Informatics

SeminVR: A WebXR Public Speaking Trainer

DIPLOMARBEIT

zur Erlangung des akademischen Grades

Diplom-Ingenieur

im Rahmen des Studiums

Media and Human-Centered Computing

eingereicht von

Andreas Macsek, BSc
Matrikelnummer 51826020

an der Fakultät für Informatik

der Technischen Universität Wien

Betreuung: Ao.Univ.Prof. Dipl.-Ing. Dr.techn. Peter Purgathofer
Mitwirkung: Projektass.in Dr.in techn. Iana Podkosova, BSc MSc

Wien, 5. April 2024

Andreas Macsek	Peter Purgathofer

Informatics

SeminVR: A WebXR Public Speaking Trainer

DIPLOMA THESIS

submitted in partial fulfillment of the requirements for the degree of

Diplom-Ingenieur

in

Media and Human-Centered Computing

by

Andreas Macsek, BSc
Registration Number 51826020

to the Faculty of Informatics

at the TU Wien

Advisor: Ao.Univ.Prof. Dipl.-Ing. Dr.techn. Peter Purgathofer
Assistance: Projektass.in Dr.in techn. Iana Podkosova, BSc MSc

Vienna, April 5, 2024

Andreas Macsek   Peter Purgathofer

Erklärung zur Verfassung der Arbeit

Andreas Macsek, BSc

Hiermit erkläre ich, dass ich diese Arbeit selbständig verfasst habe, dass ich die verwendeten Quellen und Hilfsmittel vollständig angegeben habe und dass ich die Stellen der Arbeit – einschließlich Tabellen, Karten und Abbildungen –, die anderen Werken oder dem Internet im Wortlaut oder dem Sinn nach entnommen sind, auf jeden Fall unter Angabe der Quelle als Entlehnung kenntlich gemacht habe.

Wien, 5. April 2024

Andreas Macsek

Danksagung

Ich möchte mich recht herzlichst bei all jenen bedanken, die mich in meinem Werdegang unterstützt haben. Sei es im Zuge meiner Ausbildung und meines Masterabschlusses oder meiner persönlichen Weiterentwicklung. Ein ganz besonderer Dank gilt dabei meiner Familie und Freunden, die mich stets motiviert und in meinem Weg unterstützt haben. Sie haben mich immer ermutigt, meinen Weg zu gehen und das Beste aus mir zu machen. In diesem Sinne möchte ich jedoch noch speziell meine liebe Freundin hervorheben, welche mich auch in den stressigsten Zeiten immer ausgehalten hat und mich mit allen ihr möglichen Mitteln unterstützt hat. Ein weiterer Dank gilt außerdem meiner phänomenalen Betreuung, ohne die meine Arbeit nie so verlaufen hätte können. Sei es in Bezug auf die vielen Ideen und die eingebrachte Expertise meiner Betreuer, Peter Purgathofer und Hannes Kaufmann, oder die vielen Stunden, die meine technische Ansprechpartnerin, Iana Podkosova, geopfert hat, um mich in meiner Arbeit zu unterstützen. Ihre kontinuierliche Hilfe und Unterstützung waren essenziell für meinen Fortschritt, welcher mir zu diesem Ergebnis verholfen hat. Vielen Dank dafür!

Acknowledgements

I want to thank all those who have supported me in my career. Be it during my studies and Master's degree or my personal development. Special thanks go to my family and friends, who have always motivated and supported me. They have always encouraged me to go my own way and become the best version of myself. With this in mind, I would like to especially mention my dear girlfriend, who has always put up with me even in the most stressful times and supported me in every way possible. I would also like to thank my phenomenal advisors, without whom my work would never have progressed as it did. Be it in terms of the many ideas and expertise provided by my advisors, Peter Purgathofer and Hannes Kaufmann, or the many hours my contact for technical matters, Iana Podkosova, sacrificed to help me in my work. Their continuous assistance and support were essential for my progress, which helped me to achieve this result. Thank you very much!

Kurzfassung

Präsentationstraining ist eine essenzielle Methode zur Vorbereitung auf das Sprechen vor Publikum. Dennoch wird es hauptsächlich privat, ohne Gesprächspartner, geübt. Entsprechend bereitet man sich mehr darauf vor, seinen Inhalt zu präsentieren, als diesen einem Publikum zu übermitteln. Aus diesem Grund haben sich Virtual Reality Präsentationstrainer etabliert, welche diesen Übungsprozess unterstützen sollen und das Üben mit einem virtuellen Publikum ermöglichen. Nichtsdestotrotz wurden diese Anwendungen oftmals nur für eine spezifische Plattform entwickelt, wodurch deren Nutzung auf bestimmte Virtual-Reality-Brillen beschränkt ist.

Um diesem Problem entgegenzuwirken, haben wir einen plattformunabhängigen Präsentationstrainer entwickelt, der sich SeminVR nennt. Basierend auf dem Web Standard WebXR ermöglicht dieser die Verwendung solch eines Trainers über das Internet und somit unabhängig von einer bestimmten Plattform. Dementsprechend ist dessen Nutzung mittels verschiedensten Virtual-Reality-Brillen möglich.

Da ähnliche Anwendungen bereits gezeigt haben, dass die Vorbereitung mit Präsentationstrainern hilfreich ist, befanden wir es als notwendig herauszufinden, ob dies auch für unseren Trainer der Fall ist. Zu diesem Zweck haben wir eine Between-Subject Studie durchgeführt, welche sich auf die qualitative Analyse der Erfahrungen unserer Teilnehmer fokussierte. Mit dieser Studie versuchten wir primär herauszufinden, ob Performanzunterschiede zwischen verschiedenen Plattformen die Wirksamkeit unserer Anwendung beeinflussen. Dementsprechend haben wir eine Herangehensweise entwickelt, welche auf Basis einer Kontrollgruppe die allgemeine Wirksamkeit unserer Anwendung ausfindig machte und so den Vergleich mit einer Experimentgruppe ermöglichte, welche die Trainingsanwendung auf verschiedenen Plattformen verwendete.

Unsere Ergebnisse zeigen, dass unser Präsentationstrainer ebenfalls als eine effektive Trainingsmethode empfunden wurde. Wir konnten jedoch feststellen, dass Performanzunterschiede zwischen Plattformen die Nützlichkeit beeinflussen können. Insbesondere in Fällen, in denen der Komfort aufgrund von trägem Verhalten der Anwendung beeinträchtigt wurde. Jedoch waren nur wenige Teilnehmer der Experimentgruppe davon betroffen. Die meisten jedoch konnten die Unterschiede ausblenden und das Training wie auf einer leistungsfähigeren Plattform verwenden. Daraus schließen wir, dass SeminVR eine gute Trainingsmöglichkeit bietet und befinden, dass WebXR eine hilfreiche Lösung für die Entwicklung von Multi-Plattform Präsentationstrainern bietet.

Abstract

Presentation training is an essential method to prepare for talking to an audience. Nevertheless, it is mainly done privately without someone to speak to. Respectively, people primarily train to get confident with their presentation's content but do not prepare themselves for talking to an audience. Therefore, virtual reality presentation trainers have been developed to enhance preparation and allow people to train with a virtual audience. However, those applications are usually created for a single platform only, which restricts their usage to a specific type of virtual reality headset.

To counteract this issue and facilitate access to such training applications, we developed our multi-platform presentation trainer, SeminVR. Based on the web standard WebXR, it allows access to our training experience via the web, making it independent from any platform.

Since other presentation trainers already showed that preparation with such trainers is beneficial, we had to ensure the same applies to our application. Therefore, we conducted a between-subject study focusing on a qualitative analysis of user experiences with SeminVR. With this study, we mainly tried to identify whether the computational differences between platforms impact our presentation trainer's usefulness. Consequently, we created an evaluation approach utilizing a control group to determine the general effectiveness of our application and compare their results to an experimental group experiencing the trainer on different performing platforms.

Our results showed that our presentation trainer was perceived as an effective training method. However, we could identify that performance differences can influence its usefulness in cases where comfort is reduced due to the lagging behavior of the application. Nevertheless, only some experimental group participants had issues with weaker performance. As for the rest, they could blend them out and enjoy training like running on a more computationally powerful platform. Therefore, we believe that SeminVR provides a great training possibility and decide that WebXR offers a good solution for developing multi-platform presentation trainers.

Contents

CHAPTER 1

Introduction

Speaking to an audience is not equally comfortable for everyone. Some people are confident talking to others, while others would rather stay quiet. Being uncomfortable talking to an audience significantly impacts people's day-to-day situations, as holding presentations and giving talks to others is common practice in various professions, education, and even private life. Fearing not performing well and running into mistakes often causes feeling less confident talking to an audience, making people insecure and tending to avoid these situations. However, it is possible to overcome this fear and gain more confidence by preparing for those circumstances.

Training to present and getting comfortable with the content one wants to inform about is an excellent way of gaining more confidence for talking to an audience. Nevertheless, individual preparation is usually done privately and fails to replicate a feeling similar to talking to real people. Respectively, this kind of preparation allows one only to get more confident with one's content but does not appropriately prepare for speaking to groups of people [BSBW23].

With Virtual Reality (VR) becoming increasingly relevant, various studies [BSBW23, TCB+19, WKH21] focused on building training applications that can prepare for situations that are usually hard to train. Due to VR's image realism allowing people to immerse themselves in the virtual world, those training applications can replicate feelings comparable to real situations. Respectively, they allow preparation for lifelike scenarios in a safe and simulated environment while leaving trainees in control to stop the experience at any time if feeling uncomfortable.

Benefiting from this realism and the possibility of creating a virtual training scenario, different studies focused on developing tools for training presentations in such safe environments [BSBW23, PCPK19, TCB+19]. Since training with an actual audience is usually not feasible, those applications benefit from VR's possibility to create virtual people, often called agents, that imitate a real audience. Therefore, those trainers allow for a more complete preparation experience, enabling interaction with people while training one's presentation content. Accordingly, those studies showed that using a VR environment in preparation for training presentations positively impacts people's performance and increases their comfort while talking to an audience [BSBW23].

Although various presentation trainers were developed in the past focusing on different aspects of training [BSBW23, PCPK19, TCB+19, GKR20], they have one thing in common. While the availability of training is a vital factor to consider when developing such applications, they are usually created for a single platform only. Since many platforms differ in their standards and capabilities, creating a single application that works for all of them is challenging. Therefore, most studies choose to develop a trainer for a specific platform. However, this makes those applications less attractive to potential users as they restrict them to a particular platform. Nonetheless, a technology has emerged in recent years that allows us to overcome this issue and develop a single application for multi-platform use.

Utilizing the WebXR API, it is possible to create web-based VR experiences that can be consumed from various devices using a web browser, independent of their platform. While this allows any Augmented Reality (AR) and VR headset to experience immersive content over the web, it makes the application also available to smartphones, tablets, and other devices that can run a web browser. Different kinds of projects have already made use of these capabilities for educational purposes to enable conducting lab experiments off-site [TSM+22] or for treatment intentions, to remotely help students with their mental health at times when COVID-19 restricted access to supporting institutions [HEF21]. However, none focused on creating a Virtual Reality Public Speaking Trainer (VRPST).

1.1 Aim of Work

The aim of this thesis is to build a VRPST that we further call SeminVR. It is intended to provide scholars with a presentation training experience in a university-like setting. Therefore, the trainer should imitate a seminar room-like setting, including an audience, to establish a realistic preparation scenario. In this regard, we will focus on developing a sophisticated audience setup to imitate life-like behavior and transport an immersive training experience.

Unlike other existing applications, SeminVR should run on various platforms independently of their capabilities. Respectively, we want to develop it using the WebXR standard to run the application on the web and enable access for any VR headset. Consequently, various devices can access the application over a web browser, thus improving the availability of such training approaches.

Previous training applications have shown that virtual presentation training is beneficial for preparation. For this reason, we want to identify if the same applies to SeminVR and if there are any differences in its effectiveness depending on various platform capabilities. Therefore, a between-subject study should allow us to identify our trainer's effectiveness on computationally high- and low-performant platforms. Respectively, it should indicate if we developed a successful presentation trainer with SeminVR and whether WebXR is beneficial when developing similar training applications.

Seeking to explain SeminVR's usefulness and understand if performance influences users, we formulated the following research questions:

RQ1: Does SeminVR show a positive training effect on people?

RQ2: Does the feeling of presence correlate with a platform's performance while running SeminVR?

RQ3: Does an increased feeling of stress correlate with a higher level of perceived presence?

1.2 Structure of the Thesis

The thesis initially provides an overview of existing VRPST applications from different areas in Chapter 2. Alongside explaining their different focus points, we elaborate on their effectiveness and findings that influenced the development of SeminVR. Additionally, a description of the WebXR API provides a general understanding of its benefits over conventional developed single-platform applications. Afterward, we discuss the established requirements and design for SeminVR in Chapter 3, detailing its essential components and the decisions involved in creating the presentation trainer. Moreover, we elaborate on our evaluation approach, by conducting an intermediate pilot study to fine-tune SeminVR and evaluate its effectiveness in a final examination. Next, a detailed description of our implementation is provided in Chapter 4 to give more insight into how our presentation trainer works. Therefore, all its components are discussed from the virtual environment to the virtual agents over to the slide integration and interactions. Subsequently, the final study and its findings are elaborated in Chapter 5, describing how we gained insight using Thematic Analysis (TA) and what we found with it. Afterward, those findings are used to address the research questions and discuss the effectiveness of SeminVR in Chapter 6. Lastly, everything is summarized in the conclusion Chapter 7, providing an overview of our thesis results and potential future research directions.

CHAPTER 2

Related Work

This chapter overviews state-of-the-art presentation trainers and their application areas to give a better understanding of their usefulness. Therefore, we initially discuss how such trainers are used for fear treatment purposes by applying them as exposure therapy. Subsequently, we examine current VRPSTs and describe their approaches to improving presentation preparation for the broad mass. Afterward, we describe the WebXR API and its benefits in creating web-based VR experiences. Therefore, this chapter should create a foundational understanding of virtual presentation trainers and their importance alongside describing the technology to integrate a multi-platform solution and why one should do so.

2.1 Public Speaking Training for Fear Treatment

People uncomfortable talking to others and fearing to present has been the topic of various studies already[REC+21, EMM20]. Those often focus on potential ways to support them, mitigate their fear, and help prepare for such uncomfortable situations. Therefore, they utilize a common practice known as exposure therapy. The concept of this therapeutic method is to repetitively expose someone to a fear-inducing stimulus to reduce their response intensity, thus helping them get more comfortable in those situations. However, such repetitive exposures are often hard to achieve as they require an audience to train with [REC+21].

Therefore, studies like Reeves et al. [REC+21] analyzed the applicability of VRPSTs to reduce the need for in-vivo therapy. Their study investigated the usefulness of 360° VR videos for treatment purposes and sought to understand whether the video content makes any difference. Therefore, they compared three settings: 360° audience video, 360° empty room video, and no treatment. Using a within-subject analysis, they showed that anxiety levels across all 51 participants significantly reduced when exposed to their

VRPST video approaches. Thus, they could confirm that virtual exposure therapy is an effective method for treatment.

Similarly, Emmelkamp et al. [EMM20] reported equivalent results, analyzing various VRPSTs and reviewing their findings. However, they argue that further research is required to compare the efficacy of virtual trainers as stand-alone treatment options since many studies combined exposure therapy with cognitive interventions. Therefore, they argue that those study's results might not exclusively report the effects of VR treatment and must investigated carefully.

Alongside reporting the positive treatment results of such applications, studies highlight their benefits as a cost-efficient way to increase therapy accessibility. While therapy commonly relies on in-vivo sessions, VR exposure therapy can be used on-demand and from anywhere. Respectively, they can provide a great alternative to regular treatment and show the importance of additional research in this direction [REC+21].

2.2 Virtual Reality Presentation Training

As for the public speaking trainers, they share the same goal as VR exposure therapy, increasing people's confidence in talking to an audience. Nevertheless, their focus is slightly different. While exposure therapy is dedicated to providing an experience targeting people with increased fear symptoms, trainers aim to provide a generally applicable tool for everyone. However, their concepts overlap in that they each try to replicate a scenario with a virtual audience transmitting a similar feeling as talking to actual people.

An example of a more generally applicable approach is the work of Takac et al. [TCB+19], which tried to identify the effects of different sizes of audiences and habituation developments when training with VRPSTs. Therefore, they designed an experiment evaluating three environments in a within-group study featuring 19 participants of various ages between 18 and 76. Their environments consisted of different room settings and audiences, creating the feeling of different sizes of meetings or talking at a conference. Respectively, each room contained various amounts of agents, who also behaved differently depending on the room size. To replicate different levels of attention in an audience, some of their agents were designed to be distracted by things like their phones, while others were constantly listening to the speaker. For their evaluation, they had each participant present in all environments and give a six-minute talk in front of every audience size. To ensure that the difficulty of presenting to different large audiences does not influence their results, they decided to randomize the order of conditions for every participant. Their findings confirmed that their application did replicate a fearful stimulus similar to public speaking distress. Additionally, they concluded that repetition of training in such an environment is essential to achieve habituation effects and allow for better preparation. Therefore, they argue that it is essential to consider multiple rounds of training exposure when evaluating similar applications.

Palmas et al. [PCPK19] proposes an approach to additional improve training with

VRPST by adding direct feedback to them. In their study, they developed an application consisting of a small meeting room with three agents listening to one's presentation. Unlike other applications, they enhanced the presenter's field of view by adding informational panels giving feedback on their performance. As shown in Figure 2.1, those panels contained compact icons highlighting the potential adaptions to improve the presentation. These indicators represented improvement possibilities by informing the presenter to change their talking speed, focus more on the audience, and adapt to how loud they speak. Therefore, they wanted to support trainees by allowing them to improve their performance as they go. To evaluate the effectiveness of their direct feedback during training, they conducted a between-subject study with 200 undergraduate students divided into two groups. One group used the presentation trainer without any additional information displayed in the user's field of view, while the other group used the same trainer with additional direct feedback added. After presenting for five minutes, all participants had to answer questions concerning the acceptance of technology alongside how they perceived the experience in terms of usefulness and ease of use. Their findings concluded that adding elements for direct feedback in presentation trainers significantly benefits users and improves their training experience. Additionally, they identified a higher level of technology acceptance for their trainer with the added feedback mechanism, indicating further that the gamification of such trainers is practical.

Figure 2.1: Presentation trainer containing direct feedback by Palmas et al., 2019 [PCPK19]

Different from the previous approaches, Bachmann et al.[BSBW23] focused on evaluating the effectiveness of training with a VRPST compared to conventional preparation for presentations. Since they identified that many studies lack evaluation of their developed presentation trainer, they wanted to determine if the positive aspects reported correlate also with better results when judged by an unknowing audience. Respectively, they designed a between-subject experiment with 42 participants randomly allocated to either prepare for a presentation in private or be supported by their VRPST. To increase comparability, they required all their participants to train for the same presentation and allowed preparation for only a limited amount of time. In their evaluation, they both focused on participants' individual perceived performance and assessment of the audience to see if any correlation exists between those. While participants using the virtual trainer reported higher perceived performance levels, they also identified them as performing better according to the audience. Therefore, their findings further support the understanding of VRPST's positive training effects and show that preparation with them is beneficial.

2.3 WebXR

Since multi-platform development usually requires additional work to build multiple variants of the same application, it is not considered feasible for small-scale projects. Mainly due to increased maintenance and required distribution to separate stores, it becomes clear why studies usually focus on single-platform development. Nevertheless, there exists a potential solution to these problems by developing applications as web-based VR experiences by utilizing the *WebXR Device API*[1] in conjunction with *WebGL*[2]. The latter is one of the most common ways to render 3D graphical content on the web. It acts as the interface that allows web browsers to access the system's GPU and render an application's visual content in the browser. Meanwhile, WebXR provides the standardized logic to communicate with different types of AR and VR headsets and handle their connection with the browser. Accordingly, it handles receiving the headset's input information about the device's position and rotation (pose) and translating this information to display the experience appropriately. Therefore, those APIs provide a standardized way to develop Augmented Reality (AR) and VR content to be distributed and consumed via web browsers, which can be accessed by different types of headsets. Thus, it is possible to develop a single application that can be used across multiple platforms, which requires only to be distributed on a website.

2.3.1 Benefits and Drawbacks

Availability One of the most significant benefits of developing a WebXR-based application is its increased range of availability. Since it runs on the web, it can be accessed from various devices using a WebXR-enabled browser and is, therefore, not limited to a specific platform. Additionally, being separated from platform-specific distribution makes the application potentially more future-proof, as it is independent of platform changes and solely relies on the continuous support of the standardized WebXR API. Moreover, it needs to be highlighted that these experiences are not restricted to solely consuming them via AR and VR enabled devices. They also allow mobile phones, tablets, and desktop computers to use the same application and display them on a conventional 2D screen. Although this might not be as immersive, it enables additional ways to use the developed experience as not everyone owns a VR headset [TSM+22].

[1] https://www.w3.org/TR/webxr/
[2] https://www.khronos.org/webgl/

Browser Support Since WebXR is still relatively new, only some browsers support its full capabilities. While most support the WebXR core functionality required to allow consuming VR content over the web, not all support features like hand input and anchors. While the former feature allows benefiting from hand-tracking within such applications, anchors are an essential part of enabling AR experiences as they allow placing and tracking objects in space. Unfortunately, not all browsers support those capabilities, so it is necessary to consider using a specific browser when accessing WebXR content. Consequently, the possibilities for developing complex applications intended for multiple platform use are also restricted, as different browser support needs to be considered during development [ZA21]. A reduced version of the WebXR browser compatibility matrix can be found in Table 2.1, containing a few example browsers and features.

Feature Name	Chrome	Safari on visionOS	Meta Quest Browser	Microsoft Edge
WebXR Core	Chrome 79	Behind a feature flag	7.0, December 2019	Edge 87 on Windows Desktop Edge 91 on Hololens 2
WebXR AR Module	Chrome for Android, 81		24.0, October 2022	Edge 91 Hololens 2 only
Hand Input		Behind a feature flag	7.1, December 2019	Edge 87 on Windows Desktop Edge 91on Hololens 2
Anchors	Chrome for Android, 85		24.0, October 2022	Edge 93 Hololens 2 only

Table 2.1: WebXR browser compatibility matrix

Performance Another aspect to consider while developing WebXR applications is the performance differences between differently powerful end devices. Since the developed experiences can run on any device supporting a WebXR-enabled browser, the number of supported devices ranges from mobile phones and tablets to standalone headsets and computer-powered VR devices [ZA21]. Therefore, it is crucial to consider which devices to target while developing an application as they define the baseline for optimization. Consequently, keeping the most limiting platform in mind is essential when aiming to create an application usable across multiple AR and VR devices. Otherwise, compatibility across platforms might not be ensured, defeating the purpose of the multi-platform application.

2.3.2 Development Environments

Since WebXR is a web-based technology, a common approach to develop applications for it is using frameworks like A-Frame[3] that are built on a combination of JavaScript and HTML. With this framework, it is possible to create virtual sceneries using straightforward code lines to add various kinds of objects. For example, to create a simple box object as shown in Figure 2.2, the code from Listing 2.1 would be necessary. However, this process might quickly become cumbersome when one is thinking of developing more advanced and complex environments. For these cases, visual editors offer many benefits to adapt environments quickly and provide immediate visual feedback on the final result.

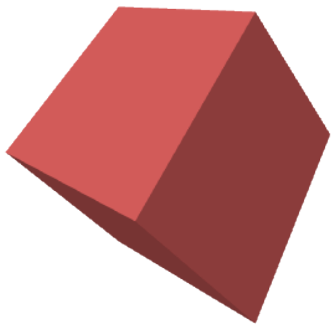

Figure 2.2: A-Frame box example scene

```
<a-scene>
  <a-box color="red" position="0 2 -5" rotation="0 45 45" scale="2 2 2"></a-box>
</a-scene>
```

Listing 2.1: A-Frame box example code

A great tool offering such a visual editor, often used in game development, is Unity[4]. The game engine stands out for its advanced capabilities when creating 3D environments and can also be used for developing WebXR applications. However, Unity does not natively support WebXR. Although it allows the creation of WebGL projects, it lacks the integration for creating WebXR experiences out of the box. Nevertheless, this can be fixed by using the Unity WebXR Exporter Package[5] provided by De-Panther, as recommended on the Immersive-Web website[6]. The package offers templates for exporting WebGL

[3] https://aframe.io
[4] https://unity.com/
[5] https://github.com/De-Panther/unity-webxr-export
[6] https://immersiveweb.dev/#unity

projects with WebXR support and contains the necessary building blocks for creating an immersive experience that can be used on the web.

Creating a VRPST including a virtual audience requires complex logic and animations to make them appear life-like. Since we had already developed applications with virtual agents in Unity, we knew how to make life-like behavior work using the game engine. Differently, we did not know how to integrate easy and smooth animation behavior for A-Frame and had trouble identifying ways to implement our audience's behavior. Consequently, we developed our presentation trainer using Unity, which seemed more versatile and better suited for our use case.

CHAPTER 3

Methodology

This chapter explains each facet we want to integrate with our presentation trainer and how to confirm its applicability. Therefore, we initially describe all the requirements and ideas that formed the baseline for developing SeminVR. Respectively, we elaborate on what look we wanted to achieve, what feelings it should transmit, and what interactions we considered essential. Alongside these descriptions, we explain our pilot study conducted during the application development and discuss the findings that further improved our final presentation trainer. Subsequently, we discuss our final study design. Therefore, we describe all our choices made to evaluate the effects of platform differences on participants' perceived usefulness and how we aimed to identify if SeminVR's training benefits users.

3.1 Requirements

Since we wanted to understand how scholars perceive training presentations in a VE, we aimed to design an application appropriate for them. Respectively, our objective was to create an experience that resembled an environment familiar to our target group, allowing them to see parallels to an actual situation and increase the likelihood of immersion. Therefore, we wanted to build a seminar room setting that is typical for universities.

As our goal was to allow training with an audience, we wanted to integrate virtual agents replicating the lifelike behavior of an attentive group of people. They were intended to imitate a similar feeling to giving an actual presentation while transporting the message of being in a safe training environment. Respectively, we targeted our application design to follow a realistic-looking cartoon-like art style to increase comfort and avoid uncanny valley feelings.

Nevertheless, all those aspects had to be carefully designed while constantly considering the application's performance on computationally low-performant devices. Therefore, we used Frames per Second (FPS) as a performance-indicating metric and tried optimizing each aspect of our application to match our target platform's capabilities best.

Based on all those characteristics, we formulated a couple of requirements necessary to fulfill while developing SeminVR:

Run on multiple platforms (R1): To improve the availability of presentation training, SeminVR should run on various platforms. Therefore, it must not be developed to be restricted to a particular headset but instead follow an approach to creating a multi-platform-supported solution. Respectively, an appropriate development approach must be selected to implement such an application while maintaining a low development effort.

Ensure low-end device support (R2): Alongside making SeminVR run on various devices, we must ensure it runs smoothly across platforms. Therefore, we should optimize our application to match the lowest-performing device we want to support. This should allow for a seamless application experience across all devices, thus ensuring a usable tool independent of the platform.

Allow training with an audience (R3): We should provide a virtual audience to simulate the preparation for an actual presentation. Nevertheless, the audience should transmit the feeling of a comfortable and safe environment so as not to make users afraid of training. Therefore, we must replicate a calm and attentive group of people imitating human behavior. Additionally, it needs to be ensured that the agents are not looking too realistic to induce uncanny feelings, which could conflict with creating a comfortable preparation scenario.

Provide a university-like training setting (R4): The application's environment design should target scholars' expectations to allow for higher levels of immersion. Therefore, the training setting should replicate a university-like seminar room that is familiar to them. Thus, it should provide a class-like setting containing table rows and a lecturer's table in the front. Additionally, all the necessary tools to hold a presentation must provided in the room, thus allowing it to replicate feelings similar to a natural environment.

Allow training with your presentation slides (R5): SeminVR should allow users to train with their presentation materials. Therefore, we must provide a way for users to access and display their presentation slides on a virtual monitor. Those slides should also be projected on the wall for the audience to see, allowing trainees to feel like they are in an actual seminar room. Furthermore, an interaction method must be supported to allow users to navigate between their slides and move along as they like.

3.2 Application Design

SeminVR aims to provide a virtual presentation training experience intended for scholars. Respectively, it is designed to replicate a seminar environment collectively known among the target group. Imitating a familiar setting is essential for users to immerse themselves better in the experience and benefit the most from training their presentations.

To provide an appropriate training environment, SeminVR features a group of agents who form the audience and listen to users. Throughout the experience, these agents imitate human-like behavior by looking around and changing their seating poses. Additionally, they get distracted occasionally and transport the feeling of losing interest, like an actual audience would. Thus, they allow trainees to prepare to talk to groups of people and get more confident doing so.

Similarly essential as providing an audience, SeminVR allows trainees to prepare with their own presentation slides. Therefore, it implements a logic to integrate the user's materials and display them in the VE. Respectively, they can see the content on a virtual screen and a projection on the wall behind them. Having those slides virtually allows users to progress through their materials using virtual controls and train their presentation appropriately. To summarize the application's components, we provide an overview in Figure 3.1 containing the most essential aspects of SeminVR.

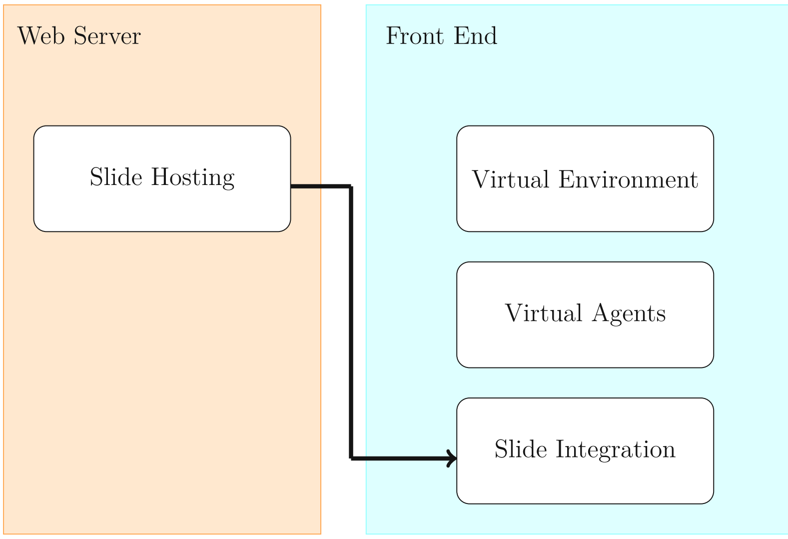

Figure 3.1: SeminVR components overview

The following sections elaborate on the three core design aspects of SeminVR in more detail. They start with the virtual environment and how it was designed to replicate a realistic experience. Subsequently, a description of the agents imitating the virtual audience and the behavior they mimic is given. Lastly, the design concludes by elaborating on the slide integration and interactions within the virtual world necessary to use the presentation trainer.

3.2.1 Virtual Environment

To make SeminVR's environment best replicate scholars' experiences with giving presentations at university, we wanted to build a seminar room that looked similar to those typical at Austrian universities. Respectively, it should allow trainees to see parallels to real scenarios and increase their immersion while training (R4). Consequently, we created a design concept of the room, as shown in Figure 3.2, on how we intended to build our environment. In this design, we depict the core layout of the room as well as illustrate details necessary to support the university-like feeling.

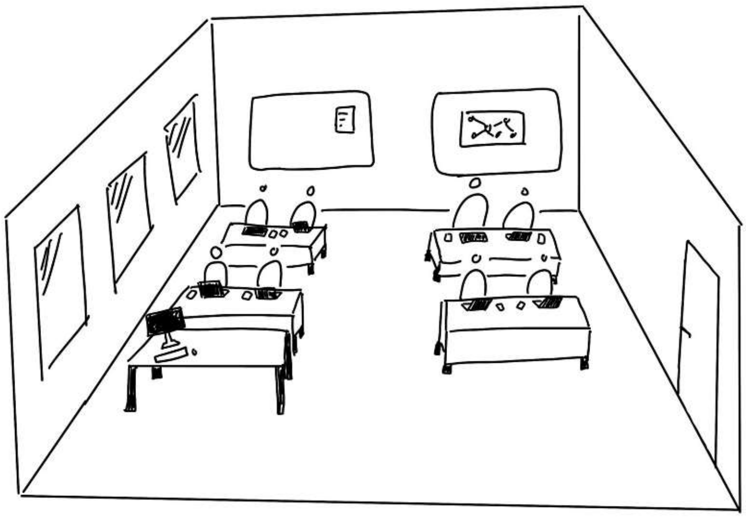

Figure 3.2: SeminVR's room concept

Like other VRPST environments [BSBW23, TCB+19], we wanted to build a room that was not too big in size. Although studies like Takac et al. [TCB+19] indicate that larger environments create a more challenging training situation for users, we decided to pick a more comfortable setting for SeminVR. Respectively, we decided to have eight virtual people forming our audience, which listens to trainees' presentations.

With an audience this size, we create a realistic feeling of holding a presentation in a seminar setting while not being too small to make it feel improbable. Accordingly, trainees should be able to prepare to interact with an actual audience and see parallels to situations they have been in.

To further enhance the realism of the room setting, each audience member should have a notebook and phone in front of them. Having those items next to them should transmit the feeling of people taking notes during the presentation and occasionally getting distracted by their phones.

Aside from the details within the audience, the room needs to contain additional enhancements not to make it look sterile. Therefore, we wanted to add a projector hanging from the roof that informs trainees that their presentation slides are projected for the audience to see. This would allow them to interact with the projection and encourage them to make references while presenting. Furthermore, some walls should have marker boards to evoke the feeling of being in a university room, as they are typical for such a setting.

The seminar room should have windows on one side to increase the depth of the environment and depict it as being surrounded by an urban environment. To further enhance this ambiance, we decided to add a matching soundscape, conveying the feeling of a city. Therefore, noises like people talking on the street and cars going by should provide the atmosphere of a busy outdoors and not have trainees placed in a completely silent environment. Consequently, it was equally important to have a matching indoor atmosphere. Therefore, we wanted to add ventilation noise, footsteps outside the room, and closing doors. Respectively, it should increase the feelings of realism and allow for a better training experience.

Although many small details were essential to making the environment appear natural, we constantly considered what parts were crucial in supporting realism and which details we could omit. Since the computational capabilities of our target platforms were limited, we had to reduce details to a minimum while not compromising realism. Otherwise, it would have resulted in our presentation trainer not running smoothly, which could have affected its usefulness. Therefore, we kept environmental objects to a minimum and made them look simplistic to ensure higher performance across our target platforms (R2).

3.2.2 Virtual Agents

As a central piece of SeminVR, the virtual audience consists of eight agents imitating human behavior (R3). They are meant to behave like an actual presentation audience in that they listen to the talk and react to certain events. Their visual fidelity should be low and replicate a cartoon-like look to avoid making them feel uncanny. Otherwise, users might get irritated looking at them, defeating the purpose of appropriate presentation practice.

To avoid making the audience look static and indifferent in behavior, agents should each be acting individually. Accordingly, they must regularly transition between different poses throughout the training to give them a distinct look. Additionally, their gaze needs to wander around the room to replicate the individual level of interest in certain things, as shown in Figure 3.3. Be it the trainee talking about their presentation or distracting objects like phones and notebooks. To make an agent's behavior look unique, their actions should rely on predefined traits related to their likelihood of being distracted by the previously mentioned objects. Therefore, the probability of their attention being drawn away by those things should be described using distraction types. Using those types, it should be possible to define how regularly an agent looks toward either point of interest, thus making them appear more or less attentive.

Additionally, the likelihood of attention drifting away during training should increase the more prolonged the presentation. Respectively, agents will behave less attentively over

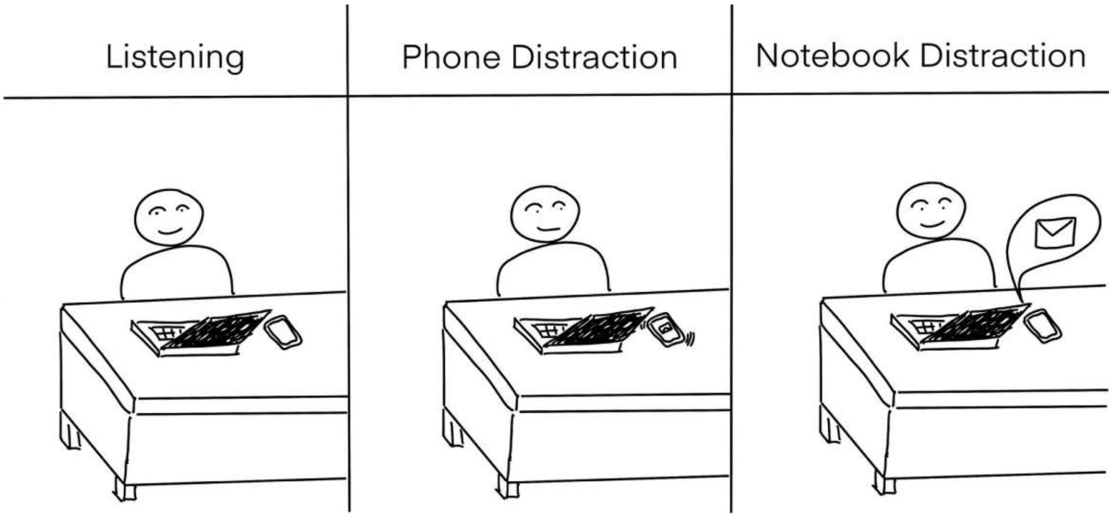

Figure 3.3: Agent attention behavior

time as they get more distracted. Thus making them appear as exhausted the longer the training takes.

The agent's traits, like distraction type and distraction level, should be pre-formulated and stored in presets to ensure all audience participants behave uniquely. Consequently, those presets allow for the same level of behavior for every training session, keeping the intensity of the preparation controllable.

Nevertheless, the presets should not always be applied the same way as users might otherwise observe the agent's behavior as a fixed sequence of actions. Therefore, at the start of each training, they should be randomly applied to agents. Thus, each audience member appears slightly different while maintaining the same level of behavior.

Further details on the design of the virtual agents and how it evolved during the integration of the presentation trainer are described in Chapter 4.

3.2.3 Slide Integration and Interactions

One of the most essential aspects to consider when designing an application is how users intend to use it and what they might expect. Since we want to keep the cognitive effort as low as possible, providing simplistic interactions is the way to go. Therefore, actions within SeminVR should be mainly based on touching interactable objects. Furthermore, we do not want to overwhelm users with extensive interaction methods. For this reason, we only provide the most essential interactions necessary for training presentations. These include loading presentation slides, as described in Figure 3.4, and navigating between them (R5).

To load the slides, a way to enter a file location within the application to communicate where those are stored is required. Therefore, it is necessary to implement a text input

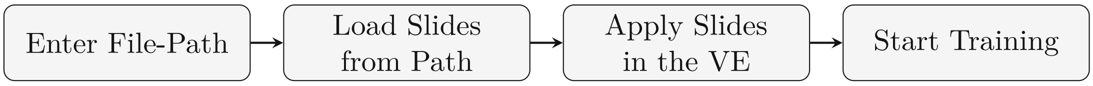

Figure 3.4: Slide integration flow

to inform the system about their location and load the slides accordingly. After loading, they get displayed on a virtual monitor in front of users and as a projection behind them for the audience to see. Thus, it should transmit the feeling of being in a seminar room and allow training like being in one.

To appropriately interact with the presentation, users need a navigation system. Therefore, we provide a virtual panel on the presenter's table featuring buttons that need to be pressed physically. Respectively, they should replicate a feeling similar to interacting with actual buttons. Additionally, we support using the controller buttons as an additional input method and allow users to experience a sensation similar to using a presenter remote. Respectively, users can choose their preferred way of interaction and are not restricted to a single way of doing things.

3.3 Pilot Study

Developing a practical training application is sophisticated, usually requiring many iterations. Since the goal of this work was not to produce an industrial-grade public speaking trainer, only one iteration was done to optimise and analyse the system's efficacy and usefulness. As a part of this process, we conducted a pilot study at an alpha state of the application, identifying its most problematic aspects that needed fixing before the final evaluation. Much effort was put into the pilot testing phase, as it already was aimed at transporting the professionalism of a final evaluation setting and preparing for the final test run adequately. Respectively, the pilot and final study design were similar. Nevertheless, some differences remain based on their different focus and goals. While the pilot study aimed to identify any issues concerning the user experience and how participants perceived the application, the final evaluation goal was to determine how the training worked for people and whether it was effective.

In the following section, the general idea of the pilot study and how we conducted it is elaborated. Subsequently, we discuss the concept of the final evaluation and its differences from the pilot study. Accordingly, those sections contain a guide on how we realized our study, thus allowing for a better understanding of the results found in Chapter 5 and making it more comprehensible how they emerged.

3.3.1 Study Design

To identify crucial aspects and potential issues with the application, we invited four colleagues to provide their views on the alpha state of the project. We ensured that neither was familiar with this project before the experiment, so they were not biased. Aside from that, all of them had a shared understanding of computer games and used VR at least once. Familiarity with VR was considered a benefit here since the pilot study aimed to identify issues specifically with our application, not the platform itself. Therefore, prior knowledge and experience with VR ensured that participants felt comfortable with the technology and could focus more on the application.

Overall, we designed the study to analyse how participants felt while presenting in this environment and if they thought anything was missing or distracting them. To do so, they were required to prepare a five-minute presentation about a topic of their choice. Holding a presentation this long in the VE was considered sufficiently long to have a general idea of the setting while not bothering participants too much with extensive preparation necessary.

Since the focus of the pilot study was not to have the most realistic setting possible but rather to let participants play around with the experience, we decided that the presentation topic was irrelevant. Instead, it was beneficial to reduce further tension in participants towards the study and the unknown situation.

Regarding the presentation, participants also had to provide slides. While supporting them in talking about their topic and presenting their arguments, the slides needed to at least contain a starting and ending slide. Even though these are standard slides to include in a presentation, they were communicated as a requirement to participants since they are coupled to the simulation logic. Otherwise, it could have happened that the end of the presentation was not correctly identified by the simulation, resulting in missing the applause at the end of the performance.

3.3.2 Metrics

To better understand what participants felt and how they perceived the experience, we came to the consensus that talking to them was the most promising way for evaluation. Therefore, we utilized a semi-structured interview approach, allowing us to dynamically explore their thoughts while covering the most essential questions we wanted to discuss. This interview examined various topics, as can be seen in Table 3.1, which contains the used interview guide. The guide mainly focused on what parts of the simulation worked for participants and what did not, though it also included some fun questions to get participants talking and make the interview more comfortable for them.

In further preparation for the final study, a demographic and Simulator Sickness Questionnaire (SSQ) were included. While the contents of the SSQ were clear due to it being a standardized questionnaire (see 3.4.1), the demographic questions asked resulted from those common in VR literature. In our case, asking about the age range, gender, and both gaming and VR experience seemed sufficient since we mainly wanted to ensure a similar experience level.

Part 1: General	
1.1.	What person was the most interested?
1.2.	How do you usually prepare for presentations?

Part 2: Presentation specific	
2.1.	How did you feel during the presentation?
2.2.	Are there any particular moments you kept in mind?
2.3.	Were you distracted by something?
2.4.	Did you feel that something was missing?

Part 3: Audience specific	
3.1.	What impression do you have from the virtual audience?
3.2.	How would you describe their attention level?
3.3.	Did you have a feeling of them being alive?

Part 4: Environment specific	
4.1.	What did you notice in the environment?
4.2.	How many different kinds of laptops were there?
4.3.	Did you see people looking at their phones?
4.4.	Did you see someone scratch their nose?

Part 5: Conclusion	
5.1.	How would you describe your experience presenting in VR?
5.2.	Is there anything you would like us to improve for further tests?

Table 3.1: Pilot study interview guide

3.3.3 Procedure

We started each test by briefing our participants about SeminVR's capabilities and goals. Subsequently, they were familiarized with the VR headset and how to use the application with it. Before onboarding and letting participants enter the application for the first time, they were asked to fill out the pre-exposure SSQ. In the onboarding phase, we allowed them to check out the environment and familiarize themselves with the surroundings. After they knew how to interact with everything, their presentation slides were loaded, and they could start with their training.

During training, we observed their actions to identify potential issues within SeminVR and take notes of any remarks. After participants finished their five-minute presentation, they filled out the post-exposure SSQ, and we conducted our interview.

The study took around one hour each. However, half of the time was spent on the interviews as we tried to gain the most knowledge possible for our improvements.

3.3.4 Findings

The pilot study showed that participants had no severe issues with using the application. None of them fell sick or showed symptoms of simulator sickness. We assured this by requiring them to fill out the SSQ before and after the experiment. Aside from that, some issues and ideas were mentioned, generally discussing how to improve the simulation.

The most important of them all, and one that got fixed immediately after the first test, was the absence of a timer in the VE. As the first participant correctly argued, "It is hard to know how much longer to present if you have way of checking the time". Therefore, we added a virtual timer for the following tests to prevent further issues due to this uncertainty. Displaying the time expired since the presentation's start, the timer shown in Figure 3.5 was implemented. With two buttons on its top side, we allowed participants to interact with the timer by resetting, stopping, and continuing the displayed time. Respectively, no other participants complained about anything similar to this issue after the first one.

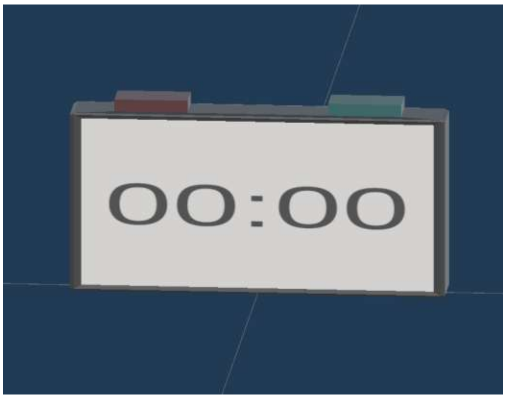

Figure 3.5: Timer model for indicating time in the VE

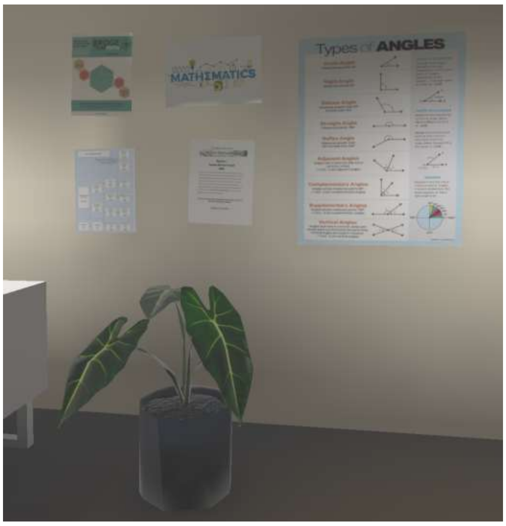

Figure 3.6: Details added for a more realistic environment

Issues with the Environment

Several times, participants mentioned the environment and issues with it, primarily arguing that it looked too sterile. In this regard, they also noted that a couple of plants can help make the environment more friendly. Therefore, we decided to add two medium-sized plants in the environment that are common for seminar rooms. Additionally, some posters were added to the walls, giving the room a more university-like touch, as seen in Figure 3.6.

We also identified issues with the virtual buttons used for changing the slides. While observing participants, we saw them not getting any visual feedback after pressing a button in the VE, resulting in them not knowing when the button was pressed. As a result, this led them to use the additional interaction method - the controller buttons. While we did not consider it a crucial issue, as another way of input was still possible, we fixed this behavior to allow participants to choose their preferred method.

Another participant highlighted that it is unlikely for an audience to have all their phones placed the same way. Thus, they perceived it unrealistic to see that phones in the environment were all lying in the middle of the desk and oriented perfectly parallel. Therefore, we randomized those for the final evaluation to prevent this feeling. While it might still be considered uncommon for every agent to have their smartphone lying in front of them, they can not be removed as they are essential to our simulation's distraction integration.

Issues with the Agents

Aside from improving the environment, the interviews showed parts of the agent simulation that need fixing. One of these issues was that agents were perceived as staring too much at participants. Most of our participants reported that they had this feeling of agents staring at them throughout the experience, especially at the presentation's start, as they were all looking directly at them. They argued that this transported the feeling of agents wanting them to start and being impatient. Trying to eliminate this effect, we decided to make agents more distracted before the start of the presentation. Thus, they look around more, focus on their notebook or phone, and not all collectively stare at the presenter at the beginning of the session.

Furthermore, some participants argued that the applause animation looked funny with the male agents. Though they still perceived it as applause, it caught their attention since they seemed only to wiggle their hands. After further inspection, we identified that this was due to a difference in the animation rig, resulting in male agents not touching their hands while clapping. To solve this issue, we modified the animations for male agents only and focused on making them touch their hands while clapping.

3.4 Study Design

With our final study, we wanted to identify SeminVR's effectiveness and elaborate on its usefulness as a preparation approach. Additionally, our goal was to explain if this efficacy is related to a platform's different performance and if this conflicts with its training purpose. To identify the general applicability of SeminVR, we considered it best to observe its effects over multiple training cycles. As Takac et al. [TCB+19] explained in their study, repetition is essential for presentation trainers to show habituation effects. Thus, we decided to repeat the training twice, allowing us to see if such habituation can be achieved.

Since we wanted to observe what benefits participants take from training a presentation in SeminVR, we chose to have them present the same presentation twice. Therefore, we required them to prepare a five-minute presentation on "Improvements for public transport in Vienna." Defining a distinct topic eliminates different intentions behind the presentation training and ensures our results stay comparable. If not considered, some participants prepare a business presentation while others create a fun slideshow for friends, making training effects hard to compare. Furthermore, we thought it essential to keep presentations short so participants do not feel overwhelmed when giving the presentation multiple times.

We conducted our study in a between-subject fashion. Seeking to explain whether SeminVR's training effects differ depending on a platform's performance, we defined two groups. One acted as the baseline to indicate what training effects to expect when using SeminVR, while the other compared different platform performances. Consequently, the control group was twice exposed to the same condition to show repetition effects. The experimental group, on the other hand, used the application on two different platforms to reveal if similar effects persist.

As platform differences might conflict with the usefulness of the presentation training, we considered it crucial to understand if less-performing platforms might not be capable of producing the same training effects. Therefore, we observed a high- and low-performant variant of SeminVR, which we compared in our experimental group. The high-performant version refers to a desktop computer-bound VR experience with a wirelessly connected Quest 2 (PCVR). At the same time, the low-performant version was considered the Quest 2 as a standalone headset (Standalone), as it is significantly less powerful than a desktop computer. Another reason for choosing these settings is that they best compare the entry-level VR experience running on a standalone headset against the more expensive setting of running it on a powerful desktop computer. Therefore, it should best replicate our interest in making presentation training available for everyone by identifying the applicability of SeminVR across platforms.

Interviewing participants about their training experience was central to identifying what they perceived using SeminVR. We considered it the best possible way to explore their thoughts and what training effects they might have experienced. Alongside determining the general impact of training, we also wanted to understand if participants noticed any performance differences. However, we considered it contradictory to directly ask how performance was perceived, as it would induce a bias for the second condition. Therefore, we had our participants explicitly focus on another aspect of the presentation trainer such that they only passively observed the changing performance. Respectively, we asked them to focus on the virtual audience's attention level and how it changed, although it remained the same throughout the conditions. Consequently, they should be unaware of the performance changes and genuinely report on any differences if observed.

As those interviews aim to follow a semi-structured approach, they allow for deviation from the guide to gain a deeper understanding. However, this makes them harder to compare, inducing the need for an appropriate analysis method. Therefore, we decided to apply TA to the knowledge gained from our interviews, aiming to create a more profound understanding of our data and allow for an elaborate interpretation of participants' perspectives. With this method, we seek to identify recurring patterns among our participant's feedback, create connections, and combine them into themes. Based on these, we aim to explain SeminVR's training effects and see if the changing conditions influenced our experimental group.

Since VR applications likely induce simulator sickness, we had to ensure our participants were not feeling sick during our experiment. Therefore, we had participants complete the SSQ multiple times throughout the study and observe its development. Additionally, we added some questions from the Igroup Presence Questionnaire (IPQ), since we aimed for a quantitative measure replicating participants' feelings towards our presentation training to provide a basis for comparison of our interview results.

3.4.1 Metrics

We decided on a combination of metrics to cover the most critical aspects relevant to understanding how participants perceive the simulation. Focusing more on the qualitative side of analysis, our primary method of generating insight was by conducting interviews. Although discussing all the essential aspects in our interviews, we considered it beneficial to have supporting quantitative measures for comparison and strengthening our findings. Therefore, we added the SSQ and IPQ to identify the participant's feeling of presence and simulator sickness while using SeminVR. Utilizing those measures in combination with interviews ensured the assessment of user comfort and their perception of the application on multiple levels, allowing us to put participants' verbal feedback and what they reported in the questionnaires into relation.

Since each participant had to perform their presentation twice, they also were required to fill out the questionnaires and be interviewed after each condition. Additionally, the SSQ was filled out before the first experience to identify initial sickness symptoms and allow for the comparison of how they evolved after their presentation. As for the IPQ and interview, asking them after each condition was sufficient since they only relate to how participants perceived the experience and do not show any effect developing over time.

The following sections cover each metric in more detail, elaborating on what they measure and how they were applied. Starting with the most important metric, we describe our interviewing technique and the questions we prepared. Afterward, a short description of both SSQ and IPQ and how we applied them in our study are given, elaborating on their necessity and their role in strengthening our findings.

Interviews

As our primary analysis method, we developed an extensive interview guide covering various questions. From questions about participants' stress levels and how they perceived their performance to discussions on the audience's attention level, those questions aim to better understand how participants perceived the application. With those questions, we aimed to formulate them as openly and unopinionated as possible to prevent bias and distortion of our results.

Although we developed a set of questions to ask, this evaluation focused on discussing people's experience with the application, making it essential to allow them to speak freely about their opinions and thoughts. Hence, we followed a semi-structured interview approach as it provided enough guidance to cover all relevant aspects while allowing us to pursue side thoughts to gain further knowledge.

The interview guide used for the evaluation can be found in the Appendix 7.1.

Perceived Stress To understand a participant's feelings of comfort, we asked them about their perceived stress levels during and after the experience. Those questions aimed to recall the most genuine response to that feeling. Hence, they were considered a good entry point for our interview. We decided that asking, "How stressed do you feel right now?" creates an opening for participants to formulate an elaborate answer containing their feelings towards the application and how it influences the perception of stress. Respectively, it also addresses whether they usually feel stressed in a situation like this. Since people's fear of presenting varies, identifying how they feel during actual presentations is highly relevant for later comparison with what they reported towards the virtual experience.

Leaving a similar room to interpret as with the first question, we continued asking whether the feeling of stress differed during the presentation and how it evolved. While it is essential to consider that asking participants how stressed they were is an opinionated question, we agreed that this bias was acceptable for exploring the influence of application performance.

Furthermore, we asked participants if they could compare their feelings to a real-life scenario and elaborate on what made a difference for them. Putting the simulation and real scenario into relation by asking participants to compare them should make the crucial aspects of the realistic simulation visible. As a result, we should see whether the application worked for them or not.

Perceived Performance Using further questions, we addressed how participants would describe their performance. Due to not observing them as explained in 3.4.6, it was necessary to rely on feedback on how they perceived everything. Hence, we aimed to discuss this topic more openly with our participants and follow their cues.

The only additional question to ask here was if they could talk about their topic without constantly looking at their slides. Although not necessarily a general indication of how well their presentation went, knowing how much they relied on their notes is a good way to see how secure one felt presenting their topic. Therefore, we asked our participants directly what they would say about this and see if any differences occurred for the second evaluation condition.

Perceived Attention In the last step, we decided to ask questions covering the participant's task at hand. Asking them to describe the audience to us and what their feeling of attention was like ensured maintaining the illusion of a changing audience. Nevertheless, it further served as a way to gain further insight into how the simulation felt to them and what role agents played in this context.

While again formulating the question openly and unopinionated, we aimed to let our participants speak freely about their feelings towards the audience. Thus, we allowed deviation in multiple directions and prevented them from answering with a finite value.

If not already discussed in the interview's progression, our guide's final question was asked to see if anything out of the ordinary happened during the experiment. Though this question specifically addressed whether any of the agents stood out to participants, the formulation of "stood out from the rest" was a cue to make participants think. Allowing them to rethink if anything weird happened, not necessarily restricted to the agents, should further show the viability of the training application and indicate improvement capabilities.

Thematic Analysis

We focus on gaining more profound knowledge based on the results of our interviews. Therefore, it is necessary to apply a method of analyzing those contents and form a better understanding of the things mentioned. We did this using the technique called TA. Known to be a valuable approach to identifying themes in qualitative data, the idea behind TA is to gather topics based on what interviewees mentioned, combine those into themes, and seek a connection between them. As the method tries to keep prior assumptions aside, it is necessary not immediately to form any assumptions but rather reiterate the data's facts. Since it is an interpretive research method, it always needs to be considered that its results closely relate to the researcher's values, experiences, and expertise. Thus, it is crucial to remember that TA conducted by a different party will likely produce different insights into the data. Nevertheless, when applied successfully, it should unveil general topics and hidden knowledge usually not visible using quantitative methods alone [BC06].

For our study, we apply TA using the transcripts of our interviews. By forming initial topics based on our participants' exact words, we start the first iteration to find a more profound meaning. Our next task will combine those and seek initial themes to identify any reoccurring patterns in our data. Overall, we'll compare the different testing conditions and each experiment run. For those, we try to identify differences in the data and aim to see if any significant disparity exists. As typical in TA, we will review our themes and reiterate our codes to ensure validity and relevance.

After we decide that our themes are elaborate enough and best represent our participant's views, we use the results to describe the effects of our experiment. By comparing it to our quantitative measures, the results of the TA should allow us to make a statement if differences exist between our training application's high- and low-performant versions.

Simulator Sickness Questionnaire

To act as a sort of insurance to secure participants' well-being and keep track of their senses, we have incorporated the SSQ [KLBL93]. The questionnaire contains standardized questions for identifying sickness symptoms with simulation applications, making it relevant for us to include in the study.

Designed to be applied as a repeated measure, SSQ combines the three core symptoms *nausea*, *oculomotor*, and *disorientation* in a total score. Each core symptom and the total score are observed regularly and should show the effects of repetitive exposure to a VE. Based on a calculation matrix, the questionnaire's three core symptoms form from 16 questions, which are further used to calculate the total score. Each of those 16 items describes symptoms relative to the three cores and is indicated on a severity scale ranging from 0 (none) to 3 (severe).

For our study, participants must fill out this questionnaire three times - *before*, *after the first*, and *after the second* presentation. As a result, it allows us to compare both the initial impact of the application and the difference between the conditions. While we need to consider that repetitive exposure influences those symptoms, we assumed that due to our short duration of presentations, only a few differences should arise.

Igroup Presence Questionnaire

Trying to understand the various aspects influencing immersion in a VE, one commonly used metric is the IPQ[1] [SFR01]. Since its development in 1997, it has been widely used inVR research [TCB+19, Ber20, WMD+21] as a way to identify how immersed people feel in VR experiences. Discussing *spatial presence*, *involvement*, and *experienced realism*, the questionnaire covers many details to identify a person's feeling of presence in the virtual space.

Although the questionnaire uses a predefined set of 14 questions, the scale for answering it sometimes varies depending on the language used. Like Melo et al. [MGB+23] describes, the German IPQ, for example, uses a 5-point scale, while the Portuguese version is based on a 7-point Likert scale.

For our study, we decided to use a 5-point Likert scale and follow Melo et al.'s [MGB+23] approach of adapting it to be indicated from 0 to 4 rather than going from 1 to 5. Since the SSQ uses a scale from 0 to 3, we thought it might be easier for participants to adapt to the IPQ range if it mostly overlaps with the SSQ range. Although both SSQ and IPQ indicate different things, their scales similarly indicate agreement with a symptom or statement. While for the former, the scale indicates the severity of symptoms ranging from *none* to *severe*, the scale for the IPQ ranges from *fully disagree* to *fully agree* with the statement. Therefore, we decided it would be a good idea to use this adapted scale to ease the burden on our participants while keeping results close to the original questionnaire.

[1] http://www.igroup.org/projects/ipq/

For simplicity and to further ease the evaluation for participants, we selected only a few questions from the original questionnaire by Schubert et al.[SFR01]. Since we also covered participant's perceptions in the interviews, we decided to be sufficient only to ask some of the questions to strengthen our understanding further. The selected questions aim to cover the three core aspects of the questionnaire while also considering their distribution. The resulting selection can be found in Table 3.2.

ID	Question
Q1	Somehow I felt that the virtual world surrounded me.
Q2	I felt like I was just perceiving pictures.
Q3	I had a sense of acting in the virtual space, rather than operating something from outside.
Q4	I felt present in the virtual space.
Q5	I was not aware of my real environment.
Q6	I still paid attention to the real environment.
Q7	I was completely captivated by the virtual world.
Q8	The virtual world seemed more realistic than the real world.

Table 3.2: IPQ question selection

As a result of this selection, the total amount of questions asked after each condition, solely coming from questionnaires, was 26. Although there are still many questions to answer, we considered a further reduction disadvantageous due to the additional distortion of questionnaire results.

Although IPQ is applied in a reduced form, we consider it essential to support our understanding of the participant's level of immersion. Especially considering that higher levels of immersion are related to a better training experience, it is crucial to discuss it sufficiently to allow us to estimate the usefulness of our application.

3.4.2 Participants

Since the simulation design replicates a seminar room in a university-like setting, we considered it best to have scholars as participants. Additionally, we decided to restrict participation to scholars of Austrian-based universities only, as the room's design was inspired by those of TU Wien (TU). Having participants from other countries accustomed to different kinds of universities could have influenced the perceived realism of the experience, thus being considered essential to think of.

Since the evaluation's goal is to assess each participant's perspective and perception of the simulation, we invited only 10 participants to ensure enough depth while retaining a manageable scope.

3.4.3 Repeated Measures

Based on the assumption that prior VR experience might vary between participants, we considered a way to incorporate those differences and ensure higher reliability of our results. Respectively, we thought of running a repeated-measures experiment. Doing so should allow us to compare iterations and distinguish between the effects directly associated with the trainer and those related to the novelty of the experience. Additionally, it should support habituation effects as Takac et al. [TCB+19] proposed in their study, improving training benefits and allowing for a better experience. Nevertheless, we did not want to overwhelm participants with too many repetitions. Thus, we decided only to incorporate two iterations of training the same presentation.

As our goal was to identify differences in SeminVR's effectiveness in correlation with different performing platforms, we considered it necessary to design a between-subject study. Doing so allows us to define a control group acting as a ground truth and compare it to an experimental group incorporating the differences performance might make. Respectively, the control group is designed to rehearse the same presentation twice under the same condition, allowing us to observe SeminVR's training effects. Those can then be compared to the results of our experimental group, which repeated their presentations, switching from one platform to another. Having participants repetitively rehearse the same presentation should show an improvement from the first to the second round, as they already could familiarize themselves with its content. If the same effect persists across both groups, we consider performance differences between platforms irrelevant for the efficacy of SeminVR.

In designing our study in a repeated-measure fashion, it was essential to consider the order of conditions. As shown in Figure 3.7, we decided our control group to repeat the same *PCVR* condition, while the experimental group compared it with the *Standalone* one.

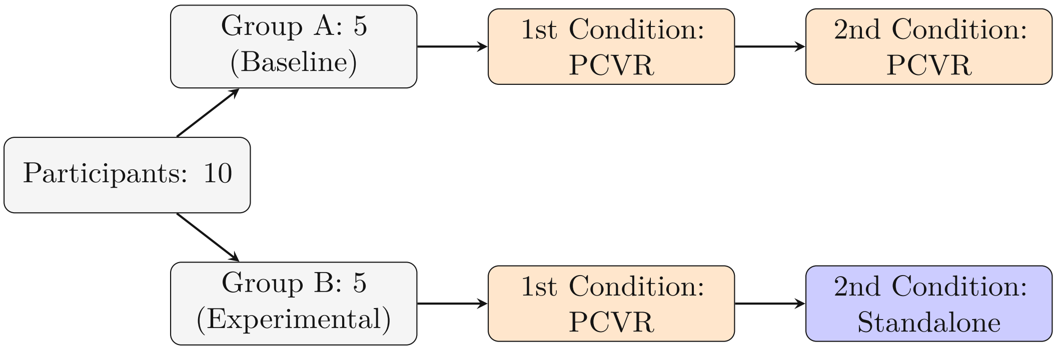

Figure 3.7: Participant group separation and order of conditions

Running SeminVR on a VR-ready desktop computer with a wirelessly connected Quest 2 headset (PCVR) was the best way to experience our application. Therefore, we wanted our control group to train under this condition to allow for the best possible presentation training. Furthermore, we had to consider which of our two platforms to start with for the experimental group. To allow us to observe differences between our groups before our experiment might affect them differently, we decided to start with the PCVR version as with the control group. Thus, the experiments are identical until the second round, which allows us to obtain potential group differences as they experience the same initial condition. Respectively, the experimental group's conditions order started with the high-performant PCVR version and had them repeat their presentation in the low-performant Standalone condition, running SeminVR exclusively on a Quest 2 without a desktop computer.

Considering the measures we want to use for identifying the effectiveness of SeminVR, we came up with the scheme shown in Figure 3.8 for conducting our research. Before each training condition, a SSQ is filled out to identify sickness levels before the exposure. Subsequently, participants will conduct training, giving their five-minute presentations. To see how sickness symptoms evolved, another SSQ is conducted after training to allow comparison with the initial values. Before starting the next round, we will ask participants to fill out the reduced version of the IPQ to identify how immersed participants felt during training. As soon as all the quantitative measures are taken, we can start with the interview and ask participants to elaborate on their feelings. After the interview, one round is completed, and the next iteration can begin.

Having the questionnaires answered shortly before and after the experiment ensures that participants are still fully aware of their experience. Additionally, having the interviews as the last measure in each iteration prevents bias participants might take from them while filling out the questionnaires.

In this regard, it needs to be mentioned that we used the post-exposure SSQs to also act as the pre-exposure questionnaire for the following round. This was done to not overwhelm participants with unnecessary questionnaires and under the consideration that the levels of sickness should not be significantly different from after training to starting the next round.

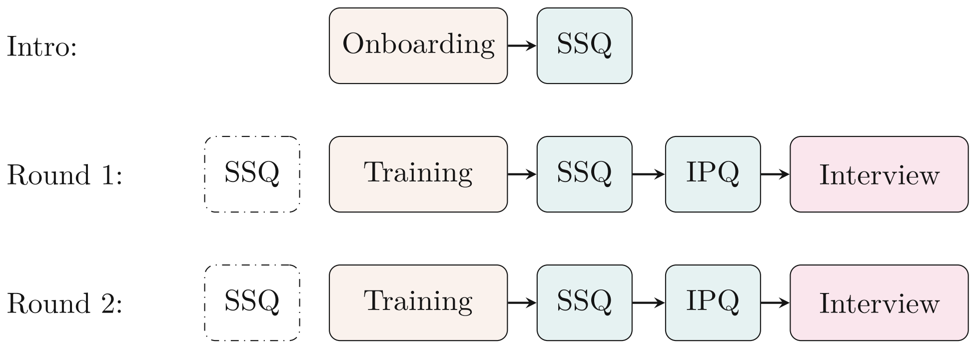

Figure 3.8: Scheme for conducting the study

3.4.4 Illusion of Changing Attention

Asking somebody whether they liked the application's performance will produce biased results. The question makes participants rethink their experience and search for better or worse performance indicators, even if none exist. Considering that asking the question in a repeated-measure setting introduces an additional bias for the second condition, it is necessary to find a way of indirectly identifying the effect of performance on participants.

Thinking of possible ways to measure potential differences indirectly, we came across the approach of Neumann et al. [NCB91]. Their work had participants rate different television viewing experiences while only modifying the audio setting. Doing so, they wanted to identify if the change between mono and stereo audio impacts how people perceive video content without directly asking them to report on it. Therefore, they had participants rate videos on how they liked them, their level of interest, and how visually and auditively attractive they were. Respectively, they could identify the effects of audio quality despite participants knowing about the exact focus of their study.

Similar to their approach, we decided to have our participants focus on another part of our presentation trainer to understand if they passively perceive performance differences. Therefore, we decided to have them under the illusion that the virtual audience's attention level was changing. In asking them to focus on how they perceived the audience and how attentive they were, participants focused on this aspect of the simulation while unknowingly perceiving the effects of the changing performance on them. In combination with not actively asking them to report on the prototype's level of performance, it should be possible to observe if this change affects them or if the results are identical to the control group.

3.4.5 Onboarding

We decided to include an onboarding step before the first evaluation to incorporate differences in VR experience and ensure participants share a similar level of knowledge using the study's VR equipment. It allows inexperienced participants to get to know VR before the evaluation and reduces potential issues emerging due to not understanding how to use the application. Respectively, we created a demo presentation for the participants, allowing them to familiarize themselves with the evaluation environment and learn using SeminVR. Therefore, the presentation contained the basic information on how to interact with the trainer, as shown in Figures 3.9 and 3.10.

Nevertheless, we must consider that the onboarding influences the participant's stress response for the first condition due to the otherwise unknown setting. However, since we wanted to identify the training effects of SeminVR and not replicate an examination-like situation, we considered this stress reduction beneficial. Having participants aware of what to expect should produce a more genuine stress response as it does not solely relate to being in an unknown setting for the first time. Therefore, the stress levels should be more comparable between conditions and not only differ due to the novelty of the situation.

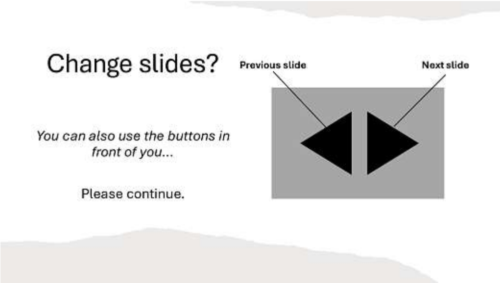

Figure 3.9: Onboarding slide informing about the slide navigation

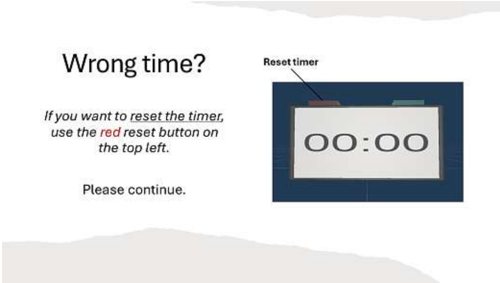

Figure 3.10: Onboarding slide informing about the timer reset capability

3.4.6 Privacy while Training

As a result of the pilot study, we identified that participants were slightly uncomfortable with being observed by somebody outside the VE. Due to VR intending to fully immerse users, we understood why participants felt weird being observed, as they had no way of knowing what was happening outside the virtual space. As a result, we decided not to observe participants in the final study in any way, meaning they were neither recorded in the VE nor real space.

We considered it more important to leave participants their privacy and allow them to train on their own rather than compromise the realism of the training application for the sake of identifying behavioral details. Respectively, we, as researchers, were leaving the room for the duration of the presentations, asking the participants to inform us as they finished.

While a virtual presentation recording might have been a possibility, it was also considered an influential factor in the participant's behavior and, therefore, not worth it from our point of view. Furthermore, a virtual recording would have caused potential performance issues for either evaluation condition, further strengthening our decision not to pursue it.

CHAPTER 4

Software Implementation

The foundation of our application was based on the *Unity WebXR Exporter*[1] project by De-Panther. Since Unity does not natively support the WebXR API, basing our implementation on this project allowed us to develop a Unity WebGL application and use the project's presets to make our application WebXR-ready. The most important of these presets was the *WebXRCameraSet*, which contained the logic to match the virtual camera feed to a VR headset's point of view and handled the controller input for each supported device. Alongside this, the central part of the exporter project was the integration of the WebXR session, meaning it contains the logic to identify if the web page is displayed on a headset, figure out what kind it is, and prepare the browser to allow showing the VR content accordingly.

Basing our application on WebXR made it possible to access it via various kinds of headsets simply using the browser. Although it was a considerable benefit to distribute only one piece of software for multiple operating systems, it also required finding a feasible way to make it work equally well on all of them. Respectively, we aimed to develop an application that is performant for standalone and computer-bound headsets.

As is typical for multi-platform development, we had to consider our worst-performing device as a baseline for optimizing our application. Therefore, we consider the computational capabilities of our targeted low-end device, a Meta Quest 2. Since this headset is powered by a smartphone processor, its performance is nothing comparable to a VR-ready desktop computer with a dedicated graphics card. Consequently, we needed to optimize our application to require as few necessary resources as possible to allow a smooth experience on the Quest 2.

Alongside aiming for a smooth experience, it was equally important to make the presentation training realistic and create an experience replicating a real-life scenario. Thus, we

[1] `https://github.com/De-Panther/unity-webxr-export`

designed a simplistic scenery containing enough details to be naturalistic while remaining minimalist and performing. In doing so, a substantial piece to add to this realism was the development of a virtual audience that replicates life-like behavior and provides an audience to talk to. Combined with a simplistic set of interactions ensuring the application's ease of use, we aimed to achieve a performant VRPST that runs on low-end devices but is elaborate enough to replicate a realistic training scenario.

In the following sections, we elaborate on how our development was restricted by aiming to make the application run on low-end platforms. Subsequently, we explain in detail the integration of our application, starting by elaborating on how we designed the virtual environment. Afterward, we discuss the development of the virtual audience and all the nuances considered in trying to imitate life-like behavior. Subsequently, details on how we integrated interactions within SeminVR are discussed. Lastly, the integration of the dynamic loading of presentation slides using a local server and the steps necessary to apply those in the virtual environment are given, marking the end of this chapter.

4.1 Performance and Platform Limitations

Although it is beneficial that WebXR allows the creation of a single application for multi-platform use, it is also restrictive in some ways. Especially when someone tries to develop a visually appealing and smooth experience, the benefit of multi-platform support also becomes a limitation. Those constraints become more evident when considering the application's ability to run on powerful machines like VR-ready desktop computers and weaker-performing standalone headsets. Respectively, it is essential to consider the computational capabilities of each platform and consider optimizing the application to perform similarly across devices. Since standalone headsets use smartphone processors, their capabilities are very restricted. Therefore, it is essential to focus development on providing a usable experience to the lowest-performing platforms since it also ensures applicability on more powerful end devices. Respectively, we considered potential ways to improve SeminVR by efficiently using available resources and reducing its complexity where possible to make it a helpful training application that is useable independently of a platform.

4.1.1 Performance Baseline

The necessity of performance improvements being a high priority became clear while analyzing the demo provided by De-Panther's Unity WebXR exporter. Testing the application by locally hosting the demo and accessing it via a Quest 2 in standalone mode showed that its FPS was already limited. Although the scene contains little complexity, showing a desert environment with some cubes and a ball, as shown in Figure 4.1, we only could achieve measuring 45 FPS. Therefore, our development basis was already relatively low. Especially when considering the recommendation for VR applications to run at 60 FPS to allow a smooth experience. Out of curiosity, we deleted all the objects inside the demo scene for comparison and tested the performance again. Surprisingly, nothing

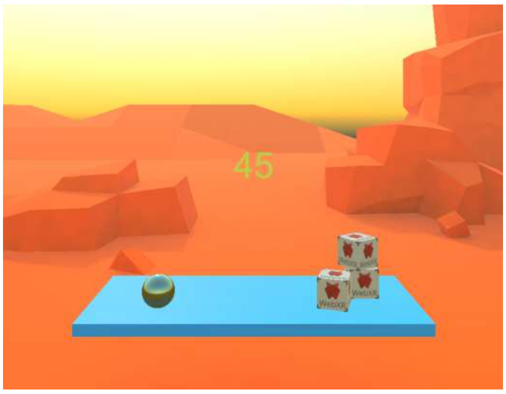

Figure 4.1: Quest 2 standalone performance for WebXR demo scene

Figure 4.2: Reduce complexity - Simplified (left) VS Original (right)

changed in this regard, which made clear that reaching higher framerates is unlikely. Respectively, we were reassured that the performance optimization of our SeminVR was crucial to providing a usable training application and might likely be a factor that might influence the experience.

4.1.2 Reduce Model Complexity

As a starting point to increase performance, we considered reducing the complexity of our scenery. Thus, we focus on the number of 3D models in the environment and each object's details. Since high-resolution models contain many polygons to provide more detailed structures, they require more computational power to render appropriately. Therefore, by reducing the number of polygons and trimming details from a model, we could simplify them and reduce computational load.

To achieve this complexity reduction, we simplified our models using the free 3D modeling software *Blender*[2]. Therefore, we imported each of our application's models into the software and modified them using the *Decimate* modifier. Using this modifier allowed us to reduce the complexity of all model details by adjusting a single slider value. In our case, we used the decimate modifiers collapse method, which lowers the number of polygons by merging their edges. The lower the slider value set in the modifier, the more edges merge. Respectively, the more simplistic the model gets. Using trial and error, we tried identifying each model's sweet spot until it was no longer realistic. Afterward, we exported the simplified model version and added it to our Unity project. That process was performed for each environmental object and agent model. The latter are the ones that profited the most from the reduction process. Figure 4.2 shows an example of its result, displaying the reduced version of an agent model on the left and the original on the right for comparison. While one might see minor differences, we kept the model

[2] https://www.blender.org/

similar to its original. Yet, we significantly reduced the number of details such that only around 20% of the polygons remained. Therefore, we achieved an approximate complexity reduction of 80% for the agent models alone, substantially impacting our application's smooth operation.

4.1.3 Occlusion Culling

Another common trick to improve game performance is occlusion culling. When applied, it eases the load on the processor by reducing unnecessary calculations of environmental objects. Occlusion culling renders only models visible to the viewer in their current position and discards everything outside their view frustum. Therefore, objects behind the user or covered by another object are excluded from rendering, as they are not visible to the user. Calculating only the parts of an environment users can see drastically enhances the application's performance due to significantly reducing the objects rendered for every frame. Additionally, it comes alongside the benefit of allowing the addition of more environmental details without worrying that performance will be reduced significantly. Nevertheless, it still needs to be treated carefully and does not allow indiscriminately adding details, as their amount still had to be kept low.

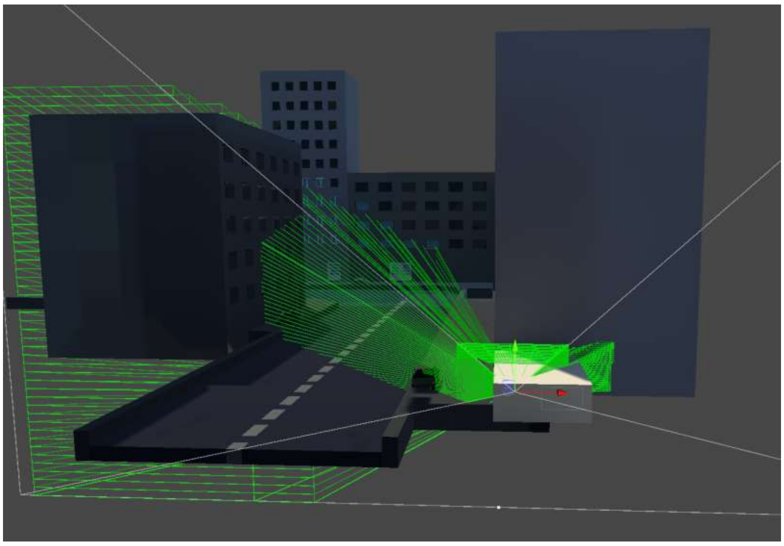

Figure 4.3: View frustum of the occlusion culling

Figure 4.3 shows the view frustum of the occlusion culling within SeminVR, indicated by the green lines. Each line that hits an object represents parts visible to the users in their current position. Therefore, all the objects colliding with such a ray are rendered, while everything outside is excluded. Accordingly, when focusing on the opposite direction the view frustum is facing, it is visible that no models are rendered in this space, as they are irrelevant at that time.

To further optimize this process, it would be possible to separate models into smaller pieces, thus allowing more of the scenery to blend out. However, this comes at the cost of adding more polygons to the objects, making them more complex and demanding while rendering. Therefore, we decided to keep the models as they were and not optimize the culling process further.

4.1.4 Lighting, Baking, and Reflections

Lighting is essential to replicate a realistic environment by improving depth and establishing a comfortable setting. While traditional computer games often contain a variety of light sources to provide the most realistic lighting effects, including real-time shadows, we had to simplify this aspect to achieve better performance with SeminVR. Although realistic light and shadow behavior make a simulation feel authentic, it comes alongside extensive calculations necessary. Since real-time lighting effects require recalculation for every frame, they constantly influence performance and must be considered accordingly. Consequently, we decided not to use dynamic lights and instead work with fixed lighting. Utilizing a method known as *baking*, we calculated the lighting in our scene before running the application. Those lighting patterns are stored in a *lightmap*, which allows us to easily apply this static lighting setting to the scene, laying it on top of the rendering. Respectively, it does not require continuous recalculating since it is a static image projected on the environment's textures. Thus, it is a great way to improve performance since it reduces the computational load on the processor while maintaining lighting effects.

Baking is also a commonly used tool in game development to enhance performance. However, in this field, it is usually combined with some real-time lighting to create more realistic experiences. While it would have been possible to include one real-time light source inside SeminVR, we decided not to add one to increase performance further. Although this did come with the drawback of having no real-time shadows, we identified them as adding little to our experience. Thus, we accepted this disadvantage in favor of reducing the computational load of our application.

4.1.5 Final Performance

After exhausting all the refinements and optimizations described above, we measured SeminVR's performance on our evaluation platforms. Therefore, we tested it running once on a Quest 2 in standalone mode and another time connected wirelessly to a desktop computer. As expected, the differences between those were significant. In the desktop computer's case, SeminVR ran consistently with 60FPS, while only 21FPS were reached for the standalone version. Nevertheless, we identified this performance as sufficient for our intended use case since presentation training is primarily static, with users mainly standing quietly in a room and not moving extensively. We also confirmed this assumption in our pilot test, as all participants used the standalone version without issues. Respectively, we consider this performance level sufficient while potentially not enough for everyone. Thus, we also wanted to find this with our study.

4.2 Virtual Environment

Focusing on replicating a seminar room-like feeling, we designed a virtual scenery consisting of tables, chairs, notebooks, phones, and other little details. The resulting design is shown in Figure 4.4, displaying the room from the speaker's perspective. Typical for a university-like setting, the audience sat in rows facing the front of the room. Additionally, there was a table dedicated to the presenter containing all the necessary tools to interact with the presentation trainer. On the left side of the room, we added some windows facing the outdoors, showing that the room was located in an urban environment to transmit the feeling of being in a city. We added notebooks and smartphones for the virtual audience to enhance realism, which we further utilized as distraction points while developing their behavior. Additionally, we added little details like posters on the walls, plants in the corner, ceiling lights, and a projector hanging from the roof to give the room more depth and further replicate a university-like setting.

Figure 4.4: Overview of the virtual seminar room

To allow for such a detailed setting and reduce work-load most of the models used in SeminVR were based on free-assets downloaded from cgtrader, SketchFab, and the Unity Asset Store. However, there were some exceptions. As for the seminar room, we built it ourselves using Blender and being inspired by a meeting room model we found on SketchFab[3]. Similarly, we did the same for the table model basing our creation on a model found on cgtrader[4]. Lastly, we also built both timer and slide control panel on our own as we did not find a free model matching our idea of those objects.

[3]https://sketchfab.com/3d-models/meeting-room-f0c870c6c73e43c48a139c93a780b363

[4]https://www.cgtrader.com/free-3d-models/furniture/table/wooden-table-9ff1d9d1-e583-448e-83b5-a6b03142433d

The complete list of all assets and their origins can be found in the Appendix 7.1.

To enhance the feeling of being at a university, we added atmospheric sounds typical for the setting. Therefore, we used an ambiance recording of a hotel room from freesound.org[5] containing ventilation, footsteps, and closing door sounds. To make the sounds less overwhelming, we played the recording quietly in the background, creating slight disturbances expected in seminar rooms. Therefore, it should transport a more realistic feeling than presenting in a tranquil environment.

4.2.1 Presenter Table

For the presenter table, as shown in Figure 4.5, we created a minimalist-looking setup that provided all the tools needed to interact with the presentation trainer. Alongside keeping it simplistic, we also thought of the placement of each tool. Respectively, for the monitor showing the presentation slides, we considered it necessary to allow users to face the audience while presenting. Therefore, we placed the monitor in front of the trainees, aiming to be within their periphery vision. Since we desired to transmit the feeling of training in front of an actual audience, we considered it essential to guide participants to face the room while also allowing them to keep track of their presentation.

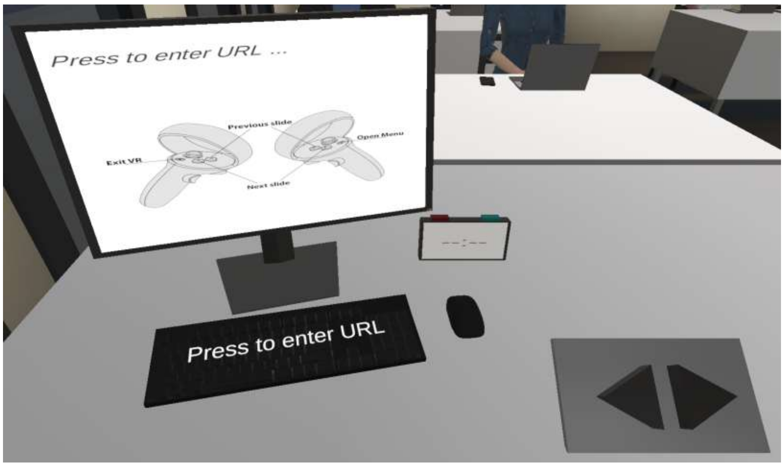

Figure 4.5: Overview of the presenter's table

Similarly, we considered the placement of the timer, another essential piece to keep users informed about the duration of their presentation. Thus, we placed it next to the monitor to ensure it is noticed while not overwhelming users with its presence. Having it also in the periphery vision but separate from the monitor allowed us not to overload the screen with information while still allowing users to keep track of time.

Lastly, the presentation slide controller was placed right in front of the trainees to be easily reachable. Additionally, putting it close to users also required them to look down

[5] https://freesound.org/people/AderuMoro/sounds/712590/

at the table to see the controls fully as they were placed at hip level. Thus, this should further enhance the feeling of a realistic experience as it is similar to using keyboard controls for changing slides during an actual presentation.

4.2.2 Outside

The outside was designed to replicate a city-like environment with residential buildings, a street with a sidewalk, and some cars. As seen in Figure 4.6, the setting was also kept minimalist and calm so that trainees would not be too interrupted by the things on the outside. Nonetheless, it added depth to the scenery, trying to transport a realistic feeling of being at an urban university site and giving a presentation there.

To enhance the realism of the outdoor setting and not make it appear deserted, atmospheric sounds were placed outside of the seminar room to make them appear to come from outdoors. These sounds were based on another recording from freesound.org[6], which captured the environmental sounds of a town center. Thus, it creates a soundscape of people talking on the streets, birds chirping, and traffic going by.

Figure 4.6: Outside view from participant's default standpoint

4.3 Virtual Agents

Modeling and animating virtual characters is a demanding task that could be the subject of a study on its own. As our focus was building an application rather than creating each asset ourselves, we decided to search for existing humanoid models. While searching, we came across *Adobe mixamo*[7], a website providing animated 3D characters for personal use. As we seek not to publish our project commercially and only use it for research purposes, we decided to use their models in our project. The significant benefit of using

[6] https://freesound.org/people/XYZsoundscapes/sounds/712672/
[7] https://www.mixamo.com

Mixamo's models over others was that they already contained a matching character rig, which was necessary for applying animations to them. Additionally, they provide various animations developed explicitly for those characters and offer adjustments in their web interface to make them match each model. Moreover, they provided a broad range of characters, allowing us to select a diverse group of humanoid models for our application.

Although not having to individually model, rig, and animate multiple characters saved much time, all those models still had to be integrated into SeminVR and equipped with intelligent behavior. Since we aimed to have agents mock simplistic human-like behaviors, we considered creating their logic without developing extensive artificial intelligence. Consequently, we restricted their actions to sitting, applauding, and looking at a point of interest. As for the latter, this meant looking at some point influencing the agent's behavior and matching their current state. Such as looking at the presenter when agents actively listening to them.

We quickly realized that developing a state machine would be the best approach to integrate our logic. Utilizing the state machine approach, we were able to define the different kinds of states that our agents can take and describe their relationships easily. Accordingly, it allowed us to define transitions and the existing restrictions between them clearly. Consequently, a managing state machine controller was developed to handle the transition logic and keep track of the agent's current behavior.

In the following section, we elaborate on general terms we used while describing our agent's logic, where they come from, and why they are necessary. Subsequently, we discuss the agent's behavior and explain the predefined set of agent behaviors. The section concludes with details on how we implemented their animations and what made them change their focus point.

4.3.1 General Terms

Presentation Reference Duration... T_{ref} When thinking about presentations, we often notice distractions increasing over time. People get tired and tend to move their attention to something else. Therefore, it was necessary to develop our application accordingly and base our audience's behavior relative to the expired time. Respectively, the reference duration acts as a central piece of the simulation logic required for calculating the current distraction level of agents and helping to calculate the likelihood of distractions occurring.

Since we designed our study to ask participants to give five-minute presentations, we correspondingly decided to match the reference time for our trainer logic to the same duration. Accordingly, the virtual audience reached their highest level of distraction at the end of the five-minute presentations.

Distraction Level To increase the likelihood of agents getting more distracted the more prolonged the presentation, we integrate a steadily rising value replicating their level of distraction throughout the simulation. Based on our assumption that attention does not decrease gradually but logarithmic, we decided to let our agents' distraction levels evolve similarly. As seen in Figure 4.7, at the beginning of a presentation, the agent's distraction levels start at zero and grow until reaching their maximum at the five-minute mark.

If users take longer than the reference duration, the distraction level is reset, making attention restart every five minutes. While this might seem odd, it must be considered that the distraction level's most significant influence is the distraction state duration, meaning that the higher the value, the longer distractions take away the agent's focus. After a reset, it does not mean that distractions will no longer occur since the variables used for those reset as well. Thus assuring that agents maintain their level of distraction likelihood, although their duration varies over time.

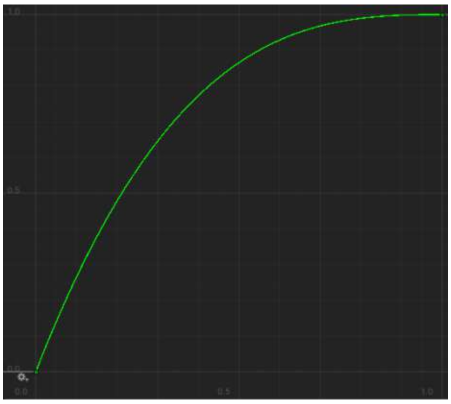

Figure 4.7: Distraction level growth over the reference duration T_{ref}

Distraction Type Although distractions sometimes appear randomly, we consider it insufficient and uncontrollable to attribute our agent's behavior solely to randomness. While considering achieving realistic distraction behavior, we quickly agreed that our interpretation needs to be more conclusive. Therefore, we conducted short literature research to identify the development of attention during presentations alongside the most common distractions and how often they happen.

In doing so, we came across the work of Bradbury [Bra16]. They discussed the common misconception that attention decreases after ten minutes and where this idea originated. We used their findings to develop our distraction model, basing the distraction level on a reference time and not making it drop after ten minutes as they suggested.

Combined with the findings of Leysens et al. [LlRP16], describing distractions during lectures, we identified a basic concept for modeling our agent's behavior. They explained that various distractions exist during lectures, mostly from notebooks and smartphones. Respectively, we decided to focus on these as our primary distraction sources. Additionally, Leysens et al. elaborated on the probability of being distracted by one or the other while observing around 120 students in their study. Based on their findings, we developed *distraction types* describing agents' personalities and likelihood of being distracted. Those were separated into five possible states as we understood from their study: being distracted *never*, *once*, *twice*, *regularly*, or *constantly*. While the former three indicate how often a distraction occurs over the presentation time frame, we interpreted the latter two concerning our reference time. Accordingly, we decided that "regular" distraction means every minute, while "constant" distraction is when an agent looks somewhere about every 30 seconds.

Those types made it possible to give agents traits describing how likely they are distracted by phones and notebooks. Storing them in two separate variables further allowed us to differentiate between agents more distracted by one or the other. Therefore, those two values represent an equivalent of personality and enable us to define unique behavior for each audience member.

Distraction Time Interval... t_{int} To make our distraction types believable, we had to think of a way to ensure regular pseudo-random intervals in which those occur. Respectively, we defined distraction time frames in which those interruptions are likely to happen. Combining the presentation reference time (T_{ref}) and the distraction amount resulting from the distraction type for each phone (d_{phone}) and notebook ($d_{notebook}$) distraction, we calculated the interval (t_{int}) in which distractions should occur.

$$t_{int} = \frac{T_{ref}}{d_{phone} + d_{notebook}}$$

Since directly using this interval would lead to a relatively distinct behavior of agents getting distracted every X seconds, we decided to include a modifier adapting the interval by a randomized amount between +/- one-sixth of the interval ($\frac{t_{int}}{6}$). This modifier is recalculated after each distraction, allowing a slight randomization of the time intervals between distractions. Respectively, it increases the likelihood that agents sharing similar distraction types will not act the same.

Participation Modifier... p For more versatility in how agents look while applauding, we decided to have two applauding animations, as depicted in Figures 4.8 and 4.9: one anticipated and the other relaxed. To improve realism, we integrated them so they can be blended together, allowing for more than just two applauding animation styles. Respectively, we added another trait to the agent's personality, describing their unique look while applauding, known as the *participation modifier* (p). Based on this value, the position between the two applauding animations is defined, representing the agent's level of participation on a scale from zero to one. Respectively, agents acting calmer had a lower participation modifier value than those behaving more actively. Having the value predefined and mixed among all agents allowed us to ensure that not all agents look the same while giving applause, thus providing a more realistic behavior.

Movement Speed Modifier... m Typically, people differ in their reaction speeds and how fast they move. Some are more relaxed and move slowly, while others are more hectic and act quickly. To include this kind of variability, we added a predefined movement speed modifier to replicate these differences and make our audience act unique.

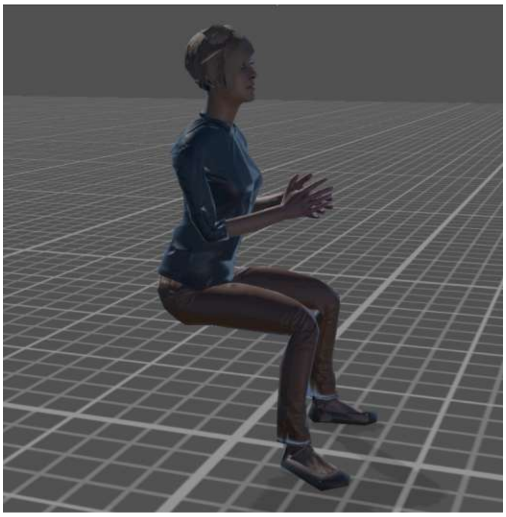

Figure 4.8: Relaxed applause pose

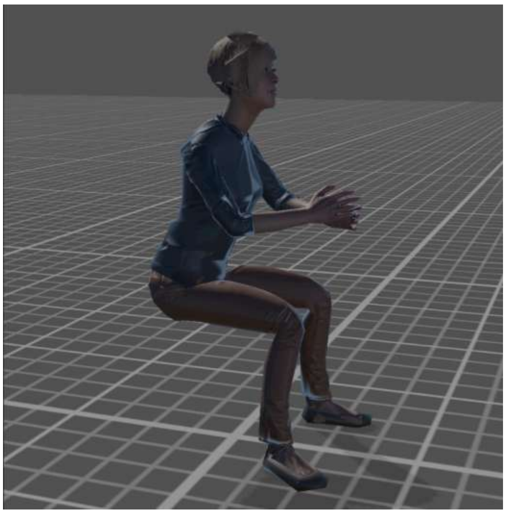

Figure 4.9: Anticipated applause pose

Therefore, the modifier was used in combination with the playback speed of the agent's animations. Adapting the default speed by applying the modifier, speeding up or slowing down their movement was possible. Since modifying the animation speed might easily result in unrealistic behavior, we carefully chose an acceptable range to adjust. To identify this range, we used a trial and error approach to see at what point animations look unrealistic. Respectively, we identified animation speeds deviating +/- 25% from our base value to be still perceived as realistic. However, since this feeling of realism closely relates to our perception, we ensured a more generalizable result by shrinking the identified range by 25%. Thus, we resulted in a modification range of +/- 20%, implying that more active agents act up to 20% faster than the baseline.

4.3.2 Behavior

Although we could have built the agent's behavior using simple queries, we considered our application to profit from a more distinct separation. Thus, we developed a state machine to allow flexibility and a clear structure of our agent's logic. While other structures might scale better than state machines, such as *behavior trees*s, we decided them to be overkill for our use case, as we tried to keep our logic simplistic.

Overall, we separated our agent's behavior into five states: four distractions and one idle state. As for the distraction states, they further split into two types: *random* and *interrupting* distractions. While the former refers to distractions that randomly occur throughout presentations, the latter describes those resulting from actions by the presenter. Respectively, both *notebook distraction* and *phone distraction* are categorized as random since they get randomly triggered throughout the presentation. On the other hand, changing the presentation slides and applauding are considered *interrupting* distractions as they relate to the presenter changing their slides. Lastly, the final piece of the logic, which is the one connecting all the states, is the idle behavior. As the state that's the most active, the idle contains the agent's core functionality to make them act like actively listening. Additionally, it provides the entry point for the simulation and acts as a middleman when transitioning between states.

The following paragraphs elaborate on each of those states in greater detail. Each contains aspects regarding behavior and describes how it relates to other states. To accompany those descriptions, Figure 4.10 shows an overview of all the states and their connections to allow for a better understanding of their relationships.

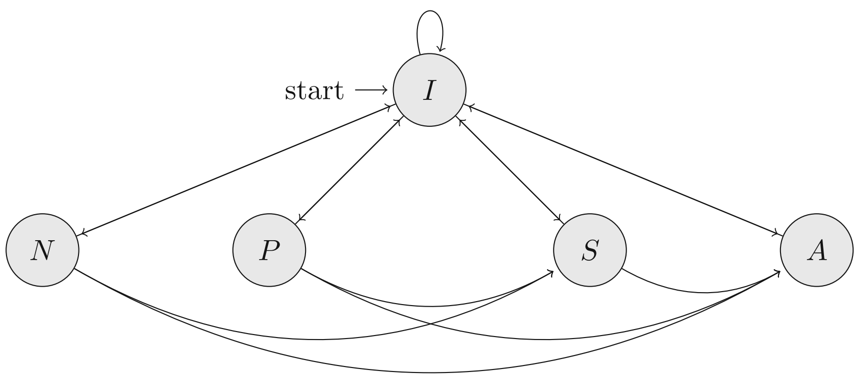

Figure 4.10: Agent behavior state machine

Idle State (I) As the central piece of the state machine, the idle state is where everything comes together. It connects all the different distraction states and contains the core functionality of agents' behaviors. Thus making them act like they are silently listening to the presenter. Throughout the simulation, agents stay in this state for as long as no distractions occur. During that time, agents sit and watch the presenter, following their head position to transmit the feeling of actively listening to them. Since the state is the one agents spend the most time in, it includes logic for changing idle animations to make them not all look the same. Accordingly, they occasionally move around slightly but share the same set of animations. Since all the other states are interruptions of this idle behavior, all of them are connected to it. Respectively, transitions to any of the other distraction states are possible. The same is true the other way around since all distractions result in a return to idle behavior as soon as they disappear.

Notebook Distraction State (N) and **Phone Distraction State (P)** The two randomized distraction states are active in pseudo-random intervals relating to the agent's likelihood of getting distracted and the last time they were active. Relative to the individually set distraction type, how often those states are happening varies for the different kinds of agents. While in these states, agents look toward the distracting object, either being their notebook or phone. Accordingly, they communicate to presenters that they are not listening to them as those objects distract them. How long agents are distracted by those depends on their current distraction level and what kind of distraction it is. For each distraction type, we defined a minimum (t_{min}) and maximum (t_{max}) duration in seconds, as shown in Table 4.1.

Distraction Type	t_{min} [s]	t_{max} [s]
Phone (d_{phone})	2	30
Notebook ($d_{notebook}$)	2	60

Table 4.1: Distraction duration range in seconds

Each time a distraction occurs, we calculate the state's duration uniquely based on the range provided for its type. Moreover, we included the distraction level in this process to incorporate the effect the presentation duration has on agents' attention. To make the level not directly influence the distraction duration but act as a reference point, we used it as the upper boundary for calculating a random modifier ($rand_{distLvl}$).

$$rand_{distLvl} = RandomValueBetween(0, distLvl)$$

Therefore, we ensured that the duration remained randomized while increasing the likelihood of distractions being longer the more prolonged the presentation. Consequently, we calculate the difference between the maximum and minimum distraction length, modify this value with the $rand_{distLvl}$, and add the result to the minimum duration.

$$t_{dist} = t_{min} + (t_{max} - t_{min}) * rand_{distLvl}$$

Respectively, we ensured that if the randomized modifier value is zero, the distraction state remains active for at least the minimum duration. Additionally, it restricted the maximum duration to be reached only toward the end of a presentation as it included the distraction level to modify the value.

Slide Distraction State (S) and **Applause Distraction State (A)** Both interrupting distractions occur due to the presenter's actions, more precisely, relative to them changing the slides. When such an event occurs, agents check their likelihood of being distracted by it relative to their current distraction level. Should they already be more easily distracted, they are more likely to be affected by a slide change than if they are when more actively listening to the presenter. Respectively, they transition to the slide distraction state accordingly or not. When transitioning, the agent's focus point is changed to the projection of the presentation slides, as shown in Figure 4.11. Thus making it look like they are reading the slides as they appear. To make this behavior more realistic, we randomized the distraction's state duration between two to ten seconds, replicating different reading speeds of people. Additionally, as it is being recalculated for each slide and agent individually, it creates a random mixture of the audience reading speeds. Thus providing a more life-like feel.

Similar to the slide change state, the applause distraction results from recognizing a change of slides, explicitly the one made to the last slide. Since the direct transition to the applause state would result in the audience applauding immediately, we introduced a

Figure 4.11: Slide distraction state

Figure 4.12: Applause distraction state

timer to delay this transition. Respectively, unless the presenter changes the slide within the timer duration, the applause distraction is triggered when the timer runs out, making agents start applauding. We integrated it like this to ensure that mistakenly switching to the last slide is not punished by immediately triggering the applause state. Additionally, the delay allowed participants to conclude their presentation on the closing slide before being thanked for their presentation with applause, as shown in Figure 4.12. Considering this integration, it becomes clear why we required participants to include a final slide in their presentations. Otherwise, it would be not possible to know when they are finished and trigger applause accordingly.

Given the five-minute presentation limit, we decided to make our applause take around five seconds, short enough to be perceived as satisfying while not too long to be unrealistic. By modifying each agent's applause duration using their participation modifier, the resulting applause setting was a mixture of different applauding styles and lengths, trying to replicate a realistic scenario.

Distraction Flow

To make agents understand when they should get distracted, they regularly update all their variables using the *Update()* function called every frame. Figure 4.13 describes the flow of what happens in this function, showing both the regular update and the interrupting flow.

On the right side, the regular flow starts by updating the agent's distraction level relative to the time expired since the start. Afterward, they check if it is time for a new distraction to occur. Nothing happens if the last distraction happened too recently, and the flow stops here. However, a new distraction is created if the time since the previous distraction surpasses the distraction time interval (t_{int}). Therefore, we use a function to calculate the most probable distraction for the agent based on their distraction history and relative to their personal distraction types. Subsequently, depending on the kind of distraction returned by the function, the transition to the new state is triggered, resulting in the agent's attention drifting away to the latest point of interest.

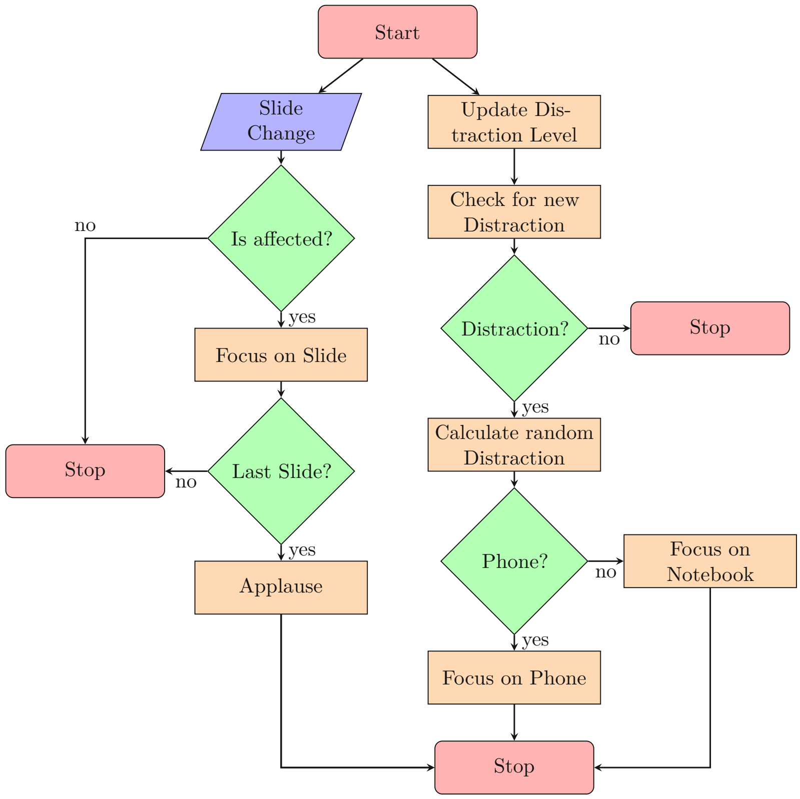

Figure 4.13: Agent's regular updates and distraction checks

In parallel to this flow, agents constantly await if a slide change appears and respectively interrupt as soon as they notice. If recognized, agents individually check whether or not they are affected by it and decide if they should transition. If not, they remain in their current state and nothing changes. However, they immediately transition their focus to the presentation slides when affected. Subsequently, the system analyzes if the change corresponds to the last slide of the presentation and decides if the applause should be triggered. If this should be the case, agents start applauding as soon as the five-second timer is over. However, if it is not the last slide, the flow stops, informing the agent to return to the idle state after they finish reading it.

Presets

We created a list of presets based on the findings of Leysens et al.'s [LlRP16] work related to students' attention in class. Each list entry is only used once by randomly allocating them to our eight agents in the virtual room. Doing so allowed us to maintain the same level of attention while mixing agents' behavior to avoid making them always act the same. While this ensures that participants have a unique experience with each iteration, reducing the likelihood of them seeing patterns in the agent's behavior, it also guarantees that the attention level stays comparable between all conditions.

As described previously, the agent's personality is mainly characterized by four traits: phone distraction type d_{phone}, notebook distraction type $d_{notebook}$, participation modifier p, and movement speed modifier m. Together, those four traits build one unique preset entry, as shown in Table 4.2, listing them all. When taking a closer look at these presets, one can see that the participation modifier correlates with higher levels of attention, making more attentive agents the ones most enthusiastic while applauding. As for the movement speed modifier, we aimed to equally distribute the range of +/- 20% among all agents to allow for the most diversity considering their movement speed.

#	d_{phone}	$d_{notebook}$	p [%]	m [%]
1	never	never	100	-20
2	once	never	80	-15
3	once	never	60	-10
4	once	once	50	-5
5	once	twice	40	0
6	twice	twice	30	+5
7	regular	twice	10	+10
8	constantly	regular	0	+15

Table 4.2: List of predefined agent behaviors

4.3.3 Animations and Rigging

In parallel to the structure of the state machine, we developed a similar layout for the agent's animation trees. Figure 4.14 shows the overview of this structure based on two core parts: the *IdleStates* block and the *Applause* triangle. Thus, the question of where the slide and applause distraction animations are remains. Since we considered that neither requires a specific movement, we implemented them on top of the idle state. Respectively, only two animation states are necessary in the animation tree. As for the idle state, represented as a single element here, it contains a substructure of animations and transitions as shown in Figure 4.15.

The distraction level and participation modifier, which represent agent personality and modify their animations, are listed on the left side of the overview. Additionally, another value is listed there, acting as a flag to inform the animator to transition to the applause state after the timer expires.

The transitions between the states are indicated as lines with arrows pointing in their transitioning direction. While most transitions have a single arrow indicating their direction, one stands out for having three arrows along the transition line. This transition suggests the three idle states that transition from the idle states sub-tree to the applause animation flow.

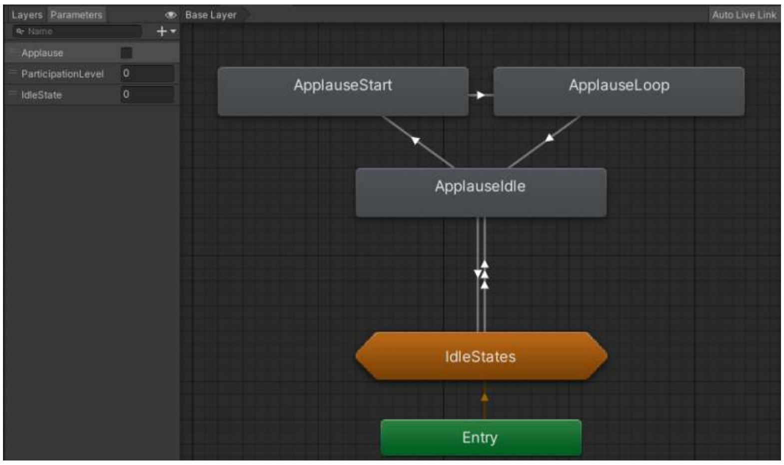

Figure 4.14: Animator overview

Idle Variants

We added three idle animations that change over time to increase the variety of looks while agents sit still and listen. As seen in Figure 4.15, all those states are connected and can transition to the *base layer* containing the applause animations. The starting point for all agents is idle animation zero, the one being the most basic, making agents sit upright and calmly breathe. Differently, the other two idle states make the agent lean back and cross their legs or lean forward with their hands on the table. Thus, they are less neutral animations and make the upright position act as the center between them.

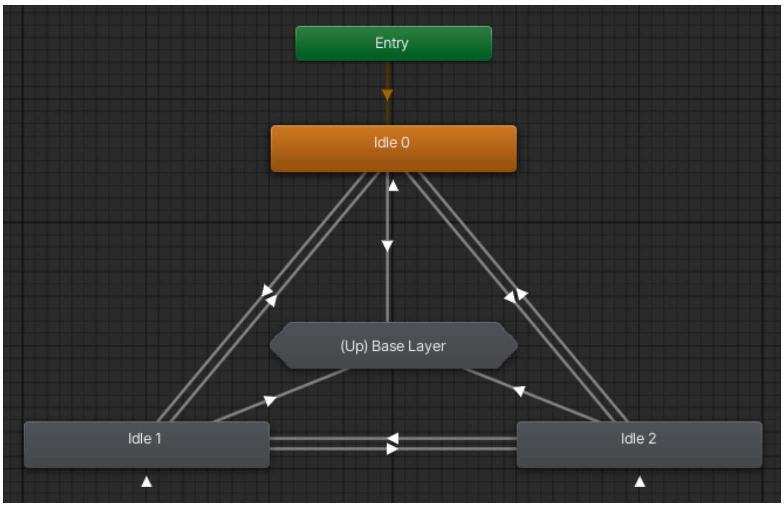

Figure 4.15: Idle animation tree

After each run through one of the idle animations, an animation event triggers, calling the agent's state machines *NextIdle(int currentState)* function, shown in Listing 4.1.

By calling this function, the next idle state gets calculated and triggers an idle state transition if necessary. Passing the current idle state while calling the function is required to prevent duplicate calls from any other idle animation. Since it might be the case that multiple animation events are triggered simultaneously while transitioning between them. Thus, ensuring that only the one coming from the latest idle state is handled is essential.

By including a probability of staying in the same idle animation for another animation cycle, the function calculating the next idle state prevents agents from constantly changing their animations. Therefore, a random number is generated and checked using a query. With this condition, we ensure that for 70% of the time, agents stay in the same animation. In the other 30%, split into two equal parts, the idle animation is changed to one of the two different states. Respectively, agents change their looks occasionally while not constantly moving around. Therefore, they look calm but not static in their behavior.

```
public void NextIdle(int currentState)
{
    // If the previous `_nextIdleState` is not equal to the 'currentState' passed by the caller, it means
    // that a different idle state is calling, making the call invalid.
    if (_nextIdleState != currentState) return;

    var rand = new System.Random();
    var newStateProp = rand.Next(101);
    var newState = currentState;

    // New state
    if (newStateProp < 15) newState += 1;
    // New other state
    else if (newStateProp < 30) newState += 2;
    // Else: stay in current state

    // Make sure the states are in bounds
    newState %= 3;
    _nextIdleState = newState;

    AgentAnimator.SetInteger(Constants.AnimatorParameter.IdleState.ToString(), newState);
}
```

Listing 4.1: Clapping sound logic

Applause

As shown in the overview, the applause animations formed a triangular connection. Splitting the applause into multiple parts was necessary because we had to separate the animation's starting, looping, and ending sequences. We also needed to ensure it looked normal to start applause transitioning from all idle states. Therefore, before starting and ending the applause loop, we made all agents run through the applause idle animation, making them sit upright and ready to applaud. Afterward, the clapping can start by making agents raise their hands in the *ApplauseStart* part of the animation.

Subsequently, the applause loop makes them clap repeatedly until the applause duration exceeds. Afterward, the transition back to the applause idle state, making agents lower their hands and get back to the idle animation variants.

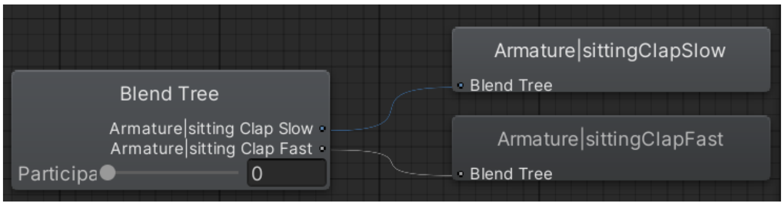

Figure 4.16: Clapping animation blending tree

As previously described, it would be weird if all agents looked the same while applauding. Thus, we introduced some variation in the applause states starting and ending points. However, we improved this by using the participation modifier to allocate a position in the transition, blending the two different applause animations, as seen in Figure 4.16. Each agent's state in the transition is set based on this predefined modifier value. Therefore, they all act uniquely since the participation levels differed for each agent, as described in Section 4.3.2.

Since clapping animation causes users to expect corresponding sounds, we introduced a simple *AgentSounds* script handling this logic. Utilizing Unity's animation events, we added triggers at each point in the applause animations where the hands are closed. At these points in the animation, the events call the *Clap(int mode)* function of the script while passing on a *mode*, as described in Listing 4.2. Since we blend between slow or faster clap animation, both can trigger the function simultaneously. Respectively, this would result in two clapping sounds playing simultaneously. Therefore, it was necessary to include the clapping mode information while calling the function to allow the decision of which of those two triggers should be played. Based on the participation level, we decided if the applause was more or less active and respectively dismissed the opposite function call. Thus, if the participation level was higher than 50%, the clapping events coming from the faster animation are considered a better match or vice versa. After deciding which of both modes is the more precise, a random clapping sound is played, selecting one out of six samples randomly modulated in pitch and volume to increase their variety further. Respectively, we ensured that no clapping sounds would constantly repeat themselves, thus making the applause sound realistic.

```
void Clap(int mode)
{
    var participationLevel = _animator.GetFloat(Constants.AnimatorParameter.ParticipationLevel.ToString());
    var clappingMode = (ClappingMode)mode;

    // Prevent multiple claps to occur if transitioning between clapping animations
    if ((clappingMode.Equals(ClappingMode.Slow) && participationLevel > 0.5)
```

```
        || (clappingMode.Equals(ClappingMode.Fast) && participationLevel <=
    0.5)) return;

    PlayRandomClap();
}

private void PlayRandomClap()
{
    AudioClip clip = clapSounds[Random.Range(0, clapSounds.Count)];
    _source.pitch = Random.Range(0.7f, 1.1f);
    _source.PlayOneShot(clip, Random.Range(0.5f, 1f));
}
```

Listing 4.2: Clapping sound logic

Head Redirection

Since our animations did not include head movement, agents constantly looked straight ahead by default. Therefore, we added an animation rig to the agent's heads, allowing us to dynamically modify their rotation at runtime. To make use of this rigging functionality, we had to include Unity's animation rigging package[8], containing all the necessary tools to create such behavior. Using the package's multi-aim constraint and adding it to the agent's head, as shown in Figure 4.17, we were able to modify their head rotation by making them aim towards *Source Objects*. Since the constraint allows it to be influenced by multiple sources, we considered it the optimal tool for replicating the behavior of looking at various points of interest. Consequently, we equipped each agent with a multi-aim constraint containing a list of focus points (source objects) relevant specifically to them. Since everyone has their own notebook and smartphone in front of them, the source objects deviated slightly from one another. However, they all share the same two core components: the presenter's head position (*CameraR*) and the position of the slide projection (*Projection*).

Each source object has a unique weight to describe its influence on the agent's gaze, representing to what degree it affects the constraint. Therefore, when setting the weight of a source object to one (its highest value), the constraint redirects the agent's head to look directly towards it. On the other hand, if agents are not interested in one of the source objects, their value must be zero.

Considering that each object's weight needs to be handled separately, we integrated a logic to ensure that only one focus point is active. Otherwise, an agent could try to look at multiple focus points simultaneously, resulting in intermediate states between those. Although this might look like agents daydreaming, users might also perceive it as weird. Therefore, we ensured this could not happen and restricted the constraints to prevent simultaneous focus points.

Alongside restricting source objects to be active once at a time, we integrated a way to transition between them smoothly. Therefore, we tried replicating Unity's animation

[8] https://docs.unity3d.com/Manual/com.unity.animation.rigging.html

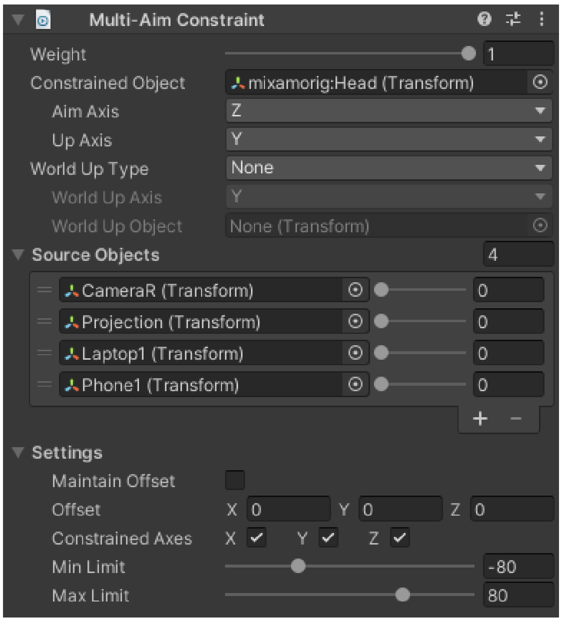

Figure 4.17: Agent's multi-aim constraint for head redirection

transitions by gradually changing the weights of an active focus point and its successor. Accordingly, we developed a *FadeIn()* and *FadeOut()* function as shown in Listing 4.3 to cross-fade between those. With these functions, the weights of both focus points are continuously updated over multiple frames. Thus making the change between them look gradual and smooth.

```
while (_fadeIn)
{
    ...
    // Fade in
    fadeInWeight = Util.InterpolateOverCurve(
            _transitionCurve,
            sources.GetWeight(_fadeInFocus.Value),
            1f,
            Time.deltaTime * _movementSpeedModifier);
        sources.SetWeight(_fadeInFocus.Value, fadeInWeight);
    // Update sourceObjects
    multiAimConstraint.data.sourceObjects = sources;

    // Check if either of the focusPoints is null, if so use the other for
    checking if the fade reached a certain threshold
    if (fadeInWeight > Constants.FadeInThreshold)
    {
        _fadeIn = false;
        _fadeInFocus = null;
        yield break;
    }

```

```
    yield return null;
}
```

Listing 4.3: Focus point fade functionality example

4.4 Interactions

To make interaction with SeminVR as simple as possible, we decided to make user input collision-based. Respectively, users would use their virtual hands to touch an interactable object and trigger its functionality. As shown in Figure 4.18, the hands are static models replicating a low-poly version of human hands. These models contained a collider property to make objects react to the hands touching them. As a result, the simulation could recognize when one of the hands hits an interactable object and make it respond to it.

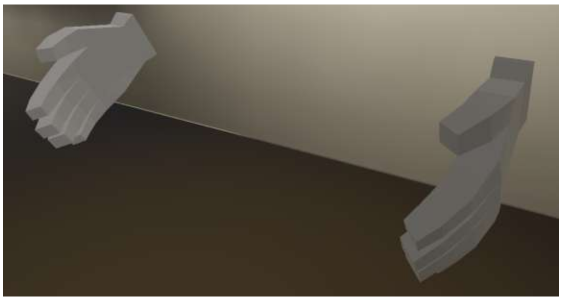

Figure 4.18: Models representing the user's virtual hands

Overall, the interactable objects in the simulation were kept to a minimum while providing all the necessary functionality for training presentations. Respectively, the presenter's table contained three primary tools: a computer, a timer, and a slide-control panel. Although minimalist, the three objects allowed the display of slides, let users navigate through them, and showed the time expired since the presentation started.

Presenter Computer The computer monitor's function was primarily displaying user slides. However, it also contained interaction functionality necessary for dynamic loading of the slides from a server. To do so, we equipped the monitor with a text input field, responding to the virtual hands colliding with it and opening the operation system's default keyboard. For the Quest 2, this results in a keyboard appearing in front of users, as shown in Figure 4.19. Using this keyboard, they can enter the path to where their slides are stored and confirm the input by touching the virtual keyboard model. Although the system keyboard contains an enter button, this event is not recognized by the input field, thus requiring users to confirm their intention of submitting the text by touching the virtual keyboard. After submitting the URL, the system recognizes the user's intent and attempts to load the slides from the provided path.

As soon as the application finished loading the slides, users were presented with their first slide and could see the world as shown in Figure 4.20. Respectively, they could

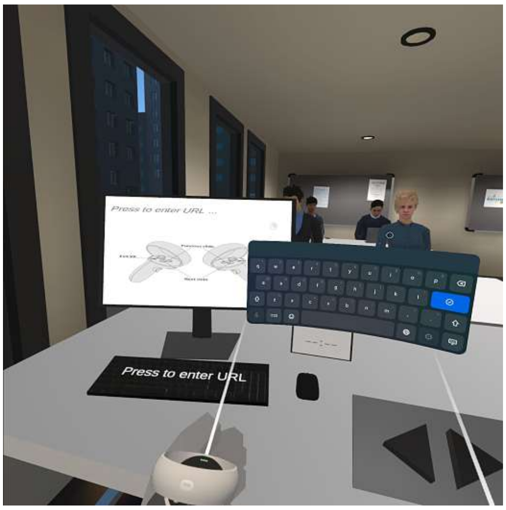

Figure 4.19: Entering file path to slide location

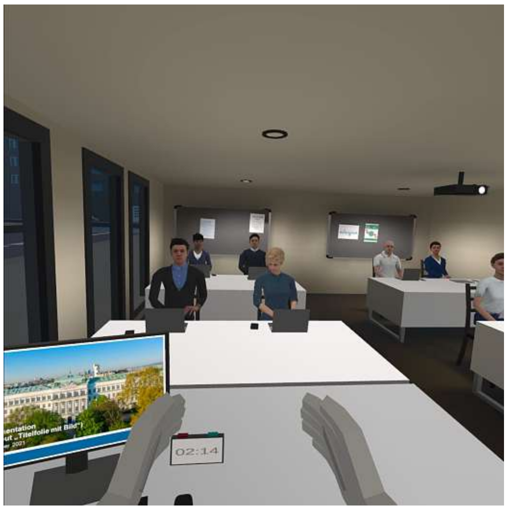

Figure 4.20: Presentation loaded and ready to start training

observe their slides on the monitor, see their virtual hands where they are holding their controllers, and face the audience ready to start presenting.

Timer To let users keep track of time and know how long they can go in their presentation, we added a timer to display how much time had expired since the start. As we integrated it to automatically start after loading the presentation slides, we needed to ensure that users could reset the timer if they intended to. We implemented it this way to ensure users are aware of the timer's existence and not require them to start time tracking independently. Consequently, we added the functionality to the timer's top left button to reset and restart the timer when touched. Additionally, the button on the right side was dedicated to pause and continue the time if one prefers.

Slide-control Panel Represented as a small panel containing two triangular buttons, as shown in the bottom right corner of Figure 4.5, we provided a simplistic tool that acts as the slide control. Pressing any of the two buttons changed the slides according to the direction they were facing. Requiring users to trigger the buttons physically to activate them was aimed at mimicking a motion similar to pressing a keyboard button when presenting in real life. Therefore, we considered the input method to support immersion by creating a parallel to the real world.

4.4.1 Controller Input

Alongside the possibility of interacting with the environment by touching objects, we also included some functionality for the controllers to allow an alternative way of interaction. Hence, we dedicated the controller's main buttons to changing the slides and trying to replicate the feeling of using a presenter remote. As shown in Figure 4.21, the layout was mirrored for both controllers, allowing users to choose their preferred side.

Alongside the description of the buttons dedicated to the slide controls, the figure also shows the essential system buttons and their functionality on our evaluation platform. While the menu button on a Quest 2's left controller usually acts as an in-game menu button, in the case of WebXR applications, it acts as a control for leaving the VR experience and coming back to the web browser view as shown in Figure 4.22. For the other controller, the same button opens the system menu, informing about the currently running application. Using this menu, users can also exit the application by confirming an alert dialog to stop the experience.

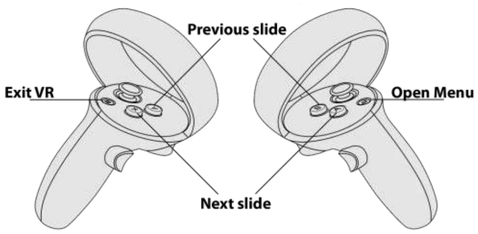

Figure 4.21: Quest 2 controller button layout

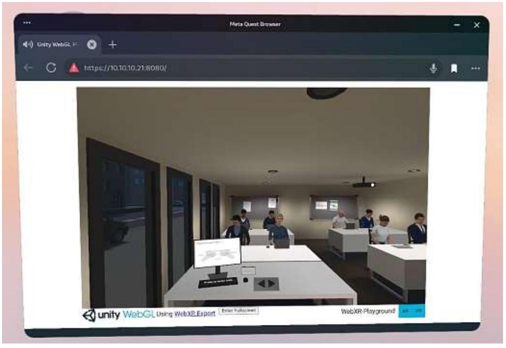

Figure 4.22: Meta quest browser web view showing SeminVR

Since the WebXR Exporter project did not yet support showing the platform-specific controller models, users only saw the virtual hand models inside SeminVR. Thus, we included a slide in our onboarding dedicated to informing users about the controller layout and letting them explore the interaction capabilities before the evaluation conditions. However, this still did not eliminate the need for them to remember the controller layout and actions behind each button. Therefore, we also provided the virtual slide control panel as an alternative in case they were unsure.

4.5 Slide Integration

To allow SeminVR to load each user's presentation on demand dynamically, we decided to store the slides separately from our application and access them using a file system. While this can be done by accessing a platform's local file system, we went for a more elaborate approach of hosting a local server. Since SeminVR is a WebXR application, it already requires hosting on a server to make it available on the local network. Thus, we decided to reuse the server to integrate slide loading. Consequently, unlike accessing files locally over a distinct path, we had to incorporate some logic to interact with our server and access its content using HTTP GET requests. While hosting slides on a server introduces more implementation effort, it comes with the benefit of building the foundation to let SeminVR run on a public web server. Thus, it makes the application more future-proof by reducing the adaption necessary when intending to distribute it publicly.

4.5.1 Hosting Local Server

For hosting a local server, we decided to use *http-server*[9], a GitHub project by http-party providing a simple web server with little configuration needed and containing all functionality required for our use-case.

To make it run on our Windows machine, we had to ensure that *nodeJS*[10] was installed to allow installing http-server using our terminal and running the command shown in Listing 4.4. Although other ways exist to install the project on a Windows machine, we decided to go with npm as it seemed the most straightforward.

```
npm install --global http-server
```

Listing 4.4: Install http-server

Although having a locally hosted HTTP server is sufficient for many WebGL projects, we must provide an HTTPS connection to allow the browsers to WebXR's capabilities to communicate with VR headsets. Since accessing VR headset's camera feed and sensors using an unencrypted HTTP connection is considered a security risk. Consequently, an HTTPS connection is required. Otherwise, consuming WebXR content using a VR headset would be impossible.

As a result, we created our own SSL certificates for our local hosting, which is necessary for providing such a secure connection. Listing 4.5 shows how we generated those certificates by running various commands in our terminal. First, we needed to navigate to the folder where SeminVR was stored. Subsequently, we created a folder to hold all the SSL certificates necessary for our HTTPS connection. Afterward, we navigated into this folder and started creating our certificates. Therefore, steps three and four show that we first made a root and server configuration. Subsequently, those were used to run the algorithm for creating the SSL certificates.

```
# 1) Navigate to the project's 'build' folder containing the application
cd ~/<projectName>/build

# 2) Create a certificate folder and access it
mkdir cert
cd cert

# 3) Create a 'root.cer'
openssl req -x509 -new -nodes -keyout root.key -out root.cer -config root.cnf

# 4) Create a 'server.cer'
openssl req -nodes -new -keyout server.key -out server.csr -config server.cnf

# 5) Create a 10-year valid SSL certificate using server and root certificate
openssl x509 -days 3650 -req -in server.csr -CA root.cer -CAkey root.key -
    set_serial 123 -out server.cer -extfile server.cnf -extensions x509_ext
```

Listing 4.5: Create certificates to host server as https

[9] https://github.com/http-party/http-server
[10] https://nodejs.org/en/download/

While those certificates were not certified by an external institution, web browsers considered the connection insecure. Thus, they prevent SeminVR from automatically being loaded. Although browsers inform about the website's security risks in this regard, one can confirm an alert making the browser load the side anyway. Even if browsers considered our certificates insecure, having those allowed us to access the headset's VR capabilities and make our WebXR application fully functioning for our local network.

After completing those prerequisites, we were ready to start our local server using the commands shown in Listing 4.6.

```
# Get to the project's build folder
cd ~/<projectName>/build

# Start http-server in "secure" mode with the previously created certificates
# Additionally, allow CORS to enable accessing the presentation slides
http-server -S -C ./cert/server.cer -K ./cert/server.key --cors
```

Listing 4.6: Start local https-server

After starting the server, SeminVR could be accessed by opening a browser window and entering the server's IP address and port. If the server is running on the same machine, this can be easily achieved by entering the local host address and adding the HTTP server's default port at the end. However, to access SeminVR from another local device, it was necessary to know the hosting machine's IP address. Listing 4.7 describes both ways of accessing the server. Alongside knowing which IP address the server is running on and what port it uses, it is essential to add the *"https://"* prefix while entering them in the address bar, as otherwise, the server is not accessed via the HTTPS connection. Thus, the WebXR capabilities would not be available again.

```
# Access the hosted web server from the local machine
https://127.0.0.1:8080

# Example for accessing the web server using the hosting machine's local IP
  address
https://192.168.0.1:8080
```

Listing 4.7: Access SeminVR via browser

4.5.2 Slide Loading

We created a folder within the application's root directory named *Resources* to host our user's presentation slides on the server. Since Unity does not support working with any presentation software's file formats or PDF, we needed to require presentation slides to be in an image format as those can be easily integrated into our application. Respectively, each presentation consisted of multiple images representing its slides. Therefore, we added a folder for each participant.

To access those image files within SeminVR, we integrated a text input field allowing us to enter the exact location where those are located. Knowing the server IP and port alongside the location of the slides, we could call the server using an HTTP GET request and ask it to provide the contents of the transmitted location. Utilizing Unity's web request integration, as shown in Listing 4.8, we created a function running as a coroutine, automatically awaiting the server's response and handling its results accordingly.

```
private static IEnumerator GetCoroutine(string URL,
                                        Action<string> onError,
                                        Action<string> onSuccess)
{
    using (UnityWebRequest unityWebRequest = UnityWebRequest.Get(url))
    {
        yield return unityWebRequest.SendWebRequest();

        if (unityWebRequest.result == UnityWebRequest.Result.ConnectionError
            || unityWebRequest.result == UnityWebRequest.Result.ProtocolError)
        {
            onError(unityWebRequest.error);
        }
        else
        {
            onSuccess(unityWebRequest.downloadHandler.text);
        }
    }
}
```

Listing 4.8: Regex for scraping the slide folder contents

The server responded to this GET call by returning the HTML page representing the submitted paths folder structure. Based on this page, we could identify each slide within the folder by scraping the page content using the regular expression shown in 4.9.

```
// Regex to identify the URL underlying the links to each slide image
public const string DefaultImageScrapeRegex =
    "<a href=\\\".*\\\">(?<name>.*\\.[^\\/]*)<\\/a>";
```

Listing 4.9: Regex for scraping the slide folder contents

After extracting all the slide-names from the page, we could load their contents utilizing *UnityWebRequestTexture*, as shown in Listing 4.10. With this method, it was possible to get the texture contents of an image file based on a provided URL. Consequently, we iterate over all slides within the presentation folder and download their contents. The returned textures from this call were then stored locally in the application's cache for further processing.

After all images were cached successfully, the first slide's texture was applied to both the virtual screen and the projection of the wall. Respectively, it indicated that the loading process was complete, and users could start with their presentation.

```
private static IEnumerator GetTextureCoroutine(string URL,
                                               Action<string> onError,
                                               Action<(string url, Texture
    texture)> onSuccess)
```

```
{
    using (UnityWebRequest unityWebRequest =
            UnityWebRequestTexture.GetTexture(url))
    {
        yield return unityWebRequest.SendWebRequest();

        if (unityWebRequest.result == UnityWebRequest.Result.ConnectionError
            || unityWebRequest.result == UnityWebRequest.Result.ProtocolError)
        {
            onError(unityWebRequest.error);
        }
        else
        {
            DownloadHandlerTexture downloadHandlerTexture =
                unityWebRequest.downloadHandler as DownloadHandlerTexture;
            if (downloadHandlerTexture != null)
                onSuccess((url, downloadHandlerTexture.texture));
            else
                onError("DownloadHandlerTexture is `null`!");
        }
    }
}
```

Listing 4.10: Function to load textures from the server

4.5.3 Change Slides

Going through the slides was achieved by keeping track of the current slide number and accessing the list of textures at the corresponding position, as shown in Listing 4.11. Therefore, when users intended to change the slide, the slide count was updated accordingly, and the texture of the list at the count position was applied. When the end of a presentation was reached, the counter was reset to either the first or last slide, depending on which direction the slide change was happening.

```
private int _currentSlide = 0;
private List<(string name, Texture texture)> _slides;

private void ChangeSlide(int direction = 0)
{    ...
    // When at the start of the list AND would step back
    // --> Show last slide
    if (_currentSlide == 0 && direction < 0)
        _currentSlide = _slides.Count - 1;

    // When at the end of the list AND would step forward
    // --> Show first slide
    else if (_currentSlide == _slides.Count - 1 && direction > 0)
        _currentSlide = 0;

    // Otherwise, just move back & forth between the slides
    else
        _currentSlide += direction;
```

```
    ...
    // Apply the current slide
    simulationManager.monitorScreen.texture = _slides[_currentSlide].texture;
    if (simulationManager.externalScreen) simulationManager.externalScreen.
    texture = simulationManager.monitorScreen.texture;
    ...
}
```

Listing 4.11: Change slide

CHAPTER 5

Findings

The results of our study provided two foundational perspectives on our participants' perception of SeminVR. On the one hand, the analysis of the interviews provides a qualitative angle on what they perceived. On the other hand the questionnaire results deliver additional details in a quantitative manner. Nonetheless, the focus was on understanding and explaining each individual's view and what influenced it. Thus, our priority was on the qualitative part of the analysis. Therefore, we analyzed the interviews using TA, which allowed us to develop a more profound understanding of our participant's opinions.

The following section starts with presenting demographics and how they were balanced between groups. Afterward, we detail how we developed a more profound understanding using TA and how it was conducted. Subsequently, its findings are presented, giving elaborate insight into our study. The chapter concludes by describing the development of quantitative values and their relation to our qualitative results.

5.1 Demographics

Our study included 10 scholars from Austrian universities; all of them completed the entire study. Respectively, 10 answers are included, separated equally between the control and experimental groups. Having participants indicated their age within five-year ranges, they collectively reported being between 23 and 27 years old. Additionally, they identified themselves as either men or women since no answers were given for inter, divers, open, or no entry. Moreover, they primarily indicated being experienced in gaming while having little to no prior VR experience. However, those minor differences were balanced between groups. Thus inferring that no effects should have resulted solely from different levels of knowledge with the medium.

5.2 Thematic Analysis

To allow for a better understanding of our process on how we were seeking knowledge in our interview data and what steps were involved in producing our results, we will elaborate on how we applied TA in our study. Since the method highly relates to personal experiences and perspectives, we seek to explain the ideas and thoughts that led to our final view of the evaluation results.

5.2.1 Generating Knowledge

Code Finding

As usual for TA, our first step was to get acquainted with the data. Therefore, we decided to transcribe all our interviews upfront to allow a more accessible exploration of what was said and seek patterns in the data.

We decided to transcribe all the interview data using software, as it is more efficient than doing it by hand. While most common transcription solutions require uploading audio files to a web server, we sought a way to do this locally and without the necessity to share our data. Especially since we wanted to be GDPR compliant, our goal was to find a tool that would not require our data to leave our computer. Respectively, we found OpenAI's project known as *Whisper*[1]. It is a *"general-purpose speech recognition model"* also used in YouTube's automatic transcriptions. Nonetheless, unlike YouTube's approach, Whisper runs on a local machine. Therefore, it allowed us to transcribe our interviews using our computer without uploading our audio files to any server.

We merged the resulting transcripts into four files, connecting each group's interviews per condition. This allowed us to see how each group changed over the two conditions and what the differences were between those groups. Since qualitative analysis like TA highly relates to people and their perspectives, we decided it is also essential to understand the differences between the groups. Thus, our approach of handling each group-condition combination as a separate piece was further supported.

Respectively, this also meant that we looked at each of the four interview sets and collected codes from them separately. Nevertheless, as we identified later, many of those codes overlap between groups, though still slightly differ from on another. We further noticed that analyzing codes for one piece influenced our perception, making us more likely to see similar patterns in the other conditions. Nonetheless, this aspect is due to the nature of this kind of analysis, making it acceptable.

Coding

After reviewing all the data and taking notes of repeating patterns, we reiterated their validity and usefulness before starting coding. At this stage, some were generalized to be more versatile and broadly cover an area. An example of that would be the transformation

[1] https://github.com/openai/whisper

of our initial code *stress level* to change to *comfort*. We adjusted this code since our participants did not exclusively talk about stress but also other feelings indicating their comfort. A list of the initial codes found in our interviews can be seen in Tables 5.1 and 5.2 showing the previously mentioned similarity of codes between the two groups.

For coding our transcripts, we highlighted and took notes in our files using *Adobe Acrobat Reader*[2]. Since it provides a wide variety of highlighting colors and a way to filter those accordingly, we decided it was best to use this software. Although other solutions exist, being supported on almost every platform was further considered a benefit of this application, as we liked not to be restricted to analyzing our data on just one operating system.

First Interview	
Group A	**Group B**
Knew that not real	Knew that not real
Compare to real	Compare to real
Understand agent	Understand agent
Comfort	Comfort
Distractions	Distractions
Visual Clarity	Visual clarity
Preparation	Preparation
Attention	Attention

Table 5.1: Codes found in the first round of interviews, separated per group

Second Interview	
Group A	**Group B**
Knew that not real	Knew that not real
Compare to real	Compare to real
Understand agent	Understand agent
Comfort	Comfort
Distractions	Distractions
Different confidence	Different confidence
Different focus	Different focus
Self-reflection	Visual clarity
More like presenting	

Table 5.2: Codes found in the second round of interviews, separated per group

After successfully coding all the interviews, we added a comment for each highlighted quote, simplifying its contents using the participant's words. The simplification was necessary to reduce the information in the participant's feedback to the essence and keep notes as simple as possible. Thus allowing us to display them easily on post-its later.

[2] https://www.adobe.com/at/acrobat/pdf-reader.html

Additionally, we added a code at the end of each note to ensure we could reconstruct its origin. This code was a combination of the participant's group, which round the quote was from, and the participant's identifier. In combination with a color coding for each code mentioned in Tables 5.1 and 5.2, it allowed us to quickly identify the statement in our transcripts and check the validity of the argument if we were uncertain. An example of such a simplified version of a participant's quote can be seen in Figure 5.1 showing a quote from participant *P03* of our control group's (Group A) first round of evaluation.

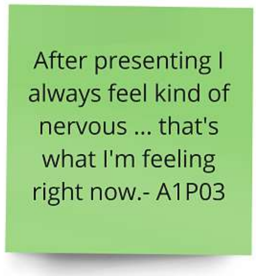

Figure 5.1: Example for simplified quotes used for further analysis

Grouping and Identifying Themes

For easier management of our notes and accessible exploration of our data, we decided to work with a virtual board using *miro*[3]. It allows the organizing of virtual post-its without space restrictions, giving them different colors, drawing connections, and multiplying them on various virtual boards. Thus, we preferred using the software-based approach over the traditional hand-written one to explore our data more freely.

Having our notes on the virtual board sorted and colored relative to the code found in the statement, we combined them into themes. We linked the notes to convey a story and tell more about the participants' feelings. Thus, initial themes emerged. As seen in Figure 5.2, those themes were closely related to the quote's associated code and acted as a way to separate each statement's positions better. Especially considering this was the first iteration of seeking knowledge, the primary goal was to organize and cluster participants' arguments and their relations.

Finding Connections

After we found our initial themes and allocated the notes to the best-matching one, we considered connecting those themes into a knowledge network. Therefore, we combined them on a level of relationship and how they can convey a story from our participant's perspectives. These networks were created for each of the four analysis conditions, resulting in multiple distinct webs of knowledge containing their own stories. However,

[3]https://miro.com

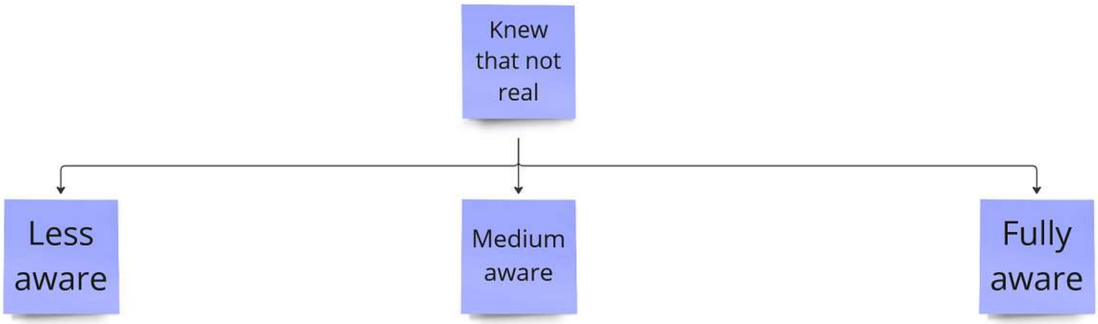

Figure 5.2: Example of an initial theme found in the data

due to similar codes used, some parallels remained. An example of connections found at that state was the relationship between our theme *knew that not real* and *compare to real*. While both handle different facets of participants' perspectives, they stand in close relation to one another, which indicates a connection between them.

We did expect a general overlap for our four networks since we asked similar questions in the interviews for both conditions. Respectively, each of the knowledge networks contained the three major topics of our interview: *comfort*, *knew that not real*, and *understanding agents*.

The first theme primarily refers to what participants reported to us about how they felt during the presentation, while *knew that not real* conveys the arguments on how they perceived their performance concerning presenting in the VE. Lastly, *understanding agents* grouped all things mentioned towards how participants perceived the audience and what they observed.

While other themes changed or merged into something else over the iterations of our knowledge formation, those three themes were the ones that remained the same. Therefore, they formed the foundation of our findings. Furthermore, those themes acted as cornerstones for the other topics. Thus, connections like, for example, the relationship between *knew that not real* and *missing parts* as well as *comfort* and *trained before?* were formed for each group's initial condition.

Since connections were unique for each condition and evolved differently, further details would go beyond the scope of generally explaining the process of finding connections. Moreover, we must point out that finding connections was already inspired by the idea of knowledge presentation afterward. Respectively, themes covering significant findings in our data were prioritized and connected to those supporting their stance.

Summarize and Reorganize

After successfully finding connections between the themes and organizing them to form a cohesive story, we reduced each theme's quotes to the most valuable arguments. Alongside eliminating duplicates, this step focused on strengthening each theme's validity and identifying significant aspects. Therefore, we further summarized our networks to the most relevant topics, reorganizing themes and connections to match each knowledge network's new state. In addition, we rearranged arguments if we felt they were better suited elsewhere.

As a final step, before comparing the conditions, we searched for headings and statements best representing the argument-theme relationships, acting as sub-themes. Those conveyed our understanding of each of those relationships and helped us to compare the conditions more efficiently, as they already contained some interpretation of the data. An example of those can be seen in Figure 5.3, where we grouped each statement for our theme *Parallel to real situation* to the sub-themes *Had similar experiences* and *Behave like I normally would*.

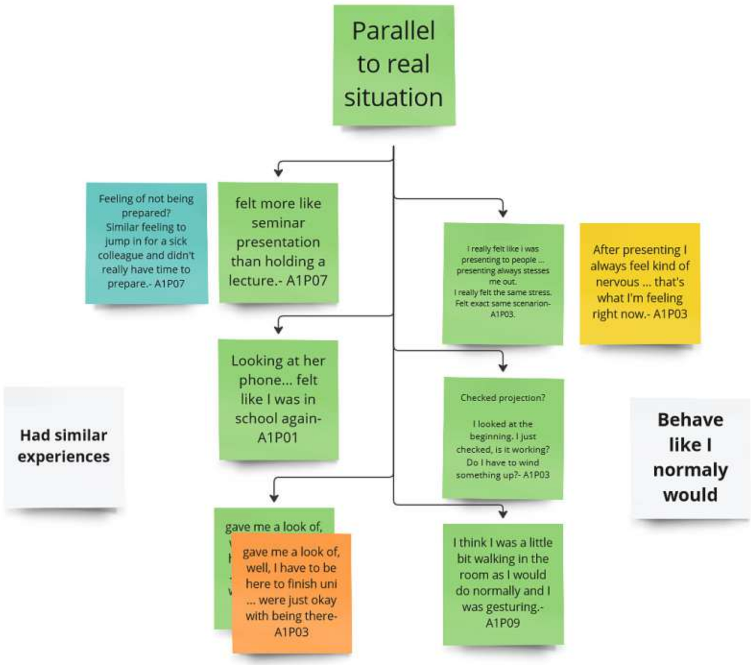

Figure 5.3: Sub-theme example for our "Parallel to real situation" theme

Comparing Conditions and Groups

Having the argument-theme relationships, as described above, allowed us to form a storyline of what we found for each evaluation condition. Comparing those for each group individually and between the groups showed us how different they were and allowed us to form a general understanding of our evaluation findings.

The results of these comparisons are described in the following sections, which contain all the details of what we found and why we see it that way. To allow an understanding of how participants' views changed over the evaluation, we first will discuss the findings of the initial condition and compare them between both groups. Subsequently, the findings of the second condition are elaborated on and put into relation with the things initially found for each group.

5.2.2 First Presentation

The participants' first impressions of presenting in SeminVR provided details on what they noticed and how they felt holding a presentation in VR. They reported things that discussed the application's realism and comfort and elaborated on its immersiveness. Overall, we perceived a mixture of feelings as some reported it to be easy to present in VR, while others were less comfortable being stared at while training for the first time. Nevertheless, a feeling of a safe space to train was transmitted, which made discomfort less severe.

Awareness of Simulation

Independent of the participant's confidence, arguments were made about how realistic the training felt and what they experienced. Generally speaking, our control group reported more positive feelings toward realism than the experimental group did. Nonetheless, they shared a similar understanding of why and how SeminVR felt realistic and why it did not.

In this regard, we observed that the onboarding and phrasing of their task introduced some bias. Since we told participants what they could expect in the simulation, they already knew that the audience consisted of agents. Primarily, since we used the terms *virtual audience* and *simulation*, we made them aware upfront that they would not perceive an authentic audience, thus having them report that they knew it was not real.

Nonetheless, this knowledge did not prevent participants from getting immersed in the training application. While all of them knew it was not real, the level of how real it felt for them was different overall. Some participants described their experiences as *"felt like you were actually there"*, while others commented on the audience that *"they are not feeling like real people"*. Many described an urge to give their best effort and do well in their presentation since they felt the audience was listening and interested in what they were saying. On the other hand, others mentioned that they did not think the audience was curious or even listening to them. Nonetheless, this feeling did not result in them having a less realistic experience presenting in the virtual space, as they elaborated on the feeling of being in the virtual world.

Realism

Across all participants, there was a balance on how realistic it felt to hold their presentation in a VE. Many of them compared their presentation to prior experiences, things they remember from real scenarios, and similarities they observed while using SeminVR. Overall, those feelings resulted in more comfort and them acting like they normally would.

Respectively, participants mentioned things like moving around a little in the virtual space and pointing at the slides to reference a picture. On the other hand, others pointed out that *"you can tell that the people are just some sort of program"*, and that they knew that they were not in a real scenario. Nevertheless, this did not deter them from getting immersed - *"I was aware that it is not a real situation, but I got nervous about it, which usually is not the case when I prepare for a presentation."*.

We furthermore observed that our decision to design the application to look less realistic, combined with participants knowing that it is a simulation, played out in favor of an increased feeling of security. One participant described it accordingly, *"it did not feel uncanny because it was not as realistic"*. However, they also added that the audience still had this *"human-esque feeling"* and that it was not like *"there was sitting a dragon and a cat"*. Overall, this allowed them to immerse themselves in the virtual world while not entirely disregarding the fact that SeminVR mocks the real world. Respectively, it was acting as something like a safe space where they could try out things.

Telling us about how they could experiment and not need to fear consequences indicated further comfort with the situation and showed us their enjoyment of the experience. However, though some of them still had this secure feeling, they were not resistant to stress and felt slightly uncomfortable occasionally. Especially when they did not know what to say or had to rethink the point they wanted to make, training their presentation in front of an audience increased their discomfort and made them nervous. Nevertheless, this discomfort was of short duration as they clarified *"it is always like 30 seconds where it is a bit more difficult, but would not call this stressed"*.

Understanding the Audience

When we asked our participants about the virtual audience and how they perceived it, their responses varied. From descriptions of them being *"nice listeners"* to saying *"they were genuinely paying attention"*, a broad spectrum of answers were given. Having one thing in common across all interviews, participants commented on the agent's calm behavior. Hence, describing how they looked around, checking the slides, and looking back at them was included in most participant's responses.

However, their interpretation of this behavior was mixed. While some did understand the agents mainly focusing on the speaker and looking around sometimes to be realistic, others perceived it as them not doing much. One participant even described the feeling as agents behaving like they knew their teacher was watching, which forced them to pay attention. Nonetheless, most agreed they did not observe them as an uncomfortable audience.

Missing Emotions

A fascinating thing we noticed while interviewing participants was that, independently of whether they perceived the audience as more or less realistic, they described their looks as staring. Although some still used the terms looking and watching, they very commonly used the term staring. Since we already observed this phenomenon in our pilot study, we were curious about elaborating what makes them say so. While a few could not tell a reason for using the term over another, a couple of participants explained this feeling to come from the absence of emotions transmitted by the audience. One participant described this situation in a nutshell, as they could not read the feelings of an audience member - *"I was not able to read which feelings he had about the presentation. I was not able to see if he is interested, enjoys it, or I do not know... wants me to say something interesting"*.

Similar to this argument, other participants also elaborated that they thought it had to do with the absence of facial expressions. Since our agents' faces are just static images, not changing in any way, neither smiling nor blinking, simply not showing any mimic. We consider this to be the cause of participants feeling this way. However, since most of our participants mentioned that they looked past it, we understood that this did not significantly reduce their immersion in the virtual space.

Missing Sounds

Another aspect participants noticed while performing their presentation was the calm environment. Not only was the audience behaving composedly, but there were little to no sounds from either them or the environment. As one participant explained, they noticed the absence of *"some clicky noises when someone is writing on the laptop"*, which made the training less realistic for them. Additionally, they expected some coughing, clicking, and other *"little noises people do every day"*. Moreover, one participant expected parts of the audience to maybe chat with each other, as they usually do during presentations. Nonetheless, others enjoyed this particular calmness of the environment and argued that it made them feel the audience was listening to them.

However, it must said that only half of the participants mentioned that the sound setting was minimalistic. Especially considering the outdoor atmospheric sounds, it was interesting to hear only one participant describing that they noticed cars on the street. While it is not certain that others did come across the more traffic-prominent parts of the outdoor atmosphere track, only one participant reported that they noticed it. Although this participant said it did not bring them out of the concept, they were uncertain of a car going by since they saw no movement on the streets.

Audio/Visual Mismatch

Although the environment and sounds were designed to be very calm, only containing minimalist atmospheric indoor sounds like closing doors, ventilation, and walking noise, the outdoor sound setting was slightly different. Replicating the atmosphere in a town center, sounds like people talking, birds chirping, trains moving, and cars going by were common in the audio track. While those are familiar sounds in an urban environment, they usually come with a visual match, allowing participants to hear and see what is happening. As one participant pointed out, *"I heard some car going by and I did* *not see some"*, we understood that SeminVR is missing some visual feedback for the outdoor sounds. Although only one participant observed this in the first round, it later showed that others had the same feeling and got distracted during the second evaluation condition.

Interestingly, this was only true for car noises. Things like people talking on the street or birds chirping did not create the same kind of distraction, as participants were accepting the noise and not looking for a visual match. Only in the case of cars going by, participants were tempted to see where the sound came from, making them look outside the windows. Since the virtual seminar room is located on street level and faces the road, participants must have expected at least to observe a matching movement in their peripheral vision. As this was not the case, they got distracted by it and reported it during the interviews in the second round.

Comfort

Regarding comfort, a mixture of answers was given throughout the interviews. While some parts of SeminVR seemed stress-inducing for a couple of participants, others did not even care about those. Things like the audience looking at them while presenting were perceived as the people wanting the presenter to say something interesting, while others interpreted them as nice listeners.

In other situations, participants were distracted by the virtual slide controls as they got irritated about whether to use the controller or the virtual buttons. However, others reported it the other way around, so they did not feel stressed at all *"because it was easy to know how to switch the slides"*. Thus they had no trouble working with SeminVR.

The latter example confirmed our position on the necessity of prior training since, even with our onboarding, participants still needed to discover how to use the application and what they could do with it. Nevertheless, the onboarding prepared them to know all the tools and interactions necessary for an unguided experience, allowing them to give their presentation without further support. Respectively, our participants reported overall comfort using the application and did not mention severe issues using SeminVR.

Technical Difficulties

Although everyone used the same high-performing condition, two participants reported issues concerning the application's visual clarity in the first round. In this regard, they mentioned that they had trouble reading their slides from the virtual monitor as it was *"a little bit unsharp"*. Nonetheless, they further explained that this issue was short-lived, as they discovered that getting closer to the screen helped get a clearer image. One of them even thought it was because of their glasses instead of coming from the technology. However, they were not sure about it.

Being an exception, one of our participants reported discomfort from wearing the VR headset, describing it as *"pretty heavy"* and arguing that they had a full of head feeling due to displays right in front of their eyes. Nonetheless, after informing us about this, they added that they generally felt pretty okay during the experiment, which we also could confirm by checking the values noted in their SSQ.

Unfortunately, the participants who mentioned issues with visual clarity and one who did not feel as comfortable with the VR headset were all in the experimental group. Thus, we must remember this circumstance while analyzing the second condition, especially since none of the other group's participants had mentioned anything similar to what they had noticed.

Preparation

During our interviews, we became curious about why some people are more confident presenting in the virtual space than others. Therefore, we asked further questions to discuss why they felt one way or the other. Alongside asking those, we noticed that most of our participants did not rehearse their presentation before coming to the evaluation. Nonetheless, one participant in each group did.

As we informed them to prepare a presentation and that they are training to hold it in a virtual space, we should have additionally clarified whether they should rehearse it before coming to the evaluation. Consequently, participants interpreted it differently since some came prepared and some did not.

We came to this assumption based on another unexplained thing we noticed in our invitation - *"What language should the presentation be in?"*. Since all our participants were German-speaking natives, the question arose about whether the presentation had to be in English. Under the consideration that all the information provided to them concerning the study was in English, we did not think it necessary to clarify the study's language further. Additionally, we assumed that no matter which language they presented, the most important thing was to consider that everyone presented the same topic to make results comparable. However, we later realized that having all participants present in one language would have also been ideal. Similarly, we considered the same to be true for requiring participants to train their presentation before the evaluation or asking them not to.

Still, an equal number of participants presented in German and English, creating a balance between groups. More precisely, two participants in the control and three in the experimental group presented in English, while the rest did it in German.

However, not presenting in their native language created situations where they could not find the right words and rethink how to phrase what they wanted to say. Respectively, they argued towards their presentation performance that, *"I think it could have gone better, but I think there are a lot of language difficulties"*. Thus, we believe that the presentation language played a significant role in the participant's confidence to hold their presentation in SeminVR.

Level of Experience

While interviewing, we sometimes felt that participants were so amazed by VR that their enthusiasm could have influenced their answers. Analyzing the demographic data showed that half of them had never used VR before, while the rest only did rarely. In this context, most of them explained that their choice of selecting *rarely* was due to it being the next smallest option since they only tried it out once or twice. However, this intention differed for two of these participants, as they mentioned that they had used the technology a couple of times already, not only once or twice. Fortunately, the experience level with VR was divided equally between groups, supporting that differences in group results were not solely based on prior experience.

Overall, this low level of previous experience with the technology supports the assumption that there was an increased amount of excitement and a high level of dedication to perform during the first evaluation condition since it was something new for participants to try out.

5.2.3 Second Presentation

During the second round of evaluation, our participants again had a mixture of feelings toward using SeminVR. Overall, they somewhat remained the same for our control group, while the experimental group reported slightly more changes. Nonetheless, we recognized a stronger tendency of participants to analyze the application's functionality, as they focused on describing how they perceived the audience changed. Additionally, comfort-wise, a difference was observed between the two groups as well. Thus, performance has a potential influence on the applicability of training. However, we believe this difference was mostly related to the experimental group's participants being more analytical than the others, as the previous condition indicated.

Bias from the First Round

One of the most prominent things observed in our data was the development of the participant's focus throughout the evaluation. For the first presentation, participants were encouraged to learn how to present in the environment, explore how it made them feel, and train their presentation. *"The fact that there were people forced me to look at them more, but I always had to look down to the slides"* best describes participant's levels of encouragement to do their best while giving the presentation. However, their focus changed for the second round, and they became more analytical about the environment. In some cases, this change was so severe that they even reported that they forgot about their presentation at some point - *"I was even so into paying attention to the audience that I even forgot my last slide"*.

Reflecting on the first condition, we identified participants' focus on training due to their tendency to explain their feelings and how they experienced their presentations. Although the interview questions were the same for the second round, it was noticeable that they were more focused on explaining to us how they perceived the experience. We observed this effect across both groups, where many reported feeling nervous, stressed, and excited while using SeminVR for the first time. Quotes like *"I was a little bit nervous at the beginning, but I felt relief after the first slide"* and *"After presenting, I always feel kind of nervous still, that is what I am feeling right now"* were common among their descriptions of how they felt initially presenting in the VE.

Nonetheless, this feeling was different for everyone. While the excitement for trying something new was observed all across participants, some did not perceive it as stress-inducing. Some elaborated, *"Considering I did not rehearse, I felt really secure"*. However, it did not mean those people felt SeminVR was less realistic. Instead, some seemed more confident talking to an audience, which they also mentioned during the interviews. One of them told us even, *"I might not be the best person to ask because talking in front of people is basically part of my job"*. Additionally, another participant responded as we asked them if they felt confident talking to the audience, *"I prefer presentations over anything else"*. Nonetheless, those two participants were an exception. Overall, we observed different levels of confidence across participants. Fortunately, we could identify those variations in confidence were balanced between the two groups, thus not distorting our results.

Regarding the second round, participants already knew what to expect. They had used the application before and were already more confident working with SeminVR. As we noticed this tendency of them being more confident the second time, we asked them to explain why. Respectively, a participant described, *"I already did it and practiced it once, so I knew what phrases I want to use"*. Since nearly all of them reacted similarly, we understood why some of them drifted away from presenting and put their focus on analyzing the application. One participant explained this understanding in a nutshell, *"I felt quite good prepared for the presentation... I thought I can a little bit more check the audience"*. Thus, we understood why their responses contained extensive details on the application rather than reporting how they felt during training.

Training Effect

As participants focused more on providing us with further details on what they perceived using SeminVR, we observed changes in their presentation performance accordingly. However, only one participant explicitly mentioned that their presentations could have gone better due to their change in focus. For the rest of them, this change seemed not to primarily influence them while presenting. Nevertheless, their performance still varied, similar to the first condition.

Although some said they already knew their slides and what they wanted to say, they felt they performed worse than the first time. One of them described it as coming from being very self-critical, which made them unhappy with their latest performance. Another participant mentioned that they struggled to find the right slide and had to jump back and forth in their presentation, making them uncomfortable and insecure. Respectively, they were happier with their previous presentation.

Nonetheless, others described a feeling of accomplishment due to their improvements the second time. In this regard, a participant in the experimental group elaborated that they felt that *"the VR experience was also better because I more tried to be in the VR world"*. Thus, we understood that their presence increased in the second round. Comparing this participant's feedback with their first round's interview showed that they previously felt less immersed due to being unable to *"trick their brain"* into thinking it was a real scenario. Although they still had a comfortable experience before, allowing themselves to be more immersed in the VE made them feel to be in the virtual world and holding a presentation in front of an actual audience.

Overall, those answers gave us the impression that most participants experienced the effects of training since they reported self-critical comments on how they could have done better and reflected on their previous performance.

Technical Difficulties

As with the first presentation, there were also some arguments concerning issues with the technology and how it diminished comfort. Generally speaking, this was again mainly noticed by the experimental group participants, who previously mentioned some difficulties after the first presentation. Nonetheless, those reports got more severe the second time.

Overall, they explained that they felt the application was lagging, somewhat perceiving pictures, and in the case of one participant, even observed an increase in sickness symptoms. As for the latter, the participant who mentioned feeling uncomfortable was the same person who reported the highest level of SSQ after the first condition. Although they assured us that the symptoms were not too severe for them to feel sick, they also noted that they *"would not use it for some hours because it would be too exhausting"*. Based on this description, we felt reassured that the increase in sickness symptoms was not too drastic, but they indeed perceived a change from the first to the second condition.

Considering that none of the other participants felt anything like this, even having in mind the reported SSQ values of our pilot tests, we consider this case to be more of an exception. However, we can not be sure that another person is not feeling the same when using SeminVR. Nevertheless, we must also consider that the participant previously described their discomfort coming from the headset pressed to their face and having displays right in front of their eyes. Additionally, it was the participant's first time using VR as well. Thus, it is likely that the participant's discomfort was already coming from using the technology alone and not necessarily only due to using SeminVR and giving a virtual presentation.

The other two individuals who had previously commented on the visual quality of the application were those who commented the most the second time around. Explaining that they were restricted by the *"input lag"* and did, therefore, consider using fewer gestures than before.

One of the two remaining participants in the experimental group also mentioned that they observed issues with the visual clarity of SeminVR. They described the feeling of their vision getting *"blurrier and blurrier"* for approximately twenty seconds, blocking them from correctly reading their slides and interrupting their presentation. Nonetheless, this effect was gone after ten seconds and no longer interrupted them for the rest of the experience.

The last participant in the experimental group did not report observing any visual or performance changes. Therefore, we understood they did not perceive a technological change in the conditions and were not directly influenced by it.

Since the two participants who reported little or no issues with the application's quality seemed very interested in using SeminVR for their presentation training, we believe they felt slightly better immersed than the rest of the group. Their description of having a better room feeling and feeling more comfortable with the audience the second time supported this assumption.

Nevertheless, we must admit that SeminVR's performance change did make a difference overall since none of the control group's participants mentioned anything similar to the performance issues described by the experimental group's participants. They all reported things equivalent to the first training condition and never mentioned any technical difficulties or discomfort.

Comfort

Although the experimental group observed performance differences, this did not mean that all of them felt affected by them. Three of the five people in this group either mentioned it as a side note or did not even complain. They also reported feeling better this time, not perceiving the audience as too distracting. Especially one described that the audience was way more into it and that they perceived an actual behavior switch from the first round. Although they noticed still that the audience was only a computer program, they mentioned, *"I was feeling like everything I said got to them... So there was not a single person who was distracted in any way"*. As they later elaborated, this increased feeling of attention was especially noticeable as they felt uncomfortable with the previous audience. During the first experience, they thought that the audience was randomly looking away, emphasizing that they particularly noticed this when making mistakes or they had trouble finding the right words. As a result, they felt insecure and ran into more mistakes during their presentation. Therefore, the audience in the second round felt way better as they did not notice anything similar. Concerning that, we consider the participant to be more confident with their presentation, resulting in fewer mistakes.

The other two participants also described a better feeling this time around. While one mentioned that they perceived their presentation to be worse than before, they clarified that this is common for them as they always try out new things over multiple iterations of training presentations. However, both describe the second experience as more comfortable, similar to the previously mentioned participant. *"I had the feeling that they are not looking that much down on the table to their phones"*, one of them described, while the other mentioned that they perceived the audience to be more listening as they were more looking at them than before. Respectively, we argue that they have benefited from the first round's training effects.

Similar to the experimental group, the reports of the control group also varied. While some described the audience as more attentive and, thus, more comforting during the presentation, others felt they did not change. More precisely, two of them reported that they enjoyed the experience of having an audience that was slightly more attentive than the one before. Although they sometimes questioned their perception and asked themselves, *"Maybe if I am even right, the people changed"*, they generally had a similar or better feeling presenting their topic the second time around. The other three remaining participants in the control group did not experience that improvement in comfort. They described how, due to their particular focus on the audience, they sometimes did not pay enough attention to their presentation and, thus, performed worse. Additionally, they reported that they felt the audience was the same as before. One of those participants described them as, *"the same people just acting differently"*.

Particularly for the control group, it was interesting to see that participants noticed that the only difference between the two conditions was agents changing roles with one another. However, we believe this was mainly related to them, primarily putting their focus on the audience and the analysis of their actions. Therefore, we think they likely would have interpreted the audience's behavior differently if they had focused more on training their presentation.

Overall, we could observe a mixture of comfort similar to the first round. While some participants perceived a change from the first to the second condition, others felt the same. Regardless, on average, the feeling of comfort stayed the same, except for one participant who reported higher sickness symptoms, as we previously described.

Comfort due to Controllers

One fascinating aspect we observed during our analysis was that two participants mentioned feeling more secure holding their presentation in this environment because they could hold onto controllers. Interestingly, both of them similarly elaborated on this. They said they usually gesture and extensively move their hands around while presenting. Therefore, they often hold onto a pen to keep their hands steady.

Since using SeminVR required controllers, they always had something in their hands, which they described as comforting. They had an increased feeling of safety, which made them more relaxed talking in front of the audience. Nevertheless, holding the controllers did not prevent them from gesturing in the virtual space, as they explained related to that. They still used gestures during their presentations, although how it was different from not holding controllers can not be said. Therefore, further research is needed to identify if this setting changes their behavior due to feeling more comfortable holding onto something.

It must be mentioned that those participants were in separate groups, thus reducing the likelihood of this effect standing in relation to the change in performance between the two conditions. Therefore, we believe this is solely related to personal confidence in talking in front of people and how one usually behaves during presentations. It also further shows that participants had an immersive experience as they felt the need to be comforted by holding onto their controllers.

Distractions

As we already elaborated a couple of times, the different feelings of comfort in the second round did not solely come from the virtual audience and participants' perceptions. In particular, three participants mentioned another reason why they felt less comfortable: being distracted by environmental events while presenting. Each of them noticed some sounds of cars going by on the street. They were so prominent that they created the urge to look out the window to check where they came from. Especially since participants had not seen or heard any cars moving initially, it was weird that they suddenly could notice corresponding sounds. One of them described this situation in a nutshell *"I could not locate the sound, and I could not see anything that caused the sound. But I paused there for a second"*.

Being noticed by one participant in the first round already, we understood that participants were confused by car sounds if they could not see one going by. Especially since they entered the VE when it was calm at a time where no distractions were either visually or audibly present, they described this circumstance as not being acclimated to the distractions. Accordingly, they compared it to a real scenario where they would be used to the things happening around them since they had to get to the seminar room first, giving them time to accustom to the environment. In the case of SeminVR, they were put in this environment with no prior knowledge about the setting. Thus, participants were confused when they perceived an unnoticed sound.

Therefore, we concluded that observing the environment where the room is situated and acclimating to its surroundings is an essential factor to consider in allowing higher levels of immersion in training scenarios. Nonetheless, it is also crucial to provide a matching visual response to the atmospheric sounds from the environment. Otherwise, it might irritate participants where the sound came from, resulting in them searching for the potential cause. As a result, they would pause their presentation, like our participants did, and wonder if something was happening around them. Thus, assuring that car sounds can only be played if a visual representation of a car is around or simply getting rid of this atmospheric sound should eliminate this confusion and increase comfort for participants.

Presenting or Training

As some participants already mentioned, while we asked them about their comfort, they felt that SeminVR was not different for the two evaluation conditions. Some of them described that the audience was behaving the same, only changing their roles, sometimes even making them feel like repeating themselves. One participant elaborated on this, describing that they *"Had more the feeling of presenting rather than training it"*, which made them feel like repeating themselves giving the same presentation twice. They compared it to giving lectures about the same topic to different groups, repeating the content over and over again. Thus, they felt *"a bit annoyed to do the same presentation again"*. However, this was only the case for one participant.

For all the other participants, it was still more of a training scenario. Although they had different feelings towards the presentation concerning how well they did and what the audience was like, they shared their understanding of SeminVR being a training application. As some participants described, they were trying out new ways to present their points or even went into further detail as they noticed they had time left.

Nonetheless, since one participant explicitly mentioned their feelings about presenting rather than training it, we decided that SeminVR might be more beneficial later in preparation for some people. Seeing it as a way to finalize one's preparation for a presentation rather than training it in front of an audience from the start might be more comfortable for some. Especially since other participants sometimes also described SeminVR as a different training scenario, mentioning it to be a hybrid between training alone and in front of people. Hence, it becomes clear why we need to consider it an addition rather than a replacement for traditional training.

5.3 Quantitative Results

Generally, the quantitative results of our SSQ and IPQ, did not result in any significant insight. Due to our small sample size, no significant differences were obtained, as expected before our study. Furthermore, we did not observe a normal distribution of our data while analyzing it, thus leading to the assumption that differences were mainly due to randomness. Respectively, the results found mostly replicate our interpretation in correlation with the qualitative results. Nevertheless, we would argue that the quantitative results support our assumption that SeminVR was immersive and not sickness-inducing. Accordingly, the presentation training was perceived well and allowed participants to prepare for a real scenario.

To further elaborate on how we came to this decision, we describe the results of both questionnaires in more detail.

5.3.1 SSQ

Utilizing the SSQ, we observed the changing levels of sickness symptoms and how they developed over time. Typical for this metric, we combined our results in boxplots for better visualization of results and to see the changes between the different states. Furthermore, those were created for each group individually to see if their symptoms developed differently and identify the influence of changing performance. Nevertheless, it must be remembered that each group only contained five participants. Thus, individual differences have a more significant impact on the boxplots.

Before starting our experiment, participants indicated their initial feelings of sickness. As shown in Figure 5.4, those were generally low and did not deviate much from one another. All scores were within a similar range, indicating that participants felt comfortable before starting the experiment. Only a minor exception can be perceived concerning the control group's total score and the experimental group's disorientation level. Some minor outliers were visible for these measures, indicated by the additional marks. Nevertheless, since the rest of the metrics were already relatively low, these outliers were not considered particularly relevant.

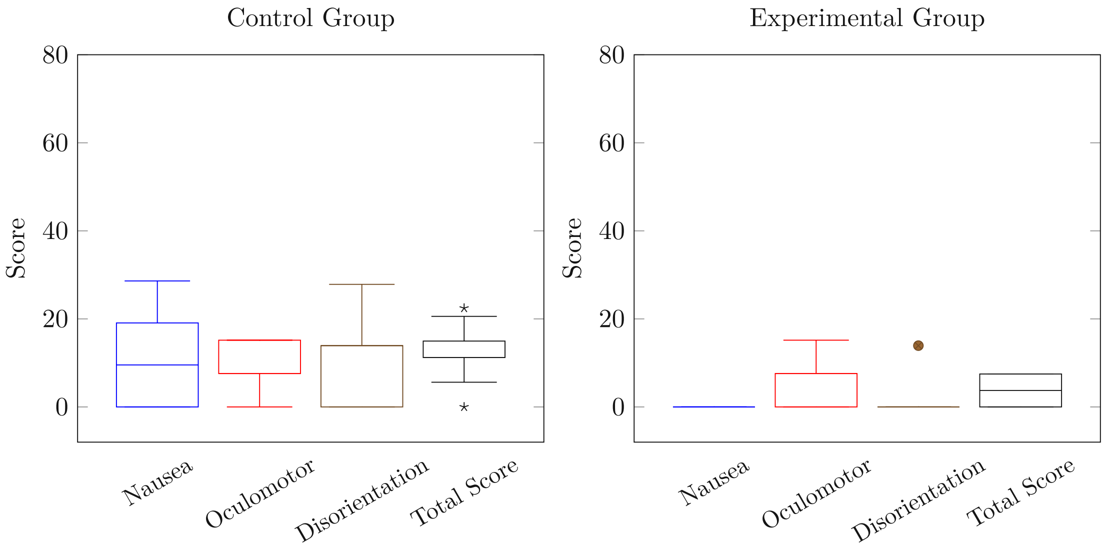

Figure 5.4: SSQs before 1st exposure

After the first presentation, a few changes were observed. The most noticeable across both groups was the change in disorientation, as seen in Figure 5.5. While this score dropped to zero for almost all participants, one participant in each group had either an equal feeling of disorientation as before or perceived a slight increase. The other score ranges shifted a little but mostly remained the same. Nevertheless, they also included some minor outliers, but not prominent ones like the disorientation measure. Although they do not show significant differences, a slight change from before can be observed. However, generally speaking, the differences between before and after the first presentation were minor and did not indicate any significant effects.

As seen in Figure 5.6, slightly more notable changes were observed than the scores reported after the second presentation. When comparing the results of both groups, it

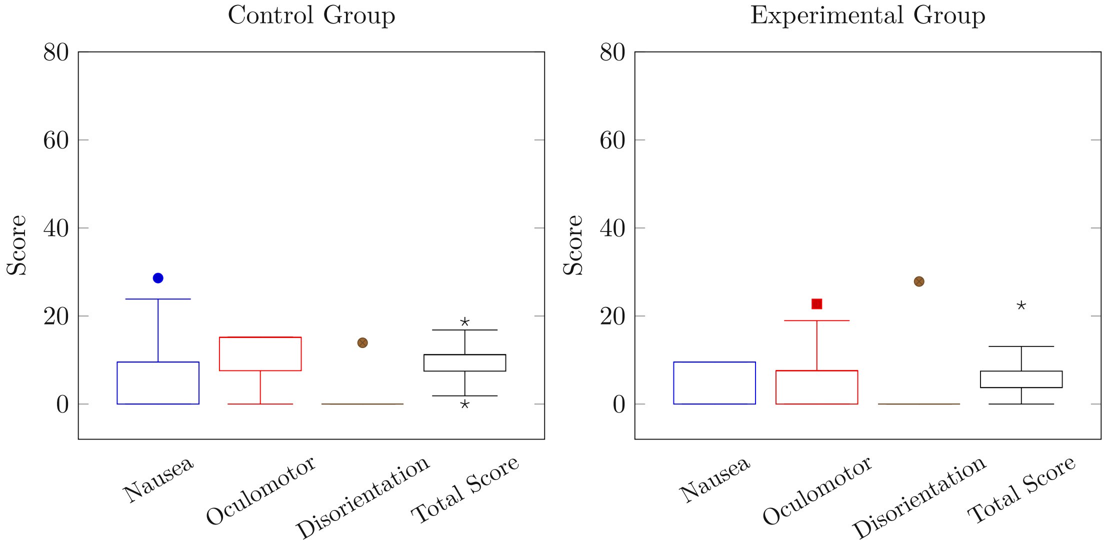

Figure 5.5: SSQs after 1st exposure

was noticeable that they differ much more than previously. However, we must consider that the reported values still were reasonably low. Considering that the severity scale of the SSQ is indicated from zero to three and the vast majority of questions were answered with zero, it still was considered a good result. Especially when observing the increased range for the experimental group, it was visible that the results mainly changed due to some more prominent outliers.

In this case, one participant indicated slightly higher perceived sickness symptoms than the rest. Because this was only one out of ten participants, and they did not demonstrate critical levels of sickness, we consider it an exception. Considering the results of the qualitative analysis, this assumption was supported by the fact that this participant already perceived discomfort with VR in general. Thus, we assumed that the increase did not solely relate to the usage of SeminVR.

Overall, the perceived levels of sickness did not drastically change throughout the experiment. Although changes were visible between the two groups, they were not distinctly different. Thus, we understand that our application was unlikely to induce simulator sickness in general. However, some exceptions exist that did not feel comfortable using the trainer on low-performance platforms. Nevertheless, this result was not generalizable due to the short duration of our experiment and only five participants for each condition. Regardless, we achieved a general level of comfort for our participants, which has been our primary goal in identifying with this metric.

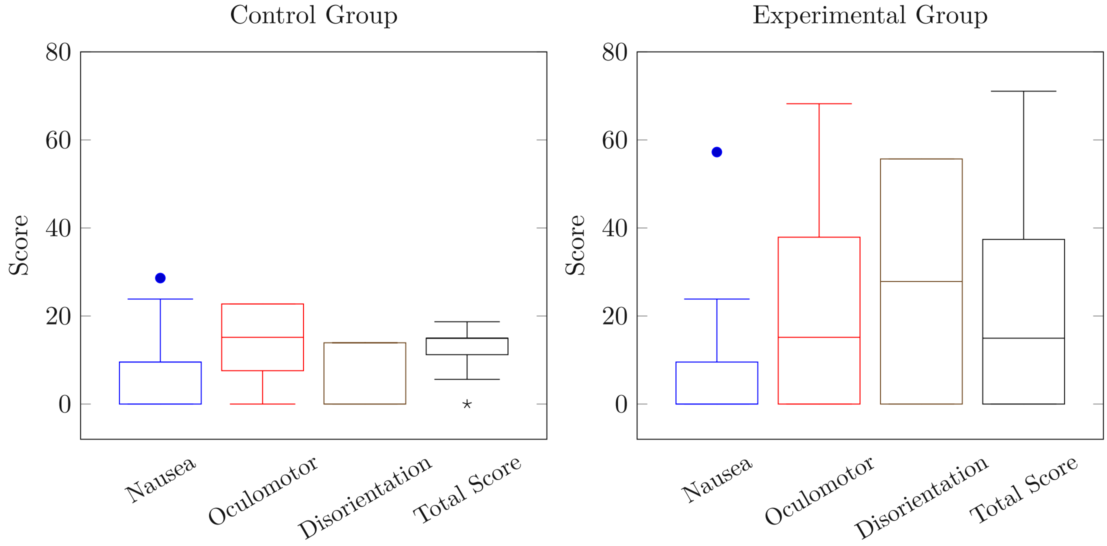

Figure 5.6: SSQs after 2nd exposure

5.3.2 IPQ

We analyzed our selection of the IPQ questions by comparing the two conditions for each group. Starting with the overall impression, we already identified that our data was not normally distributed. Thus, we analyzed only the median values, which best show the difference between conditions. However, due to the small number of participants, the significance of each of those values was again low. As seen in Table 5.3, those changes in median values mostly stayed the same and did not drastically vary for either group.

		Ctrl.		Expr.	
ID	**IPQ-Question**	**C1**	**C2**	**C1**	**C2**
Q1	Somehow I felt that the virtual world surrounded me.	4	4	4	3
Q2	I felt like I was just perceiving pictures.	1	1	0	1
Q3	I had a sense of acting in the virtual space, rather than operating something from outside.	4	4	4	4
Q4	I felt present in the virtual space.	4	4	4	3
Q5	I was not aware of my real environment.	4	4	3	4
Q6	I still paid attention to the real environment.	0	0	0	1
Q7	I was completely captivated by the virtual world.	4	3	3	4
Q8	The virtual world seemed more realistic than the real world.	1	1	1	1

Table 5.3: IPQ medians for both groups and conditions

Overall, we identified a general understanding of perceived presence across both groups since participants answered the questions in favor of the experience's immersiveness. Comparing the first and second conditions (C1, C2) for each group (Ctrl., Expr.), we observed that this feeling did not significantly change for either experience. Nevertheless, we saw more changes in the experimental group than in the control group. However, those were only slight differences and did not show any particular tendency. Respectively, we believe that participants had shared feelings concerning the application's immersiveness, indicating that the scenario was realistic.

Since the selected questions covered different aspects of immersion, which was valid for a better understanding of how participants felt, we further elaborated on what they indicated and what we understood from them.

Q1-Q4 Spatial Presence Concerning spatial presence, the first four questions showed that participants felt like they were in a virtual space rather than in the real world. Respectively, we argue that the application environment was realistic enough to allow this feeling of immersion. Even though this feeling slightly changed for the second condition, no significant difference occurred in general, indicating any drastic shifts.

Q5-Q7 Involvement The reported values for the involvement-related questions indicate that our participants could focus on their training in the VE and tried blending out the real world. While this feeling slightly changed for the second condition, it was much more likely that this change was due to some randomness.

Q8 Experienced Realism With the final question in our reduce IPQ, the experienced realism was covered, being comparably less precise as it only consists of one question. However, the fact that the median value for both conditions remains the same indicates that nothing drastic has changed. Respectively, we believe the environment was realistic enough for training, but it has room for improvement.

CHAPTER 6

Discussion

This chapter discusses the effectiveness of our developed presentation trainer based on the evaluation findings. We begin with formulating initial interpretations of the most substantial findings to create the foundation for answering the research questions. Subsequently, those questions are addressed and answered based on our results. Additionally, the limitations observed while conducting our research are discussed, highlighting aspects that influenced the study's outcome.

6.1 Main Interpretations of Findings

Changes in the Focus of Attention As we already described in our findings section about the participant's different focus for each condition (see 5.2.2), we observed that their attention changed throughout the experiment. Due to the questions asked during the first interview, we transmitted to participants the feeling that they did not extensively focus on the environment and changes in the audience. Thus, they focused more on how they perceived the application while somewhat neglecting their presentation in the second round.

We assume that conducting the interviews more freely, without any pre-formulated questions, could have prevented this shift. Since it would not have restricted participants to answering questions we previously defined, it would have likely allowed them to focus more on what they felt most important. Consequently, it should have allowed for less influenced application exploration and improved their presentation training experience.

Additionally, communicating to participants upfront that they do not get recorded and that study results rely on what they tell us must have further strengthened the idea that they must focus more on the environment than anything else. On the other hand, this enabled us to collect more specific details on what participants felt was good or missing in SeminVR, allowing us to elaborate more on the application's realism.

Realism Throughout the evaluation, most participants reported that SeminVR replicates a realistic scenario. Accordingly, they made various comparisons on how it imitated actual presentation settings and how that supported their feelings of immersion. Although they knew the presentation trainer was just a simulation, they felt like being in the environment and preparing their content. Thus, we understand that they perceived it as a safe training space where they could explore their presentation skills.

Nevertheless, participants reported some issues that appeared throughout the experience that conflicted with its level of realism. An example was the virtual agent's calm behavior being interpreted differently by various trainees. In this regard, some participants described the audience as pleasant and calm listeners, while others perceived them as being passive. Although it did not significantly distract them from training, it made the scenario less believable for some participants.

Comfort The participants' overall feeling of comfort varied between conditions. It was primarily affected by their confidence in talking in front of an audience and the perceived realism of the situations. In this regard, they mentioned higher levels of comfort and a better overall experience in cases where they allowed themselves to be more in the virtual world.

Additionally, participants reported the scenario as being very calm and not containing many distractions, which allowed them to focus on training. Nevertheless, this quiet setting also appeared strange to some participants since they reported missing typing noise and agents chatting. Although some would have liked a more active setting, others preferred the calm behavior and were comforted by it. Respectively, we understand that making the setting more active would have been counterproductive.

Training Effects Overall, we could observe the training effects of participants while using SeminVR. Reports on how they tried to optimize their presentation content and focus on doing their best while presenting showed that participants were encouraged to train and improve their presentation skills. Although this effect was not the same for everyone across both conditions, it was noticeable that participants gained more confidence talking about their topic.

The difference between conditions was significantly influenced by the focus change between the two conditions. Since we introduced a bias with our interview questions in the first round, more participants focused on analyzing SeminVR's integration instead of optimizing their presentation skills in the second round. Accordingly, some participants felt there were fewer benefits of training the second time, as they focused more on observing and analyzing the experience.

Impact of Performance None of the control group's participants mentioned any effects of SeminVR's performance influencing their training experience. They generally reported how they perceived the training and felt about talking in the virtual space. Although their focus shifted to be more analytical in the second round, they did not report on the application's visual clarity or performance. Thus, we understand that they did not consider it an influential factor.

On the contrary, participants of the experimental group reported that SeminVR's performance had impacted their training. Already, while using it for the first time, two participants highlighted being affected by its visual clarity. Although everyone used the same high-performant version for the initial condition, they had trouble reading their slides, which made them criticize SeminVR's visual quality. Those reports increased for the second condition as more participants noticed the change in performance. Although most participants noticed the performance change, only two of the five participants were particularly affected by it. As for the rest, they seem to have blended out the effects of the performance change and trained equally efficiently as in the first round. Nevertheless, the two participants who were most affected could no longer effectively train as they felt the experience was lagging and uncomfortable.

6.2 Findings in Connection to Research Questions

6.2.1 RQ1: Does SeminVR show a positive training effect on people?

During our evaluation, participants reported different feelings on how they perceived their training. Most notably, this was influenced by a change of focus as they analyzed the application more during the second round. Therefore, they concentrated less on training their presentation skills and could only benefit from it partially. Due to the combination of us asking participants to report on the experience and them being more confident with their slides, they could allow themselves to focus more on exploring SeminVR. However, this resulted in many of them reporting that their presentations were better the first time. Independently, some still reported on how they tried different things during their presentation and used it as a chance to optimize their performance.

Based on those findings, we understand that our study design has negatively influenced training effectiveness by shifting participants' focus too much on analyzing the application. However, we could still observe participants benefiting from training, indicating that SeminVR can be helpful for training presentations in front of an audience.

6.2.2 RQ2: Does the feeling of presence correlate with a platform's performance while running SeminVR?

Although most experimental group participants noticed that the application's visual quality and frame rate changed, not all of them got distracted by it. Nevertheless, two of the five participants felt restricted due to lagging movements and even perceived it as less comfortable. Consequently, they did not like using SeminVR on the less performant platform. However, it must be mentioned that the experimental group seemed more analytical overall when compared to the control group. While no participants from the control group commented on the application's performance in either of the conditions, two participants from the experimental group did so already in the first round.

Participants across both groups also reported that they would like to see a more active audience alongside some distractions to make the setting feel more lifelike. Thus, they would have enjoyed seeing cars going by on the street, people talking, or individuals randomly looking around in the room. Interestingly, those affected by performance changes even mentioned those changes to be more meaningful than performance improvements. Thus, we understand that realism is sometimes considered more important than performance.

From these results, we conclude that the platform's performance running SeminVR can impact participants' ability to feel present in the virtual world. This is especially noticeable in cases where performance reduces their comfort and prevents participants from appropriate training. However, we also understand that those effects can be compensated for if participants think the situation is realistic. The overall recommendation of the audience was to improve the depth of the application rather than focus on making it run faster. Consequently, we believe that various people can benefit from our presentation trainer if platform performance does not interfere with their ability to feel immersed.

6.2.3 RQ3: Does an increased feeling of stress correlate with a higher level of perceived presence?

Concerning feelings of stress throughout the experience, participants generally agreed that SeminVR felt like a safe space to train in. Although some did report feelings of stress, they described it as coming from them feeling like presenting to an actual audience. However, we could not see a tendency for people who feel more stressed to be more immersed than others. Instead, we identified that stress correlates with the individual's confidence to talk in front of an audience. Thus, people who are usually confident talking to an audience feel the same comfort talking to the virtual audience and vice versa. Therefore, we argue that stress might not be a good indication of the perceived presence, as people feel differently in this regard.

At the same time, we observed that presence correlates with how realistic participants perceived the simulation. This is primarily because those reporting similarities to a real scenario and describing parallels also reported the most on how they tried optimizing their presentation skills. Hence, we concluded that they perceived the setting as natural and could immerse themselves better. Therefore, we consider analyzing how realistic the situation felt for participants as a better metric for identifying presence than stress levels.

Although studies like Takac et al.'s [TCB+19] argue that higher stress levels correlate with an increased feeling of presence, we can not say the same for our evaluation. Nevertheless, we must admit that this can not be generalized due to our small sample size.

6.3 Limitations and Reflections

While conducting our study, we identified various limitations essential to mention and discuss for future studies. Consequently, future research on multi-platform presentation trainers should consider these restrictions and avoid potential issues resulting from them.

6.3.1 Evaluation Hardware

While hardware with higher resolution and more performance already exists, we conducted our study using a comparatively old headset. Although the Meta Quest 2 is still a highly relevant entry-level VR headset, its performance could be better compared to state-of-the-art devices. Nevertheless, we utilized its weaker performance to our benefit by identifying if our multi-platform presentation trainer is usable when run on it. However, we must consider that our results would have looked differently on a state-of-the-art headset due to running the application more smoothly and delivering a better visual quality.

6.3.2 Unity WebXR Exporter Performance

Although the Unity WebXR Exporter project already supports many of the API's capabilities, we identified that its performance is limited. While we tried identifying its maximum FPS running on the Quest 2 in a standalone, as described in Section 4.1.1, we found that the baseline with 45 FPS was already low. Respectively, it did not leave much room for creating a computationally extensive application like ours.

Unless the WebXR Exporter project's baseline performance improves significantly, we recommend considering using more native tools for WebXR development. Since native approaches are more lightweight, they are likely to produce more performant applications and allow for better results. Nevertheless, we cannot disregard Unity's benefits in this context, as it provides advanced integration possibilities which were necessary in cases like ours. Therefore, the choice of the development platform should still be made based on the application's goals.

6.3.3 Personal Progression

Another aspect to consider regarding the depth of our data is the personal progression and experiences made throughout the evaluation. As I, the researcher, became more confident in my interviewing, the later interviews deviated from the structure and covered the topic in greater depth. However, it also needs to be considered that earlier participants were interviewed with a more open mindset, as no previous experiences influenced the researcher's perspective.

Nonetheless, we believe conducting our evaluation by alternating the two conditions from participant to participant should have reduced the effect of forming assumptions. Primarily due to not being exposed to the same assessment setting twice in a row, it should have reduced forming assumptions from one to another experiment as they were not wholly the same. However, this does not eliminate the fact that a general understanding was formed over time, thus influencing how we understood what participants mentioned in the interviews.

CHAPTER 7

Conclusion

In our thesis, we developed and evaluated SeminVR, a WebXR-based public speaking trainer designed for presentation training with a virtual audience. Alongside creating a realistic experience to allow for appropriate training, our goal was to make this application usable on different platforms. Accordingly, we decided to use the WebXR API, allowing the development of a single application that can be accessed via a web browser. Although it eliminates the need to develop the same application for multiple platforms, we had to ensure the single application runs equally well on various end devices. Therefore, we decided to focus on the least powerful devices and optimize our training in a way that is usable on those. As a result, the presentation trainer had to be simplistic to allow smooth performance while maintaining a high level of realism to allow immersion.

To determine whether SeminVR is a helpful preparation method for presentation training independent of any platform, we evaluated its usefulness in a between-subject study. With a control group defining a baseline for the effects perceived using our trainer, we were able to compare those with an experimental group using different platforms and decide whether performance made a difference.

Our results suggest that SeminVR can be a helpful addition to conventional preparation while training for presentations. Nonetheless, it is essential to consider that performance differences between various platforms can influence people's feelings of comfort and affect how useful they perceive the training. However, we observed that higher levels of immersion allow for blending out performance differences as long as those introduce no discomfort. We take from this that performance is critical to ensure a comfortable and smooth experience but can be partly compensated by providing a more realistic experience.

We conclude that developing a presentation trainer using WebXR is a good and efficient way to create an experience that can be used across various devices. Although performance-efficient development is essential for smooth experiences across platforms, it is a great way to create multi-platform-supported applications.

7.1 Future Work

Currently, SeminVR imitates a calm and attentive audience actively listening to trainees. Although agents sometimes look around and get distracted by things like phones and notebooks, they mainly focus on what is said. Some of our participants reported that they sometimes perceived that behavior as too quiet and expected more distractions. Therefore, developing a more elaborate agent behavior model might be helpful. Nonetheless, it should not replace the calm behavior of SeminVR as it is right now but rather be an addition that users can select if they are seeking a more demanding training scenario.

Closely related to agents getting more distracted, additional events in the surroundings could help people prepare for various settings, such as giving a presentation close to a heavily used road or a construction site. Nevertheless, we consider those additions as things that users should individually choose, as they otherwise overload the experience and compromise their comfort.

Another aspect that would benefit users is to allow a better understanding of the audience and receive feedback from them. Since a couple of our participants mentioned that they could not understand if the audience liked their presentation and what they should change, it is recommendable to integrate some feedback mechanism to improve training. Potentially, this could be similar to the feedback system developed by Palmas et al. [PCPK19], but also go in the direction of adding mimics to agents.

List of Figures

List of Tables

Acronyms

AR Augmented Reality. 2, 8, 9

FPS Frames per Second. 14, 36, 39, 91

GDPR General Data Protection Regulation. 66

IPQ Igroup Presence Questionnaire. 25, 28, 29, 31, 82, 85, 86

SSQ Simulator Sickness Questionnaire. 20–22, 25, 28, 31, 75, 78, 82, 84

TA Thematic Analysis. 3, 25, 27, 65, 66

TU TU Wien. 29

VE Virtual Environment. 13, 15, 20, 22, 23, 28, 33, 69, 71, 77, 78, 81, 86, 95

VR Virtual Reality. 1, 2, 5, 6, 8, 9, 20, 21, 24, 25, 28, 30, 32, 33, 35, 36, 59–61, 65, 71, 74–76, 78, 84, 91

VRPST Virtual Reality Public Speaking Trainer. 2, 3, 5–8, 11, 16, 36

Bibliography

[BC06] Virginia Braun and Victoria Clarke. Using thematic analysis in psychology. *Qualitative Research in Psychology*, 3(2):77–101, 2006.

[Ber20] Borbála Berki. Experiencing the sense of presence within an educational desktop virtual reality. *Acta Polytechnica Hungarica*, 17(2):255–265, 2020.

[Bra16] Neil A Bradbury. Attention span during lectures: 8 seconds, 10 minutes, or more?, 2016.

[BSBW23] Manuel Bachmann, Abimanju Subramaniam, Jonas Born, and David Weibel. Virtual reality public speaking training: effectiveness and user technology acceptance. *Frontiers in virtual reality*, 4, 2023.

[EMM20] Paul MG Emmelkamp, Katharina Meyerbröker, and Nexhmedin Morina. Virtual reality therapy in social anxiety disorder. *Current psychiatry reports*, 22:1–9, 2020.

[GKR20] Alice Gruber and Regina Kaplan-Rakowski. User experience of public speaking practice in virtual reality. In *Cognitive and affective perspectives on immersive technology in education*, pages 235–249. IGI Global, 2020.

[HEF21] Mhanaj Hossain, Daphne Economou, and Jeffrey Ferguson. Work-in-progress-webxr to support student wellbeing and anxiety. In *2021 7th International Conference of the Immersive Learning Research Network (iLRN)*, pages 1–3. IEEE, 2021.

[KLBL93] Robert S Kennedy, Norman E Lane, Kevin S Berbaum, and Michael G Lilienthal. Simulator sickness questionnaire: An enhanced method for quantifying simulator sickness. *The international journal of aviation psychology*, 3(3):203–220, 1993.

[LlRP16] Jean-Louis Leysens, Daniel B le Roux, and Douglas A Parry. Can i have your attention, please? an empirical investigation of media multitasking during university lectures. In *Proceedings of the Annual Conference of the South African Institute of Computer Scientists and Information Technologists*, pages 1–10, 2016.

[MGB+23] Miguel Melo, Guilherme Gonçalves, Maximino Bessa, et al. How much presence is enough? qualitative scales for interpreting the igroup presence questionnaire score. *IEEE Access*, 11:24675–24685, 2023.

[NCB91] W Russell Neuman, Ann N Crigler, and V Michael Bove. Television sound and viewer perceptions. In *Audio Engineering Society Conference: 9th International Conference: Television Sound Today and Tomorrow*. Audio Engineering Society, 1991.

[PCPK19] Fabrizio Palmas, Jakub Cichor, David A Plecher, and Gudrun Klinker. Acceptance and effectiveness of a virtual reality public speaking training. In *2019 IEEE international symposium on mixed and augmented reality (ISMAR)*, pages 363–371. IEEE, 2019.

[REC+21] Rachel Reeves, Adam Elliott, David Curran, Kevin Dyer, and Donncha Hanna. 360 video virtual reality exposure therapy for public speaking anxiety: A randomized controlled trial. *Journal of Anxiety Disorders*, 83:102451, 2021.

[SFR01] Thomas Schubert, Frank Friedmann, and Holger Regenbrecht. The Experience of Presence: Factor Analytic Insights. *Presence: Teleoperators and Virtual Environments*, 10(3):266–281, 06 2001.

[TCB+19] Marcel Takac, James Collett, Kristopher J Blom, Russell Conduit, Imogen Rehm, and Alexander De Foe. Public speaking anxiety decreases within repeated virtual reality training sessions. *PloS one*, 14(5):e0216288, 2019.

[TSM+22] Yuji Teshima, Akihiro Sakaguchi, Yuto Mitsue, Nobutomo Uehara, and Kazuhide Sugimoto. Development of webxr-based environment for engineering experiments. In *2022 IEEE Frontiers in Education Conference (FIE)*, pages 1–5. IEEE, 2022.

[WKH21] Sebastian Weiß, Nelly Klassen, and Wilko Heuten. Effects of image realism on the stress response in virtual reality. In *Proceedings of the 27th ACM Symposium on Virtual Reality Software and Technology*, pages 1–10, 2021.

[WMD+21] Erik Wolf, Nathalie Merdan, Nina Dölinger, David Mal, Carolin Wienrich, Mario Botsch, and Marc Erich Latoschik. The embodiment of photorealistic avatars influences female body weight perception in virtual reality. In *2021 IEEE Virtual Reality and 3D User Interfaces (VR)*, pages 65–74. IEEE, 2021.

[ZA21] Misbahu Zubair and Nnamdi Anyameluhor. How long do you want to maintain this thing? understanding the challenges faced by webxr creators. In *Proceedings of the 26th International Conference on 3D Web Technology*, pages 1–4, 2021.

Appendices

Final Evaluation - Consent Form

TU WIEN Informatics   Human Computer Interaction Group

Consent form for study participation

The diploma project "WebXR Public Speaking Trainer" focuses on the analysis of a virtual presentation trainer. The diploma project is conducted under the supervision of Peter Purgathofer at the Vienna University of Technology (TU Wien).

Please confirm your participation by checking the applicable statements:

I confirm that I would like to participate in the study in the form of an evaluation of the prototype, and that my participation is voluntary.	
I confirm that I have received a written explanation of the research project as well as a copy of the privacy statement, and that I have had the opportunity to seek oral clarification for any questions and uncertainties.	

____

First- and Lastname in block letters

________________________   ________________________

Location, Date   Signature participant

____

Signature researcher

Interview Guide
"WebXR Public Speaking Trainer"

By Andreas Macsek

Perceived Stress Level

1. How stressed do you feel right now?
2. Did you feel more stressed during the presentation?
3. Is this feeling similar to a real presentation?
4. Are you exhausted?
5. How tense do you feel right now?

Perceived Performance Level

1. How would you describe your performance?
2. Did you speak freely?

Audience Attention Level

1. What was the audience's level of attention like?
2. Were there any participants that stood out from the rest?

Final Evaluation - Questionnaires

WebXR Public Speaking Trainer Questionnaires

Demographic Data					Participant- ID:	P00
Name						

Please indicate your decission using an "x" in the boxes provided below.

	18-22	**23-27**	**28-32**	**33-37**	**38+**
Age					

	Male	**Female**	**Inter**	**Divers**	**Open**	**No Entry**
Gender						

Please indicate how frequently you ...

	Never	**Rarely**	**Sometimes**	**Almost every time**	**Every time**
Play Games?					
Use VR?					

Before 1st Presentation

Simulator Sickness Questionnaire (SSQ)

Please indicate how you are feeling on a scale from **0** (none) - **1** (slight) - **2** (moderate) - **3** (severe)

Symptom	
General discomfort	
Fatigue	
Headache	
Eye strain	
Difficulty focusing	
Increased salivation	
Sweating	
Nausea	
Difficulty concentrating	
Fullness of head	
Blurred vision	
Dizzy (eyes open)	
Dizzy (eyes closed)	
Vertigo	
Stomach awareness	
Burping	

After 1st Presentation

Simulator Sickness Questionnaire (SSQ)

Please indicate how you are feeling on a scale from **0** (none) - **1** (slight) - **2** (moderate) - **3** (severe)

Symptom	
General discomfort	
Fatigue	
Headache	
Eye strain	
Difficulty focusing	
Increased salivation	
Sweating	
Nausea	
Difficulty concentrating	
Fullness of head	
Blurred vision	
Dizzy (eyes open)	
Dizzy (eyes closed)	
Vertigo	
Stomach awareness	
Burping	

After 1st Presentation

IGroup Presence Questionnaire (IPQ)

Please indicate how you are feeling on a scale from **0** (fully disagree) - **1** (disagree) - **2** (neutral) - **3** (agree) - **4** (fully agree)

Questions	
Somehow I felt that the virtual world surrounded me.	
I felt like I was just perceiving pictures.	
I had a sense of acting in the virtual space, rather than operating something from outside.	
I felt present in the virtual space.	
I was not aware of my real environment.	
I still paid attention to the real environment.	
I was completely captivated by the virtual world.	
The virtual world seemed more realistic than the real world.	

After 2nd Presentation

Simulator Sickness Questionnaire (SSQ)

Please indicate how you are feeling on a scale from **0** (none) - **1** (slight) - **2** (moderate) - **3** (severe)

Symptom	
General discomfort	
Fatigue	
Headache	
Eye strain	
Difficulty focusing	
Increased salivation	
Sweating	
Nausea	
Difficulty concentrating	
Fullness of head	
Blurred vision	
Dizzy (eyes open)	
Dizzy (eyes closed)	
Vertigo	
Stomach awareness	
Burping	

After 2nd Presentation

IGroup Presence Questionnaire (IPQ)

Please indicate how you are feeling on a scale from **0** (fully disagree) - **1** (disagree) - **2** (neutral) - **3** (agree) - **4** (fully agree)

Questions	
Somehow I felt that the virtual world surrounded me.	
I felt like I was just perceiving pictures.	
I had a sense of acting in the virtual space, rather than operating something from outside.	
I felt present in the virtual space.	
I was not aware of my real environment.	
I still paid attention to the real environment.	
I was completely captivated by the virtual world.	
The virtual world seemed more realistic than the real world.	

SeminVR Asset List

Model Name	Origin
Room and Interior	
Seminar Room	Self-Built - Inspired by model on sketchfab.com
Table	Self-Built - Inspired by model on cgtrader.com
Chair	cgtrader.com
Room-Details	
Timer	Self-built
Slide Change Panel	Self-built
Posters	cgtrader.com
Projector	cgtrader.com
Monitor	cgtrader.com
Keyboard	cgtrader.com
Notebooks	cgtrader.com, cgtrader.com
Phone	cgtrader.com
Plant	cgtrader.com
Pinboard	cgtrader.com
Outdoor	
City	assetstore.unity.com
Cars	cgtrader.com, cgtrader.com
Sounds	
Outdoor Atmo	freesound.org
Indoor Atmo	freesound.org
Clapping	freesound.org