Adaptive VR Educational Game Framework

Student Name: Logan Armstrong
Supervisor Name: Jingyun Wang
Submitted as part of the degree of BSc Computer Science to the
Board of Examiners in the Department of Computer Sciences, Durham University

Abstract—This project presents the design, implementation, and evaluation of a novel virtual reality (VR) educational game framework that incorporates an adaptive questioning system based on a new interpretation of the Elo-rating system and user-initialized customization through an in-game shop. A case study, called Cyber Decider, was developed as a concrete implementation of the proposed framework, focusing on teaching users about cyber security through multiple-choice questions. Results of an experimental study indicated that participants found the game to be a useful learning tool, with increased motivation to play the version featuring the adaptive questioning system. This research contributes to the field of virtual reality education by providing a comprehensive framework for developing VR educational games and introducing an innovative adaptive questioning system with proven effectiveness. The findings have the potential to inform future studies and developments in the rapidly evolving domain of immersive learning technologies.

Index Terms—Educational games, Adaptive and intelligent educational systems, Knowledge personalization and customization, Virtual reality

1 Introduction

The rapid development of technology has opened up new avenues for education and learning, with digital educational games becoming increasingly common in all manner of topics. Digital educational games, or serious games, refer to a branch of video games that have the purpose of teaching the user a given topic, instead of just providing entertainment [1]. Research continuously shows their pedagogical effectiveness to be just as good as traditional teaching methods, if not better [2]. Many researchers point toward increased levels of stimulation and motivation as factors for their success as learning tools [3].

Given the extensive evidence of positive learning outcomes, much research now looks towards extending the approach with various technologies. One of the main directions of current research is adaptive systems. These allow for learning experiences to be automatically personalized to the learner's capabilities and preferences [2].

There is no single method that is optimal for building an effective adaptive environment. Many well-researched systems that show strong results are not feasible to implement when a short development cycle is involved. For example, dynamic Bayesian networks (DBNs) can be unfeasible due to the level of expert knowledge required for their implementation [4]. On the other hand, utilizing an Elo-rating system (ERS) would allow for easy implementation, but the amount of user data required to have the system work can be too large to collect within a short time limit [5]. An adaptive system that is able to capture the adaptivity of one of these systems while also requiring less time to set up would be a valuable contribution to the field. Thus, The first objective of this project is revealed: **to define a novel adaptive system that retains the benefits of larger systems while being feasible to implement in a shorter development cycle**.

Along with adaptive systems, the two other most well-studied technologies in the field of digital educational games are (1) customization, which refers to either a user or supervisor being able to directly make changes to the game; and (2) virtual reality learning environments, which utilize real-time interactive graphics with 3D models combined with display technology (for example a VR headset) to give users a sense of immersion in a model world [6].

A common theme across many studies focused on digital educational games, particularly virtual reality, is that they take a loose exploratory approach - not backing up their proposals with experimental data that can point towards a strong conclusion [7]. A likely cause of this is the large amount of work required to construct the systems that would allow for rigorous study. That is, multiple versions of an educational game (with and without the independent variable of the investigation) typically need to be produced in order to conduct a meaningful experiment.

One way to combat this issue is to define a framework that can allow for more efficient implementation of VR educational games. Such a framework would benefit from being designed to utilize systems proven to be pedagogically effective, such as adaptivity and customization. One paper in the literature describes a framework focused on customization, that poses questions on the topic of cybersecurity [8]. This framework, called MEMORABLE, is shown to be pedagogically effective and provide high levels of motivation to users. However, as it is the framework lacks support for adaptive systems and virtual reality. An improved version of this framework with the aforementioned features could be a great tool to allow for more rigorous experimental studies and therefore the production of more effective VR educational games.

With this in mind, the second objective of this project is **to provide a VR educational game framework that can be**

used to efficiently implement ideas and allow for more conclusive experimental studies across the field. Aside from virtual reality capabilities, this framework should also include some form of an adaptive system and customization features in order to be relevant to as much of the serious games research domain as possible.

The two research questions we ask are: (1) 'Can we present an adaptive questioning system suitable for use in digital educational games with short development cycles' and (2) 'Can we define a pedagogically effective VR educational game framework that can allow for more efficient experimentation of new technologies in the domain of VR educational games?'.

1.1 Deliverables

In order to answer this, we present a VR educational game framework with support for adaptive systems and customizable features. This framework builds on top of the previously mentioned MEMORABLE framework, but with a new focus on modularity of design. The best tools and development environments are discussed for constructing each component, with a focus on future-proofing and ease of implementation behind these choices.

Next, we introduce a concrete implementation of the framework, named Cyber Decider, in which users are able to answer multiple-choice questions about various cyber-security topics. These questions are posed to users based on a novel adaptive questioning system that is designed specifically for use in a short development cycle. This is based on an extension of the previously mentioned Elo-rating system (ERS). It utilizes preset difficulty ratings at system initialization, as well as difficulty updates local to each player. In addition, we also propose the use of a "streak system" to counteract slow skill rating convergence.

In order to assess the effectiveness of the adaptive system and game framework as a whole, we design an experimental study involving thirty participants. These participants play two versions of the Cyber Decider game - one with the adaptive questioning system and another with a random questioning system. Results in learning outcomes and levels of enjoyment are analyzed to make evaluations on the extent that these deliverables meet our research goals.

2 Related Work

This section looks at previous work carried out on lines of research that have outcomes relevant to our research questions. Specifically, we observe the current technologies being explored or improved for use in digital educational games. Then, we determine how these trends apply to the deliverables of our project.

2.1 Digital Educational Games

There is a large and continuously expanding body of scientific literature which focuses on digital game-based learning as an effective method of teaching. Digital educational games or 'serious games' are interactive learning tools which have the primary goal of integrating education, as opposed to providing entertainment [1]. Compared to traditional teaching methods, educational games offer a more motivating and stimulating learning environment which helps to preserve the user's attention for extended periods of time [3]. Because this duration is spent being genuinely productive by the user, educational games can be defined as separate from the hedonistic nature of traditional video games. Essentially, educational games attempt to capture the increased motivation and attention that digital games offer but then apply that to a given learning task.

While this method seems to have a clear pedagogical advantage over traditional teaching methods, some studies show that this is not always the case [9]. With that being said, it should be made clear that there is a consensus across the field that serious games are never substantially worse than traditional teaching methods. That is, serious games perform at least as well as traditional teaching in studies that compare the two approaches [1], [10], [11].

With the baseline effectiveness of educational games well established, work in the field now looks towards improving the approach with new technologies. Future research will benefit from experimental studies that systematically explore which of these technologies are most effective at supporting learning [7].

2.2 Technologies

2.2.1 *Adaptive Systems*

A useful characteristic of educational games is that they produce a large amount of data, which can be captured and analysed to build adaptive systems. These systems can then be used to build adaptive learning environments, where the aim is to provide learning experiences personalised to the learner's cognitive capabilities, knowledge levels, and preferences, among other factors [2], [12].

The effectiveness of adaptive serious games compared to non-adaptive serious games has mixed findings. One study found negligible differences in knowledge retention between the two for young children studying basic maths and reading [13]. However, the rate at which the children obtained new knowledge was in fact higher for the adaptive version of the game. This indicates that although adaptivity may not be required for successful learning, it can allow for a more efficient learning process.

There are several methods that can be used to successfully build an adaptive environment, and no singular technique has been found to be clearly better than others in the literature [2]. One such method is to use a dynamic Bayesian Network (DBN) that works with learning materials including feedback content and instructions to control the difficulty of the game [4].

An advantage of dynamic Bayesian networks is that they can function well even with sparse data if the conditional probabilities in the DBN are specified in advance by subject matter experts [14]. A downside of this method is that the complexity of implementing such a model may make it unsuitable for a time-sensitive project.

As an example, Hooshyar uses this method in an educational game to decide which algorithm enemy non-player-characters should follow, and what kind of feedback or hint should be provided to the player to allow them to progress [4]. Results showed that the increase in students' learning

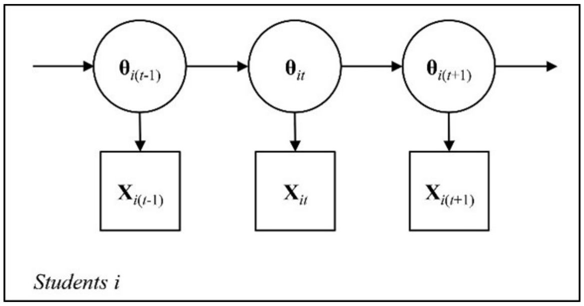

Fig. 1. Basic depiction of a dynamic Bayesian network, with latent variables (θ) and observable variables (X), for each student i, at various time points denoted by t. [14]

interest using the adaptive educational game was statistically significantly higher compared to those who learned with the traditional technology-enhanced approach.

Another method that can be used to build an adaptive system for an educational game is the Elo rating system (ERS). The ERS allows for on-the-fly estimation of item difficulty and person ability parameters [15]. This is done by assigning each student s and question item i ratings θ_s and d_i respectively. The correctness of an answer by student s for item i is denoted as correct$_{si} \in \{0, 1\}$. The probability that a student will answer a certain question correctly can be determined using the following shifted logistic function:

$$P(correct_{si} = 1) = 1/(1 + e^{-(\theta_s - d_i)}) \tag{1}$$

With this system in place, questions can be posed to students based on these probabilities. Upon answering a question, θ_s and d_i are updated as follows:

$$\theta_s := \theta_s + K \cdot (correct_{si} - P(correct_{si} = 1)) \tag{2}$$

$$d_i := d_i + K \cdot (P(correct_{si} = 1) - correct_{si}) \tag{3}$$

Where K is typically either a fixed constant or uncertainty function. The positive aspects of this system largely come from its simplicity; implementing the above equations requires little effort, and their intuitive nature makes it very easy to extend functionality with new ideas [15]. One example of this is an extension of the logistic function to work with multiple choice questions that have k possible answers:

$$P(correct_{si} = 1) = 1/k + (1 - 1/k)/(1 + e^{-(\theta_s - d_i)}) \tag{4}$$

A downside of the ERS is that it typically requires a large amount of user data to work well [5]. In the context of a research study, this would mean a high sample of participants would be required to test the system's effectiveness.

Adaptivity in educational games is not limited to adjustments in-game difficulty. An alternative example is having an adaptive hints system, which offers direction to struggling players without necessarily changing any difficulty metrics of the game or posing harder questions to the user [16]. These hints can be based on progress assessment, which can be measured in several ways depending on the type of game. One method is to study players' movement behaviour when they would like to receive a hint [16]. Data could be collected from many players and a machine learning model can be produced from this which predicts when a player wants help. The study which proposed this idea suggested its use in a Virtual Reality setting, where a headset's orientation and angular velocity would be tracked. [16].

2.2.2 Virtual Reality

Virtual reality (VR) technologies have consistently evolved over the years, becoming more prominent in many different domains. VR can be described as real-time interactive graphics with 3D models combined with display technology to give users a sense of immersion in a model world which provides direct manipulation of objects; and the illusion of participation in a synthetic environment [6]. This technology has made its way into the educational sector, with significant uptake during the global pandemic in 2020 [6]. Consequently, a new type of educational game emerged, which many researchers refer to as Game-Based Immersive Virtual Reality Learning Environments (GIRVLE) [11].

The effect of this technology on educational games has been observed in several studies. One such study looks at whether differences in educational value exist between immersive virtual reality games and traditional desktop games [17]. The study found that significant differences were found in measurements of presence, motivation, and immediacy of control. These affective properties allowed users to see the value of a lesson more easily, and therefore be more willing to exert cognitive effort on the task at hand.

Another study focusing on the effectiveness of Virtual Reality found improved knowledge retention when compared to more traditional digital teaching methods [11]. The main reasoning behind this was a significant enhancement of motivation in the group that used virtual reality. Increased motivation has previously been shown to facilitate learning in educational games [18].

Virtual reality allows for several different pedagogical approaches to be used when designing a serious game. One study compares four types of approaches to determine the best methods for learning [19]. It was found that an explicit story structure resulted in a higher retention of content. However, an implicit story structure led to higher feelings of presence and cognitive interest. These findings indicate a trade-off between guidance and freedom which should be balanced carefully when designing an educational game. Another aspect of this trade-off is development cost/time. Games with more exploratory mechanics will likely require more assets and therefore more development.

The development of a GIRVLE poses its own unique challenges, such as users being initially unfamiliar with the interfaces of the devices [20]. This along with other domain-specific problems must be considered when designing an educational game in the virtual reality domain. One way to account for this is to have lots of iterative user testing and expert feedback throughout the development cycle [21]. A downside of this method is that it can be quite time-consuming, so may not be suitable for time-sensitive projects.

One way to reduce development time is to utilize a pre-existing framework. There exists one framework in the literature that could be used as a foundation to a well-functioning GIVRLE [8]. This framework, called 'MEMORABLE' is an educational game framework that allows instructors to customize the game learning contents (see 2.2.3 Customization) [8]. While not directly related to virtual reality, the framework is adaptable and could be extended to facilitate this technology.

2.2.3 Customization

Within the domain of educational games, customization can be split into two sub-sections. The first of these is user-initiated customization. This refers to the user making choices about a given aspect of the game, such as visual and audio attributes [22]. One study that evaluates this type of customization focused on the use of custom avatars in an educational game for children [23]. It was found that the avatar served as a great factor for the children's learning because the customization was integrated into the reward system. That is, children completed learning tasks more readily because it allowed them to customize their avatars more.

In contrast, another study that focused on environmental customization in a children's educational game had mixed findings [22]. Results showed that environmental cosmetic customization features were not necessarily instructionally beneficial to the children. The contrast in results between [22] and [23] may imply that customization is only useful when integrated into the reward system of the game.

The second type of customization found in educational games is supervisor-initiated customization. This type of customization refers to an external administrator – typically a teacher or parent – making choices about aspects of the game that change the user's experience. An example of this would be a teacher being able to enter questions into a database that would then appear in the game world for the user to answer [10]. As described above, the 'MEMORABLE' framework is an example of this customization technique. In this framework, instructors can enter cybersecurity-related questions into a database which will then be answered in-game by the users. Feedback from users suggests that the game framework 'achieves the goal of stimulating learning interests and motivating players to learn the cyber-security content, through rewards and punishment' [8]. The customization available to the instructor allows them to keep the game usable for years to come, even as technology moves forwards and previously relevant questions become obsolete.

2.3 Summary and Application

Looking firstly at adaptivity, the main contribution this theme offers in educational games is increased efficiency of learning. Although improvement of knowledge retention has mixed findings, improved learning efficiency is in consensus across the field. The adaptive systems most dominant across the literature have downsides when it comes to fast execution, stemming from either large implementation or data-gathering requirements. The Elo-rating system suffers from the latter, but is very easy to extend with new ideas. Because of this, it makes sense to attempt to extend the functionality of the Elo-rating questioning system in such a way that data-gathering requirements are reduced. If successful, this would help to answer our first research question.

Virtual Reality has been shown to offer increased immersion and motivation for users, consequently improving their cognitive performance. There are different approaches that can be taken to developing a virtual reality educational game, with a direct storyline and passive interaction being the least costly on development time. Interestingly, this game design approach could be easily integrated into the 'MEMORABLE' framework discussed above to further save development time. There are currently no papers discussing a game framework specifically for virtual reality educational games, which makes it a good topic for the project.

User-controlled customization shows positive results for learning only when it is linked to a reward system in the game. Instructor-controlled customization helps to maintain the usability of the system over long periods of time even when fields such as cyber security change very quickly over the years. Also, this customization allows for solutions to be generalized to work in other subject domains. This is because instructors can include questions from any topic area they want. With all this in mind, it would be wise to allow for the addition of customization features in an educational game framework. The MEMORABLE framework serves as a good example of this, and can act as a rough guide in designing the framework for this project.

3 Methodology

For our solution, we provide a full implementation of a VR educational game framework, with both user-initiated customization and adaptive capabilities. We call this implementation "Cyber Decider". There are two main purposes of this solution: (1) to showcase that the framework design can allow for the efficient testing of various technologies in the domain of digital educational games, and (2) to directly evaluate an adaptive questioning system that utilizes a novel variation of the Elo-rating system. To begin, an overview of the framework design is given with explanations of why specific platforms were chosen to house each component. After this, we provide a detailed look at our implementation, iteratively walking through each component of the game.

3.1 Framework Design

The general design of the game framework is shown in Fig. 2. It essentially is comprised of three main components: the virtual reality environment with an in-game shop; the cloud database which stores all data required for the game to function; and finally the adaptive questioning system which essentially ties the two together.

The role of the virtual reality environment is to act as the point of interaction for the user. To interface with this environment, users must use a VR headset and controllers. With this, they are able to choose between answering a question (chosen by the adaptive questioning system) or accessing an in-game shop, where they can obtain rewards for correctly answered questions.

Both the virtual reality environment and in-game shop are built on top of the Unity game engine. Unity is well

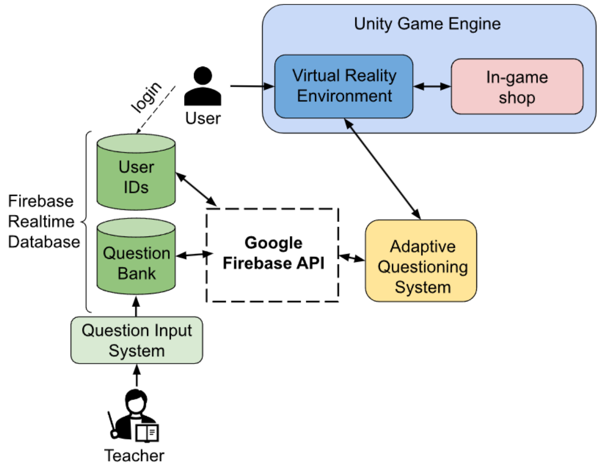

Fig. 2. The overall structure of the game framework.

established as a leader in VR development, supporting all popular VR headsets such as the Oculus Quest, Oculus Rift and HTC Vive. Furthermore, a combination of an active community, rich feature set and intuitive interface make it likely to continue to be at the forefront of future advances in VR game development.

Data on all users and questions is stored on a Firebase Realtime Database. This is a NoSQL, cloud-hosted database that stores data in a JSON-like format. It synchronizes data in real-time between clients, which means that any change made on one device is automatically propagated to all other connected devices. This platform was chosen over other cloud-based solutions because of its intuitive API, and direct support for Unity projects. Attached to the database is a question input system, from which a teacher or other type of supervisor is able to add or remove questions from the database.

The adaptive questioning system sits in a script connected to the virtual reality environment, acting as a middleman between the environment and the real-time database. It is used to determine which question should be posed to the user based on some algorithm specific to a given implementation of the framework.

For further clarity on the functionality of this framework, a generalized version of the user journey for any implementation is portrayed in Fig. 3.

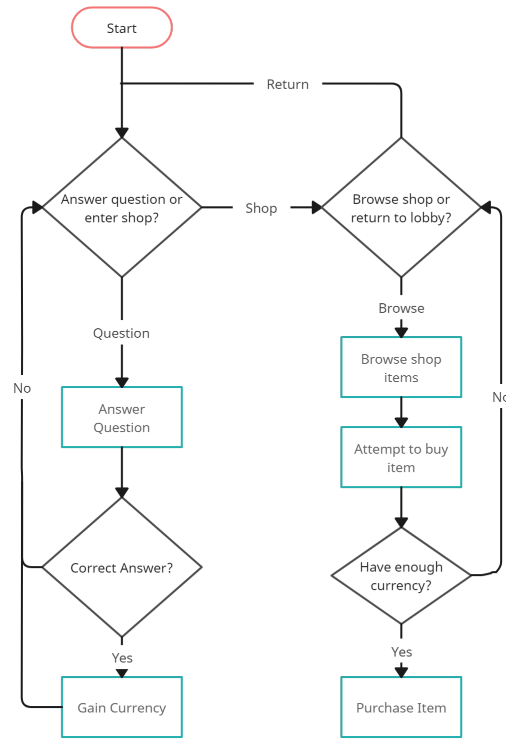

Fig. 3. The general user journey of any game that utilizes the framework. Note that there will likely be specific events within a given implementation that are not covered in this diagram.

3.2 Implementation

We constructed a concrete implementation of the framework above to demonstrate the effectiveness of a new adaptive questioning system and the framework itself. This implementation takes inspiration from the previously discussed MEMORABLE framework, posing multiple-choice questions to the user based on cyber security topics [8]. To answer these questions, the user must navigate from a central lobby area to one of four doors that they think has the correct answer above it. Upon walking through this door, the user submits their answer by pressing a button found in the room. This process fits neatly into the question-answering portion of our framework, and is well-suited to be extended to include an in-game shop and adaptive questioning system.

Since the main aim of this implementation is to evaluate the effectiveness of a novel adaptive questioning system, two versions of the implementation were built: one with the adaptive questioning system, and another with a random questioning system.

The development cycle of this project followed an iterative approach. With each iteration, additional features were designed, developed, and tested until there was a fully functional game. When possible, user testing was performed on features with up to 5 participants of varying technological backgrounds. This allowed for the system to be tested for ease of use from a wide of perspectives.

3.2.1 *Virtual Environment*

The first stage of development focused on constructing the virtual environment, from which the player can interact with the system. All scene construction was performed in the Unity editor, with a package called ProBuilder. Assets used within the environment were either created using primitives from the Unity engine or acquired from the Unity asset store.

The first section of the virtual environment to be constructed was the main lobby. This is where the player spends the majority of their time and engages with all other aspects of the game. Many deviations were made throughout the development of this lobby. The first of these was a rearrangement of the main components of the lobby. In the first iteration of the lobby, question rooms sat behind each wall of the lobby, requiring the user to turn excessively. This caused motion sickness in some users when testing, so we resolved this by placing all question rooms directly in front of the user's start position.

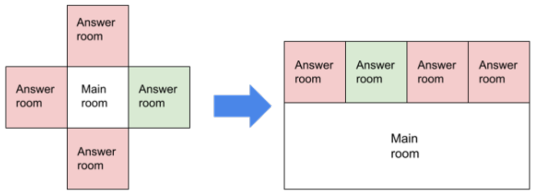

Fig. 4. The difference between the lobby arrangement in the original version of the lobby, and our new layout.

In order to actually answer questions, the user needs to be able to activate a trigger. In the original lobby, this was done by pressing a button in the centre of the room. Here we take a similar approach, but adjust the position of the button to sit closer to the back wall of the lobby. This is to account for the fact that all question doors now sit along the front wall of the room. Instead of sitting alone, the button is also part of a larger control panel (as seen in Fig. 5), which allows the user to easily access information about their play session and look at the in-game store.

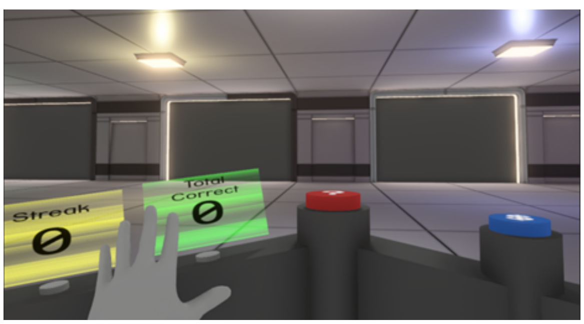

Fig. 5. The default position of the player. From here they are able to use the control panel to either answer a question, enter the store or walk around the lobby.

With the primary components of the lobby in place, the next phase was designing and constructing the four answer rooms of the game. Instead of having these rooms be a basic extension of the lobby, we chose to have each answer correlate to a world that the player must teleport to. The reasoning behind this was to diversify the environments the user was interacting with and therefore increase their overall engagement with the system. A button was also placed in the centre of each world, so that the user can submit their answer.

Each of these worlds (as seen in Fig. 6) is accessible by their respective portals (equivalent to doors) in the lobby. A custom texture was created for these portals using the Unity shader graph tool to improve levels of immersion.

Upon completing the main parts of the virtual environment, work began on allowing users to interact with it. We used a Unity package called the XR interaction toolkit to track user input data. This package supports all leading virtual reality headsets, meaning that the game can be played from any such device. There were three main modes of interaction required for the full functionality of the game:

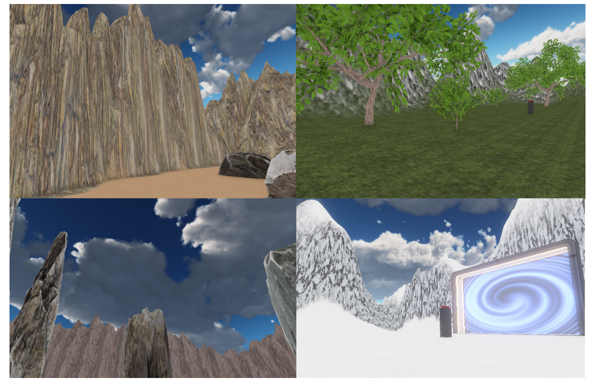

Fig. 6. The four possible ”worlds” that the user can enter when answering a question.

(1) **Full tracking of the player's orientation and translation in their real-life physical space**. This is to allow for an immersive experience in which the player is able to look around the virtual spaces in an intuitive manner. The Unity game engine with the XR interaction toolkit handles this requirement seamlessly. Very little user testing was required for this function as the user experience is already very optimized by the toolkit functions.

(2) **Collision response between the player controllers and interactable objects in the game world**. This allows for the handling of triggers such as pressing buttons or detecting entering rooms. Setting this up required adding box colliders to the in-game controller positions, as well as any interactable objects. Rounds of user testing was required here in order to ensure box colliders were positioned and sized in a way that resulted in realistic feedback for the user.

(3) **A locomotion system that allows for the movement of the player beyond the limitations of their real-life physical space**. This allows the player to move around the virtual spaces, which is vital for being able to answer questions in the game. The XR interaction toolkit has built-in functionality to support this, such that the user can use a joystick to move their avatar around the game world. We chose to assign the joystick on the left-hand controller for this purpose, as is common in many modern video games. Some rounds of user testing were required to improve this system.

The first issue was related to the speed of the player movement. User feedback was gathered to find a speed that was fast enough such that moving around wasn't tedious, but also slow enough to prevent motion sickness.

The second issue was preventing the player from moving out of bounds. The intuitive solution to this problem is to add a collider to the player that directly follows their real-life position. However, this can result in motion sickness. Take an example of a player walking towards an in-game wall using real-life motion. Their in-game movement will be halted, while their real-life movement will continue. To get around this issue, we instead fix the player collider to the origin of their real-life space. This way, their real-life movements will always match up with in-game movements.

However, this is not a perfect solution. If the player moves too far off-center, then collisions will happen away from the player's observed position. This means that real-life movements must be kept within a relatively small area to maintain game functionality.

3.2.2 Game Loop

With the virtual environment and basic interaction complete, the next phase was to implement full functionality of the main game functions. This essentially meant making sure that the game followed the user journey described in Fig. 3. A number of scripts were written to allow for this:

1) **ButtonVR**: Allows for buttons to provide an appropriate response upon collision with the player. This response is dependent on the specific button the script is tied to, and is used in three different ways within this implementation.
2) **DoorMechanism**: Controls whether the doors to the portals in the main lobby are open or closed. Essentially, this grants/denies the user access to the answer worlds. This script is triggered via instances of the ButtonVR script. Doors are set to open upon pressing the question button (the red button in Fig. 5), and close after pressing an answer button (as seen in Fig. 7).

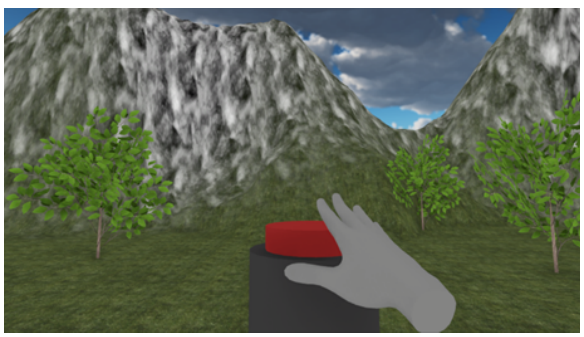

Fig. 7. The view of the player when they are about to submit their answer to the current question.

3) **CabinetMechanism**: Similar to DoorMechanism, controls whether the main control panel or cabinet in the main lobby is available for use. This script is also triggered via an instance of a ButtonVR script attached to the question button.
4) **NextQuestion**: Essentially a placeholder script that loads a dummy question into the game's virtual environment when the question button is pressed. The question text is placed infront of them, just below eye level, while the four possible answer texts are placed above each portal. This is shown in Fig. 8.
5) **PortalTeleporter**: Allows for the user to teleport to answer worlds upon colliding with their respective portals in the lobby. There were some initial problems with this system in the original version of the script. This version used the same collider as the one used for the locomotion system. However, as discussed in section 3.2.1, this collider is prone to become slightly offset from the user's observed position when real-life movements are involved. In order to get around this issue, we use a new box collider directly attached to the user's headset position in order to allow for seamless teleportation.

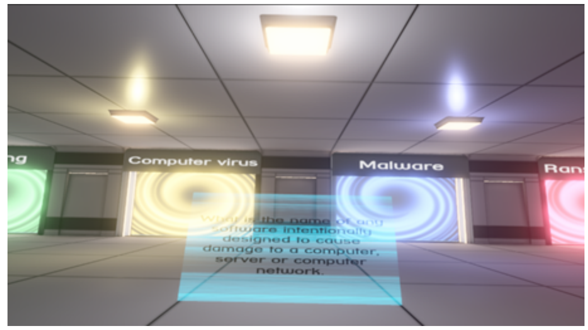

Fig. 8. The view of the player upon beginning to answer a question. They are able to read the question on a hologram slightly below eye level and study possible answers in front of them, slightly above eye level.

6) **MovementToggler**: Controls whether the locomotion system is enabled for the user. This script is used when the user submits an answer, in order to prevent them from offsetting their position before being teleported back to the lobby.
7) **ResetPlayerPos**: Determines the origin point of the user, and teleports them to this point upon submitting an answer to a question. This essentially allows for the game loop to repeat as many times as required.
8) **ResultQuads** Displays a small message box to the user when they submit an answer. This box is either green or red, depending on whether they answered correctly. If correct, it also tells them how much gold they earned from correctly answering the question. If wrong, it provides the user with the correct answer. Visualizations of these message boxes are in Figures 9 and 10.

Fig. 9. The pop-up that the user is greeted with upon answering a question correctly.

There were also additional scripts which were used to improve general usability. The **Fader** script causes a gradual fade to black whenever a player teleportation occurs. This is to reduce motion sickness for the user, and allow for a more seamless transition overall. The **AnimateHolograms** script essentially deals with all animations within the game, including the pop-up animation for the in-game store menu and question box. These animations improve the aesthetics

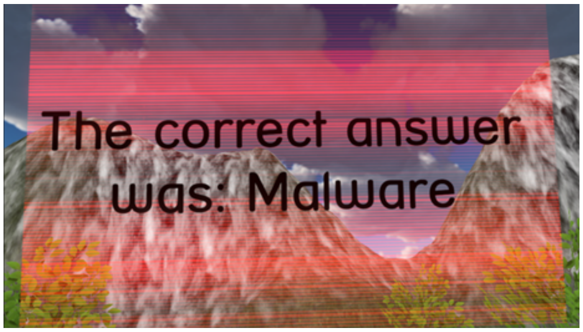

Fig. 10. The pop-up that the user is greeted with upon answering a question wrongly.

of the game, providing the user with a more immersive experience. Finally, the **PointTowards** script causes menu items to point towards the position of the user. Again, this improves general usability of the game.

All components of the game loop were designed to be self-contained, such that all components were modular in nature. This was done by following good OOP practices such as encapsulation, inheritance, and abstraction.

3.2.3 Back-end Functionality

Upon completetion of the main game loop, the system was ready to begin using legitimate cyber security questions. In order to maintain flexibility of the framework, it was important to have questions be accessed from a server seperate to the client, instead of being hard-coded. This allows for new questions to be seamlessly added to or removed from the system. In fact, it allows for the system to be adapted to completely different question topics altogether, contributing to us meeting our second research objective.

A Firebase Realtime Database was used to store user and question data. Real-time databases are one of the services offered by Firebase, a cloud-based platform for developing online and mobile applications. The Realtime Database is a NoSQL cloud database that supports real-time client synchronisation and saves data in JSON format. We followed best practises in the construction of our JSON tree, flattening the data structure to minimize nesting and improve querying efficiency. Data was also organized in such a way that suited our application's querying needs. The general structure of this JSON file is shown in the Fig. 11.

In order to interface with the data, We used the Firebase SDK for Unity/C#, which offered a complete set of tools for reading, writing and updating values. This API was accessed in a client-side script called **DatabaseManager**. The main functions of the script are as follows:

1) **GetCurrentUser**: Gather the data relating to the current user. The most important item fetched here is the user's skill rating, which is used within many other functions. This action is performed using the GetValueAsync() method from the firebase API, which automatically handles failed requests and retries them.
2) **GetAllQuestions**: Very similar to the GetCurrentUser method, but instead fetches data on all questions and their associated difficulties specific to the current user. Again, the GetValueAsync() method is used to handle failed requests.
3) **SelectRandomQuestion**: Randomly selects a question number that corresponds to a question ID in the database. However, it only selects this question if it wasn't already posed in the previous 5 iterations of the game loop. This function is only used in a version of the game that does not involve an adaptive system.
4) **SelectQuestionWithBestProbability**: This holds all functionality of the SelectRandomQuestion method, but it uses an adaptive system to find a viable question. This system is explained in detail in section 3.2.4.
5) **UpdateRatings**: This function is responsible for updating the skill ratings and difficulty ratings of the player and recently answered question. It does this using the SetValueAsync() method from the firebase API.

```
{
  "questions": [
    null,
    {
      "answer1": "answer1 is always correct",
      "answer2": "this is always wrong",
      "answer3": "this is always wrong",
      "answer4": "this is always wrong",
      "presetDiff": 1,
      "questionText": "This is an example question?"
    }

  ],
  "users": [
    null,
    {
      "skillRating": 0,
      "username": "TestUser",
      "localQuestionDiffs": {"1":1}
    }
  ]
}
```

Fig. 11. The general structure of the JSON data used to store user information and question data.

A total of 30 questions were added to the database in total.

3.2.4 Adaptive System

The implementation utilizes a new extension of the Elo-rating system (ERS) to pose questions to users based on their skill level. The decision to base the system on the ERS was due to the simplicity of implementation and extension of capabilities. A problem with the ERS is that it needs a large amount of user data to be able to function correctly. This was particularly a problem for this project as the time constraints placed on it made it unfeasible to gather enough data to achieve sufficient functionality. Furthermore, the greater complexity of alternative adaptive systems such as dynamic Bayesian networks meant that a similar amount of time would be lost by trying to use such an approach.

In order to circumvent this problem, we first propose a variation of the ERS which utilizes **preset difficulty values**. Instead of having all questions initialized with a difficulty rating of 0, we set them to 1 of 3 values based on expert-level knowledge of the cyber security domain:

- 0 - The question can be answered with little knowledge of cyber security, possibly even just with common sense.
- 1 - The question requires a good understanding of cyber security concepts.
- 2 - The question requires a good understanding of cyber security concepts and understanding of the current best practices in the industry.

Each preset difficulty category had 10 questions each. A trade-off of this approach is that expert-level knowledge is required for the system to function effectively. This is a downgrade from the vanilla Elo-rating system in which questions are sorted automatically.

This alone does not fully solve the issue of requiring large amounts of data for the system to function correctly. Although these presets will help sort questions initially, difficulty ratings will still become easily skewed when few users have played the game. For example, if the first user struggles with questions relating to phishing attacks, difficulty ratings for these questions will increase. Now, it will take some time for phishing-related questions to reduce to a more globally acceptable difficulty rating, negatively affecting the experience of future users.

Our next proposed extension of the ERS is to switch to a **local updating system**. Given the set of all students S and all question items I, a vanilla ERS implementation would have each student $s \in S$ have an estimated skill rating θ_s, and each question item $i \in I$ have an estimated difficulty rating d_i. Instead of questions using this global difficulty rating, we propose a set of difficulty ratings D_i for each question item, where each $d_{si} \in D_i$ is used against a specific student s. This means that equations (3) and (4) were updated as follows:

$$P(correct_{si} = 1) = 1/k + (1 - 1/k)/(1 + e^{-(\theta_s - d_{si})}) \tag{5}$$

$$d_{si} := d_{si} + K \cdot (P(correct_{si} = 1) - correct_{si}) \tag{6}$$

Note that for this implementation, in Equation 5, $k = 4$ at all times due to the fact that all questions have 4 possible answers.

The initial local difficulty ratings for each user are set to be identical to the presets discussed earlier. This means that the system is able to provide a linear progression of difficulty straight out of the box. As the user continues to play, the system becomes more tailored to their specific strengths and weaknesses.

One topic yet to be discussed is the value of K in Equation 6. The use of a local update system and preset difficulty values has the potential to cause many questions to be missed out if the value of K is too high. This is because questions will initially be clustered in 3 groups, and a user may quickly progress past the first and second groups. A common value used in traditional implementations of the ERS for educational games is $K = 0.4$ [24], [25]. So, to reduce jumping across groups too fast, we use a reduced value of $K = 0.1$. This value was initially decided through intuition and then verified through a series of user tests comparing K values ranging from 0.01 to 0.8.

Questions are selected for users based on the estimated probability the user will answer correctly, via Equation 5. Typically the question that is 75% likely to be answered correctly (or as close to this as possible in the database) is selected for the user [26]. However, this poses a problem to highly skilled users. Since K is so low, high-skill users may become stuck answering questions unsuited to their knowledge. To reduce this problem, we introduce a **streak system**, in which users are posed harder questions dependent on how many questions have been answered correctly in a row. This is done by using the following formula to get an adjusted probability P_{streak}.

$$P_{streak} = 0.75 - w \cdot streakCount \tag{7}$$

Where $streakCount$ is the number of questions the user has answered correctly in a row, and w is a weighting term. We then search for a question in the database that has a probability equivalent to P_{streak} of being answered correctly (or as close as possible to this). This means that question difficulty increases more rapidly as more questions are answered correctly in a row, helping to boost high-skilled players to their appropriate level within a shorter time period.

A value for w had to be found which allows for a difficulty acceleration rate that benefits high-skilled players while not causing problems for lower-skilled players. To do this, user testing took place in which a range of values were tested. From this, a value of $w = 0.03$ was determined. This means that the difficulty of the question posed to the user increases by approximately 3% for every consecutive correct answer.

In summary, our adaptive questioning system utilizes an extension of the Elo-rating system, with preset question difficulties and question updating local to each user. This is to get around the issue of requiring large amounts of user data before the system is functional. We utilize a streak system to counteract the downsides of a low K in the update functions. This system was implemented within the **databaseManager** script of our game, described in the previous section.

3.2.5 *In-game Store*

The final stage of development was implementing customization features. We integrate user-initialized customization via an in-game store. In this store, currency can be exchanged for cosmetic items that appear in the lobby of the virtual environment. These items are all related to different cyber security topics, such as computer viruses and internet bots. This way, the rewards themselves also act as learning tools.

In order to obtain currency, users must correctly answer questions. This is an example of gamification and is done to create a satisfying game loop in which users want to continue to answer questions.

To encourage obtaining a streak of correct answers (and therefore increase the size of question difficulty jumps), we include the streak count in the calculation of how much currency the user receives. This calculation is performed using the following equation:

$$gainedCurrency = min((10 + 5 \cdot streakCount), 40) \quad (8)$$

By limiting the maximum gained currency to be 40 units, we prevent the user from obtaining currency too quickly and ending their incentive to continue playing.

This store appears in the virtual environment as a pop-up window that points toward the player's camera position. The primary aim when designing the store menu was ease of use. To meet this aim, we used the heavily optimized menu functionality available with the XR interaction toolkit. With this, the user is able to interact with the store by pointing toward the window and pressing purchase buttons using the VR controller trigger.

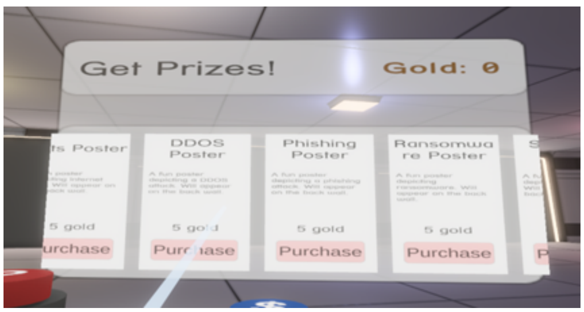

Fig. 12. The menu that the user is greeted with upon entering the shop. From here they are able to purchase in-game items.

A large focus when designing this feature was ensuring a modular design such that additional shop items could be added in the future. To facilitate this, we utilized scriptable objects to represent individual items. Scriptable objects are a powerful feature in Unity that allows for the creation of custom, reusable data assets. By representing each shop item as a scriptable object, we were able to define their properties, such as name, description, and price, in a single place. This approach promotes code reusability and simplifies the addition of new items to the shop.

The integration of these scriptable objects into the shop interface was achieved by automatically assigning them to panels. The shop interface consists of a dynamic panel system that populates itself with the scriptable objects available in the project. Each panel is responsible for displaying the relevant information about an item, such as its name, description, price, and icon. This dynamic system enables seamless addition and removal of items in the shop interface, simply by adding or removing the respective scriptable objects.

The implementation of the shop feature adheres to several key OOP principles, such as encapsulation, inheritance, and abstraction. Encapsulation is observed in the organization of the scriptable objects, which keep the item data and related methods self-contained. Inheritance enables the creation of specialized item types, such as consumables or equipment, that inherit from a base item class. Abstraction is demonstrated by the separation of the shop interface and the underlying item data, which allows for easy modification of either component without affecting the other.

By adhering to OOP principles, our modular shop design offers several benefits, including:

1) **Code maintainability**: The modular design promotes a clean and organized codebase, simplifying updates and maintenance.
2) **Scalability**: The system allows for easy addition and removal of items, making it adaptable to the changing needs of a project.
3) **Code reusability**: The scriptable objects and OOP principles facilitate code reuse across different projects or game modes. This is particularly relevant to our second research question.

Unfortunately, due to time constraints and the fact that this feature was not deemed essential to providing sufficient answers to our research questions, supervisor-initiated customization was not implemented within this game. However, future iterations of the game would benefit from incorporating this type of customization, as is discussed in the conclusion.

4 RESULTS

In order to assess the effectiveness of the adaptive questioning system and the educational game as a whole, an experimental study was conducted involving 30 participants.

To begin, participants were individually informed that they would be playing an educational game focused on cyber security, and that they would play two different versions of the game. One version of the game utilizes the adaptive questioning system described in section 3.2.4, while the other uses a random questioning system.

In order to minimize the risk of demand characteristics influencing our results, they were not informed of the differences between the versions of the game. Next, each participant was given a cyber security help sheet, from which they could learn the basics of the topic. This was done so that all participants were at least able to correctly answer the easier questions in the game. All participants were given the opportunity to read the helpsheet for five minutes, although some participants of a more technical background opted to waive this right.

After this, each participant played a version of the game for 10 minutes. 50% of the participants started with the adaptive version of the game, while the other 50% started with the non-adaptive version. This was done to prevent carryover effects impacting the investigation. As soon as the time limit was reached, the participant started playing whichever version of the game they hadn't played for a further 10 minutes.

Upon completion of this, every participant was asked to fill out a questionnaire detailing their perceived differences between the two versions of the game, as well as their experience with the game as a whole. A visualization of this experimental study can be found in Fig. 13:

4.0.1 Effectiveness of the Adaptive System

In determining the effectiveness of the adaptive system, we first look from a pedagogical perspective. To do this, we compare the percentage of correctly answered questions in the adaptive version of the game versus the non-adaptive

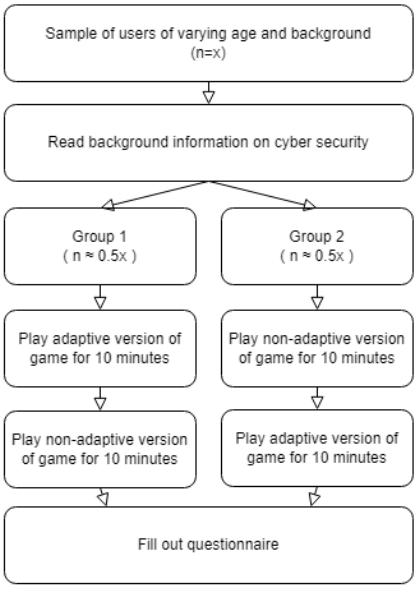

Fig. 13. The structure of the experimental study used to assess the effectiveness of the adaptive system and game as a whole.

TABLE 1
Descriptive Statistics of the percentage of correct answers by participants in the adaptive and non-adaptive versions of the game.

	N	Minimum	Maximum	Mean	Median	Range	IQR	Std. Deviation
% Correct Adaptive	30	60	100	81.498	81.653	40	10.564	10.178
% Correct Non-Adaptive	30	23.077	90.909	55.371	54.792	16.913	25.045	16.913

version. This allows us to account for potential variations in the total number of questions answered by each participant between the different game versions. Descriptive statistics on these percentages can be found in table 1.

Before conducting a paired t-test to compare the differences in percentages, we needed to verify the normality of the data. We performed a Shapiro-Wilk test on the differences between the adaptive and non-adaptive percentage scores. The results produced a p-value of 0.942, higher than the usual significance level of 0.05. This result led us to the conclusion that the data is approximately regularly distributed.

The results of the t-test showed a significant difference in the percentage of correctly answered questions between the two versions $(t(29) = 9.579, p < 0.001)$. The mean difference in performance was 26.13% (95% CI: $[20.55, 31.71]$), with the adaptive version showing higher correctness compared to the non-adaptive version. While this does not necessarily imply improved learning outcomes (as the difficulty of questions answered is not accounted for), it could indicate increased efficiency of learning.

Next, we turn to the role of the adaptive system in player motivation. We do this by first looking at the total number of questions answered by participants in each version of the game. If this number is substantially larger for the adaptive version, this could imply a higher level of engagement with the adaptive version, and therefore a potentially more motivating experience for the participant.

Upon performing a Shapiro-Wilk test on the differences between the number of questions asked, we found a p-value of 0.023. This makes a t-test unsuitable for analysis as the data is not sufficiently normally distributed. Instead, we use a Wilcoxon signed-rank test. This indicated a significant difference between the medians of the adaptive and non-adaptive versions of the game $(Z = -3.511, p < 0.001)$, with the adaptive version having a higher median number of questions answered than the non-adaptive version.

Some results from the questionnaire also have implications for the apparent effectiveness of the adaptive system. All of the participants stated that they noticed one version of the game adapted to their skill level, while the other one didn't. This implies that the level of adaptivity was sufficient for users of all skill levels to notice. Furthermore, all agreed that they would choose to play the adaptive version of the game over the non-adaptive version. This shows that the difficulty adjustment was tuned such that players of all skill levels could have an enjoyable experience.

All participants also stated that they felt more motivated to answer questions in the adaptive version of the game. However, there was some variability in how strongly participants felt about this, with one participant only opting to ”Slightly Agree” with the statement. A similar pattern occurred when participants were asked whether the adaptive version of the game was a more enjoyable experience. The vast majority of participants either ”Agree” or ”Strongly Agree”, with one participant responding ”Neutral”. The modest relevance of this variation means we can treat it as an anomaly and draw the conclusion that participants found the adaptive questioning system to improve levels of motivation and enjoyment.

An area of slightly greater contention in the questionnaire was whether participants noticed an increase in difficulty as their streak of correct answers increased. Four participants responded ”Neutral” to this statement, indicating that further work may be required in determining a good weighting value for the streak feature.

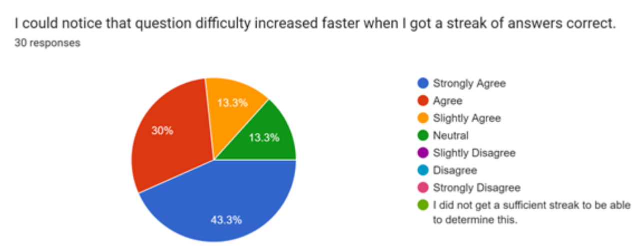

Fig. 14. Results from the questionnaire on how much participants noticed their streak count affecting the question difficulty. This is the area of the adaptive system with the least convincing results.

4.0.2 *Effectiveness of the Game as a Whole*

The questionnaire also contained questions relating to the overall effectiveness of the game as a learning tool. In general, responses show very positive feedback for the game, with very little data on the contrary. All participants either agreed or strongly agreed that the game interface was attractive and appealing and the learning contents were provided in a vivid way. This means that the virtual environment is successful in providing the user with an immersive experience, which has a positive impact on their motivation to answer questions.

Most participants agreed that little effort was required to understand the purpose of the game. This is beneficial to learning as it means the game loop is not distracting the user from focusing on cyber security. However, five participants responded neutral to the statement and one slightly disagreed. A common trait that all of these participants shared is that they had no past experience with virtual reality. A conclusion to draw from this is that the game features may be too overwhelming for users new to virtual reality.

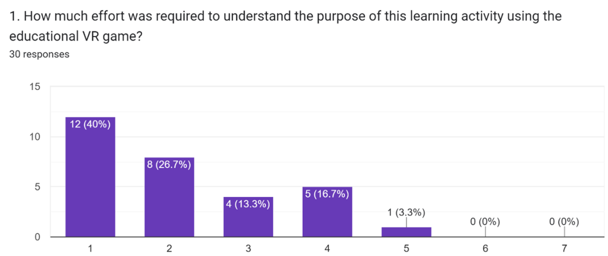

Fig. 15. Results from the questionnaire on how much effort participants thought was required to understand basic game functionality. 1 represents strongly agree and 7 represents strongly disagree. 4 means the participant is neutral to the comment.

All participants reported that they did not feel frustrated during or after their play session. Furthermore, at no point did any participant feel discouraged or stressed. This supports the idea that the user journey is easy to understand and suitable for a comfortable learning experience.

Turning attention towards the in-game shop, all participants at least slightly agreed to have found it to provide motivation to continue answering questions. The difference in whether participants only slightly agreed with this statement or strongly agreed followed no discernible trend. This could imply that the effectiveness of user-based customization may be largely down to personal preference.

Some participants also left text feedback to highlight certain aspects of the game. One participant wrote: "The shop encourages others to continue gameplay and learning. This is very motivational". However, in contrast, two other participants stated: "Quite easy to buy everything in the shop quickly so it might need more items/ be harder to get them." and "I would like more items to be available from the shop." The combination of these responses implies that while the shop feature is an effective motivator for learning, it may be necessary to increase the number of items available to maintain a good level of motivation.

Another trend in text feedback left by participants involved improving the look of the terrain in the answer worlds. This feedback came from users who had previous experience with virtual reality and therefore were more savvy to current VR graphics capabilities. Since immersion is a driving factor in learning motivation, it may be necessary to take note of this feedback.

5 Evaluation

We now reflect on our two research questions: (1) 'Can we present an adaptive questioning system suitable for use in digital educational games with short development cycles' and (2) 'Can we define a pedagogically effective VR educational game framework that can allow for more efficient experimentation of new technologies in the domain of VR educational games?'. We will critically discuss the suitability of our approach in addressing these research questions, the strengths and limitations of the system, and the organization of the project. This comprehensive evaluation will help identify areas for improvement and future research directions.

5.0.1 Suitability of Approach

In order to ensure the suitability of our approach in answering the research questions, we employed a combination of literature review, system development, and empirical testing.

The adaptive questioning system was designed using principles from the Elo-rating system (ERS), a thoroughly proven method of adaptivity in educational games. In extending the functionality of the ERS, we incorporated additional features and modifications that were tailored to the specific context of our educational game, while maintaining the core principles of adaptivity and accurate skill assessment. By conducting an extensive review of existing adaptive systems and learning theories, we ensured that our approach built on top of previous work in a meaningful way.

An argument against the suitability of our approach could relate to the dismissal of lesser-known adaptive systems in the literature. Although many adaptive systems were considered, most were dismissed based on the fact that they were shown to be less pedagogically effective than the more well-known systems such as the ERS. In hindsight, some of these systems could have been extended to achieve better learning outcomes if sufficient experimentation took place. Conversely, such experimentation may have been ill-suited to the time constraints of this project.

The VR game framework was designed based on an established educational game framework [8], as well as drawing on the successes and challenges faced by existing VR educational games. By conducting a comprehensive review of the uses of VR technologies in education, and educational game design, we ensured that our framework was grounded in current research and best practices. Additionally, we incorporated features and modifications that were tailored to the specific needs of VR educational games, enabling rapid integration of new technologies and facilitating more efficient experimentation in the domain.

Development of the VR game framework took an iterative approach in which testing took place after the completion of each module of the game. This suited the project well as an aim of the system was to be modular in nature, such that sections can be swapped out to allow for testing. Therefore, this testing process rarely became a hindrance, instead continuously reaffirming that we were making progress in answering the second research question.

One possible downside of this approach was the amount of time spent on user testing. While absolutely necessary in some cases - such as deciding on the value of a weighting term in our adaptive system - there may have been too much time spent on user testing during the development of the virtual environment. The time spent on this may have been better suited for implementing supervisor-based customization in the form of a question input system.

We designed an experimental study to determine the effectiveness of the VR educational game framework. We implemented suitable control measures and selected appropriate statistical tests to account for the specific characteristics of our study. Our experimental study not only contributed to the validation and refinement of our developed systems but also provided valuable insights that can inform the design and implementation of future educational games and adaptive learning systems in the context of VR and beyond.

An argument against the suitability of our experimental study design lies in its lack of a pre and post-test procedure. Without these, the effect of the adaptive system on learning outcomes was impossible to assess. However, implementing such tests would have required a complete redesign of the study, with participants no longer being able to play both versions of the game. This would make it harder to draw conclusions on the effectiveness of the game as a whole since participants would not have the same experiences with the system. Essentially, there is a trade-off between these two studies where valuable data is both lost and gained.

Another aspect of the experimental study to consider is the background information of the participants. A diverse participant pool with varied backgrounds can help ensure the generalizability of the study's findings and offer insights into how different users may interact with the system. Our experimental study included participants with diverse backgrounds in terms of gender, educational level, and prior experience with digital educational games and VR technologies. This diversity ensures that the results are more representative of the wider population. However, a possible source of bias in the study could relate to age groups.

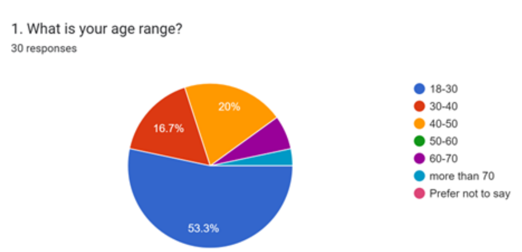

Fig. 16. Results from the questionnaire on the age range of participants. A significant

As can be seen in Fig. 16, over half of the participants involved in the study were aged 18-30. This raises concerns about how reflective the study is to people outside of that age group. Unfortunately, this factor must be considered when making any conclusions about the data.

5.0.2 Strengths of the System

1) **Successful Adaptive System**: Our extension of the Elo-rating system has been shown to successfully adapt to the skill rating of the user such that they are able to learn more efficiently and be more motivated to engage with the learning contents. The local updating system with preset question difficulty ratings allows for an immediately functional system, waiving the requirement of extensive data collection typical of an Elo-rating system. This makes the system suitable for projects with short development cycles that have little time for data collection or extensive implementation requirements.
2) **Modularity**: The VR game framework was designed with modularity in mind, allowing for the easy integration of new technologies and the ability to swap out different modules as needed. This flexibility enables more efficient experimentation and adaptation to advancements in the field.
3) **Grounded in Research**: Both the VR game framework and the adaptive questioning system were developed using established principles and best practices from relevant literature, ensuring that the system is well-founded in current research.
4) **Empirical Validation**: The experimental study conducted allowed us to validate the effectiveness of the system and identify small areas for improvement, contributing to the refinement and optimization of the system for future work.

5.0.3 Limitations of the System

1) **Scalability**: Although the system is designed for flexibility and modularity, there may be limitations in terms of scalability, particularly when considering the tedious task of entering large-scale question banks into the database. It is here that a question input system would be extremely valuable.
2) **Undertuned Streak System**: Although most participants of the experimental study were able to notice question difficulty increase as their streak of correct answers grew, a significant proportion were only neutral to the possibility of it. This implies that the weighting term w in the streak system might not have been set to an optimal level.
3) **Rapidly Evolving Field**: As VR technology and educational game design continue to advance at a rapid pace, maintaining the relevance and effectiveness of the system may require many ongoing updates and refinements. This is particularly an issue for VR hardware, which may not immediately be supported by the packages used to track player input.
4) **Limited Scope of Testing**: The experimental study conducted to validate the system primarily focused on the adaptive questioning system and the overall effectiveness of the VR game framework. Further research may be necessary to evaluate other aspects of the system, such as user journey, and long-term learning outcomes.

5.0.4 Organisation of the Project

One of the strengths of the project organization was the well-structured phases, including planning, literature review, system development, experimental study, and evaluation. This clear structure allowed the team to focus on specific tasks and objectives at each stage, ensuring efficient progress. The iterative testing methodology used in the project facilitated a flexible and adaptive workflow, enabling quick responses to new insights, challenges, and opportunities. This approach contributed to the timely completion of the project.

In terms of areas for improvement, time management could be further enhanced by more closely prioritizing tasks and monitoring progress. Improving time management practices could further streamline the project workflow and potentially accelerate the development process. Another area for improvement is documentation consistency. Although progress reports were maintained throughout the project, clear documentation of certain parts of the codebase were lacking, causing delays when attempting to return back to them after a long period.

6 CONCLUSION

This project has successfully explored the design, implementation, and evaluation of a virtual reality educational game framework that integrates an adaptive questioning system based on a novel interpretation of the Elo-rating system and user-initialized customization through an in-game shop. The resulting framework provides a foundation for the development of immersive, engaging, and personalized learning experiences that can cater to individual learners' needs and preferences.

The case study, Cyber Decider, served as a tangible example of the proposed framework in action, focusing on teaching users about cyber security through multiple-choice questions. The experimental study conducted to address the research questions demonstrated the effectiveness of the adaptive questioning system in enhancing users' learning efficiency and motivation to play. It also showed the instructional effectiveness of the framework as a whole, and at the same time demonstrated the fact that the framework is well suited to efficient experimentation of ideas.

The findings from this research contribute to the growing body of knowledge in the field of virtual reality-based education. The proposed framework and its successful implementation in Cyber Decider provide a solid foundation for future research to build upon, while the adaptive questioning system offers a promising approach to creating personalized learning experiences in VR educational games with small development cycles.

As with any research, this project has its limitations, and there are opportunities for further exploration and development. One potential direction of future research is to compare our extension of the Elo-rating system with a traditional, fully trained Elo-rating system. Such a comparison could allow for direct measurement of the level of personalization and difficulty rating accuracy lost or gained between the two versions. On the other hand, we could look at further improving the VR educational game framework as a whole. One example of this would be adding a question input system that streamlines the process of adding new questions to the database. This would add supervisor-based customization to the framework, opening up a new set of possible studies. Additionally, researchers might consider directly investigating the impacts of the proposed framework on learning outcomes and retention, as well as examining its effectiveness in different cultures and age groups.

In conclusion, this paper represents a step forward in the design and implementation of virtual reality educational games that adapt to individual learners and provide a personalized and engaging learning experience. As technology continues to advance, the potential for VR-based education to transform the way we teach and learn becomes increasingly apparent, and this research contributes to unlocking that potential.

REFERENCES

[1] M. Kordaki and A. Gousiou, "Digital card games in education: A ten year systematic review," *Computers & Education*, vol. 109, pp. 122–161, 2017.

[2] M. Ninaus and S. Nebel, "A systematic literature review of analytics for adaptivity within educational video games," in *Frontiers in Education*, vol. 5. Frontiers Media SA, 2021, p. 611072.

[3] D. R. Michael and S. L. Chen, *Serious games: Games that educate, train, and inform*. Muska & Lipman/Premier-Trade, 2005.

[4] D. Hooshyar, M. Pedaste, Y. Yang, L. Malva, G.-J. Hwang, M. Wang, H. Lim, and D. Delev, "From gaming to computational thinking: An adaptive educational computer game-based learning approach," *Journal of Educational Computing Research*, vol. 59, no. 3, pp. 383–409, 2021.

[5] R. Pelánek, "Applications of the elo rating system in adaptive educational systems," *Computers & Education*, vol. 98, pp. 169–179, 2016.

[6] N. Noah and S. Das, "Exploring evolution of augmented and virtual reality education space in 2020 through systematic literature review," *Computer Animation and Virtual Worlds*, vol. 32, no. 3-4, p. e2020, 2021.

[7] E. A. Boyle, T. Hainey, T. M. Connolly, G. Gray, J. Earp, M. Ott, T. Lim, M. Ninaus, C. Ribeiro, and J. Pereira, "An update to the systematic literature review of empirical evidence of the impacts and outcomes of computer games and serious games," *Computers & Education*, vol. 94, pp. 178–192, 2016.

[8] J. Wang, R. Hodgson, and A. I. Cristea, "Memorable: A multiplayer customisable serious game framework for cyber-security learning," in *Intelligent Tutoring Systems: 18th International Conference, ITS 2022, Bucharest, Romania, June 29–July 1, 2022, Proceedings*. Springer, 2022, pp. 313–322.

[9] S. Vanbecelaere, K. Van den Berghe, F. Cornillie, D. Sasanguie, B. Reynvoet, and F. Depaepe, "The effects of two digital educational games on cognitive and non-cognitive math and reading outcomes," *Computers & Education*, vol. 143, p. 103680, 2020.

[10] R. Mathew, S. Malik, and R. Tawafak, "Teaching problem solving skills using an educational game in a computer programming course. 18 (2), 359–373," 2019.

[11] A. Shi, Y. Wang, and N. Ding, "The effect of game–based immersive virtual reality learning environment on learning outcomes: designing an intrinsic integrated educational game for pre–class learning," *Interactive Learning Environments*, vol. 30, no. 4, pp. 721–734, 2022.

[12] K. Mangaroska and M. Giannakos, "Learning analytics for learning design: A systematic literature review of analytics-driven design to enhance learning," *IEEE Transactions on Learning Technologies*, vol. 12, no. 4, pp. 516–534, 2018.

[13] S. Vanbecelaere, F. Cornillie, D. Sasanguie, B. Reynvoet, and F. Depaepe, "The effectiveness of an adaptive digital educational game for the training of early numerical abilities in terms of cognitive, noncognitive and efficiency outcomes," *British Journal of Educational Technology*, vol. 52, no. 1, pp. 112–124, 2021.

[14] R. Levy, "Dynamic bayesian network modeling of game-based diagnostic assessments," *Multivariate behavioral research*, vol. 54, no. 6, pp. 771–794, 2019.

[15] S. Klinkenberg, M. Straatemeier, and H. L. van der Maas, "Computer adaptive practice of maths ability using a new item response model for on the fly ability and difficulty estimation," *Computers & Education*, vol. 57, no. 2, pp. 1813–1824, 2011.

[16] T. Drey, P. Jansen, F. Fischbach, J. Frommel, and E. Rukzio, "Towards progress assessment for adaptive hints in educational virtual reality games," in *Extended Abstracts of the 2020 CHI Conference on Human Factors in Computing Systems*, 2020, pp. 1–9.

[17] G. Makransky and L. Lilleholt, "A structural equation modeling investigation of the emotional value of immersive virtual reality in education," *Educational Technology Research and Development*, vol. 66, no. 5, pp. 1141–1164, 2018.

[18] N. Saleh, E. Prakash, and R. Manton, "Factors affecting the acceptance of game-based learning," *International Journal of Computer Applications*, vol. 92, no. 13, pp. 1–10, 2014.

[19] C. Ferguson, E. L. Van den Broek, and H. Van Oostendorp, "On the role of interaction mode and story structure in virtual reality serious games," *Computers & Education*, vol. 143, p. 103671, 2020.

[20] D. Checa and A. Bustillo, "A review of immersive virtual reality serious games to enhance learning and training," *Multimedia Tools and Applications*, vol. 79, pp. 5501–5527, 2020.

[21] A. Wang, M. Thompson, D. Roy, K. Pan, J. Perry, P. Tan, R. Eberhart, and E. Klopfer, "Iterative user and expert feedback in the design of an educational virtual reality biology game," *Interactive Learning Environments*, vol. 30, no. 4, pp. 677–694, 2022.

[22] O. Javora, F. Děchtěrenko, T. Tetourová, K. Volná, and C. Brom, "Customization in educational computer games and its effect on learning: Experimental study with primary school children," *Journal of Computer Assisted Learning*, vol. 37, no. 5, pp. 1370–1382, 2021.

[23] S. P. Rose, M. J. Habgood, and T. Jay, "Designing a programming game to improve children's procedural abstraction skills in scratch," *Journal of Educational Computing Research*, vol. 58, no. 7, pp. 1372–1411, 2020.

[24] K. Wauters, P. Desmet, and W. Van Den Noortgate, "Item difficulty estimation: An auspicious collaboration between data and judgment," *Computers & Education*, vol. 58, no. 4, pp. 1183–1193, 2012.

[25] M. Antal, "On the use of elo rating for adaptive assessment," *Studia Universitatis Babes-Bolyai, Informatica*, vol. 58, no. 1, pp. 29–41, 2013.

[26] J. Papoušek and R. Pelánek, "Impact of adaptive educational system behaviour on student motivation," in *Artificial Intelligence in Education: 17th International Conference, AIED 2015, Madrid, Spain, June 22-26, 2015. Proceedings 17*. Springer, 2015, pp. 348–357.