 game developer participants in terms of usability, technology acceptance, immersion, and presence. The quantitative comparative analyses on the game generator, game developer-based games, and their virtual environment versions were analyzed thoroughly, followed by qualitative comments and suggestions from the users. The flowchart of this study can be seen in Fig. 1.

3 Materials and methods

In this section, the details of the scenario-based game generator (Figs. 2, 3, 4, 5), two serious games that were developed by both the game designers and the game generator, and their evaluation are explained in detail. The developed serious games are BioGarden and Hospital game scenarios based on the joint activity as part of eNOTICE organized in Belgium and France in 2018. The games are developed as PC and VR versions by graduate students of the Middle East Technical University (METU) Multimedia Informatics program. Although the main objective is focused on research purposes, it was also intended to use these games as training tools for CBRNe professionals. Once the game development was completed, the scenario-based game generator was utilized to re-create the PC and VR versions. The basis of the evaluation is a comparison of these games using user-experience questionnaires and comparative performance outcomes. This

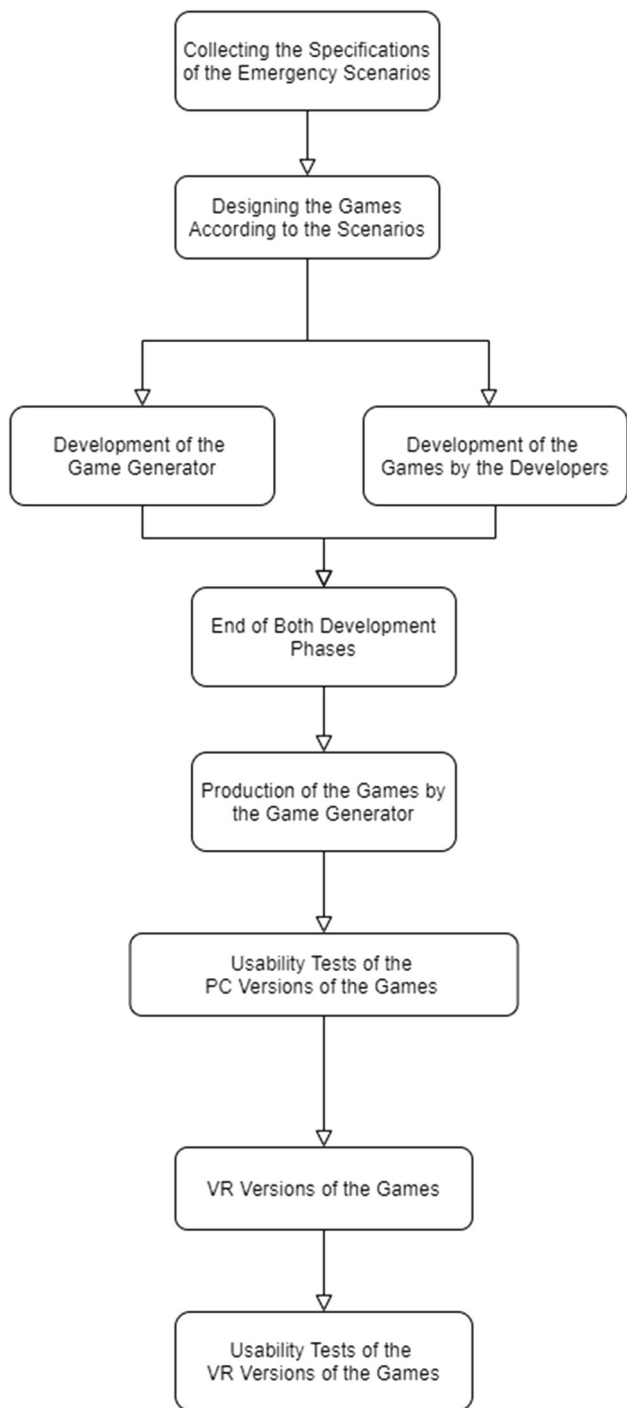


Fig. 1 Flowchart of the study

study was initialized during the 15th Summer Workshop on Multimodal Interfaces (eNTERFACE’19), and the preliminary results of the initial prototype were presented in the eNTERFACE’19 report [34].

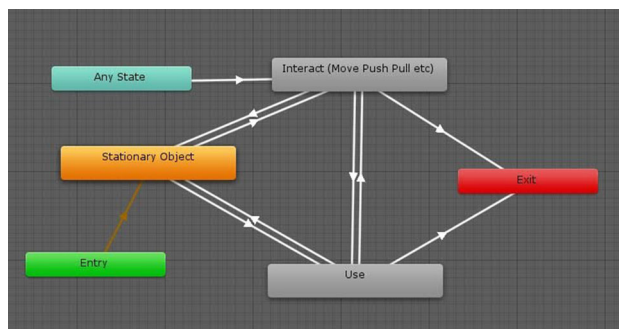


Fig. 2 Initial interaction mechanism including Entry and Exit States, a Stationary Object interacting (i.e., Move, Push, Pull, etc.) and using stationary objects

3.1 Scenario and task definitions

The theme of the scenario, active players, location, and interaction mechanisms was collected in advance from the practitioners via a brief questionnaire (See Appendix) before the eNOTICE joint activities in 2018. The workflow and state diagrams were used to create a detailed scenario where different roles of users and entities interact with each other. The game development process started with mapping the two different CBRNe scenarios that were based on real practices to the workflow and state diagram structures. One of the scenarios is based on a version of the BioGarden exercise (i.e., linear version of the scenario in which the players do not change the flow of the events), which was held in June 2018 in Belgium as part of eNOTICE joint activities. The other scenario is based on the Nimes exercise, which was held in France in January 2018 again as part of the same joint activities.

The scenarios of the exercises were designed by the contribution of various institutions, such as fire departments, research centers, and hospitals. The scenarios already exist, and they were physically performed by the practitioners during the eNOTICE project’s joint activities in Brussels and Nimes in 2018. The scenarios can take place in already existing environments or the environments can be designed specifically for the scenarios. For example, in the BioGarden scenario, a clandestine lab is designed specifically for the game as an artificial location. In the Hospital scenario, places such as airports and hospitals are designed as non-existing environments. Thus, breaking the scenarios down into actions and events was a crucial step so that the game mechanics, reward mechanisms, and scoring could be systematized. Also, different roles in the scenarios were assigned to different player types so that the active role of the player and the role of non-player characters (NPCs) were clarified.

Before starting to implement the Hospital game, a detailed survey, which was briefly mentioned above [28], was conducted on 14 professionals and researchers from the CBRNe

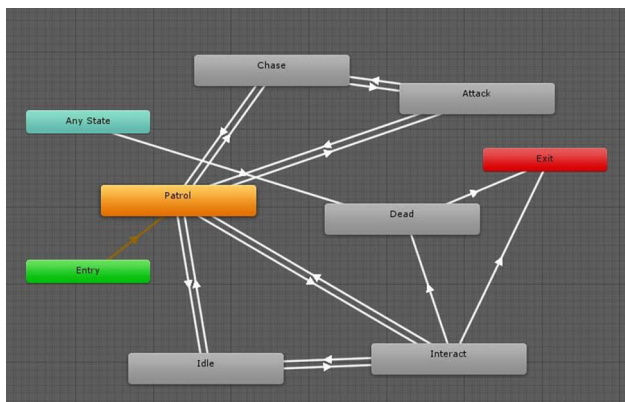


Fig. 3 In an attack scenario, chasing, attacking and interacting use cases are modeled

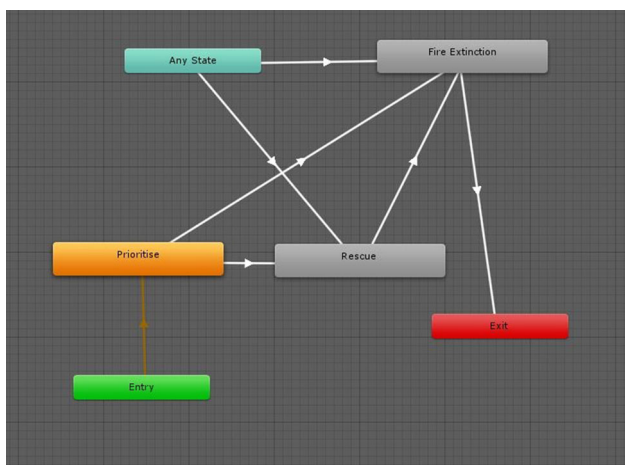


Fig. 4 Firefighting scenario includes prioritizing the steps, extinguishing the fire and rescuing the affected people

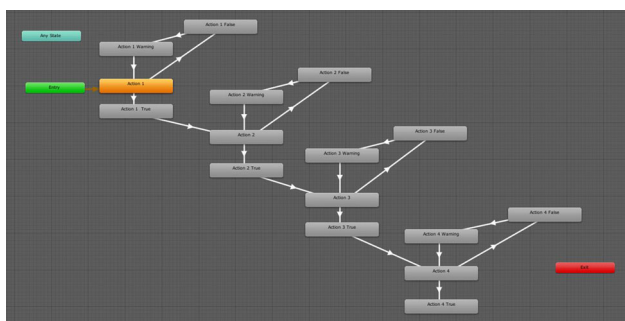


Fig. 5 A generic scenario including various different states

field working in several European Union projects—7 of them being game players. The scope and the purposes of the study were as follows: (1) Learning the user’s gamer profile, (2) Understanding the user’s perspective on serious games, (3) Retrieving the expectations of the user, (4) Clarifying the differences between the video/serious games and simulations, and (5) Asking for suggestions. The initial results of the survey showed that the expectations were highly positive, and

the community was willing to adopt new technologies to their daily tasks. The gamer profiles of the participants involved playing strategy games and multiplayer games with an aim to learn new skills and relieve stress. A tutorial mode was added to the initial game prototype after the initial results were analyzed in detail. This was a preliminary study on the perceptions of CBRNe practitioners. After the survey results demonstrated that there were positive expectations regarding using serious games in CBRNe, the current research was initialized.

3.2 Scenario-based game generator

The game generator is based on a finite state machine concept used to achieve the targeted generalization. The finite state machine is a concept for designing sequential logic in computer science. In this way, it becomes clear which states can be passed from the current state. In this study, these finite state machines were created using Unity 3D’s Animation Controller State Machine feature. Although the feature is intended for animations, it is possible to use it for different requirements in case a state machine model is required. For example, Fig. 3 shows a state machine representation of an attack scenario that includes common situations, such as chasing or attacking. The rectangular shape indicates the state, and arrows in between them indicate the transitions with aspects. It enables control of the sequence of any character in the same way in an attacking scenario.

The scenario-based game generator is developed in Unity Real-Time Development Platform’s Animator Controller tool and can be defined by four different components: (1) Main Code, (2) Control Code, (3) Transition Code, and (4) User Interface. Main Code is where the state definitions and structures—the definitions of the final consequences of the actions—are initialized. The actions and consequences in this part are based on task definitions of the scenarios that were created previously. Control Code works as a mechanism to form and map action methods and their related states. This part of the code checks the current state and condition if there is an action that creates any transition. Then, Transition Code is the link where the game generator works with Unity’s Animator Controller. Transition Code maintains a connection between previous codes and the Animator Controller by triggering the animations. Finally, additional user interface mechanisms such as feedback, scores, and health points are added and grouped under the User Interface component (Fig. 6). Due to the needs of the VR environment, the user interface of the virtual reality versions of the games has some differences in terms of locations and size of the elements, and camera location.

While developing the scenario-based game generator tool, the following steps are executed: (1) Creating an environment, (2) Defining the state diagrams, (3) Creating anima-

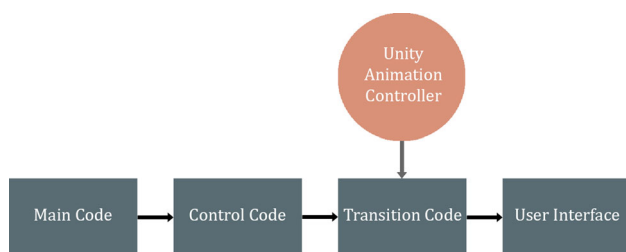


Fig. 6 Structure of the scenario-based game generator

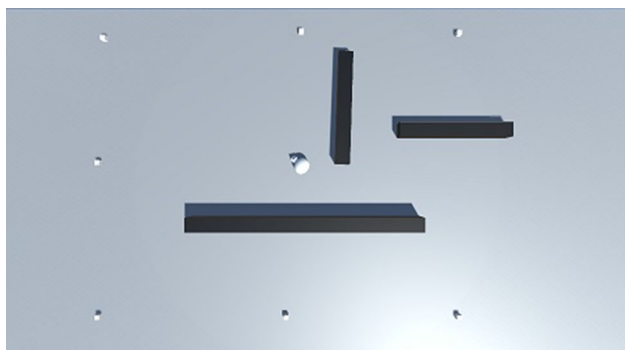


Fig. 7 Initial tests of the scenario-based game generator were performed on simple game objects

tions, (4) Adding basic artificial intelligence (AI) to states, (5) Resolving player and NPC interactions and (6) Adding basic AI for interactions.

In this scenario-based game generator, only linear scenarios, where the decision making of players do not modify the outcomes of the actions, are implemented. The scenario-based game generator is used to generate the duplicates of Hospital and BioGarden games. First, the game scenarios are tested in prototypes by using simple 3D game objects such as cubes, spheres, and capsules (Figs. 7 and 8). Then, after checking that the scenario works correctly, initial 3D game objects are automatically replaced with game assets using tag information of the assets.

3.3 Development environment

Unity3D has been used in game development by both the game generator and game developers. Unity3D is a game engine that offers 2D and 3D game development environments for different platforms. It also has VR and augmented reality support with plug-ins and SDK support for various VR glasses and headsets. The SteamVR SDK developed by Valve [35] is one of the supported SDKs, and it enables playing and developing VR games by using the headsets such as Oculus Rift and HTC Vive. In Phase II of the study, the games were switched to VR versions using the SteamVR SDK, and the HTC Vive headset was used during the usability evaluation.

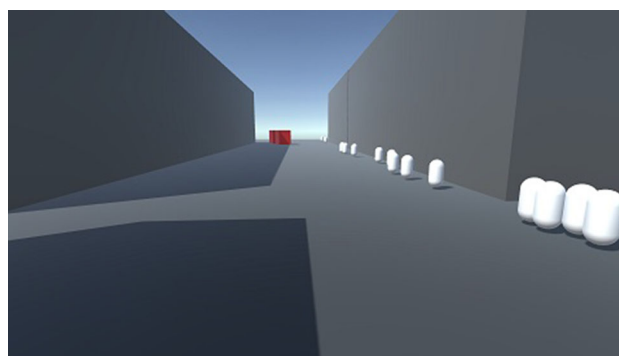


Fig. 8 Initial tests of the scenario-based game generator were performed on simple game objects such as cubes and capsules

3.4 Design patterns

In both games, players choose a role at the beginning of the game. The tasks that they have to perform according to the selected role and the interaction with the game world are different. For example, if the player is a nurse in the Hospital game, the doctor and other health personnel act as NPCs following the requirements of the scenario. Likewise, if the player chooses to be a doctor, other characters in the scenario act as NPCs. Thus, some characters can be both a playable character and an NPC, which depends on the player's choice of role. The developers used the State Design Pattern to manage this interaction network in the game world.

Design Patterns [36] are general solutions used for common problems in software engineering. For instance, if only one instance of a class is required and used throughout the application, the Singleton Pattern ensures this. Even if the content of the application changes, the Singleton Pattern can be used for the same requirement. Hence, these solutions can be considered as templates that can be customized by the requirements of the project. State Design Pattern provides an answer to such a problem: In the games, when the player moves to the next step in the scenario, the NPCs' behaviors change accordingly. For example, the player, as a doctor, should perform the diagnosis of the patient. After that, the nurse returns to the patient to make the diagnostic requirements. According to the decision of the doctor, the nurse acting as an NPC completes the patient's hospitalization procedures by interacting with the patient and secretary, which are both NPCs, too. However, if the player as a doctor decides that the patient is okay, the nurse just sends the patient back. If the doctor determines that the patient is fine, the nurse just sends them home. Therefore, the actions of NPCs vary depending on the activities of the player. When the games were developed by the game developers without the generator, the flow of interaction was needed to be managed effectively, so the State Design Pattern was used. In software design, if there are objects with different states, and

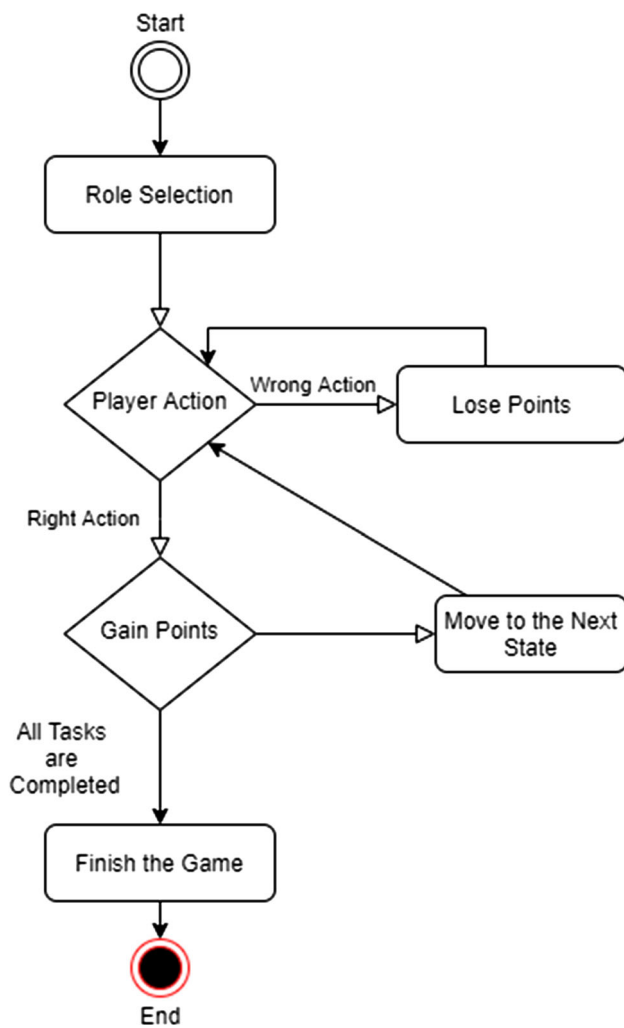


Fig. 9 The general flow of the game loop

these objects behave differently depending on the state, these behavior transitions can be resolved using State Design Pattern. In this study, the states are the actions of the player, and the behaviors are what NPCs should do in the current state. Thanks to the pattern, when the player acts, NPCs automatically perform their actions. The use of the pattern in the game design has saved both from the workload and time loss, requiring coding the playable and NPC versions of each role separately (Fig. 9).

This design pattern also provides a solution to manage the interaction network in the games. The objects are characters, and the state of a character is defined based on whether the character is an NPC or not. If it is not an NPC, the tasks are expected to be performed by the player. However, an NPC automatically performs tasks in accordance with the player's progress in the scenario. The use of this design pattern in the game development enabled us to decrease the amount of workload necessary to design the playable and NPC versions of each role separately.

3.5 Hospital game

The Hospital Game is based on the Nimes scenario performed during the eNOTICE joint activity in January 2018. The main purpose of this scenario is to train medical staff for CBRNE circumstances. In this game, players can get assigned different roles like doctors and nurses to learn how to take security measures, such as using masks and gloves, blocking the entrance of the hospital, and applying decontamination procedures. In the game, the player, as a nurse, meets the patients and asks questions about their symptoms and complaints. The nurse makes the decision about the room in which the patient will be located with the guidance of the doctor. The player listens to the symptoms and complaints of the patients via the nurse NPC while playing as a doctor. After checking the resources of the hospital, such as rooms and beds, with the help of the secretary of the hospital, the doctor decides whether the patient will be sent to a room. If the player chooses the right actions, she/he gets points. The game is based on a linear scenario where the player needs to follow strict hospital regulations (Figs. 10 and 11). The player gets informed about the necessary security measures at different levels, such as using gloves, using masks, and blocking the entrance of the hospital, and applying decontamination procedures. Players can choose between different roles of a doctor, nurse, and secretary. The game is based on a linear scenario, and when the player makes the wrong choices, they lose game points. This game is developed for the VR platform as well in order to make it more realistic, engaging, and an immersive simulation to help the users learn crucial information in a near-realistic environment. In both versions of the game, Quadart's Hospital Lowpoly pack [37], which provides several realistic modular assets, was used.

3.6 BioGarden game

BioGarden game (Figs. 12 and 13) is based on the eNOTICE joint activity, which was organized in Belgium in June 2018. Although it had a nonlinear scenario, only linear parts of the scenario were implemented so that a comparison with the scenario-based game generator would be possible. In the scenario, there are different laboratories with different structures and responsibilities. The role-playing part is composed of decontamination, role assignment, and evaluation.

There are two-player options in the game—Local Investigation Team and Decontamination Team. The Local Investigation Team is responsible for detecting samples, collecting samples, sending samples with a drone to the laboratory, and evaluating the sample results. The Decontamination Team is responsible for the decontamination of the teams that are contaminated with the materials in the laboratory and sending the results of the contamination to the laboratory. There are two scenes in the game: The menu scene and the game scene.



Fig. 10 Screenshots from the Hospital game (Game Generator version)



Fig. 11 Screenshots from the Hospital game and a demo of Dialogue menu (Game Developer version)



Fig. 12 The interior design of the Clandestine lab from the BioGarden game (Game Generator version)

The menu scene has options such as; player selection and scenario summary. The user interface of the game has several elements, such as; back button for the main menu, a camera selection button, which toggles the game camera as first-person/third-person, and a panel which shows the current player in the game. The bottom bar has an objective list and the score of the player. A right mouse click opens an action menu (Fig. 13), which has all possible actions for the current team. Whenever the player selects an action, if the action is correct, the character moves towards the objective area, the



Fig. 13 Action menu of the BioGarden game (Game Developer version)

information panel shows the description of the action, and the points of the player increase. If the action is wrong, the information panel indicates that this action is wrong, and the player loses points. Two different camera options are implemented for different platforms since the first-person camera can provide better immersion if the game is played on a VR headset. There are several differences between the two versions of the BioGarden game. The PC version can be played with a mouse, but the VR version can be played with an HTC Vive Controller. In the PC version, the action menu is overlaid to the screen, but due to the restrictions of the VR platform, the action menu is placed in front of the camera. This can cause problems such as an action menu that can get located behind an object and cannot be interacted. Further improvements like; removing the action menu, implementation of a different interaction mechanism such as pressing a button when the object is nearby, first-person view camera for VR which can be controlled by users, touching and interacting with objects using HTC Vive controllers can add new levels of immersion especially for the VR versions.

As in the case of the Hospital game, the BioGarden game was developed by the METU Multimedia Informatics program, and the same game scenario was also used as an input to the scenario-based game generator. In both versions of the games, 3LB Games' Low Poly laboratory pack [38], which provides several realistic models, textures, and diffuse maps, was used. Both versions of the game were prepared in the same version of the Unity Real-Time Development Platform and were presented to the same game developers to collect feedback, as discussed in the Results section.

3.7 Virtual reality and multimodality

The games developed by both generator and developers have been adapted to VR by using SteamVR. SteamVR is a framework that enables both playing and developing VR games with headsets such as HTC Vive. When transferring to VR, only the interfaces and the controller methods were modified.

The main difference is that a mouse and keyboard are used in the PC versions, and controllers provided by the HTC Vive headset are used in the VR-adapted versions.

As technology and computers are continuously improving, possibilities and methods for human-computer interactions expand further [39]. Currently, there are many methods to interact with a computer. The Conductor is a cross-device interaction where users collectively use different devices to complete the desired task [40]. Microsoft Kinect takes input from the users in the form of body movements [41]. HoloLens is an Augmented Reality tool that scans the real environment and creates a virtual space within the real space [42]. It uses body movements and gestures as user interaction methods. HTC Vive has two controllers, one for each hand, and users interact with the computer using the controllers' movements and using the buttons of those controllers. In a study done by Ergun et al. [43], they asked participants to use VR and MR in architectural design with different setups. Results showed that users felt more natural with the setup, including the HTC Vive and two controllers.

In terms of multimodality, the game generator has the potential to become an adaptive generation tool for both PC and VR games. The transition from the graphical user interface to a multimodal interface is possible with the game generator. The multimodal interface combines two or more user inputs, such as manual gesture, gaze, and touch, coordinated with multimedia systems [44]. Even though the research on the traditional graphical user interfaces (GUIs) has solved several user problems, immersive virtual environments still need improvements on multimodal interactions and multimodal interfaces. The proposed scenario-based game generator offers an adaptable interface framework for both GUIs and multimodal user interfaces (MUIs). The computer versions of the games are using the click-and-play system with a graphical user interface. On the other hand, VR versions of the games use similar interface design with different user input systems. In this system, users play the game with HTC Vive's controllers and headset, where the click-and-play system is kept with the addition of body and head movement inputs to the virtual environments. The importance of this flexibility is that the game generator easily allows developers to adapt their multimodal input system. Hospital Game and BioGarden games are adapted to the virtual environments by the developers of the game generator within 2 days.

4 Results and discussion

4.1 Comparative performance outcomes

In this section, the performance outcomes of the scenario-based game generator and the two serious games that were

Table 1 CPU usage of the hospital game generated by the scenario-based video game generator versus hospital game

CPU usage	Hospital (generator) (ms)	Hospital (game)(ms)
CPU	7	4

Table 2 CPU usage of the BioGarden game generated by the scenario-based video game generator versus biogarden game

CPU usage	BioGarden (generator) (ms)	BioGarden (game) (ms)
CPU	10.6	8.5

Table 3 Memory usage of the hospital game generated by the scenario-based video game generator versus hospital game

Memory	Hospital (generator) (GB)	Hospital (game) (GB)
Used total	0.85	0.45
Reserved total	1.08	0.63
System memory usage	1.65	1.38

Table 4 Memory usage of the BioGarden game generated by the scenario-based video game generator versus biogarden game

Memory	BioGarden (generator) (GB)	BioGarden (game) (GB)
Used total	0.32	0.28
Reserved total	0.58	0.49
System memory usage	1.34	1.27

developed by the game developers were compared in terms of CPU usage, rendering time, memory usage, and game development pipeline. All the tests were performed on a laptop having Intel Core i7 9750HQ CPU, 16GB RAM, and NVIDIA GeForce GTX 1660TI graphics.

Tables 1 and 2 present the comparisons on CPU usage (i.e., game generator's output vs. developer-based game), Tables 3, and 4 on memory usage, and finally, Tables 5 and 6 present a comparison using rendering parameters. The rendering profile uses the SetPass Calls, Draw Calls, Total Batches, Triangles, and Vertices as parameters. SetPass parameter is defined as "the number of rendering passes" [45], a Draw Call as a "call to the graphics API to draw objects" [45] and Batch as a "package with data that will be sent to the GPU" on the Unity's Renderer Profiler page [45].

It took three weeks to develop and implement the scenario-based game generator. The most time-consuming part was the state transitions and handling the outcomes of the actions. After the game generator was built, it took three and a half

Table 5 Rendering results of the hospital game generated by the scenario-based video game generator versus hospital game

Rendering	Hospital (generator)	Hospital (game)
SetPass calls	106	136
Draw calls	252	298
Total batches	217	243
Triangles (K)	463.9	504.9
Vertices (K)	319.5	361.1

Table 6 Rendering results of the BioGarden game generated by the scenario-based video game generator versus BioGarden game

Rendering	BioGarden (Generator)	BioGarden (Game)
SetPass calls	1826	2200
Draw calls	2584	3007
Total batches	2584	3007
Triangles (M)	3.1	3.2
Vertices (M)	2.3	2.4

hours to generate the Hospital game and four hours to generate the BioGarden game using the game generator.

The development of the original BioGarden game took 25 days in total: one week for the scenario clarification and role assignment; one week for the text-based decision mechanisms, 9 days to finish the user interface and menus, and 2 days to add the assets to the game.

The original Hospital game was first refactored from the initial prototype, which took one week. Then, it took another two weeks to adapt to the new scenario, merging different game modes, and dialogue generation.

In this study, real-life exercise scenarios of two eNO-TICE joint activities were developed by game developers as well as a scenario-based game generator—developed during this study. All the frameworks used the same scenario-to-game mechanics mapping. All four versions of the games (i.e., developed by the game developer vs. generated by the game generator) were compared in terms of CPU usage, memory usage, rendering, and game development timeline perspectives. Scenario-based game generator’s CPU usage and memory usage were higher when compared with the developer-based games. Due to the tag searching process and the need for more resources to trigger Unity’s Animator Controller, the game generator uses higher resources in terms of memory and CPU. The rendering performance results of both versions of the games were very similar because the working principle of the game generator was not dependent on the visual contents of the games. Although the game generator used higher memory and CPU, the rendering time was similar while the game development timeline efficiency highly outperformed the game developers (i.e., four hours vs. three

weeks). This is a highly promising outcome [46] that will enable further exercise scenarios to be mapped into games in a short period of time. This outcome can benefit the practitioners in two ways: (1) Visualizing the action-state diagrams of the exercise so that they can see the flaws or unassigned roles of their exercises, and (2) Having a rapid game prototype which becomes a fast, interactive testbed and training tool.

4.2 Usability tests on PC versions: phase I

Technology Acceptance Model (TAM) questionnaire covered descriptive questions to determine the level of technology acceptance representing the benefits obtained from the computer environment versions of the games together with profile questions to classify users. The age profile of the participants in the questionnaire is between 20 and 36, with equal distribution for age ranges of 20–26 and 27–36. This enabled the researchers to investigate the relationship between age and technology acceptance with equal weights. Participants of the workshop hold a Bachelor of Science degree in different disciplines related to computer engineering, informatics, computer science, and others with varying experience in game development.

The experience profile regarding game development is dominated by the participants of the workshop on multimodal interfaces with 70%. A similar profile of these participants could also be observed from 55% of the responses about having experience in VR environments. In a typical week, 25% of the participants spend four or more days playing at least 30 minutes of video games, whereas the remaining spend 0–1 days or 2–3 days on a regular basis. The participants responded to a total of 28 questions in the survey classified as “system usability” and “technology acceptance.” System Usability Scale (SUS) level is investigated by the first ten questions, and the remaining 18 questions are used to define the basis of the Technology Acceptance Model (TAM) of the two different games, Hospital and BioGarden (See “Appendix”).

4.2.1 Phase I: hospital game

The participants completed three different questionnaires after completing gaming sessions for the Hospital game as follows:

1. Questionnaire about the Hospital game (HG)
2. Questionnaire about the Hospital game with the plain game generator (HG-P)
3. Questionnaire about the Hospital game with asset game generator (HG-A)

Questions are grouped into positive and negative concepts as even and odd-numbered that are interpreted in different ways for the calculation of the SUS scores. These scores are rated between the values of 0–100, but they should not be

Table 7 Basic statistical evaluation of the answers about SUS

Game	Measurements	Q4	Q2	Q1 Q3
		Q10	Q6 Q8	Q5 Q7 Q9
HG	Mean	1.81	1.71	4.23
	SD	0.22	0.08	0.36
HG-P	Mean	2.13	1.92	3.36
	SD	0.32	0.23	0.13
HG-A	Mean	1.94	1.69	4.00
	SD	0.01	0.35	0.37

Table 8 Two-sample t-test scores of HG and HG-P

Game	Mean	SD	SE Mean
HG	7.64	2.09	0.49
HG-P	6.03	2.06	0.49

interpreted as percentage values. The SUS scores calculated are 81 for HG, 67 for HG-P, and 78 for HG-A. Grade rankings of SUS scores are evaluated based on the scale developed by [47,48] and classified as “OK” for HG-P and “good” for HG and HG-A [49]. It is stated by Brooke [33, 47] that the systems with higher SUS scores are better in terms of usability. This indicates that the respondents considered the Hospital Game with a plain game generator as a comparably lower usable game in the dedicated game session.

Similar to the positive concept related questions, responses to the even-numbered questions were evaluated for the learnability of the system and the usability of the system. The basic statistical evaluation of the questionnaire results, which was grouped based on this point of view, is given in Table 7.

As seen in Table 7, questions related to learnability (Q4–Q10) indicate that the plain game generator had higher scores in the negative concept representing a lower perception by the respondents. Responses to the odd-numbered questions in the positive concept for the Hospital Game were comparably higher than the game with plain and asset generator. This indicated that the respondents found the Hospital Game to have higher usability. This conclusion was also supported by the remaining eight questions divided into two as positive and negative questions for evaluation. The responses to the positive questions were found to be close to the highest scale value of five, and the answers to the negative questions were found similarly to be close to the lowest scale value of one for the SUS model. The questionnaires were also evaluated for the TAM, representing the three different versions of the Hospital Game.

There are various methodologies that can be followed to define the reasons for users to accept and prefer to adapt to new technology. One of the most commonly utilized concepts is the Technology Acceptance Model (TAM), developed by

Davis [32]. It is based on a theory of information systems and explains how people or society accept a technology with a theoretical model. The evaluation with TAM evaluates the theories behind the application and acceptance of new technologies, especially in information technologies (IT) based on the theory of reasoned action [50]. The theory is considered to be developed during studies on attitude in expectation value models [31] that explain perceived usefulness and perceived ease-of-use together with the intention to use and user satisfaction/perceived enjoyment. The second part of the questionnaire is related to the TAM evaluated by a 10-point scale (1—Strongly Disagree, 10—Strongly Agree). The statistical evaluation of the TAM related responses started with a reliability analysis that aims to assess the consistency between the responses to questions with a similar objective on an ordinal scale. According to [51], the Cronbach’s Alpha (α) value for reliability should be greater than 0.7 as a threshold, and $\alpha > 0.8$ indicates that the questionnaire has good reliability. The reliability test for the TAM related questions was calculated as 0.9631, indicating good reliability. Another evaluation of the TAM related part of the questionnaire was conducted by comparing the mean values of the responses.

The groups HG versus HG–P and HG versus HG–A are found to be statistically zero, considering the responses to build the TAM by two-sample t-test (independent t-test). The null hypothesis (H_0) is defined as “the difference of the mean values between HG and HG–P is equal to zero.” In other words, the technology acceptance level of both groups is equal for these games ($H_0: \mu_{HG} - \mu_{HG-P} = 0$). The alternative hypothesis (H_1) is two-tailed as the assumption is that the difference is not equal to zero ($H_1: \mu_{HG} - \mu_{HG-P} \neq 0$). Table 9 summarizes the two-sample t-test scores for the Hospital Game and the game with the plain game generator.

The null hypothesis H_0 is rejected based on the p value (0.026) lower than 0.05, the significance level (cutoff) value set for 95% confidence interval, and the t-two-tailed value (2.032) is lower than t value (2.327). In conclusion, the difference between HG and HG-P was found to be statistically significant, and the hypothesis “ H_1 : There is a statistically significant difference between HG and HG-P regarding technology acceptance level of this game” is accepted. The individual and box plots are represented in Fig. 14.

The potential outliers within the responses can be seen in Fig. 14, and they demonstrate the question that has a negative concept represented by a lower score compared to the remaining positive concept. Therefore, the data were considered to be suitable for performing a t-test. Similarly, the Hospital Game was evaluated together with the game with the asset generator.

The null hypothesis is failed to be rejected based on the statistical data given in Table 11 as the p value (0.600) is greater than the significance value of 0.05, and the t-two-tailed value (2.032) is greater than the t -value (0.530). This

Table 9 Two-sample t-test scores of HG and HG-P

Game	t	t-two tailed	df	p	95% CI Lower	95% CI Upper
HG	2.327	2.032	34	0.026	0.202	3.015
HG-P						

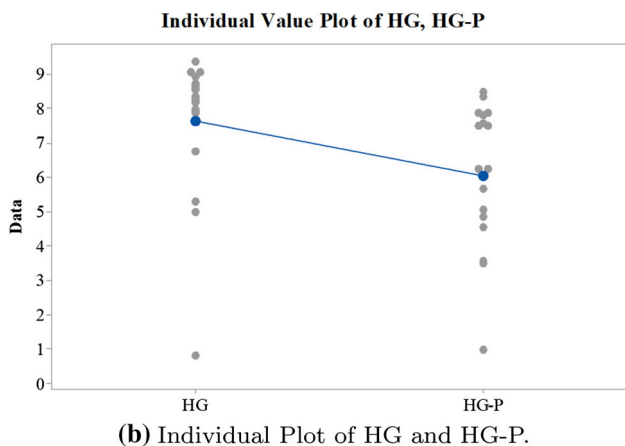
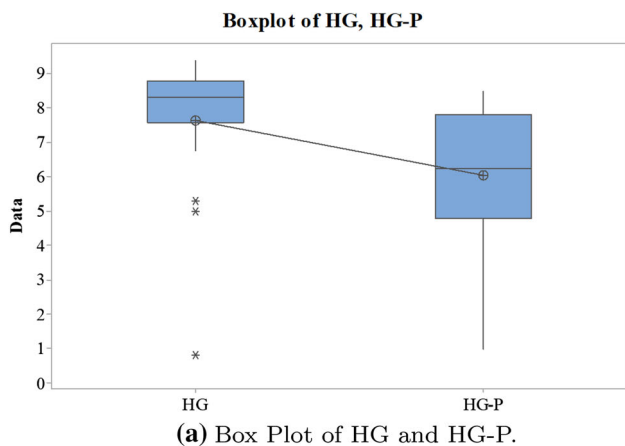


Fig. 14 a Box Plot of HG and HG-P and b Individual Plot of HG and HG-P

can be explained as the difference between HG and HG-A not being statistically significant where “H1: There is a statistically significant difference between HG and HG-A regarding technology acceptance level of this game” is rejected. The individual and box plots for HG and HG-A are given in Fig. 15.

The question with the negative concept is representing the lower value, and it was considered that the responses were suitable for applying the t-test. In conclusion, the respondents of the questionnaire indicated that the plain game generator was more adaptable than the asset game generator version of the Hospital Game. This was also observed in the similarity between the responses related to HG-A and HG, whereas the responses of HG-P are relatively different. The second game related to the BioGarden scenario was also evaluated in detail.

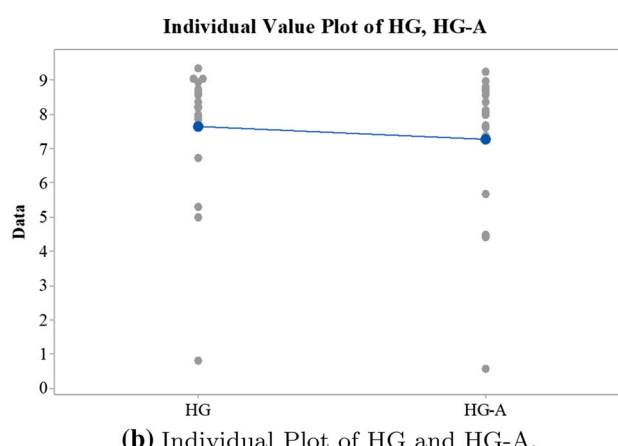
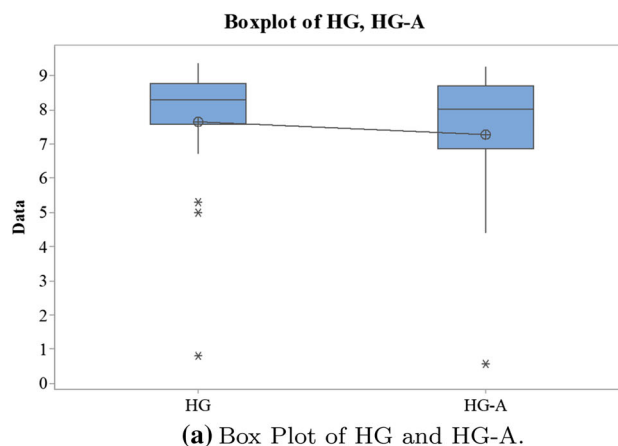


Fig. 15 a Box Plot of HG and HG-A and b Individual Plot of HG and HG-A

Table 10 Two-sample t-test scores of HG and HG-A

Game	Mean	SD	SE Mean
HG	7.64	2.09	0.49
HG-A	7.26	2.19	0.52

4.2.2 Phase I: BioGarden game

The gaming sessions were finalized with the following questionnaires completed by the participants;

1. Questionnaire about the BioGarden game (BG)
2. Questionnaire about the BioGarden game with the plain game generator (BG-P)
3. Questionnaire about the BioGarden game with the asset game generator (BG-A)

Table 11 Two-sample t-test scores of HG and HG-A

Game	t	t-two tailed	df	p	95% CI Lower	95% CI Upper
HG	0.530	2.032	34	0.600	-1.074	1.831
HG-A						

Table 12 Basic statistical evaluation of the answers about SUS

Game	Measurements	Q4 Q10	Q2 Q6 Q8	Q1 Q3 Q5 Q7 Q9
BG	Mean	1.78	1.69	4.08
	SD	0.08	0.11	0.36
BG-P	Mean	1.81	1.54	3.41
	SD	0.11	0.28	0.21
BG-A	Mean	1.69	1.65	4.06
	SD	0.09	0.06	0.30

Table 13 Two-sample t-test scores of BG and BG-P

Game	Mean	SD	SE Mean
BG	7.48	2.06	0.48
BG-P	5.60	2.06	0.48

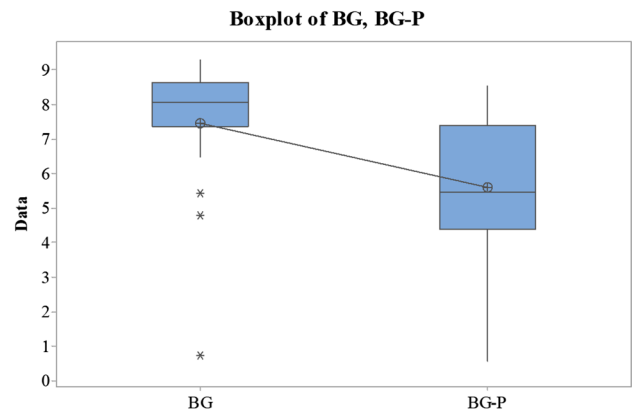
The SUS scores were calculated as 79 for BG, 72 for BG-P, and 80 for BG-A, which indicated that the plain game generator was evaluated to have lower usability compared to the other versions. Table 12 summarizes the basic statistical evaluation of the questionnaire results for the BioGarden game.

The plain game generator was scored higher by the respondents for the questions related to learnability (Q4–Q10). The BioGarden game had higher scores in the questions defined in the positive concept indicating that the usability was higher than the versions with game generators. These results are similar to the responses of the Hospital Game and point out the consistency of the questionnaire results. The questions representing the usability of the system were found to be close to the highest scale value of 5 for the positive concept and similarly close to the lowest scale value of 1 for the negative concept questions. The perspective of the respondents was used as the basis of TAM for the BioGarden game.

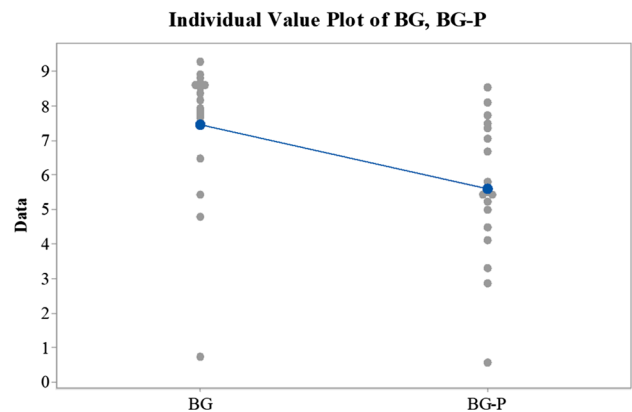
The questionnaire part related to the TAM was first analyzed for the reliability of the results. The Cronbach’s alpha value of TAM responses for the BioGarden game is calculated as 0.9705, representing a good level of accuracy. This

Table 14 Two-sample t-test scores of HG and HG-A

Game	t	t-two tailed	df	p	95% CI Lower	95% CI Upper
BG	2.739	2.032	34	0.010	0.483	3.272
BG-P						



(a) Box Plot of BG and BG-P.



(b) Individual Plot of BG and BG-P.

Fig. 16 a Box Plot of BG and BG-P and b Individual Plot of BG and BG-P

evaluation was followed by a two-sample t-test for BG versus BG-P and BG versus BG-A. The null hypothesis (H0) was defined as the difference of the mean values of the responses related to TAM of BG and BG-P is equal to zero. Test scores of this comparison are summarized in Tables 13 and 14.

The null hypothesis is rejected as the calculated p value (0.010) is lower than the cut-off value of 0.05, and the t-two-tailed value (2.032) is lower than the t value (2.739). The difference between BG and BG-P was found as statistically

Table 15 Two-sample t-test scores of BG and BG-A

Game	Mean	SD	SE Mean
BG	7.48	2.04	0.48
BG-A	7.10	2.19	0.48

significant, and it was concluded that the alternative hypothesis, “There is a statistically significant difference between BG and BG-P regarding technology acceptance level of this game,” is accepted. Figure 16 represents the individual and box plots of the BG and BG-P.

This distribution of the responses supports the concept that the t-test is suitable for the evaluation of these responses. A similar assessment was performed for the TAM related responses of BG and BG-A with the same hypothesis definition and is summarized in Table 15.

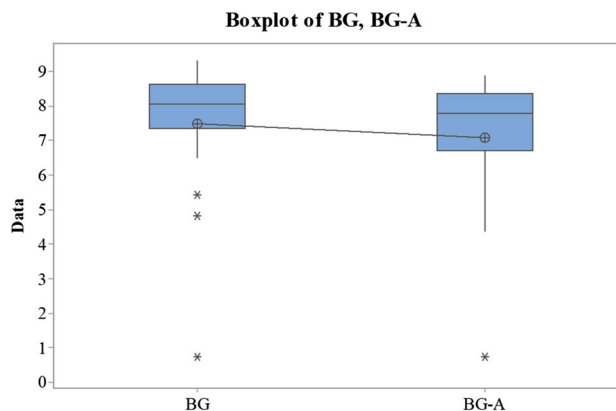
Based on the data given in Table 16, the H0 hypothesis is accepted as the *p* value (0.584) is greater than the cut-off 0.05 for 95% confidence interval and the t-two-tailed value (2.032) is greater than the t value (0.584). As a result, the difference between BG and BG-A was not statistically significant, and the hypothesis that there is a statistically significant difference between BG and BG-A regarding the technology acceptance level is rejected. The individual and box plots of the compared game versions are given in Fig. 17.

As a conclusion of the Phase-I evaluation of the developed games, the plain game generator was found to be more suitable than the asset game generator. This could be assessed from the similarity between the responses to BG and BG-A, together with the differences of the responses given to questions related to BG-P. A similar result could also be observed for the Hospital game and its versions with the plain game generator and the asset game generator.

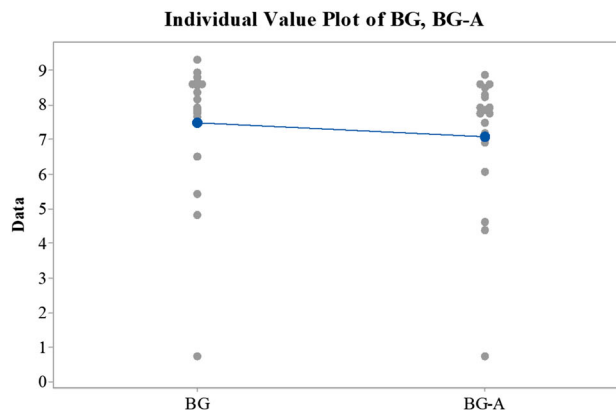
4.3 Usability tests on VR versions: phase II

The gaming sessions were finalized with the questionnaires that aimed to evaluate the experience of the participants in a systematic manner. The Phase-II of the questionnaire was related to the VR games developed for the Hospital and BioGarden scenarios. The participants were asked to complete questionnaires about the VR games and their versions with the asset game generator.

Similar to the Phase-I of the questionnaire, the VR version of the Hospital game (HG-VR) and the version with the asset



(a) Box Plot of BG and BG-A.



(b) Individual Plot of BG and BG-P.

Fig. 17 a Box Plot of BG and BG-A and b Individual Plot of BG and BG-A

Table 17 Basic statistical evaluation of the answers about SUS

Game	Measurements	Q4 Q10	Q2 Q6 Q8	Q1 Q3 Q5 Q7 Q9
HG-VR	Mean	1.75	1.81	3.94
	SD	0.60	0.28	0.28
HG-A-VR	Mean	1.57	1.69	3.91
	SD.	0.39	0.39	0.54

game generator (HG-A-VR) were assessed according to their usability. The SUS scores were calculated as 79 for HG-VR and 80 for HG-A-VR, which are both rated as “good” based on their performance in terms of usability. The basic statistical evaluation of the questionnaire results grouped based on this point of view is given in Table 17.

Table 16 Two-sample t-test scores of BG and BG-A

Game	t	t-two tailed	df	<i>p</i>	95% CI Lower	95% CI Upper
BG	0.553	2.032	34	0.584	-1.011	1.766
BG-A						

Table 18 Two-sample t-test scores of HG-VR and HG-A-VR

Game	Mean	SD	SE Mean
HG-VR	7.36	1.48	0.37
HG-A-VR	7.31	1.63	0.41

The questions Q4–Q10 are related to learnability and indicate a comparably lower score for HG-A-VR, representing a better end result as these questions are negatively oriented. The odd-numbered questions that have a positive concept can be evaluated as HG-VR having a better score. The remaining questions aim to consider the usability of the system where the positive concept questions were responded with a score close to the highest scale value of 5, and the responses to the negative concept questions were close to the lowest scale of 1. The technology acceptance model (TAM) for the HG-VR and HG-A-VR was also evaluated together with the reliability of this part of the questionnaire.

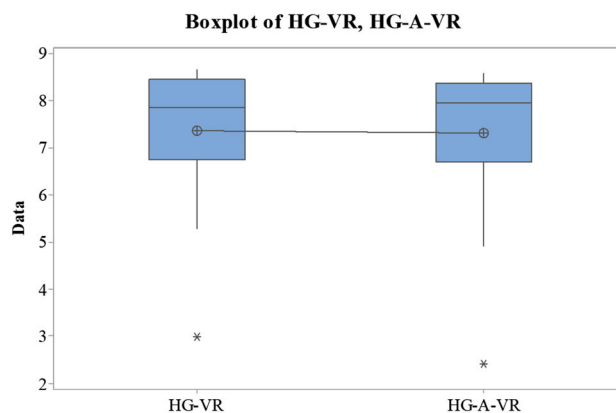
The Cronbach’s alpha value of TAM responses for HG-VR and HG-A-VR is calculated as 0.9954. The difference between the mean values of the responses was evaluated by a two-sample t-test to consider whether it is statistically zero. The null hypothesis (H0) is defined as the condition of “the difference of the mean values between HG-VR and HG-A-VR is equal to zero.” This could also be interpreted as the technology acceptance level of both groups being equal for the Hospital game. In this case, the alternative hypothesis (H1) is two-tailed because it assumes that the difference is not equal to zero. The results are summarized in Tables 18 and 19.

The null hypothesis is failed to be rejected as the *p* value (0.929) is greater than 0.05, and the t-two-tailed value (2.042) is greater than the t value (0.090). As a result of the two-sample t-test, the difference between HG-VR and HG-A-VR was not statistically significant. Therefore, “H1: There is a statistically significant difference between HG-VR and HG-A-VR regarding technology acceptance level of this game” is rejected. The individual and box plots are given in Fig. 18.

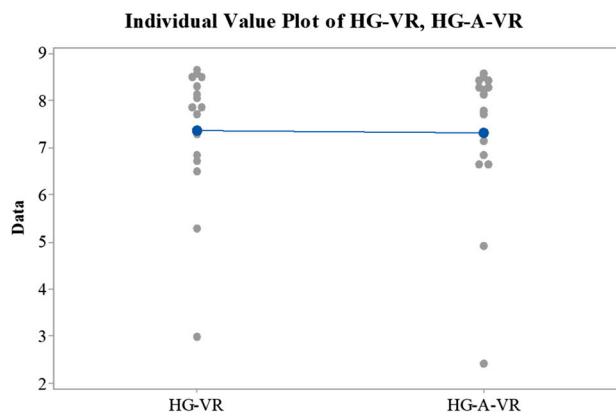
The responses provided for the questionnaire related to HG-VR and HG-A-VR were considered as suitable for the implementation of the two-sample t-test. A similar methodology was also used for the evaluation of the VR version of the BioGarden game (BG-VR) and the version with the asset game generator (BG-A-VR).

Table 19 Two-sample t-test scores of HG-VR and HG-A-VR

Game	t	t-two tailed	df	<i>p</i>	95% CI Lower	95% CI Upper
HG-VR	0.090	2.042	30	0.929	– 1.079	1.178
HG-A-VR						



(a) Box Plot of HG-VR and HG-A-VR.



(b) Individual Plot of HG-VR and HG-A-VR.

Fig. 18 a Box Plot of HG-VR and HG-A-VR and b Individual Plot of HG-VR and HG-A-VR

Table 20 Basic statistical evaluation of the answers about SUS

Game	Measurements	Q4 Q10	Q2 Q6 Q8	Q1 Q3 Q5 Q7 Q9
BG-VR	Mean	1.75	1.74	4.09
BG-VR	SD	0.60	0.08	0.30
BG-A-VR	Mean	1.57	1.71	4.12
BG-A-VR	SD	0.39	0.27	0.42

The SUS scores were calculated as 81 for BG-VR and 81 for BG-A-VR, classified as having a “good” ranking. The learnability of the system was assessed by the remaining questions, and the results are summarized in Table 20.

BG-A-VR was considered as having a better experience compared to BG-VR, whereas the positive concept questions

Table 21 Two-sample t-test scores of HG-VR and HG-A-VR

Game	Mean	SD	SE Mean
BG-VR	7.56	1.66	0.42
BG-A-VR	7.61	1.77	0.43

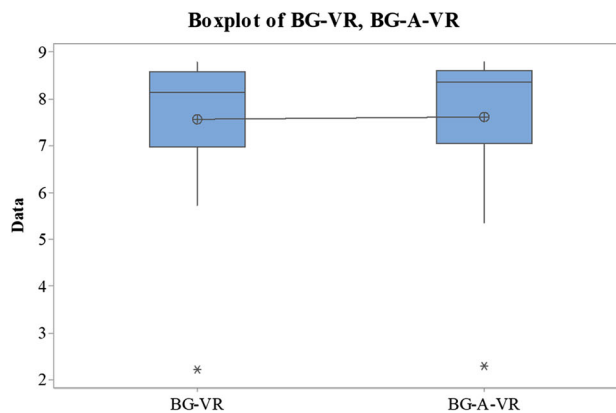
indicated that the participants classified BG-VR as having a higher score. This could be interpreted as the fact that the VR versions of the BioGarden game were highly similar by structure, and the participants could not differentiate the usability of the system distinctly. The Technology Acceptance Model (TAM) was also assessed for BG-VR and BG-A-VR.

The Cronbach’s alpha value of TAM responses is calculated as 0.9977, indicating a high level of reliability. The evaluation continued with an independent t-test to compare the two versions of the BioGarden game based on the null hypothesis (H0), “the difference of the mean values between BG-VR and BG-A-VR is equal to zero.” The results of the t-test are presented in Tables 21 and 22.

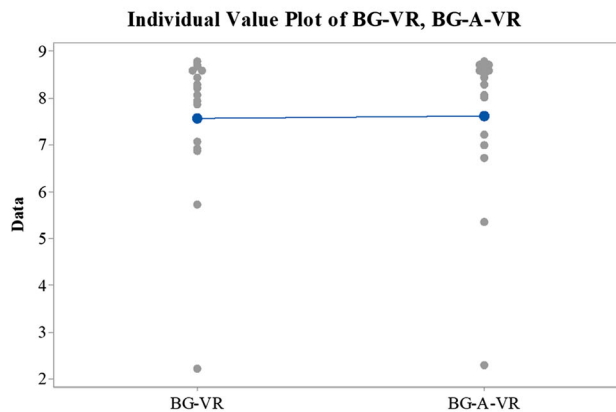
The null hypothesis defined for the t-test failed to be rejected as the *p* value (0.924) is greater than 0.05, and the t-two-tailed value (2.042) is greater than t value (0.097). As a result, the two-sample t-test concluded that the difference between BG-VR and BG-A-VR was not statistically significant. The individual and box plots are given in Fig. 19.

The assessment of the developed games continued with the presence evaluation of the virtual environments. The Immersive Tendencies Questionnaire (ITQ) was originally introduced to measure the capability or tendency of participant involvement. The Presence Questionnaire (PQ) aims to measure the experience of the participants by means of presence in a virtual environment together with the possible contributing factors. The questionnaire has a seven-point scale based on the semantic differential principle [52]. Participants are commonly asked to place a mark in the appropriate box of the scale in accordance with the question content and descriptive labels. Twenty-eight questions were available in the questionnaire from which 14 are related to control factors (CF), eight are related to sensory factors (SF), five are about distraction factors (DF), and 1 question represents the realism factors (RF). The results are summarized in Tables 23 and 24.

The results indicated that both of the developed games have no significant difference by means of presence based on the participant responses. This is interpreted as the outcome



(a) Box Plot of BG-VR and BG-A-VR.



(b) Individual Plot of BG-VR and BG-A-VR.

Fig. 19 a Box Plot of BG-VR and BG-A-VR and b Individual Plot of BG-VR and BG-A-VR

Table 23 Two-sample t-test scores of HG-VR and HG-A-VR

Category	Hospital Score	SD	Hospital Asset Generator Score	SD
PQ Main Factor				
Control	64.64	1.58	62.86	1.72
Sensory	32.71	1.91	33.29	1.95
Distraction	19.86	1.82	21.00	1.58
Realism	4.43	1.60	4.43	1.83
Total Score	121.64		121.57	

of similar VR development methodology for both games and the almost identical level of experience of the participants related to VR. The last stage of the evaluation used ITQ items with 14 questions with a distribution of 7 for focus, 5

Table 22 Two-sample t-test scores of HG-VR and HG-A-VR

Game	t	t-two tailed	df	<i>p</i>	95% CI Lower	95% CI Upper
BG-VR	0.097	2.042	30	0.924	-1.276	1.161
BG-A-VR						

Table 24 Two-sample t-test scores of HG-VR and HG-A-VR

Category PQ Main Factor	BioGarden		BioGarden Asset Generator	
	Score	SD	Score	SD
Control	66.21	1.54	65.71	1.69
Sensory	34.50	1.81	34.86	1.88
Distraction	20.71	1.68	21.07	1.63
Realism	4.93	1.21	5.14	1.10
Total Score	126.36		126.79	

Table 25 Two-sample t-test scores of HG-VR and HG-A-VR

Category ITQ Subscale	Hospital		Hospital Asset Generator	
	Score	SD	Score	SD
Focus	34.21	1.52	34.36	1.53
Involvement	22.71	1.71	22.71	1.71
Game	8.29	2.10	8.29	2.10
Total Score	65.21		65.36	

Table 26 Two-sample t-test scores of HG-VR and HG-A-VR

Category ITQ Subscale	BioGarden		BioGarden Asset Generator	
	Score	SD	Score	SD
Focus	34.29	1.52	34.29	1.52
Involvement	22.71	1.71	22.71	1.71
Game	8.29	2.10	8.29	2.10
Total Score	65.29		65.36	

for involvement, and the remaining 2 for the game itself. The results are summarized in Tables 25 and 26.

Similar to the PQ, the results of the ITQ represented a trend where the scores calculated for the responses of the participants have no significant differences. The tendency to become involved in activities and the tendency to maintain focus on current activities were evaluated identically by the participants. This indicates the similarity between the development stages of the VR versions of the game.

4.4 User comments and suggestions

Participants played the game generator's games, BioGarden game, and Hospital game both in computer and VR environments. Besides the abovementioned questionnaires, open-ended questions regarding the general comments and suggestions were also asked the users. The users could not spot differences between the outcomes of the game generator- and game developer- based games. One user particularly mentioned:

Besides the graphics, the main concept is completely the same. This is the strength of the game generator as it can easily provide a framework to prepare mini-games.

There were mixed feelings about the plain game generator. While some of them found it very difficult to follow with cubes and simple shapes, several users underlined its potential:

I was very impressed while playing plain games. They were almost abstract, and they show the potential of the generator - when I was playing the game generator with assets, I think that this game is just one of the possible solutions.

Some of the users found the assets to be more explanatory:

Assets are beautiful, they definitely help to create a flow state and presence feeling. Assets also help me to understand what is happening in the games and what I should do. They make it easier to play games.

The menu items on the UI mechanism were found to be useful, but the number of the menu items on the screen was suggested to be reduced. Different actions were suggested to be triggered by different menu icons to simplify the UI. Most of the users preferred the VR version more, and they said that it was more fun and immersive.

VR version is better. When the character moves, looking around is good. It can be better when needing UI components in order to interact with the environment is decreased.

I enjoyed the VR version. It feels like you are much more involved in the environment.

5 Conclusion

In this study, a scenario-based video game generator, which targets the scenarios in the CBRNe domain, was developed. This initial version of the game generator used linear scenarios that were based on the joint activities of the eNOTICE project. The effectiveness of the game generator was tested in comparison with two serious games, which were developed by the game developers. Even though the performance of the game generator lacked on the memory usage and CPU usage aspects, its rendering performance was similar to the developer-based games while it highly outperformed the game development pipeline of the game developers. Besides, the computer version and VR version of the games and the game generator outcomes were produced and tested by 15 game developer participants. System usability, technology acceptance, immersion, and presence aspects of the outcomes were analyzed thoroughly. The results show that the proposed game generator produces usable and immersive games in a short amount of time. This is a highly promising result that will enable the practitioners to visualize their scenarios while also generating prototype games rapidly both for computer and VR environments so that the training of CBRNe personnel will become thoroughly enriching and immersive. As future work, the number of participants will be increased, participants from different backgrounds will test the games,

and the proposed game generator framework will be adapted to nonlinear scenarios.

Acknowledgements This study was developed during the 15th Summer Workshop on Multimodal Interfaces (eNTerFACE'19) which was held at Bilkent University, Ankara, Turkey between the dates of July 8 and August 2, 2019. This framework is fully supported by European Network Of CBRN TraIning Centers (eNOTICE) project funded under EU H2020 (Project ID: 740521). The authors would like to thank Dr. Olga Vybornova (Center for Applied Molecular Technologies, Université Catholique de Louvain, Brussels, Belgium) and Prof. Gilles Dusserre (IMT Mines Alès, Alès, Languedoc Roussillon, France) for their help in scenarios of BioGarden and Hospital games, respectively. The authors would also like to thank Oğuzcan Ergün (Multimedia Informatics, Middle East Technical University, Ankara, Turkey) for his help and feedback on VR instrumentation.

Compliance with ethical standards

Conflicts of interest The authors declare that they have no conflict of interest.






References

- Santos R, Magalhaes C, Capretz LF, Neto JC, da Silva FQB, Saher A (2018) Computer games are serious business and so is their quality: particularities of software testing in game development from the perspective of practitioners. arXiv preprint [arXiv:1812.05164](https://arxiv.org/abs/1812.05164)
- Michael DR, Chen SL (2005) Serious games: games that educate, train, and inform. Muska & Lipman/Premier-Trade, Dubai
- Pirovano M, Surer E, Mainetti R, Lanzi PL, Borghese NA (2016) Exergaming and rehabilitation: a methodology for the design of effective and safe therapeutic exergames. *Entertain. Comput.* 14:55–65
- Susi T, Johannesson M, Backlund P (2007) Serious games: an overview. University of Skövde, Skovde
- Crookall D (2010) Serious games, debriefing, and simulation/gaming as a discipline. *Simulat. Gaming* 41(6):898–920
- Tinati R, Luczak-Roesch M, Hall W (2017) An investigation of player motivations in Eyewire, a gamified citizen science project. *Comput Hum Behav* 73:527–540
- Curtis V (2015) Motivation to participate in an online citizen science game: a study of Foldit. *Sci Commun* 37(6):723–746
- Dorval M, Pepin M (1986) Effect of playing a video game on a measure of spatial visualization. *Perceptual Motor Skills* 62(1):159–162
- Nelson R, Strachan I (2009) Action and puzzle video games prime different speed/accuracy tradeoffs. *Perception* 38(11):1678–1687
- Tobler-Ammann BC, Surer E, de Bruin ED, Rabuffetti M, Borghese NA, Mainetti R, Pirovano M, Wittwer L, Knols RH (2017) Exergames encouraging exploration of hemineglected space in stroke patients with visuospatial neglect: a feasibility study. *JMIR Serious Games* 5(3):17
- Tobler-Ammann BC, Surer E, Knols RH, Borghese NA, de Bruin ED (2017) User perspectives on exergames designed to explore the hemineglected space for stroke patients with visuospatial neglect: usability study. *JMIR Serious Games* 5(3):e18
- Milgram P, Takemura H, Utsumi A, Kishino F (1995) Augmented reality: a class of displays on the reality-virtuality continuum, Telemanipulator and Telepresence Technologies
- XVR Simulation | Incident command training tool for safety and security (2019). Retrieved from <https://www.xvrsim.com/en-cn/>
- Djaouti D, Alvarez J, Jessel J, Rampoux O (2011) Origins of Serious Games. In: Ma M, Oikonomou A, Jain LC (eds) *Serious games and edutainment applications*. Springer, London, pp 25–43
- Abt C (1970) *Serious games*. University Press of America, Lanham
- Laamarti F, Eid M, El Saddik A (2014) An overview of serious games. *Int J Comput Games Technol* 2014:1–15
- Michael D, Chen S (2006) *Serious games: games that educate, train, and inform*. Thomson Course Technology PTR, Boston
- Heldal I (2016) *Simulation and serious games in emergency management: experiences from two case studies*. 22nd International Conference on Virtual System & Multimedia (VSMM), pp. 1–9. IEEE
- Lukosch H, van Ruijven T, Verbraeck A (2012) The participatory design of a simulation training game. In: *Proceedings of the winter simulation conference, winter simulation conference*
- Ingrassia PL, Ragazzoni L, Carezzo L, Barra FL, Colombo D, Gugliotta G, Corte FD (2012) Virtual reality and live scenario simulation: options for training medical students in mass casualty incident triage. *Crit Care* 16(1):P479
- Ragazzoni L, Ingrassia PL, Echeverri L, Maccapani F, Berryman L, Burkle FM, Corte FD (2015) Virtual reality simulation training for Ebola deployment. *Disaster Med Public Health Preparedness* 9(5):543–546
- Ferdinandus GR (2012) Automated scenario generation, coupling planning techniques with smart objects, Master's thesis
- Niehaus J, Riedl MO (2009) Scenario adaptation: an approach to customizing computer-based training games and simulations. In: *Proceedings of the AIED 2009 workshop on intelligent educational games*, vol 3, pp 89–98
- Unity Asset Store - Storyteller (2020) Retrieved from <https://assetstore.unity.com/packages/tools/utilities/storyteller-97715>
- Hjelseth S, Morrison A, Nordby K (2015) Design and computer simulated user scenarios: exploring real-time 3D game engines and simulation in the maritime sector
- User Expectations Questionnaire for eNotice Serious Game (2019) Retrieved from <http://ec.europa.eu/eusurvey/runner/eNoticeSeriousGamePreDevelopment>
- European Network Of CBRN TraIning CEnters—Official Website (2019) Retrieved from <https://www.h2020-enotice.eu>
- Surer E, Atalay TB, Demirkan DÇ, Düzgün HŞ (2019) *Serious gaming in CBRNe domain: a survey on user expectations, concerns and suggestions*. In: 3rd international conference CBRNE—research & innovation, Nantes, France, pp 43
- Garrett J (2011) *The elements of user experience*. New Riders, Berkeley
- Witmer B, Singer M (1998) Measuring presence in virtual environments: a presence questionnaire. *Presence Teleoper Virt Environ* 7(3):225–240
- Davis F (1986) A technology acceptance model for empirically testing new end-user information systems: theory and results, Doctoral dissertation, MIT Sloan School of Management, Cambridge, MA
- Venkatesh V, Davis F (1996) A model of the antecedents of perceived ease of use: development and test. *Decis Sci* 27(3):451–481
- Brooke J (1996) SUS-A quick and dirty usability scale, usability evaluation in industry. CRC Press, Boca Raton
- Surer E, Erkayaoglu M, Oztürk ZN, Yücel F, Bryık EA, Altan B, Senderin B, Oguz Z, Güner S, Düzgün HS (2019) Developing a scenario-based video game generation framework: preliminary results. In: 15th international summer workshop on multimodal interfaces, pp 19

35. Valve Corporation (2020) Retrieved from <https://www.valvesoftware.com/en/>
36. Bjork S, Holopainen J (2005) Patterns in game design. Charles River Media, Hingham
37. Unity Asset Store: Hospital LowPoly (2019) Retrieved from <https://assetstore.unity.com/packages/3d/environments/hospital-lowpoly-82552>
38. Unity Asset Store: low poly laboratory pack (2019) Retrieved from <https://assetstore.unity.com/packages/3d/environments/low-poly-laboratory-pack-47677>
39. Hollan J, Hutchins E, Kirsh D (2000) Distributed cognition: toward a new foundation for human-computer interaction research. *ACM Trans Comput Hum Interact* 7(2):174–196
40. Hamilton P, Wigdor DJ (2014) Conductor: enabling and understanding cross-device interaction. In: Proceedings of the SIGCHI conference on Human factors in computing systems, pp 2773–2782
41. Villaroman N, Rowe D, Swan B (2011) Teaching natural user interaction using OpenNI and the Microsoft Kinect sensor. In: Proceedings of the 2011 conference on Information technology education, pp 227–232
42. Markus F, Kritzler M, Michahelles F (2017) HoloLens is more than air Tap: natural and intuitive interaction with holograms. In: Proceedings of the seventh international conference on the internet of things, pp 1–2
43. Ergun O, Akin S, Dino IG, Surer E (2019) Architectural design in virtual reality and mixed reality environments: a comparative analysis. In: 2019 IEEE conference on virtual reality and 3D user interfaces (VR). IEEE, pp 914–915
44. Furht B (ed) (2008) Encyclopedia of multimedia. Springer, Berlin
45. Rendering Profiler: Unity Documentation (2019) Retrieved from <https://docs.unity3d.com/560/Documentation/Manual/ProfilerRendering.html>
46. Surer E, Erkayaoğlu M, Öztürk ZN, Yücel F, Bıyık EA, Altan B, Şenderin B, Oğuz Z, Güner S, Düzgün HŞ (2019) *Developing a Scenario-Based Video Game Generation Framework: Preliminary Results*, arXiv preprint [arXiv:1911.07380](https://arxiv.org/abs/1911.07380)
47. Bangor A, Kortum PT, Miller JT (2008) An empirical evaluation of the system usability scale. *Intl J Hum Comput Interact* 24(6):574–94
48. Bangor A, Kortum PT, Miller JT (2009) Determining what individual SUS scores mean: adding an adjective rating scale. *J Usability Stud* 4(3):114–23
49. Brooke J (2013) SUS: a retrospective. *J Usability Stud* 8(2):29–40
50. Fishbein M, Ajzen I (1980) Understanding attitudes and predicting social behavior. Prentice-Hall, Upper Saddle River
51. Tavakol M, Dennick R (2011) Making sense of Cronbach's alpha. *Int J Med Edu* 2:53
52. Dyer R, Matthews JJ, Stulac JF, Wright CE, Yudowitch K (1976) Questionnaire construction manual, annex literature survey and bibliography. Operations Research Associates, Palo Alto

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Affiliations

Elif Surer¹  · Mustafa Erkayaoğlu²  · Zeynep Nur Öztürk³ · Furkan Yücel¹  · Emin Alp Bıyık¹  · Burak Altan¹ · Büşra Şenderin¹ · Zeliha Oğuz⁴ · Servet Güner² · H. Şebnem Düzgün⁵ 

Mustafa Erkayaoğlu
emustafa@metu.edu.tr

Zeynep Nur Öztürk
nur.ozturk@ug.bilkent.edu.tr

Furkan Yücel
furkan.yucel@metu.edu.tr

Emin Alp Bıyık
emin.biyik@metu.edu.tr

Burak Altan
burak.altan@metu.edu.tr

Büşra Şenderin
busra.senderin@metu.edu.tr

Zeliha Oğuz
zeliha.oguz@ug.bilkent.edu.tr

Servet Güner
e119605@metu.edu.tr

H. Şebnem Düzgün
duzgun@mines.edu

¹ Department of Modeling and Simulation, Graduate School of Informatics, Middle East Technical University, 06800 Ankara, Turkey

² Department of Mining Engineering, Middle East Technical University, 06800 Ankara, Turkey

³ Department of Computer Engineering, Bilkent University, 06800 Ankara, Turkey

⁴ Department of Psychology, Bilkent University, 06800 Ankara, Turkey

⁵ Colorado School of Mines, 1500 Illinois Street, 80401 Golden, Colorado, USA