

UNIVERSIDADE DE LISBOA
FACULDADE DE CIÊNCIAS
DEPARTAMENTO DE INFORMÁTICA



Full-Stack Application for Controlling and Directing Virtual Reality Experiences

Ricardo Filipe Baltazar Pedro

Mestrado em Engenharia Informática

Trabalho de Projeto orientado por:
Prof. Doutor Manuel João Caneira Monteiro da Fonseca

2024

Acknowledgments

I would like to begin by thanking my supervisor Manuel Fonseca, the excellent Professors of the Faculty of Sciences of the University of Lisbon, and my supervisor Tupac Martir for all the technical support, all the patience and also the good times we spent.

I would also like to thank my university friends, Diogo, Inês, João, Robert, Luís, Shiny, Gul'dan, and Zágora for all their help in my academic career and preparation for the beginning of my professional career, and especially Michael Batista, for all his help with the application developed in this thesis.

I would also like to express my gratitude to my friends, Tunes, Morgado, António, Christian, Valente, Rodrigo, Bernardo, Alexandre, and Beatriz for always being there for me in times of instability and confusion, but also in moments of happiness and fun.

Thank you to ALL the members of my family, especially my mother, father, sister, brother-in-law, uncles, and cousins for their comfort and unimaginable help during my academic phase.

Finally, thank you to my Psychologist Miguel, and my Jiu-Jitsu Coach João for all their teachings to make me a better human being.

Resumo

As tecnologias de Realidade Virtual (VR) estão em rápida evolução. O Meta Quest 3, o Apple Vision Pro e o Meta Codec Avatar, que integram Inteligência Artificial (IA) e Head-Mounted Displays (HMDs), são tecnologias inovadoras para criar experiências imersivas. Estas inovações permitem interações avançadas, como avatares comunicativos e assistentes virtuais personalizados.

Este trabalho insere-se no projecto europeu TransmiXR, que reúne sete universidades europeias e catorze empresas para desenvolver ferramentas de VR aplicadas às Artes Performativas, ao Património Cultural e ao Jornalismo, desenvolvido no Satore Studio

O projecto centra-se num caso de uso nas Artes Performativas, com foco no desenvolvimento de uma aplicação para controlar dispositivos de realidade estendida (XR) em experiências imersivas.

O objectivo principal deste trabalho foi criar uma aplicação capaz de gerir remotamente as experiências de XR no Unreal Engine. No âmbito dos objectivos específicos, uma das funcionalidades desenvolvidas foi o controlo de dispositivos XR, que permite aos operadores realizar tarefas essenciais, como configurar a conectividade em rede dos headsets e executar acções como pausar, reiniciar ou terminar experiências de forma sincronizada em todos os dispositivos conectados. Outro objectivo específico focou-se no direccionamento de experiências XR, promovendo a personalização de experiências multiutilizador.

A aplicação possibilita a modificação de conteúdos exibidos em tempo real, quer de forma colectiva quer individual. Para tal, utiliza comandos que acedem a um servidor local para fazer download de 'assets' personalizados, como fotografias, vídeos ou modelos 3D, adaptando a experiência de XR a cada utilizador ou a todo o grupo.

Enquadramos a parte teórica e tecnológica do projecto, detalhando as tecnologias envolvidas, as suas aplicações práticas e uma comparação a ferramentas externas. Este capítulo aborda os conceitos essenciais para compreender a experiência de XR desenvolvida, incluindo a aplicação prática das tecnologias e a análise de plataformas existentes com funcionalidades similares. Exploramos também conceitos como a Realidade Estendida (XR), que engloba Realidade Aumentada (AR), Realidade Virtual (VR) e Realidade Mista (MR). Conceitos que interligam-se com o nosso trabalho e que combinam o mundo digital com o físico.

Este projecto aplica-se às Artes Performativas, onde os participantes interagem com Isaac, um avatar digital humano, num cenário físico que recria uma casa. Utilizando headsets de XR,

fundem-se elementos físicos e digitais, criando uma experiência híbrida. Os participantes são depois conduzidos a um "Mundo Imaginário" totalmente virtual, onde exploram memórias e experiências de Isaac, proporcionando uma narrativa imersiva.

Para gerir e dirigir os dispositivos XR, foi desenvolvida uma aplicação externa que sincroniza múltiplos dispositivos, garantindo que todos os participantes iniciem a experiência ao mesmo tempo. A direcção permite distribuir conteúdos personalizados, como modelos 3D e vídeos, para cada headset, utilizando o protocolo OSC (Open Sound Control) para comunicação em tempo real, garantindo experiências adaptadas a cada utilizador.

Foram analisadas ferramentas externas como ArborXR, Protokol e Meta Quest for Business, detalhando as suas funcionalidades e limitações. Estas plataformas oferecem soluções para gestão de dispositivos XR e criação de experiências imersivas, mas apresentam constrangimentos, como falta de personalização de conteúdos ou integração limitada com o Unreal Engine.

A aplicação proposta neste trabalho destaca-se pela flexibilidade, permitindo o controlo completo de dispositivos XR, personalização de conteúdos e gestão dinâmica das cenas. A integração com o Unreal Engine, aliada à eliminação de barreiras financeiras e geográficas, torna esta solução mais adequada para projectos que exigem escalabilidade e flexibilidade, superando as limitações das plataformas existentes.

A seguir, descrevemos o desenvolvimento da aplicação "SatoreStudio", desde a definição de requisitos até a criação e melhoria de protótipos, superando desafios técnicos para garantir uma solução eficiente para a gestão de experiências XR.

Durante o processo, os protótipos iniciais destacaram limitações técnicas e de usabilidade, conduzindo a uma transição para tecnologias mais modernas, como Angular.js para o front-end e Express.js para o back-end, garantindo flexibilidade e responsividade.

A interface do utilizador foi iterativamente refinada, melhorando indicadores de status, design responsivo e comandos centralizados para sincronização e personalização.

Uma limitação identificada durante o desenvolvimento foi a interface do sistema operativo dos dispositivos Meta Quest, que exige interação dos operadores para configurar os dispositivos antes do início da experiência XR. Esta exigência pode gerar atrasos desnecessários e comprometer a eficiência operacional. Para mitigar esta questão, propõe-se o desenvolvimento de uma aplicação Android que colocaria os dispositivos XR em "modo quiosque", restringindo as suas funções à execução da experiência XR. Esta solução eliminaria a necessidade de interação dos operadores com o sistema operativo, simplificando significativamente o processo de configuração.

O processo de avaliação dos dados adquiridos no User Testing centrou-se na otimização da usabilidade do Modo Operador, tornando-o acessível e intuitivo para utilizadores com diferentes níveis de capacidade técnica. Os testes foram realizados em três fases principais: um teste formativo, com seis utilizadores de diferentes níveis de experiência técnica, para identificar problemas de usabilidade e refinar a interface; um teste piloto, com dois utilizadores adicionais que validaram as melhorias implementadas; e um teste final, com vinte e seis utilizadores, para assegurar a acessibilidade e eficácia da aplicação.

Durante os testes, identificaram-se problemas específicos, como confusão na navegação, que foi resolvida ao definir o Modo Operador como aba padrão e a falta de indicadores de experiência ativa, solucionada com a adição de indicadores visuais como fundos verdes para sinalizar experiências de XR ativas. Apesar disso, algumas limitações ainda precisam de ser abordadas, como a barreira linguística, para a qual se propõe a tradução da interface para português e melhorias nos mecanismos de prevenção de erros, e o tamanho dos botões, sugerindo-se o seu aumento e um maior contraste visual para melhorar a acessibilidade.

A análise comparativa entre os testes formativos e finais revelou progressos significativos, com um aumento médio de 0,38 valores percentuais na satisfação, uma redução de 235% na frustração, refletindo uma experiência mais positiva, e uma melhoria de 23% na facilidade de uso, acompanhada por uma redução de 61% no tempo gasto em correções de erros. Outro aspeto analisado foi o desempenho de utilizadores com mais de 35 anos, que demonstraram maior dificuldade em tarefas que envolviam elementos pequenos da interface, embora o tamanho reduzido da amostra limite a validade estatística destas conclusões.

Em suma, o processo iterativo de design e testes foi crucial para o aprimoramento da aplicação. Apesar das limitações, como barreiras linguísticas e dificuldades de interação com o teclado, os resultados demonstraram uma interface mais eficiente, intuitiva e ajustada às necessidades dos utilizadores.

Palavras-chave: Desenvolvimento Full-Stack, Desenvolvimento Front-End, Realidade Virtual (VR), Experiência do Usuário (UX), Realidade Estendida (XR).

Abstract

This thesis investigates the integration of Virtual Reality (VR) technologies within the TransmiXR EU project, focusing on the development of a Full-Stack application for managing multi-user Extended-Reality (XR) experiences in Performance Arts. The work addresses challenges in XR device management, user interaction, and content customization through a system designed for live control and real-time adjustments of XR environments.

The proposed solution includes a management platform developed with Angular.js, providing a modular and responsive user interface that communicates with an Express.js back-end, which handles device communication and management. Integration with Unreal Engine enables scene management and AI-driven interactions, demonstrated through the use of Isaac, a virtual character in the performance. Comparison with existing XR management tools, including ArborXR and Meta Quest for Business, highlights the adaptability and cost-effectiveness of the developed application, with specific capabilities in content delivery and user interaction.

The thesis also examines related technologies such as volumetric video capture and motion-captured animation, which complement the directing functionality of the application. These tools facilitate content creation by incorporating 3D video and character animation, supporting the director's control over XR experiences.

User experience testing evaluates the usability of the application, including data analysis of its performance across different age groups, which is a significant consideration for systems with multiple operators of varying ages.

This work contributes to the field of VR and XR technologies by providing a solution for managing interactive and collaborative experiences in creative industries.

Keywords: Full-Stack Development, Front-End Development, Virtual Reality (VR), User Experience (UX), Extended-Reality (XR)

Contents

List of Figures	xiv
1 Introduction	1
1.1 Motivation	1
1.2 Context	2
1.3 Objectives	2
1.4 Document Structure	3
2 Background and Related Work	5
2.1 Concepts and Terminology	5
2.2 The Experience	6
2.2.1 Physical Environment and Setting	7
2.2.2 Extended Reality Object Interactions	7
2.2.3 Transition to an Imaginary World	8
2.3 Overview	9
2.4 External Components	10
2.5 Similar Applications	15
2.5.1 ArborXR	16
2.5.2 Protokol	17
2.5.3 Meta Quest for Business	18
3 SatoreStudio Application	21
3.1 Requirement Gathering	21
3.2 Early Application Prototype	22
3.3 Application Architecture	23
3.3.1 The Local Server Environment	24
3.3.2 Application Back-End	24
3.3.3 Application Front-End	26
3.4 UI Prototypes	26
3.4.1 First UI Prototype	27
3.4.2 Second UI Prototype	27
3.5 Functional Application	28

3.5.1	Early Functional Application	28
3.5.2	Final Functional Application	28
3.6	Future Work	31
4	User Experience	33
4.1	Procedure	34
4.2	Formative Tests Observations	36
4.3	Pilot Tests Observations	40
4.4	Final User Test Observations	41
4.5	Data Analysis	43
4.5.1	Data Demographics	44
4.5.2	Overall Usability Evaluation and User Feedback	44
4.5.3	Comparison of Task Completion Times Between Users Aged Below and Above 35	47
4.5.4	Comparison of UMUX Data Between Formative and Final User Tests	49
5	Conclusions	53
	Bibliography	56
A	User Testing Data	57
B	Procedure Document	71

List of Figures

1.1	Virtual Reality conversation of two users with their avatar replica.	1
1.2	Extend Reality setting, where one or four users will talk to Isaac, on the right. . .	2
2.1	Virtual Reality continuum proposition.	6
2.2	Talking with our digital human, Isaac, while a digital frame is displayed on the wall to the right.	7
2.3	Digital assets are overlaid in a physical object, in this case, a table.	8
2.4	Overview of the interaction between the application and the multiple extended reality headsets.	9
2.5	Unreal Engine demonstration.	10
2.6	Meta Quest Pro showcase.	11
2.7	Meta Quest 3 showcase.	11
2.8	Example of digitizing a sofa.	12
2.9	Screenshots capturing a person from various angles within a volumetric capture.	13
2.10	Commercial application of a volumetric video.	14
2.11	Tracking human body movements to then apply into a 3D model.	14
2.12	Motion-capture used in big industry movies like 'The Lord of the Rings.'	15
2.13	ArborXR website preview.	16
2.14	Protokol application preview.	17
2.15	Meta Quest for Business website preview.	19
3.1	Early prototype developed in C++ using "ImGui", "OSCpack", and "Vulkan".	22
3.2	Unreal Engine experience receiving OSC commands from the C++ application.	23
3.3	Architecture of the communication between the application and the multiple XR headsets.	24
3.4	Windows directory containing assets for retrieval during the preparation of the experience.	25
3.5	Central artwork displayed on the wall, adjustable by the director in real-time.	26
3.6	User Interface layout for the first Prototype.	27
3.7	User Interface layout for the second prototype.	28
3.8	Early functional application	29
3.9	Final functional application in the operator mode.	29

3.10	Final functional application in the director mode.	30
3.11	The Meta Quest operating system interface that is used before launching the XR experience.	31
4.1	A participant engaging with the devices and with the scripted tasks outlined in the procedure form.	35
4.2	Multiple terminals opening when starting Local Server.	37
4.3	One terminal that holds all the information.	37
4.4	MongoDB application opening when executing the laptop application.	38
4.5	Application didn't automatically open in operator mode.	38
4.6	Initial interface where the experience state was unclear, as no visual indicator confirmed its status.	39
4.7	Visual indicator showing the green background applied to list items, confirming that the XR experience is activer.	39
4.8	The updated form to add new XR device information.	40
4.9	Tablet keyboard configuration used for the forms.	42
4.10	The other tablet keyboard configuration that could have been used.	43
4.11	Age distribution and corresponding histogram.	44
4.12	UMUX 1: Satisfaction – Final group Likert scale evaluation.	45
4.13	UMUX 2: Frustration – Final group Likert scale evaluation.	45
4.14	UMUX 3: Ease of Use – Final group Likert scale evaluation.	46
4.15	UMUX 4: Correction of Errors – Final group Likert scale evaluation.	46
4.16	Task 1: Comparison of time taken by users below and above 35 years, with corresponding data distribution.	47
4.17	Task 2: Comparison of time taken by users below and above 35 years, with corresponding data distribution.	48
4.18	Task 3: Comparison of time taken by users below and above 35 years, with corresponding data distribution.	49
4.19	Task 4: Comparison of time taken by users below and above 35 years, with corresponding data distribution.	49
4.20	Comparison of UMUX1: Satisfaction Values for the formative and final group. . .	50
4.21	Comparison of UMUX2: Frustration Values for the formative and final group. . .	50
4.22	Comparison of UMUX3: Ease of Use Values for the formative and final group. . .	51
4.23	Comparison of UMUX4: Correction of Errors Values for the formative and final group.	51

Chapter 1

Introduction

This chapter outlines the motivation, context, and primary objectives that underpin the research presented in this thesis. It begins by exploring the rationale for developing solutions in the rapidly evolving domain of Virtual Reality (VR) technologies, emphasizing the advancements in hardware and software that enable immersive and interactive applications. Subsequently, the chapter situates the work within the collaborative framework of the TransmiXR EU project, highlighting its significance in driving innovation across multiple sectors, including Performance Arts, Cultural Heritage, and News Publishing. Following this contextualization, the specific objectives of the thesis are articulated, focusing on the design and development of an application to facilitate the operation and direction of multi-user Extended Reality (XR) environments. Lastly, the chapter overviews the proposed solution and the document's structure.

1.1 Motivation

The world of Extended Reality technologies is rapidly evolving. The recent launch of devices such as Meta Quest 3 [1], the Apple Vision Pro [2], and the forthcoming Samsung XR device [3], yet to be unveiled, demonstrates the growing trend of corporate investment in the VR industry.

Another highlight of VR technologies evolving was shown in The Lex Fridman Podcast, where Lex Fridman and Mark Zuckerberg showcase one of the state-of-the-art research use cases with Meta Codec avatar [4], as shown in Figure 1.1.



Figure 1.1: Virtual Reality conversation of two users with their avatar replica.

In the podcast, they were using the Meta Quest Pro in a Virtual Reality environment to talk to each other's avatar, highlighting the current advancements in VR applications and providing a glimpse into the future potential of immersive technologies that combine Head-Mounted Displays (HMD), 3D avatars, and artificial intelligence (AI) to create immersive interactive experiences.

This synergy of AI and HMDs opens up a diverse range of possibilities, including AI avatars with communication capabilities, personalized AI virtual assistants, and immersive XR experiences where these avatars use AI to generate coherent speech in our physical environments.

1.2 Context

This synergy forms the foundation of the TransmiXR EU project [5]. This project brings together seven universities and six industry partners aiming to explore and advance the potential of XR technologies, including tools for photogrammetry, volumetric video, and motion-captured AI tools. Ultimately, these diverse tools will be integrated and utilized by eight media partners responsible for creating multiple XR experiences. This synergy between academic and industry expertise fosters innovation and ensures a comprehensive, high-quality development of XR applications.

The media partners in the TransmiXR project are focusing on three different Immersive technology use cases: Performance Arts, Cultural Heritage, and News Publishing.

This master's thesis work was conducted at Satore Studio, a media partner in the consortium that has been developing VR experiences since 2015, and is responsible for building the XR experience for the Performance Art Case.

For this European project to be fulfilled, an application is needed where we can control, and direct the multiple XR headsets during the immersive experience

1.3 Objectives

The main goal of this master's work is to develop an application that will support the main XR experience being developed in Satore Studio. The XR experience consists of one to four participants using XR devices to talk to an AI character, Isaac, in an Extended Reality (XR) setting, who will narrate and chat about multiple life events, as depicted in Figure 1.2.

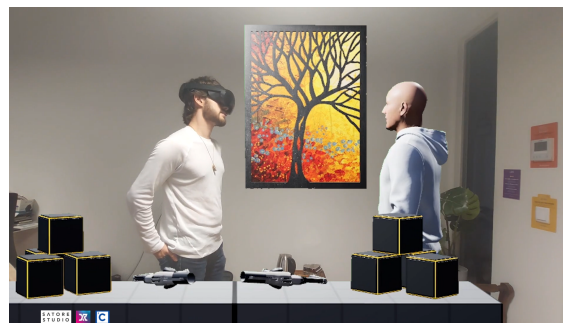


Figure 1.2: Extend Reality setting, where one or four users will talk to Isaac, on the right.

This application should offer a unique user experience by enabling live remote directing and control of three to four XR instances running Unreal Engine. It also should allow a director to issue commands manually or through an automated algorithm to gather custom assets, enhancing flexibility in content direction and asset management.

The primary objective of the application is focused on enabling two key functionalities:

Control XR Headsets

The application should enable an operator to setup the XR device's network configurations so that the operator can pause, restart, or stop the experience as needed for all headsets synchronously.

Direct XR Headsets

The application should allow the creation of multi-user personalized experiences by enabling both system-driven and director-led modifications to assets displayed to all participants collectively or to individual users selectively. The application performs command-based communication with XR devices, allowing them to retrieve custom assets—such as photos, videos, and 3D objects—from a local server.

1.4 Document Structure

The document is structured as follows:

Chapter 2 introduces the theoretical framework and examines related works pertinent to the application supporting the XR experience.

Following this, Chapter 3 delves into the development of the SatoreStudio application, detailing its architectural design, the requirements elicitation process, and an evaluation of the prototypes and functional versions.

Finally, Chapter 4 focuses on the user experience, providing an analysis of the testing phases, a review of the procedures and outcomes, and an overview of the final interface design improvements.

Chapter 2

Background and Related Work

This chapter provides context for understanding the development of the XR experience presented in this document. It is structured to guide the reader through the foundational concepts and the technologies used, as well as a review of similar applications in the field, offering a comprehensive view of both the theoretical and practical aspects of the project.

First, Section 2.1 introduces key concepts and terminology central to this document, establishing a basis for the reader to follow the rest of the document.

In Section 2.2 we explain the XR experience itself, highlighting the technologies and principles underpinning this immersive environment. This section is intended to give the reader a comprehensive understanding of the XR-specific requirements and challenges addressed in this project.

Next, Section 2.3 describes the objectives and scope of the project, elucidating the necessity for a supportive application to complement the XR experience.

Following that, Section 2.4 provides a summary of the tools and platforms utilized, illustrating how these resources contribute to achieving the XR experience final goals.

Finally, Section 2.5 reviews existing applications in the market that share functional similarities with our project. This review identifies gaps in current offerings, particularly those features essential for achieving the objectives outlined in Section 1.3.

2.1 Concepts and Terminology

This section outlines fundamental concepts and terminology related to the technologies applied in this work. A key concept in this work is the Virtual Reality Continuum, which is frequently referenced and encompasses the full spectrum of immersive technologies.

The continuum spans a range of experiences from augmented reality to fully immersive virtual reality, illustrating the degrees of digital content integration with the physical world. It highlights how various technologies blend physical and virtual environments to create diverse user experiences. Understanding these concepts is essential, as this project is focused on immersive technologies.

There are currently multiple definitions for the Virtual Reality Continuum, as no universally

agreed term exists. One possible illustration for the continuum is shown in Figure 2.1.

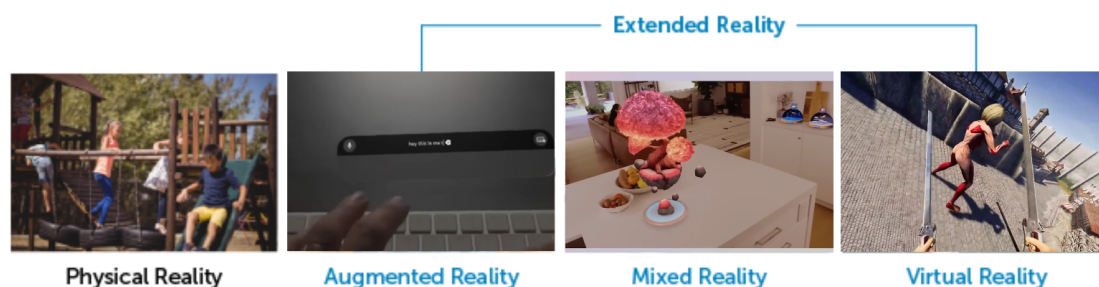


Figure 2.1: Virtual Reality continuum proposition.

To better understand the continuum, the following explanations clarify each of the mentioned terms:

Extended Reality (XR): XR is a generic term that encompasses the full spectrum of physical and virtual environments, including Virtual Reality, Augmented Reality, and Mixed Reality. It expands the concept of virtual immersion to include both fully immersive and partially immersive experiences. XR aims to blend the physical world with digital elements, creating new ways for users to interact with information and their physical or digital environment.

Augmented Reality (AR): Augmented Reality enhances the user's perception of reality by overlaying digital content onto the physical world. This integration allows digital information to integrate with physical objects. For example, as shown in Figure 2.1, a user can type on a physical Bluetooth keyboard while viewing a digital text output. This scenario demonstrates how digital enhancements enrich physical interactions in AR.

Mixed Reality (MR): MR refers to merging virtual content with the natural world to allow users to interact with digital and physical objects, individually or simultaneously. This involves placing virtual objects in the physical environment. MR often requires specialized software, such as Spatial Scanning. This technology captures and understands the physical space, allowing it to accurately place virtual objects into physical ones.

Virtual Reality (VR): VR refers to a fully immersive digital environment that completely replaces the physical world. Users wear a Head-Mounted Display and are transported into a computer-generated world, where they can interact and explore different virtual environments.

2.2 The Experience

This section provides an in-depth overview of the XR experience developed for the Performance Arts Case. Designed to merge the physical and digital, the experience utilizes interactive, immersive storytelling techniques, enabling participants to engage with Isaac, a Digital Human Avatar. The setting and structure are constructed to facilitate a narrative that evolves through different stages of Isaac's life, specifically at ages 22, 34, 45, and 66. The following subsections detail the environment, technological framework, user interactions, and narrative mechanisms that define

this experience.

2.2.1 Physical Environment and Setting

The physical setting that will be exhibited in the museum displays a domestic environment, specifically a dining room and a living room that represent Isaac's home. This choice of location aims to enhance the sense of intimacy and relatability within the narrative, placing participants in a familiar, life-like setting.

Upon entering this physical space, visitors begin their onboarding process, where a museum operator will guide them. With the help of our application, they will configure the XR devices and confirm if the multiple visitors are starting the immersive experience at the same time.

After starting the experience, they will be approached by Isaac, who will have dynamic conversations with the participants at different points in his life, namely 22, 34, 45, and 66. As seen in Figure 2.2, the setting could include digital assets, adding extra context to the narrative environment.



Figure 2.2: Talking with our digital human, Isaac, while a digital frame is displayed on the wall to the right.

The combination of physical and digital elements within this environment establishes an Extended Reality experience, where physical and virtual objects coexist. This blended setup allows the physical space to ground the experience, creating a tangible and meaningful setting for participants to explore and interact with digital assets overlaid on real-world objects.

2.2.2 Extended Reality Object Interactions

Participants explore a room populated with physical objects while wearing a head-mounted display equipped with spatial scanning capabilities. This feature of the HMD enables digital objects to be precisely overlaid onto physical surfaces, providing the integration of virtual elements within the real-world environment. Figure 2.3 illustrates this overlay, showing digital assets superimposed

on a physical table, an essential aspect of the Extended Reality capabilities of XR devices.



Figure 2.3: Digital assets are overlaid in a physical object, in this case, a table.

This overlay allows participants to interact with both physical and digital elements, which in turn elicits reactions from Isaac, further enriching the interaction. For instance, when a participant approaches or interacts with a digital representation of a refrigerator, Isaac humorously responds, "Have a beer, but please don't finish all of them." The interplay between physical and virtual elements also adds a sensory dimension to the interaction, blending the tactile of the real world with the qualities of digital storytelling with digital assets.

2.2.3 Transition to an Imaginary World

A core component of the XR experience is the transition to an "Imaginary World," a fully virtual environment designed to immerse participants in Isaac's memories and experiences. As participants progress through the narrative, Isaac seeks their attention, setting the stage for pivotal moments in his life. This transition is marked by an immersive narrative that transports participants into VR, allowing them to witness and experience the events narrated by Isaac as though they are unfolding in real time.

To transition from the extended reality setting into this VR environment, visitors are prompted to sit, and the HMD's built-in cameras are deactivated. By disabling these cameras, participants are effectively removed from the physical space, isolating them within a fully digital environment. This design choice is intended to enhance the depth of immersion, as participants are no longer visually connected to the physical room. By engaging solely with the virtual space, users become fully absorbed in the storytelling, enabling a profound level of narrative engagement that blurs the line between reality and the virtual domain.

Within the imaginary 3D worlds, participants experience narrated stories delivered by a human actor whose voice embodies Isaac's character. This design choice is integral to the shift from an interactive environment to a cinematic, passive engagement. By utilizing a human actor's

voice, with Isaac’s distinct vocal characteristics, the experience is imbued with a high level of authenticity, making the participants think they are still narrating with Isaac.

In the next section, Section 2.3, an analysis of the purpose and need for a supportive application will be provided, emphasizing the technical and narrative components that underpin the experience.

Following that, Section 2.4 will delve into the external tools and technologies referenced in Section 2.3, offering a review of how these components interact with the application to create a cohesive and engaging XR experience.

2.3 Overview

Figure 2.4, provides an overview of the integration with the main XR project, that will run on one to four XR devices at the same time. This figure highlights the requirement for an external application that fulfills the two primary functions defined in Section 1.3: Control and Directing.

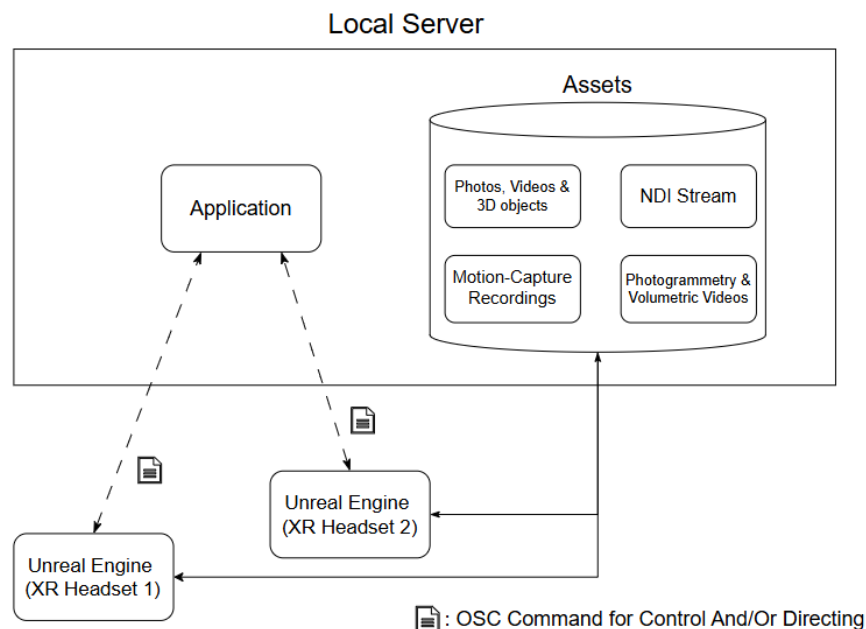


Figure 2.4: Overview of the interaction between the application and the multiple extended reality headsets.

The Control function plays a crucial role in synchronizing all connected head-mounted displays, ensuring that all users initiate the experience simultaneously. This synchronization is vital for creating a cohesive and unified experience, particularly in settings where multiple participants are involved in a shared immersive environment.

The Directing function manages the distribution of digital assets to individual devices. Through an OSC (Open Sound Control) command, the function coordinates the delivery of appropriate assets from the Local Server Assets database to each device. By dynamically selecting assets from a server-hosted library, the application is able to personalize the experience for each participant. The

assets may include Network Device Interface (NDI) streams, 3D models, and multimedia content such as images and videos. This on-demand asset deployment ensures that each device receives tailored content, enhancing engagement by providing contextually relevant media throughout the experience.

A deeper examination of these components and their functions is provided in the following section. This will elucidate the technical and operational details that enable the application's core functions of Control and Directing.

2.4 External Components

This section provides an overview of the external components integrated into this project, detailing their roles and contributions.

Unreal Engine

The Unreal engine application, as seen by the example in Figure 2.5, is a versatile and powerful 3D engine essential for creating games, film productions, and, in our example, high-quality XR experiences. As shown in Figure 2.4, it integrates all of the assets stored on the local server into the XR experience. This engine operates directly on the XR headsets, enabling the use of diverse media assets, including motion-capture recordings, photogrammetry, volumetric videos, NDI streams, 3D objects, and multimedia content such as photos and videos.



Figure 2.5: Unreal Engine demonstration.

Meta Quest Pro

The Meta Quest Pro, shown in Figure 2.6, launched in 2022, represented Meta's high-end XR headset at the time, featuring advanced capabilities, most notably its pass-through functionality.

The Meta Quest Pro was instrumental in running our initial prototype XR experience developed in Unreal Engine, providing a robust platform for testing and refining key functionalities essential to the project's goals. This device played a crucial role in the initial stages of developing our XR experience, as it was used to conduct the first Open Sound Control (OSC) communication tests with the Unreal Engine instances.



Figure 2.6: Meta Quest Pro showcase.

Meta Quest 3

The Meta Quest 3, represented in Figure 2.7, released in 2024, represents a significant upgrade over its predecessor, the Meta Quest Pro. It offers enhanced XR capabilities, including improved pass-through, better graphics performance, and more ergonomic design. This headset was chosen for the final user experience due to its superior hardware and software advancements, providing users with more immersive and responsive XR environments. Its improved battery life and comfort also make it ideal for longer sessions, ensuring a better overall experience for end-users.



Figure 2.7: Meta Quest 3 showcase.

Digitalization of Physical Assets

The digitalization of physical assets involves the use of advanced techniques to capture and transform real-world objects and environments into digital formats. These techniques include photogrammetry and volumetric video, both of which enable the creation of accurate 3D representations of physical assets.

These technologies are of particular relevance to the ongoing research initiatives at our university partners, who are exploring their potential applications for enhancing XR environments. The assets produced through photogrammetry and volumetric video capture will be stored on a local server and subsequently transmitted to XR devices for use in immersive experiences. This process allows for the integration of real-world objects and environments into virtual spaces, enhancing the realism and interactivity of XR applications.

Photogrammetry

Photogrammetry is a process that converts sequences of photographic images into accurate, three-dimensional digital models, as seen in Figure 2.8. Through careful image alignment, this technique captures even the finest structural details of an object, environment, or building, making it indispensable across multiple fields, including XR applications, architecture, archaeology, and forensics.



Figure 2.8: Example of digitizing a sofa.

In XR, photogrammetry allows developers to incorporate lifelike, high-quality 3D objects into virtual or physical environments, as seen in applications like Polycam [6]. This type of software enables users to digitize physical objects with a smartphone camera, facilitating the integration of real-world assets into digital experiences. Cultural heritage preservation also benefits from photogrammetry by creating digital archives of fragile artifacts, enabling detailed examination and conservation without risking physical damage. This technique is particularly valuable for documenting endangered sites, such as ancient ruins or unique geological formations, in accessible 3D models that can later serve educational and preservation purposes.

In construction and civil engineering, photogrammetry provides precise topographical data

and 3D models, improving project accuracy and cost-effectiveness. By using drones to capture aerial photographs, developers can create realistic site models that facilitate planning and real-time project adjustments. This technology minimizes reliance on traditional surveying methods, reducing costs and project timelines significantly, especially in large-scale developments with complex landscapes.

Forensic science employs photogrammetry to document and recreate crime scenes, allowing investigators to analyze detailed 3D reconstructions without disturbing physical evidence. This approach is particularly advantageous for capturing fragile or time-sensitive evidence, enabling virtual re-examinations and potentially providing a clearer perspective during court reviews.

Volumetric Captures

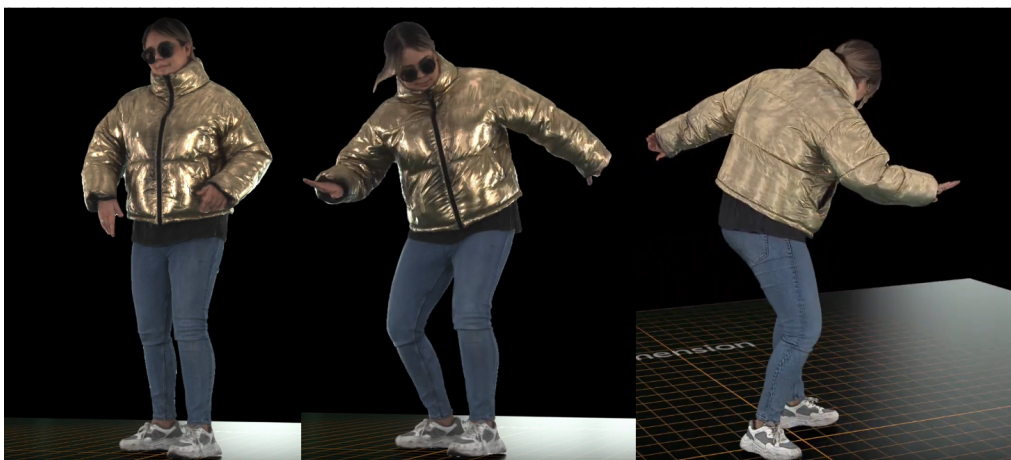


Figure 2.9: Screenshots capturing a person from various angles within a volumetric capture.

Volumetric capture, or volumetric video, is a technique for recording a three-dimensional space in real-time, enabling an immersive digital reproduction of physical spaces or live performances, as exemplified in Figure 2.9. This method is particularly advantageous for capturing dynamic scenes, such as a person dancing, which can then be reproduced as interactive, navigable 3D content in various digital environments, including games, film production, and virtual reality.

The technology behind volumetric capture involves the use of multiple cameras arranged to capture all angles of a subject, which are then processed to create a volumetric video—a fully three-dimensional, viewable asset that allows users to interact with the scene from any angle. This approach brings distinct benefits to sports broadcasting and events, offering viewers the ability to explore detailed replays from multiple perspectives, such as a player’s point-of-view (POV), as shown in Figure 2.10, an area being researched by our Intel partners [7].

Despite its transformative potential, volumetric capture faces several technological challenges. One of the primary obstacles is the need for specialized equipment capable of capturing high-resolution images from all angles. This often necessitates an extensive array of cameras and substantial computational power, which can be resource-intensive. Processing these large datasets



Figure 2.10: Commercial application of a volumetric video.

to achieve realistic textures and accurate representations further amplifies the computational demands. As such, significant research efforts are currently being focused on enhancing computational efficiency [8], improving capture accuracy, and minimizing the hardware requirements to make this technology more accessible and practical for a broader range of applications.

This is a key area of focus within the TransmiXR group, where universities involved in the project are conducting research to address these challenges. Their work aims to refine the techniques and infrastructure needed to improve the scalability and effectiveness of volumetric capture technologies in real-world applications. As these advancements continue, volumetric capture holds the potential to play an increasingly important role in creating immersive media experiences across diverse fields, including XR, gaming, and entertainment.

Motion-Captured Animation

Motion capture is a process used to record the movement of objects or people, translating these movements into digital data for animation or analysis. In its most common use, it enables the animation of digital characters by tracking the movements of human actors, which is essential in industries like filmmaking, video games, and XR experiences. The core benefit of motion capture lies in its ability to produce realistic, fluid movements by directly capturing real-world actions, making it a valuable tool for creating lifelike animations and enhancing immersive media experiences. It can be executed through systems like body trackers and AI-powered common smartphone cameras, as shown in Figure 2.11.

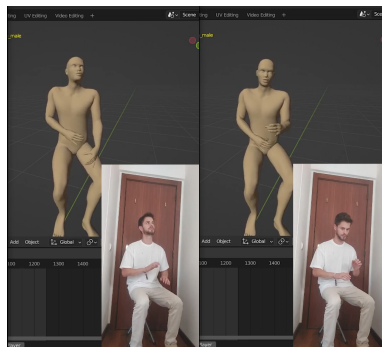


Figure 2.11: Tracking human body movements to then apply into a 3D model.

Mocap technology is applied across various sectors, including entertainment, sports, and health-care. In sports, motion capture can optimize an athlete's performance by providing detailed motion analysis, which can be used for injury prevention and improving technique. Furthermore, in the medical field, it aids in rehabilitation and diagnosis by analyzing movement disorders or monitoring recovery. Commercially, companies like Rokoko [9] and MOVE.AI [10] are leading in the motion of these capture technologies and making this technology more accessible by enabling it in standard smartphone devices.

In filmmaking, for instance, allows for the animation of characters like Gollum in *The Lord of the Rings*, as seen in Figure 2.12 captured from the Behind The Scenes of the film ¹, by recording human movements and translating them into digital characters.



Figure 2.12: Motion-capture used in big industry movies like 'The Lord of the Rings.'

While motion capture offers many benefits, it also presents some challenges, primarily in terms of cost and precision. High-end motion capture systems often require sophisticated setups, including cameras, sensors, and suits, which can be expensive and complex to operate. The precision of tracking, particularly for finer details like finger movements, remains a limitation, especially when using less advanced systems such as smartphone-based cameras.

The future of motion capture seems to lie in further refining these technologies to increase accuracy and decrease hardware requirements. MOVE.AI is trying to make multi-person tracking with a single smartphone or other compact devices that could significantly expand its applicability, particularly for smaller studios and indie developers.

2.5 Similar Applications

This section examines various applications that share functionality with the system developed for this XR project. By comparing these solutions, we highlight their respective features, capabilities, and limitations. The goal is to provide a broader context for understanding the strengths and weaknesses of similar systems and how they inform the development of our own solution.

¹<https://www.youtube.com/watch?v=wz7YUyCEGE>

2.5.1 ArborXR

ArborXR² is an advanced Mobile Device Management (MDM) platform specifically designed for the management and deployment of XR devices in enterprise environments. It offers a suite of functionalities to manage XR devices, which is valuable for enterprises aiming to streamline their XR hardware usage. ArborXR enables administrators to perform key tasks remotely, such as updating device software, configuring system settings, and monitoring device performance metrics. These capabilities provide a more structured and efficient management process, allowing organizations to ensure that all devices function effectively and maintain security standards throughout the deployment cycle.

In particular, ArborXR's remote management features allow centralized control over a dispersed network of devices, reducing the time and technical labor associated with in-person device maintenance. This feature is advantageous for large-scale or geographically distributed deployments, where device performance monitoring and software updates are needed across multiple devices. Figure 2.13 provides a preview of ArborXR's user interface, which reflects its aim to simplify XR device administration through an intuitive management portal.

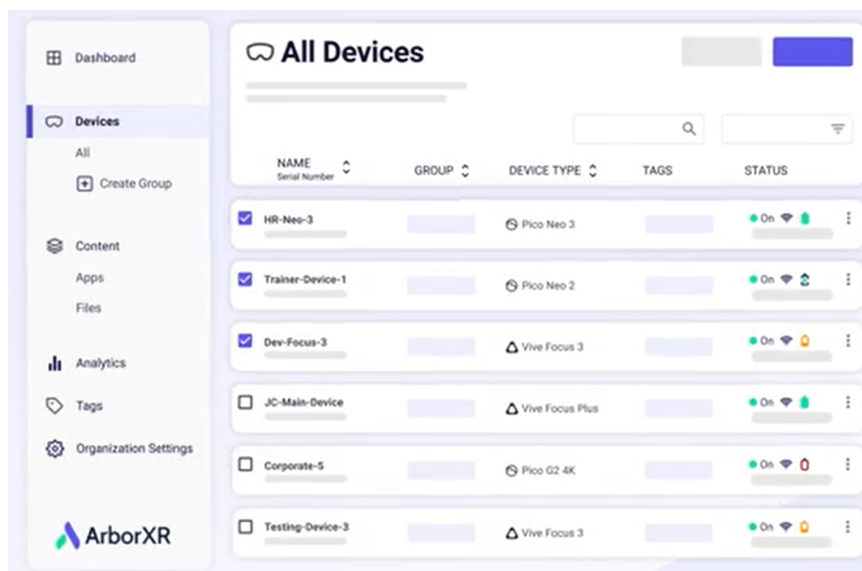


Figure 2.13: ArborXR website preview.

Despite its comprehensive device management capabilities, ArborXR has limitations that may affect its suitability for all XR projects. A significant constraint of ArborXR for certain XR projects is its inability to support the Directing objectives referred to in Section 1.3. This means that the system lacks functionality for sending personalized content to individual devices in real-time, such as delivering customized media packages to specific users within an Unreal Engine-based XR experience. This lack of content differentiation prevents the platform from being used to control interactive storytelling elements or personalized user experiences in a XR environment, which

²<https://arborxr.com/>

may be essential for XR applications requiring adaptable, user-specific content deployment.

Furthermore, ArborXR's pricing model can be a consideration for smaller-scale enterprises or projects with budget constraints. The platform's trial version is priced at €7 per device per month³, which may accumulate to significant costs for organizations deploying large numbers of devices. This pricing model could limit its feasibility for smaller deployments where cost efficiency is a priority.

In summary, ArborXR is a powerful tool for MDM of XR devices, offering efficient remote control, configuration, and monitoring. However, its lack of individualized content deployment and its subscription pricing model are critical considerations for enterprises evaluating its compatibility with specific XR project needs.

2.5.2 Protokol

Protokol⁴, developed by Hexler, is a cross-platform application used for monitoring and logging various control protocols, including MIDI (Musical Instrument Digital Interface) and OSC (Open Sound Control). This versatility makes Protokol valuable for a range of applications that require real-time communication with and between devices. Figure 2.14 shows a preview of the Protokol interface, illustrating its streamlined design that facilitates control and data tracking.

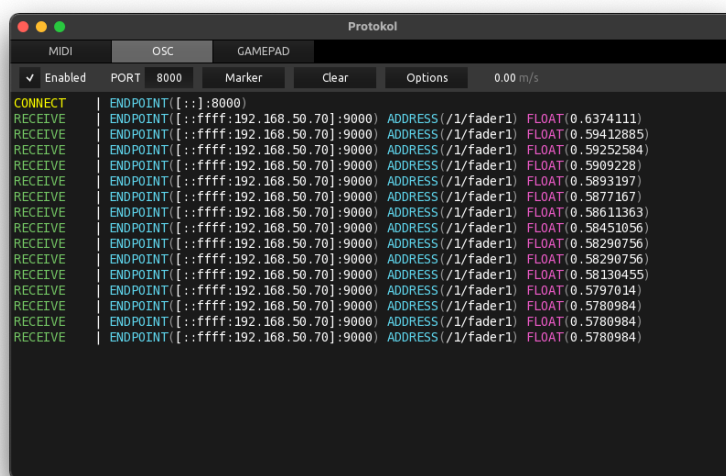


Figure 2.14: Protokol application preview.

Originally intended for internal use to aid in app testing and troubleshooting in studio environments, Protokol has since expanded to support an extensive array of functionalities. The application enables developers and operators to log and troubleshoot communication protocols, which is especially useful in settings that rely on seamless control and integration, such as audio-visual production environments, interactive media installations, and multi-device XR applications.

³<https://arborxr.com/pricing/>

⁴<https://hexler.net/protokol>

Its utility as a protocol monitoring tool makes Protokol a beneficial component in the workflow of professionals who need precise control over protocol-based interactions.

However, Protokol also has limitations in the context of our XR application. Although Protokol supports directing capabilities by allowing users to monitor and manage control protocols, it lacks controlling features for XR devices, objectives referred to in Section 1.3. For instance, Protokol does not provide functionality to track the operational status of XR devices, check their online or offline status, or send global commands across multiple XR devices. This limitation restricts its effectiveness in XR experiences where operators must have centralized control over multiple XR headsets to check their status, synchronize actions, or deploy commands across devices.

In summary, Protokol is a robust application that supports a wide range of protocol-monitoring needs across multiple platforms, with particular strengths in tracking and logging MIDI and OSC communications. Nevertheless, its inability to control XR device status and execute network-wide commands limits its applicability for XR experiences requiring real-time multi-device control. Thus, while it is an effective tool for protocol management, its constraints in XR device management reduce its suitability for more complex XR deployments.

2.5.3 Meta Quest for Business

Meta Quest for Business⁵ is a subscription-based service developed by Meta, designed to optimize enterprise use of Meta Quest XR and XR devices. By enabling comprehensive device, user, and application management, the platform supports XR-focused training, collaboration, and productivity within professional environments. The service includes a range of operating features, such as access control through a Shared Mode, which allows devices to be shared among multiple users while securing app access with session-specific passcodes. Additionally, a Casting feature enables the display of in-headset experiences on external monitors, facilitating real-time demonstrations, collaborative sessions, and training presentations. Figure 2.15 provides a preview of the Meta Quest for Business interface.

Despite its capabilities, Meta Quest for Business presents some limitations relevant to small-scale enterprises and region-specific deployments. For instance, the platform is not currently available in Portugal⁶, which restricts its accessibility for projects, such as this XR experience, that are being developed and deployed there. Additionally, the subscription model involves a cost of €15 per user per device. This pricing structure may be prohibitive for smaller enterprises or for those deploying a large number of devices, where per-device costs accumulate significantly over time.

In summary, while Meta Quest for Business offers a robust suite of management and collaboration tools tailored for professional XR use, its accessibility limitations and cost considerations may impact its adoption, especially for smaller organizations or deployments outside of supported

⁵<https://forwork.meta.com/quest/business-subscription/>

⁶<https://work.meta.com/help/307276701907179>

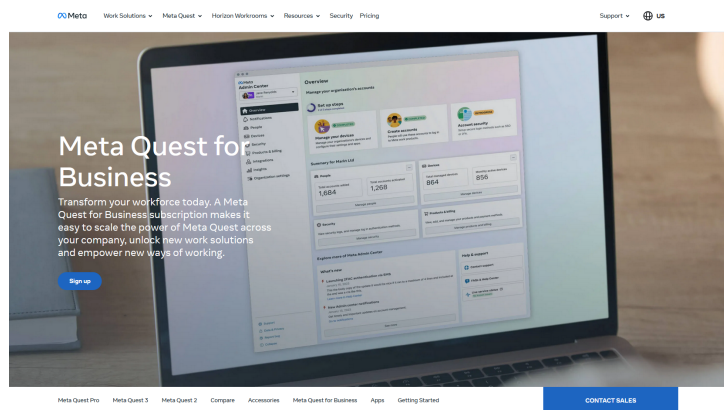


Figure 2.15: Meta Quest for Business website preview.

regions. This highlights the need for alternative solutions that can meet similar requirements while offering greater flexibility in terms of regional availability and cost.

Discussion

Both Meta Quest for Business and ArborXR are solutions that offer specific functionalities for managing XR devices and environments. However, they show limitations in terms of customization, extension, and integration with specific project requirements. One key restriction is their limited ability to facilitate custom solutions within Unreal Engine. These platforms do not support the dynamic management of different scenes or facilitate the personalized delivery of assets, such as images or other media done in the Director Mode of our application. Additionally, these platforms require users to dedicate significant time to learning their interfaces and functionalities, with ongoing monthly subscription fees that may present financial barriers for smaller-scale enterprises or projects. Such factors make these solutions less ideal for customized or budget-conscious XR deployments.

In contrast, Protokol by Hexler offers substantial capabilities for directing, especially in the management of control protocols like MIDI and OSC. While it supports the orchestration of content and sequences, it lacks the ability to manage or control XR devices themselves. This limitation means Protokol is better suited for specific tasks like scene directing or audio-visual control but is not applicable for comprehensive device management in an XR environment. As a result, Protokol, while functional in specific scenarios, cannot address the operating requirements asked by the stakeholders.

Our application, on the other hand, aims to provide a more flexible and integrated solution. Unlike the proprietary platforms, it will support full control over XR devices, offering extensive customization options that can be tailored to the specific needs of the project. This flexibility extends to its seamless integration with Unreal Engine, enabling users to implement custom scenes and manage the dynamic delivery of content across different devices. By eliminating financial barriers and providing a higher degree of flexibility, our solution could enhance the overall func-

tionality and scalability of XR experiences, aligning more closely with the diverse needs and preferences of both developers and end-users. This makes it an ideal choice for enterprises looking to implement or scale XR management solutions without the constraints imposed by the other researched tools.

Chapter 3

SatoreStudio Application

In this chapter, we detail the requirements gathering, architecture, user interface (UI) prototyping, functional prototypes, and user testing phases involved in the development of the application. Each section provides a comprehensive look at the methodologies, design considerations, and iterative processes that inform the final product, focusing on the alignment between stakeholder objectives and technical implementation.

3.1 Requirement Gathering

Since the beginning of application development, stakeholders, namely the project director, outlined a range of objectives that evolved as the project progressed. During the initial three months, the team explored and tested multiple tools referred to in Section 2.4. This exploratory phase was essential in establishing the application's core requirements, ultimately confirming the need for a solution that would effectively support Control and Directing functionalities.

Throughout this period, the team conducted several informal interviews with stakeholders to clarify requirements. These conversations were invaluable in shaping the application's functionality and user interface by capturing insights into specific project needs.

During the design iterations, we developed a functional prototype using C++. This prototype successfully transmitted OSC messages, demonstrating the application's foundational communication capabilities. However, due to significant limitations within the framework, this version was ultimately set aside, as it will be discussed in Section 3.2.

These early prototyping efforts were instrumental in refining the project's direction, revealing specific design and functionality gaps, and guiding the selection of a more versatile technology stack.

As a result, the team opted to rebuild the application using a modernized architecture that better facilitates dynamic interactions, real-time UI (User Interface) updates, and improved display flexibility. This new approach enhances the user experience by allowing more responsive, adaptable interfaces that meet the unique demands of XR environments. This updated stack, detailed in Section 3.3, leverages Angular.js for responsive front-end development, and Express.js for streamlined, efficient back-end handling. This approach not only improves the application's

ability to manage intricate interactions but also strengthens its adaptability to the unique demands of immersive XR experiences setups.

To align the design with stakeholder expectations, non-functional prototypes were presented, providing visual and interactive representations of potential UI configurations. These initial UI designs offered stakeholders an early understanding of the anticipated User Experience (UX), laying the groundwork for iterative improvements in the design phase. Further details of this iterative process and its outcomes are presented in Section 3.4.

3.2 Early Application Prototype

This section details the development and iterative refinement of the prototype created using the Open Sound Control (OSC) protocol. The primary objective of this prototype was to evaluate various technical approaches and assess the feasibility of different technologies to support and enhance the XR experience effectively.

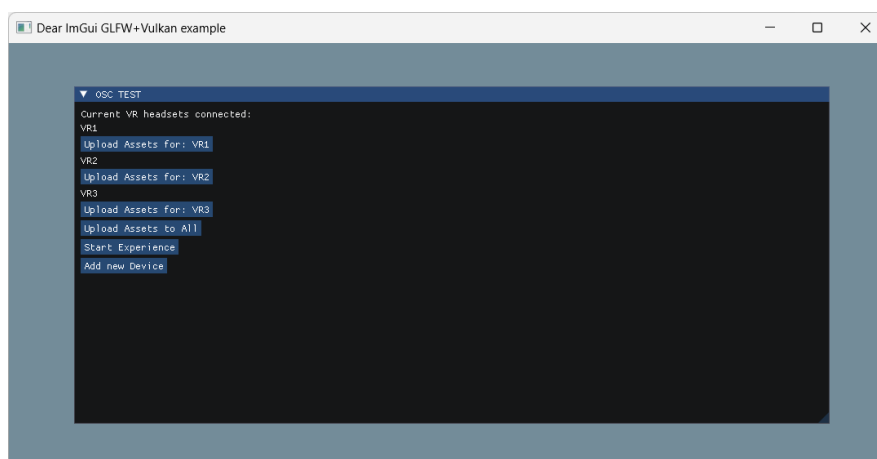


Figure 3.1: Early prototype developed in C++ using "ImGui", "OSCpack", and "Vulkan".

Our prototype, as seen in Figure 3.1 was built with the following tools:

- **ImGui:** A minimal, lightweight graphical user interface (GUI) library for C++ applications that prioritizes rapid development with few dependencies. ImGui was chosen to create a simple, interactive interface for testing the control requirements of the XR experience.
- **OSCpack:** A C++ library designed for packing and unpacking OSC packets, enabling seamless communication between systems. This tool allowed the prototype to transmit commands efficiently to Unreal Engine, a core requirement for the XR project.
- **Vulkan:** A cross-platform, high-performance API for 3D graphics rendering, providing developers with extensive control over GPU resources. Vulkan was chosen to enhance rendering speeds in Director Mode, allowing for faster visual updates and facilitating the efficient transmission of 3D assets to multiple XR headsets.

As shown below in Figure 3.2, the initial prototype was functional and successfully integrated into an XR experience, with the C++ application effectively sending OSC commands to Unreal Engine.



Figure 3.2: Unreal Engine experience receiving OSC commands from the C++ application.

These tools were chosen during the requirements gathering phase to support potential changes to the NDI live stream displayed across various XR headsets in operator mode. For Director Mode, using the Vulkan API was deemed optimal to ensure faster image rendering, providing directors with a more responsive visual experience. Additionally, Vulkan would facilitate plans to transmit 3D assets to multiple XR headsets with greater efficiency. However, after the initial implementation and further research, it became evident that this implementation of the application could not provide a satisfactory user interface or experience using these tools, ultimately rendering it impractical for regular use by operators and directors.

As shown in Figure 3.1, there is little flexibility in the design, accompanied by significant difficulties and restrictions for the interface. The implementation only allowed for vertical and inline elements, which limits the creativity and functionality of the user interface. This constrained layout made it challenging to create an engaging and user-friendly experience for operators and directors, ultimately hindering the application's practicality and usability in real-world scenarios. Given these limitations identified in the interface, a new final application was developed using an updated technology stack, which we will explore in the next Section 3.3.

3.3 Application Architecture

This section outlines the architectural structure of the final architecture application, covering the core components and their interactions, including data flow and local server interaction with the application in the tablet and XR headsets. The architecture was designed to support control and directing functionalities as specified in the requirements. Detailed diagrams and technical explanations highlight the architecture, ensuring the system can meet evolving project needs.

The architecture, as shown in Figure 3.3, consists of multiple components that will be discussed in the next subsections.

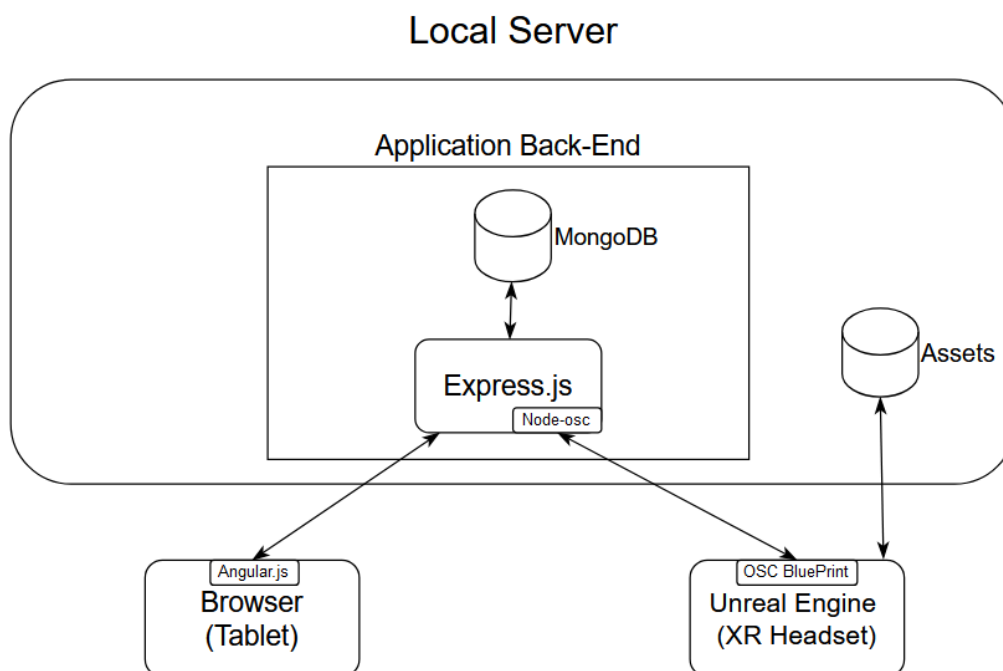


Figure 3.3: Architecture of the communication between the application and the multiple XR headsets.

3.3.1 The Local Server Environment

The system is primarily deployed within a local server environment, a requirement specified by the stakeholder to support the XR experiences. Given that each experience is situated on-site, a dedicated external computer serves as the local server. This setup ensures that all devices remain synchronized, allowing the operator to control the experience and confirm simultaneous initiation across all participants. Additionally, this server enables directors (where applicable) to directly manage the experience, including making real-time adjustments to asset folders on the local machine or server, as we'll see in the next subsection 3.3.2. For instance, directors can change the images and photos displayed within the experience. Hosting the system locally also minimizes latency, both critical for an optimized and controlled user experience.

3.3.2 Application Back-End

The system's back-end is the core component, responsible for handling the application front-end requests, processing the configurations and synchronization of the headsets, managing interactions with the database, and specifying what assets to download. This back-end is built using Express.js

¹, a fast, minimalist web framework for Node.js ². Express.js facilitates the creation of robust APIs and handles routing, middleware, and server-side logic efficiently.

Express.js: Serving as the central framework of the back-end, Express.js facilitates the management of HTTP requests from client-side applications, processes these requests, and dispatches appropriate OSC (Open Sound Control) responses to XR headsets. Its modular and scalable architecture supports the integration of various middleware, enabling efficient and adaptable server-side development. Notably, Express.js allowed the integration of the node-osc library³, which provides the functionality needed to communicate with XR headsets. These headsets utilize an OSC Plugin Blueprint⁴ within Unreal Engine, enabling them to send and receive OSC commands over the UDP protocol. This setup ensures seamless real-time communication between the server and XR devices.

MongoDB: The database management system utilized in this architecture is MongoDB ⁵, a NoSQL database recognized for its high performance, scalability, and flexibility. MongoDB stores XR headset data in JSON-like documents, capturing essential information such as IP addresses, status, and device names.

Assets: The system utilizes various external files, as referred to in Section 2.3. These assets are stored and controlled by the stakeholder/director of the Performance Art Case, and its architecture is presented in Figure 3.4.

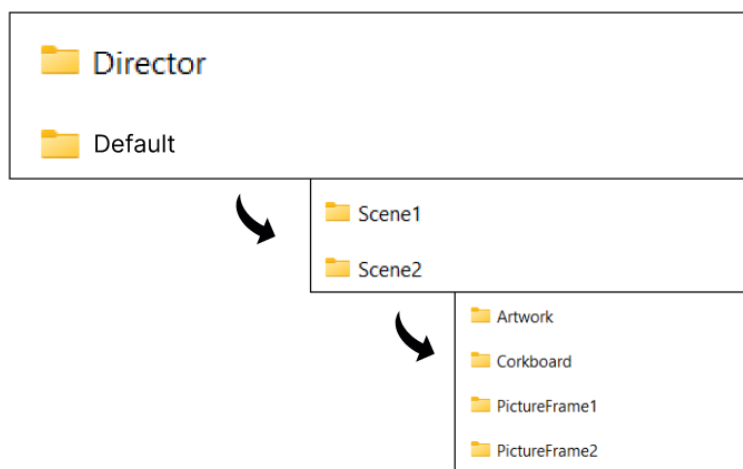


Figure 3.4: Windows directory containing assets for retrieval during the preparation of the experience.

When the director is unavailable on-site, the XR headsets automatically retrieve data from the default folder on the server.

Specifically, this means that the application accesses relevant assets stored within subfolders,

¹<https://expressjs.com/>

²<https://nodejs.org/en>

³<https://www.npmjs.com/package/node-osc>

⁴<https://dev.epicgames.com/documentation/en-us/unreal-engine/osc-plugin-overview-for-unreal-engine>

⁵<https://www.mongodb.com/>

such as the artwork folder, which contains image files designated for display on the virtual artwork digital asset inside the experience, as seen by Figure 3.5.

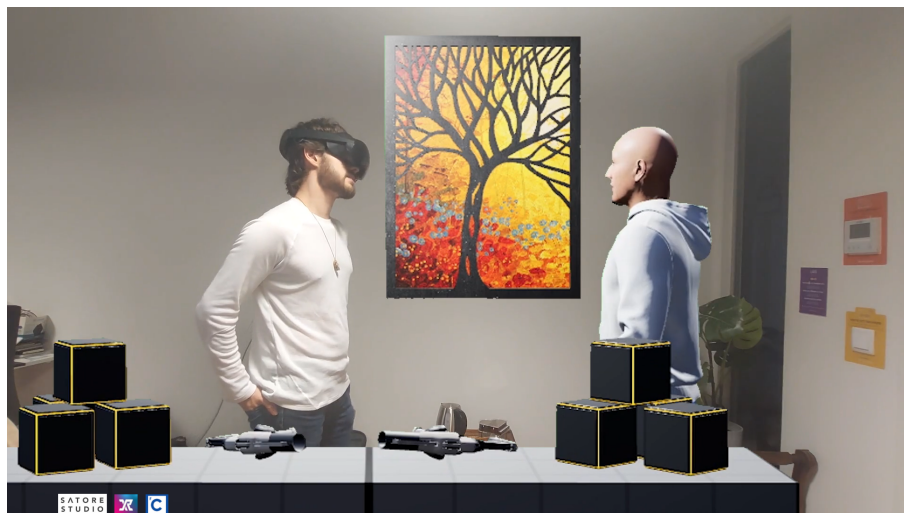


Figure 3.5: Central artwork displayed on the wall, adjustable by the director in real-time.

This system enables the headset to present pre-defined media content even in the absence of direct input from the director, ensuring that critical visual elements—like the artwork in this example—are consistently available for presentation within the environment.

3.3.3 Application Front-End

The client-side interface represents the front-end component of the system, which is responsible for user interaction and presentation. This interface is built using Angular.js ⁶, a powerful JavaScript framework for developing dynamic web applications.

Angular.js The Angular.js application runs on a browser, and its UI was specifically designed for tablet devices. This application communicates with the Express.js back-end to fetch data, display it to the user, and send user inputs back to the server. Angular.js provides a rich set of features, including two-way data binding, dependency injection, and modular architecture, enabling the creation of highly interactive and responsive user interfaces.

3.4 UI Prototypes

Multiple UI prototypes were developed throughout the design phase. This section presents two key iterations—a preliminary version and a more refined version—to illustrate the evolution of the UI based on the stakeholder feedback. Notable differences between these versions highlight the progression toward a user-centered design, reflecting improved usability and functionality. The final design will be presented in Section 3.5.

⁶<https://angular.dev/>

3.4.1 First UI Prototype

As illustrated in Figure 3.6, a comparison with Figure 3.7 reveals that the initial user interface remained significantly underdeveloped. This immaturity is evident in several key areas, particularly in how the experience was presented and interacted with.

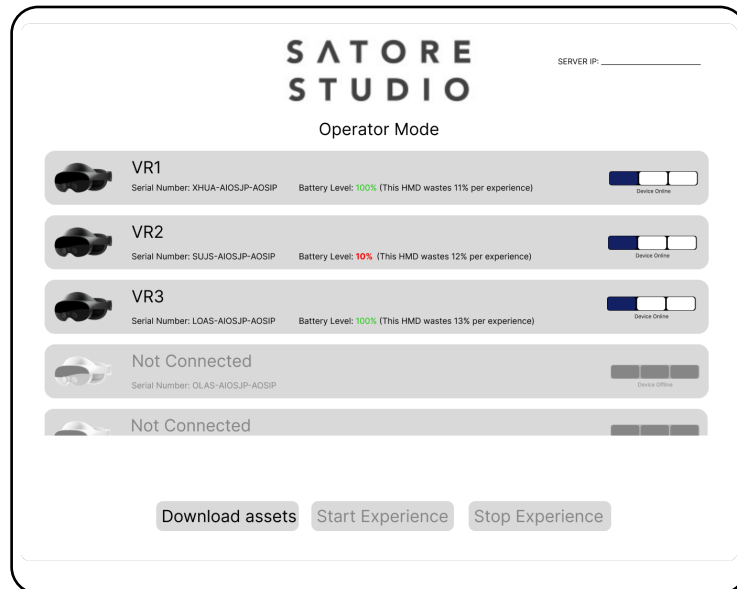


Figure 3.6: User Interface layout for the first Prototype.

Firstly, the absence of dedicated buttons for each headset to configure either a standard or personalized experience highlighted a significant oversight in the user experience design of the directing mode.

Additionally, the user interface did not include buttons for configuring headset settings, which are crucial for changing the information of each headset by an operator or director controlling the XR experience, namely the XR IP address, port, and name.

Moreover, the experience control buttons at the bottom of the interface were overly simplified, resulting in a lack of intuitiveness. This design choice fails to facilitate a seamless experience, essential for operators and directors to correctly set up the XR Experience, making the experience more frustrating.

Finally, the overall aesthetic of the application left much to be desired. The design did not effectively engage users. These deficiencies highlighted the need for further refinement and thoughtful consideration in developing the user interface to enhance usability and visual appeal.

3.4.2 Second UI Prototype

As we can see in Figure 3.7, the previously reported problems were solved. The application now clearly shows the current status of the headsets. It now includes settings buttons for further customization of the headsets. It also has intuitive controlling controls, and the overall aesthetic was improved by the overall look and extra animations presented in the app.

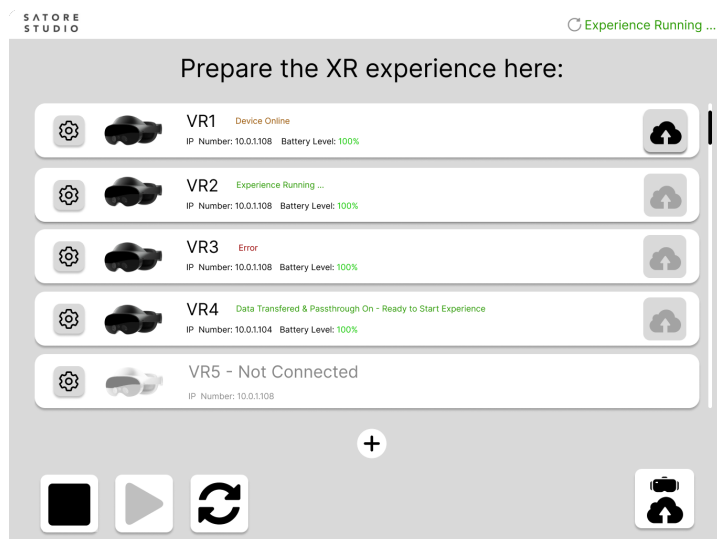


Figure 3.7: User Interface layout for the second prototype.

3.5 Functional Application

In developing an effective user interface for XR device management, the design and functionality of the application evolved significantly through multiple stages. The early functional applications were primarily focused on achieving basic functionality, with minimal emphasis on user experience or visual coherence. This section presents multiple versions of the functional application.

3.5.1 Early Functional Application

The initial functional application for the Operator Mode (Figure 3.8) represents the first working prototype with a basic and minimal user interface, prioritizing functionality over visual appeal or user experience refinement. This version was created with a limited set of design elements, focusing primarily on the core capability to manage and control XR devices within the operator's environment. Although functional, this prototype had several limitations in terms of flexibility and intuitiveness, which impacted the usability and scalability of the application.

3.5.2 Final Functional Application

After multiple iterations and user feedback sessions, the final version of the functional application was developed, significantly enhancing both Director Mode and Operator Mode. As we'll see in the next subsections, these refined versions of the UI address many of the initial prototype's limitations, namely that it lacked clear headset status indicators, no options for headset configuration, overly simplified control buttons, and a visually unrefined design. In this section, we can see the introduce an improved, user-centered design tailored to the specific needs of both directors and operators.

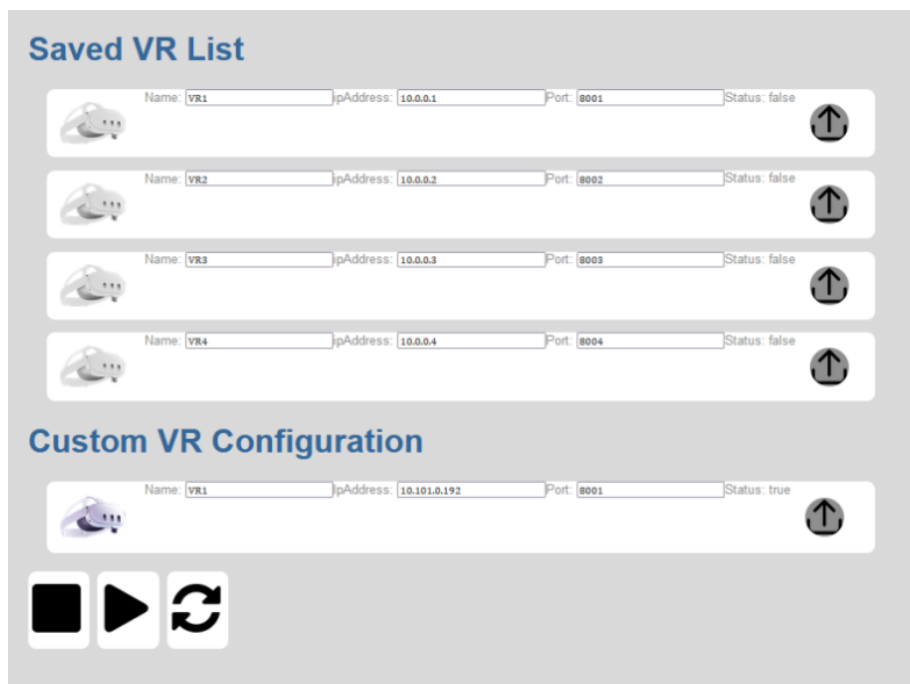


Figure 3.8: Early functional application

Operator Mode

The Operator Mode has similarly been redesigned to incorporate user-friendly elements that prioritize device connectivity and monitoring tasks, as shown in Figure 3.9. A primary enhancement in this mode is improved device connectivity management. The final version of Operator Mode clearly delineates the status of connected devices through indicators such as "online," "offline," "error," and "Experience Running," providing immediate clarity on each device's availability and operational status. This improvement ensures that operators are fully informed of device readiness at all times, thereby supporting the efficient preparation of XR experiences.

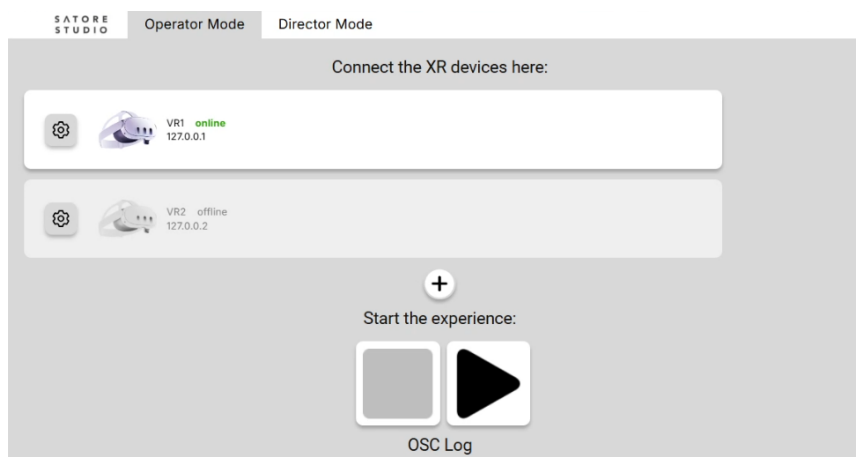


Figure 3.9: Final functional application in the operator mode.

Furthermore, the design consolidates essential functions, minimizing the steps required to

manage device states and promoting a more intuitive user experience. An emphasis on aesthetic improvements and consistency has also been applied across both Director and Operator modes. This includes a cohesive color scheme based on what status the device is, an aligned layout, and a consistent button design, all of which enhance the application’s overall usability. This visual and functional consistency enables users to transition between roles or tasks seamlessly, without the need for extensive reorientation, thus contributing to a smoother and more efficient workflow.

In addition, the application features a responsive user interface that adapts to different tablet sizes, ensuring an optimal user experience across various device dimensions. This responsiveness is essential for operators and directors working in dynamic environments where they may need to access the application on tablets of varying screen sizes, thereby maintaining usability and functionality regardless of the device used.

Director Mode

The final Director Mode UI, (see Figure 3.10), offers an organized and visually engaging layout tailored to meet the operational needs of a director overseeing multiple XR devices. Key improvements include a comprehensive device status display, where each XR device is accompanied by a clear status indicator, such as “experience running,” and a color-coded background that visually distinguishes active from inactive states. This design allows the director to quickly assess the operational status of each device at a glance. Additionally, the interface now supported both individual device control and synchronized commands. A “Sync All” toggle button provides an efficient mechanism to direct the experience and allow to use a different folder for directing practices.

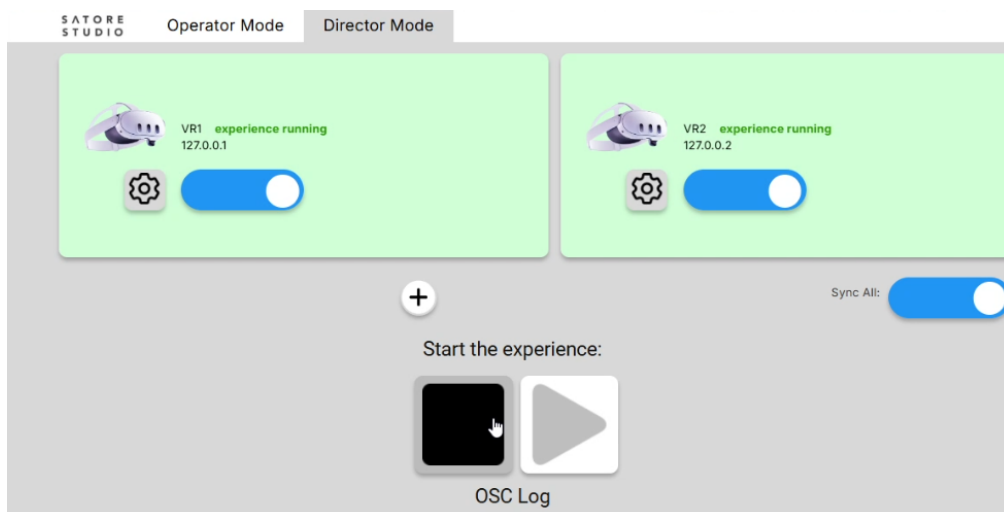


Figure 3.10: Final functional application in the director mode.

The UI also incorporates a well-defined visual and functional hierarchy, where primary actions—such as starting the experience—are prominently displayed through larger, contrasting buttons. This layout draws the director’s attention to essential controls while streamlining navigation. These design enhancements collectively made Director Mode more intuitive and responsive, align-

ing its functionality closely with the needs of directors managing complex XR setups.

Before reaching the final version of the UI of the Operator Mode, multiple tests with Users were made, and are reported in Chapter 4.

3.6 Future Work

One area for potential future development is the need for a more intuitive, streamlined, and specialized interface for operators and directors managing the XR experience. Specifically, there is an evident requirement to eliminate the need for operators to familiarize themselves with the Meta Quest devices' operating system (OS) UI, before launching the Unreal Engine application and preparing the XR devices for operation, as seen in Figure 3.11 .



Figure 3.11: The Meta Quest operating system interface that is used before launching the XR experience.

The current setup necessitates a learning curve for operators to navigate through the Meta Quest OS interface before the XR devices can be used for the intended XR experience, which could introduce unnecessary delays and operational inefficiencies.⁷

To streamline this process, future work could focus on developing an Android-based application capable of transforming the XR device into a dedicated kiosk mode. In this mode, the XR device would automatically launch the intended Unreal Engine experience, eliminating the need for operators to interact with the device's standard UI or navigate its OS settings. This would significantly reduce the complexity of the setup process and minimize delays, enabling operators to concentrate more on managing the immersive experience itself.

The introduction of an Android-based management application would streamline the operational workflow by providing a simple, user-friendly interface that allows operators to configure and control the XR devices remotely. Such an interface could also include features like remotely starting the XR experience, pausing or restarting the session, and making other real-time adjustments to the experience, without requiring direct interaction with the Meta Quest UI. This would

⁷<https://www.youtube.com/watch?v=MnCOWfFr5Y>

enhance the efficiency of the preparation process and reduce the cognitive load on the operator, ensuring that they can focus on managing the experience rather than dealing with the complexities of the device's native operating system.

Additionally, enabling the kiosk mode through an Android application would not only simplify the operator's tasks but also increase the overall reliability and security of the XR system. By restricting the device's capabilities to only the essential XR experience, the risk of accidental misconfiguration or interference with other system settings would be minimized. This would ultimately contribute to a more controlled and stable environment for the participants, ensuring that the focus remains on the immersive content rather than on the technicalities of device management.

In conclusion, while the current solution addresses the core needs of the project, the development of an Android-based application capable of transforming the XR device into kiosk mode would significantly improve the operator's workflow. This enhancement would lead to a more efficient, user-friendly, and secure experience, reducing the preparation time and allowing for smoother execution of the XR experience. This solution aligns with the broader objectives of the project to provide a more intuitive, seamless, and operationally efficient system for managing immersive XR environments.

Chapter 4

User Experience

In the development of modern applications, the roles of User Experience (UX) and User Interface (UI) design are particularly significant, especially when users with a wide range of technical proficiencies interact with complex systems. Effective UX design allows users to navigate the application intuitively, while a well-planned UI design ensures accessibility and operational efficiency. These principles are especially relevant in our Use-Case, where non-technical staff may need to operate applications without specialized training. Emphasizing a user-centered design approach not only improves usability but also minimizes the need for training, ultimately supporting more streamlined operations and reducing the likelihood of user error.

This section details the iterative testing process conducted during the development of the Operator Mode, showing how user feedback informed each design iteration. Testing was critical to ensuring that the application could meet users' practical needs, and the continuous refinement process allowed for the adjustment of design elements based on observed user behavior patterns.

Director Mode involved a distinct set of UI/UX considerations, as it was designed to meet specific requirements set by the company. To validate its effectiveness, multiple rounds of user testing were conducted in collaboration with the company's director and key stakeholders. These tests evaluated the application's functionality in managing and overseeing XR sessions and were essential in ensuring that Director Mode met all operational requirements for directing XR experiences effectively.

The testing procedure for **Operator Mode** is explained in Section 4.1, and the User Experience analysis was structured in four stages:

Formative Tests(Section 4.2): Conducted with six participants, this initial testing phase aimed to identify usability challenges, errors, and areas for improvement in the design. Feedback from these sessions was critical in highlighting aspects of the interface that were unclear or unintuitive for users. The observations collected here provided a foundation for subsequent design improvements.

Pilot Tests (Section 4.3): Following the formative tests, a pilot testing phase was conducted with two additional participants to verify that all issues raised in the formative phase were addressed. This stage ensured that the adjustments made had effectively resolved the identified problems and confirmed the interface's usability enhancements.

Final User Tests (Section 4.4): The final user test gathered data from 26 participants to ensure the application was well designed. Before delving into the data analysis of these results, we highlight key observations made during the testing phase. These insights, derived from user interactions and test sessions, providing valuable context for understanding the upcoming analysis and help identify potential limitations encountered during the evaluation.

Data Analysis (Section 4.5): The data analysis section will examine key usability metrics based on feedback from the final user test. It focuses on evaluating user satisfaction, task completion times, and frustration levels, with particular attention to differences between younger and older users. Additionally, comparisons between the formative and final test phases will be explored, using UMUX (Usability Metric for User Experience) data to assess overall usability improvements. The analysis aims to confirm the effectiveness of the iterative testing process and the modifications made to the interface.

This iterative approach to user testing provided an evidence-based process to refine and optimize the operator mode. User feedback informed each development phase, allowing for design modifications that enhanced usability and ensured alignment with both the operator's practical needs and the director's specialized requirements. The structured, multi-phase testing strategy helped ensure that the application met the expectations and functional needs of a diverse user base while supporting the seamless management of XR sessions.

4.1 Procedure

To ensure consistency and clarity for the users during the User Testing process, we developed a structured Procedure Form. This form outlined all essential details and steps required for conducting the tasks and was used across the Formative Tests, the Pilot Tests, and the Final User Tests.

The Procedure Form began by giving context to the user, which provided operators with a clear understanding of the application's intended usage. This included an explanation of the test environment and objectives, situating the operators within a physical space to be determined, where they were responsible for onboarding two participants, and managing their experience as application operators. The configuration for the procedure can be seen in Figure 4.1.

A PC displayed the Procedure Form for reference, while a tablet was used by the operator to interact with the application. A mobile phone recorded the time taken for each task, and an XR headset was used to simulate the onboarding process.

Following the context description, a detailed script was provided to the operators. This script outlined four specific tasks to be completed during the testing phase, as summarized below:

1. Power on the executable of the local server on the computer and initialize the application on the tablet.
2. Configure the XR IP addresses within the tablet interface.

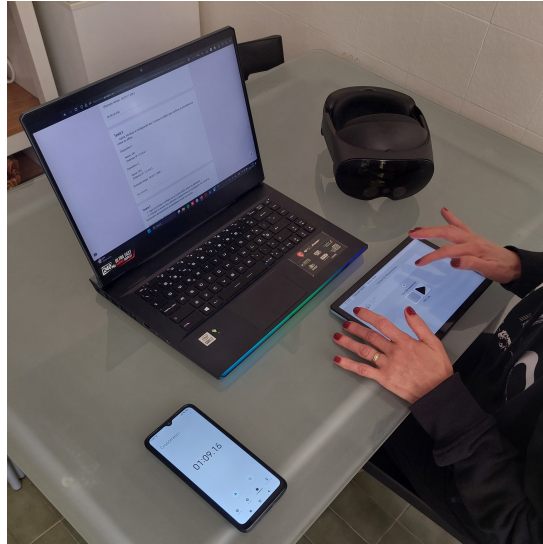


Figure 4.1: A participant engaging with the devices and with the scripted tasks outlined in the procedure form.

3. Power on the XR headset (simulating two headsets with a single device) and verify their status via the tablet.
4. Initiate the XR experience for two hypothetical users.

All the finer details can be found in our Procedure Document, in Appendix B. Additionally, The Procedure Form incorporated a final questionnaire designed to gather both qualitative and quantitative feedback from participants. This included the standard UMUX questions, which provided a structured evaluation of the application's usability. In the development of the Procedure Form, the choice to use the full UMUX questionnaire rather than UMUX-lite was driven by the need for a more detailed and structured evaluation of the application's usability. While UMUX-lite offers a simplified version with fewer questions, the full UMUX provides a more comprehensive set of questions that cover both usability and satisfaction more thoroughly. This richer data set is particularly useful when aiming to assess both the functional and emotional aspects of user experience in greater detail.

The questionnaire also collected demographic and usage-related metrics, such as the participant's age and their prior experience or duration of use with similar devices. These metrics are critical for understanding the relationship between user characteristics and their interactions with the application, offering valuable insights into how diverse user profiles influence usability and overall satisfaction.

Limitations

One limitation of our study was the unavailability of conducting the procedure with two XR devices, which prevented a complete simulation of the onboarding process for multiple headsets.

Additionally, the user had to imagine two people available to perform the testing, further restricting the ability to fully replicate the multi-user onboarding scenario.

Despite this constraint, the testing scenario involving a single XR device provided valuable insights. Users were able to interact with the application and perform tasks related to setting up the first device. This interaction likely provided sufficient guidance for users to extrapolate the same steps for configuring a second headset, as the procedure would be identical. Consequently, while the absence of a second XR device and user limited the scope of the test, we think that it didn't critically impact the validity of the findings regarding the onboarding process.

Nonetheless, it is important to acknowledge this limitation for our future evaluations. Testing with multiple XR devices and participants would provide a more comprehensive assessment, ensuring the application's performance and usability are robust in real-world multi-user scenarios. This adjustment would also allow for the identification of potential challenges unique to simultaneous multi-device setups.

4.2 Formative Tests Observations

For the initial testing phase of the application, six users participated: three with expertise in software development and three with limited experience in technology. This formative test aimed to gather feedback on the usability of the operator mode and identify areas for improvement.

The following observations and comments were recorded during the tests, based on the hand-out script and the participants' application use. These notes and user verbal feedback from the participants informed subsequent improvements and feature refinements in the application. Here, we present the main results captured from the 6 users:

User: “Confusing script. Do I need to use the application on the PC or the tablet?”

Action Taken: Improved script text.

The user expressed confusion regarding the setup process, questioning whether the application should be operated from the PC or the tablet. This feedback highlighted a lack of clarity in the instructional content, which led to misunderstandings. The script was revised to ensure that it clearly indicated which platform (PC or tablet) should be used for specific tasks within the XR experience.

Note: User expressed facial confusion by multiple terminals opening.

Action Taken: Transformed multiple terminals into one.

The user appeared overwhelmed by the simultaneous opening of multiple terminals, as shown in Figure 4.2. This visual complexity created a sense of confusion, which was not explicitly communicated but could be inferred from the user's facial expressions. To address this, all terminal interfaces were consolidated into a single one unified terminal, as shown in Figure 4.3, that displays all necessary information, simplifying the user experience and improving usability. By

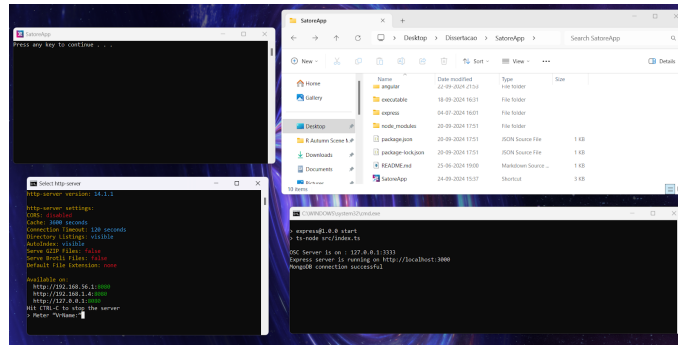


Figure 4.2: Multiple terminals opening when starting Local Server.

reducing the number of terminals, the application was made more intuitive and user-friendly. This unified terminal is used to log communication details between the application and the connected XR devices.

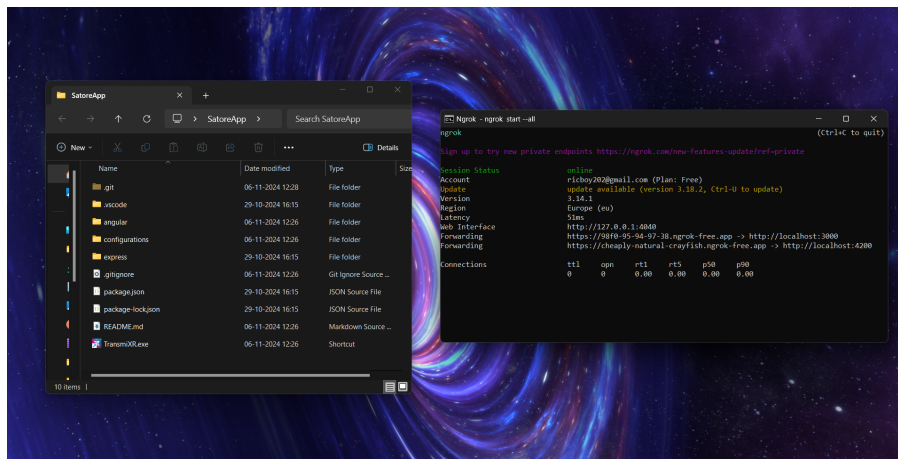


Figure 4.3: One terminal that holds all the information.

User: “Oh, what just happened? And why did MongoDB open?”

Action Taken: Removed MongoDB application from opening.

The user expressed confusion regarding the unexpected opening of MongoDB application during the experience, as seen in Figure 4.4 which had previously been an issue for other users (in the “Multiple Terminals” issue above). This was identified as a system inconsistency and was rectified, ensuring that MongoDB no longer opens unnecessarily during the preparation of the XR experience. This improvement prevented distractions and kept the user focused on the task at hand.

User: “Where do I need to go to?”

Action Taken: Application automatically opens operator mode tab.

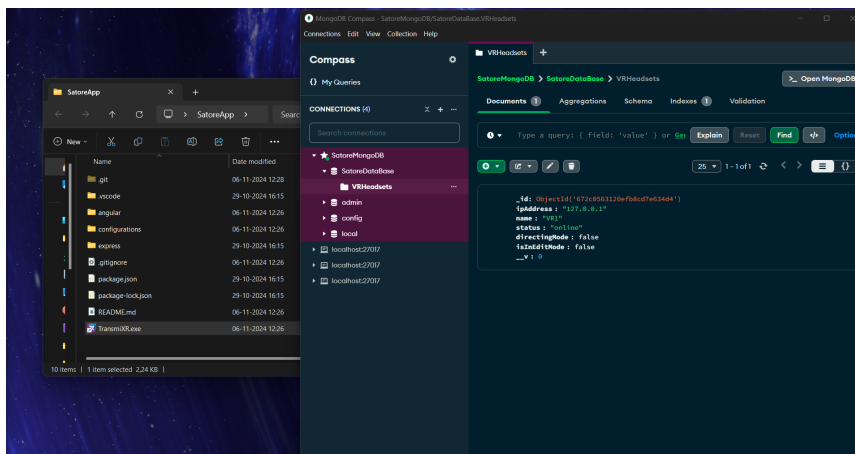


Figure 4.4: MongoDB application opening when executing the laptop application.

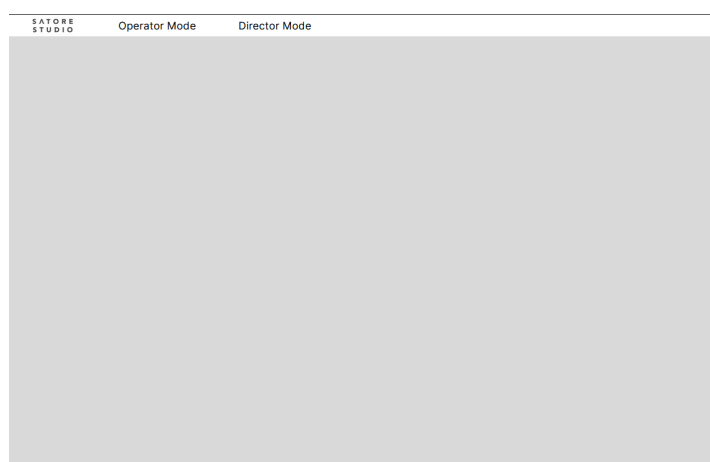


Figure 4.5: Application didn't automatically open in operator mode.

The user found the interface less intuitive than expected, primarily due to the unclear navigation and having the option to choose a tab, as seen in Figure 4.5. Additionally, the feedback highlighted the need for the application to default to operator mode upon startup to streamline the user's workflow. Based on this suggestion, the application was reconfigured to open directly in operator mode, minimizing unnecessary navigation and further improving usability.

User: “Has the experience started?”

Action Taken: Applied green background on each online XR headset.

Initially, the application lacked a distinct visual cue to indicate when the experience had started, as seen in Figure 4.6. To address this, a green background was applied to the list items corresponding to each XR headset, as seen in Figure 4.7, providing a clear, visual indication that the experience was active. This color-coded indicator, specifically the green background for each headset's list item, reassured the user that the experience was successfully initiated. The addition of this visual cue enhanced the user experience, making it more intuitive—particularly for novice

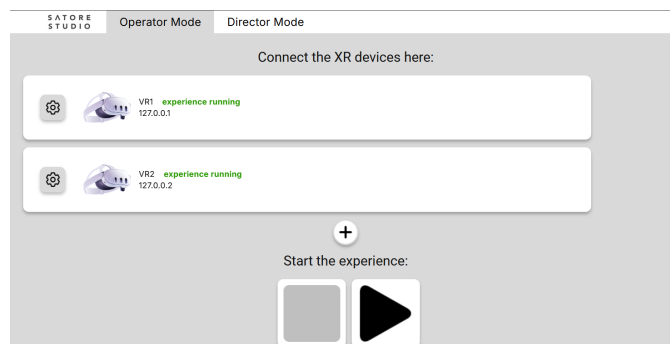


Figure 4.6: Initial interface where the experience state was unclear, as no visual indicator confirmed its status.

users who may not recognize subtler feedback mechanisms.

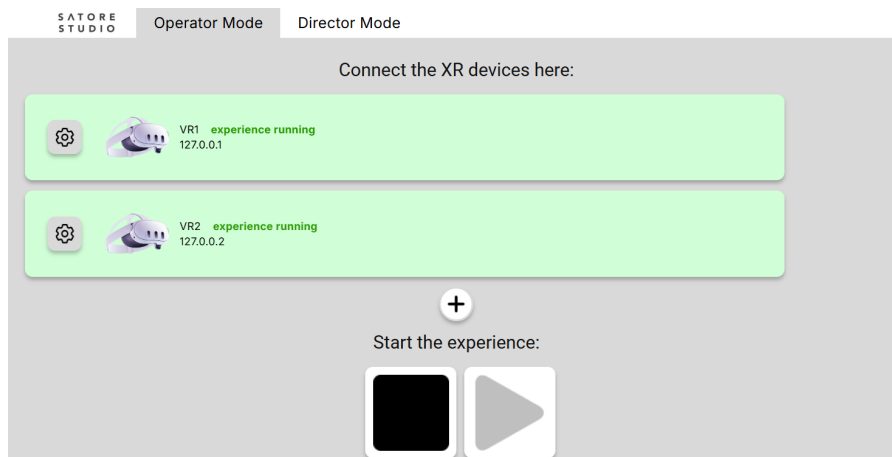


Figure 4.7: Visual indicator showing the green background applied to list items, confirming that the XR experience is activer.

User: "What do you mean by participants? Am I the participant?"

Action Taken: Updated terminology in script.

To address this feedback, the term "participants" in the Procedure form was replaced with "visitors." This adjustment was made to better align with the context of the performance art use case, where the individuals interacting with the XR devices are more accurately referred to as visitors rather than participants. This change in terminology enhances clarity by accurately reflecting the user's role in the experience, providing a more intuitive and consistent language throughout the application. Such updates improve the overall user understanding, particularly for novice users who may find the term "visitors" more relatable because of the fact they were operators in the physical location and were receiving visitors.

4.3 Pilot Tests Observations

The pilot testing phase, while intended primarily to confirm that issues identified during the formative test phase, revealed additional design adjustments required to enhance user interaction and reduce confusion. Key improvements were made to better align the user interface (UI) with users' expectations and ensure smooth operation of the application, especially for non-technical users.

The specific changes implemented following pilot testing feedback included:

Removal of Port Address Field

Initially, the interface included a field for port addresses, which created additional complexity for users. After reviewing the application's communication requirements, it was determined that all devices could reliably use a unified, standard IP port for data transfer, as seen in Figure 4.8. Removing the port field simplified the interface, decreased potential user confusion, and streamlined the form completion process, ensuring a more efficient data entry experience.

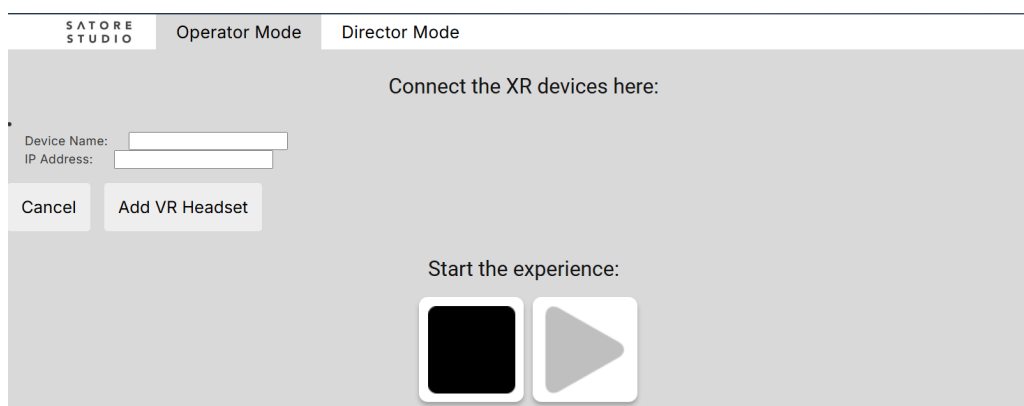
The image shows a user interface for adding XR device information. At the top, there is a header with the logo 'S A T O R E S T U D I O' on the left and two mode tabs: 'Operator Mode' and 'Director Mode'. Below the header, the main content area has a light gray background. At the top of this area, it says 'Connect the XR devices here:'. There are two input fields: 'Device Name:' and 'IP Address:'. Below these fields are two buttons: 'Cancel' and 'Add VR Headset'. At the bottom of the form, it says 'Start the experience:' followed by two buttons: a black square button and a white play button.

Figure 4.8: The updated form to add new XR device information.

Adjustment to Form Behavior on Submission

In the initial design, after clicking the "Add" button and submitting a form, the form remained open, as seen above in Figure 4.8. causing pilot users to assume further interaction with it was necessary. To address this, we adjusted the design so the form automatically closes upon submission, prompting the "Add" button to appear to the user again, aligning with user expectations for feedback on task completion. This modification is consistent with established usability principles, where predictable responses enhance user confidence and reduce error rates by clearly signaling task completion.

These modifications reflect insights from Human-Computer Interface studies, which suggest that users tend to follow predictable patterns and expect consistency in action-result sequences. By refining the design to accommodate these patterns, the application offers a more intuitive and accessible experience. This approach supports users in efficiently navigating tasks without unnecessary complexity. Feedback from pilot testing emphasized the importance of aligning design

with user expectations, particularly regarding action-based feedback and information hierarchy. These adjustments ultimately enhanced the UI, meeting functional requirements while improving the overall user experience by fostering a clear, structured workflow.

4.4 Final User Test Observations

Before delving into the analysis of the results, it is important to outline certain limitations encountered during the execution of the Final User Test. These limitations are believed to have significantly influenced the data analyzed in Chapter 5 and offer valuable context for understanding the outcomes.

The identified issues were primarily associated with Task 2, a critical component of the user workflow and a key factor in the time analysis graphs.

The absence of clear guidance for initiating Task 2

A key issue identified during user testing was the difficulty some participants faced in initiating the second task: configuring the XR headset IP addresses in the system. Some users mistakenly believed that clicking the start button was necessary to begin the configuration process. This lack of clear guidance disrupted the workflow and led to delays as users were unsure of how to proceed.

To address this, a potential solution is to implement a permission-based workflow that restricts access to the toggle button features until all required setup steps are completed. For instance, the experience activation toggle buttons could remain disabled until the system detects that the XR headset configuration has been successfully entered. This adjustment would ensure users are guided through the application in a sequential and logical manner, preventing errors or omissions in critical setup tasks.

This approach aligns with principles of guided interaction and progressive disclosure in user experience design, which advocate for limiting user access to only the functions relevant to their current step in the workflow. By doing so, the application can reduce cognitive load, minimize errors, and enhance user efficiency. Additionally, incorporating visual cues, such as a progress indicator or instructional prompts, could further reinforce this structured workflow, making the application more intuitive and user-friendly.

A significant language barrier for users unfamiliar with English

A notable limitation identified during the user testing was the language barrier faced by users who were not proficient in English. This issue was particularly relevant during Task 2, which required users to engage with the application and begin interacting with the system. The default language of the interface, being in English, posed significant challenges for non-English-speaking participants, who struggled to interpret system prompts and instructions effectively. As a result, this language barrier hindered users from successfully completing the task and reduced overall system usability.

All of the users in the User Experience test were Portuguese, and the language barrier was especially pronounced because some were not familiar with the English language, and also not familiar with tech-related text. This limitation not only led to delays in completing tasks but also resulted in errors and frustration. It highlights the importance of designing systems that consider linguistic diversity to ensure that the targeted users can interact with the interface without unnecessary obstacles.

To address this, translating the interface into Portuguese, would make the application more accessible to users with limited proficiency in English. This would help reduce cognitive load and enhance user experience by allowing participants to complete tasks more easily and accurately.

Issues with overall button size for older group age users

Another limitation identified during testing related to the physical design of interface elements, particularly the size of the form input, text, and keyboard size. Some users, particularly individuals over 35, showed difficulty interacting, and expressed their frustrations with the buttons due to their small size as seen by Figure 4.9.

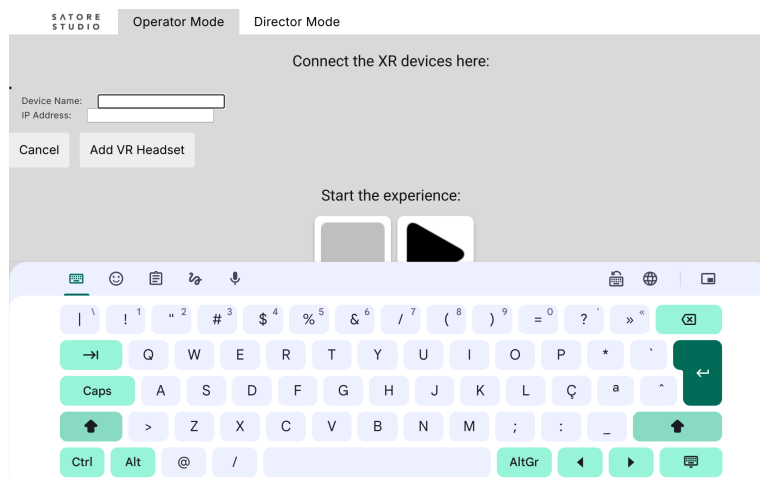


Figure 4.9: Tablet keyboard configuration used for the forms.

This issue highlights the need for accessibility considerations in user interface design, as older adults may experience reduced dexterity or visual impairments, making small buttons harder to interact with. To address this challenge, the design could be modified by increasing the size of the interactive elements, like the input boxes, ensuring they are both visually prominent and easy to engage with.

Another solution that could have improved the easiness of use of the application would be to use another simplified and bigger input number that was found on the keyboard settings during the User Testing phases, as seen by Figure 4.10.

Difficulty in learning how to input numbers with the tablet keyboard

Another issue identified during testing was users struggling to locate and input numbers using the tablet keyboard layout, the one shown in Figure 4.9. This challenge often resulted in pro-

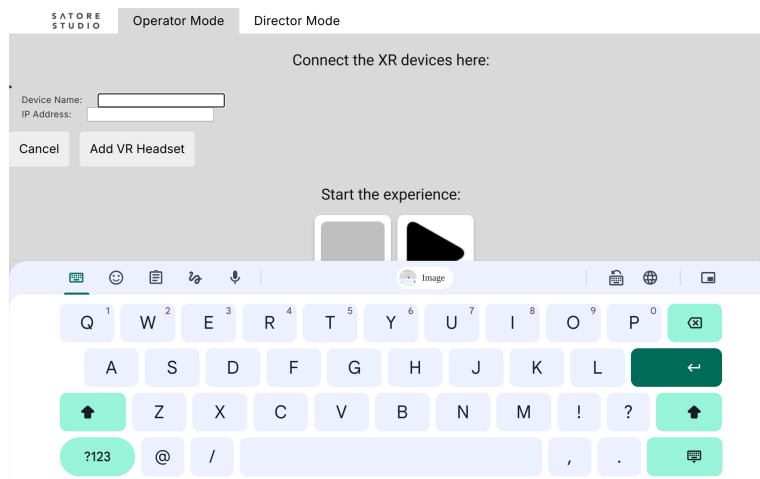


Figure 4.10: The other tablet keyboard configuration that could have been used.

longed interaction times and increased frustration, particularly for users unfamiliar with switching between keyboard modes to access numerical input.

A potential solution to address this issue is to implement input masking for numerical fields. By automatically triggering a numeric-only keyboard, similar to those used in banking applications, users would be presented with a simplified keypad containing only numeric characters. This adjustment would streamline the input process, minimize errors, and reduce the learning curve for users.

This design change leverages the principle of context-sensitive interfaces, where the system adapts input options to the task at hand, thereby enhancing user efficiency and satisfaction.

By addressing these concerns, the future overall user experience can be significantly improved, especially for users with specific needs, such as older adults or non-native language speakers. These changes would help create a more inclusive interface, ensuring the application is accessible and functional for a broader range of users, and enhancing usability across diverse user groups.

4.5 Data Analysis

The final phase of this project involved validating the findings from the final user test and testing several hypotheses related to the usability and user satisfaction of the application. The analysis aimed to assess critical user experience metrics, such as satisfaction, frustration, ease of use, error correction, and task completion times, based on the feedback gathered during the final user testing phase. This section highlights the primary actions taken to explore these aspects: overall usability evaluation and user feedback, comparison of task completion times between users aged below and above 35, and comparison of UMUX data between formative and final user tests.

4.5.1 Data Demographics

The demographic analysis of participants is essential for contextualizing the results of the user testing phase. This section provides an overview of the characteristics of the user group, focusing on gender, age, and technology usage patterns. These demographic factors help establish a baseline for interpreting variations in task performance, system usability, and overall user experience.

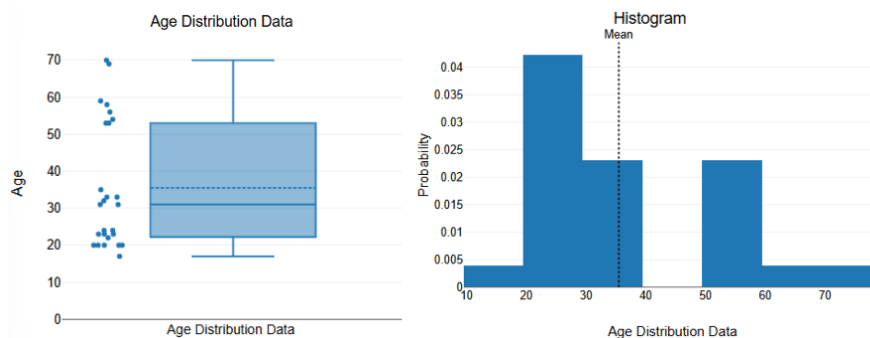


Figure 4.11: Age distribution and corresponding histogram.

The mean age of participants was calculated at 35.5 years, with a standard deviation of 17.05 years, illustrating a wide age range within the participant group, as seen by Figure 4.11. Statistical analysis revealed that 18 (68%) of the 26 participants, fell within one standard deviation of the mean, with ages ranging from 18.45 to 52.55 years, while 25 (95%) participants, fell within two standard deviations, spanning from 1.4 to 69.6 years. A Shapiro-Wilk test for normality yielded a p-value of 0.001, indicating that the age data was not normally distributed.

Of the 26 participants involved in the user testing, 14 (53.8%) identified as female, and 12 (46.2%) identified as male. This nearly balanced gender distribution offers a representative perspective on system usability across genders.

In terms of technology usage patterns, 20 (76.9%) of the 26 participants, reported no daily use of tablets, highlighting limited familiarity with the primary interface device used in testing. Despite this, all participants reported daily smartphone usage, with 11 (42.3%) using their smartphones for over three hours per day. Additionally, 24 (92.3%) of participants used a computer at least once daily, reflecting widespread familiarity with general digital interfaces.

A more detailed breakdown of all of the demographic data is provided in UserTesting Data our Appendix A.

4.5.2 Overall Usability Evaluation and User Feedback

The overall usability of the application was assessed through several measures, using a Likert scale to quantify user feedback on various aspects of the user experience. One of the most critical aspects of this evaluation was user overall satisfaction with the application.

The data indicates that 18 (69.2%) of the 26 users, rated their satisfaction at the highest level 7, while 8 users (30.8%) rated it at level 6, as seen in Figure 4.12 above. This suggests a pre-

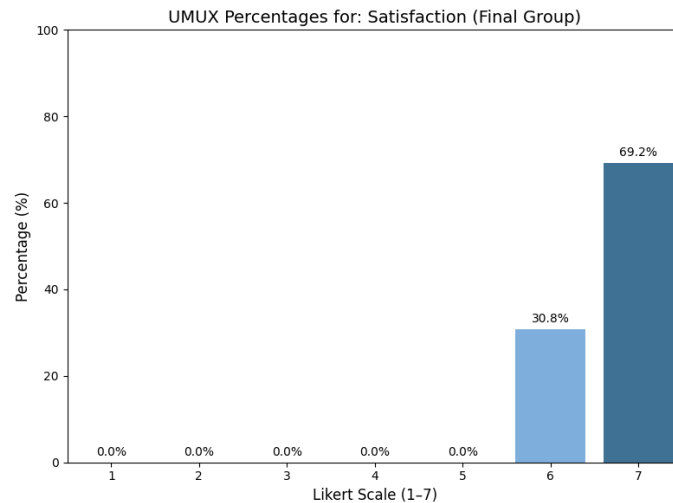


Figure 4.12: UMUX 1: Satisfaction – Final group Likert scale evaluation.

dominantly positive reception of the application, with a high proportion of users expressing a high degree of satisfaction.

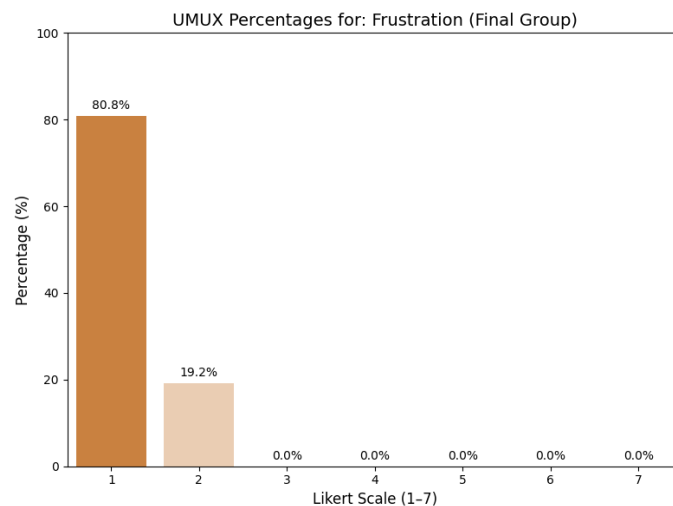


Figure 4.13: UMUX 2: Frustration – Final group Likert scale evaluation.

In terms of frustration, the results show that 21 (80.8%) of the 26 users rated their feelings of frustration at the minimum level 1, as seen in Figure 4.13, and 5 (19.2%) rated it at level 2. This indicates that the majority of users experienced minimal frustration while interacting with the application. The relatively small proportion of users who reported higher frustration levels is consistent with the overall positive satisfaction ratings, further supporting the conclusion that the user experience was largely smooth and effective.

The perceived ease of use was another important metric evaluated during the final user testing. According to the data, 19 (73.1%) of the 26 users rated the application's ease of use at the highest level 7, as seen above in 4.14, while 7 (26.9%) rated it at levels 5 and 6. This suggests that the

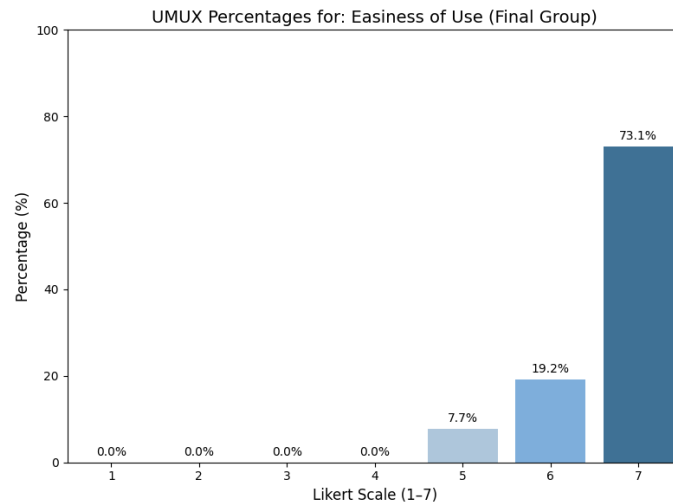


Figure 4.14: UMUX 3: Ease of Use – Final group Likert scale evaluation.

application was generally considered easy to use by the majority of users. However, the ratings of 5 and 6 indicate that a portion of users found certain aspects of the application moderately challenging, probably do to the aspects for improvement that were identified before in Section 4.4.

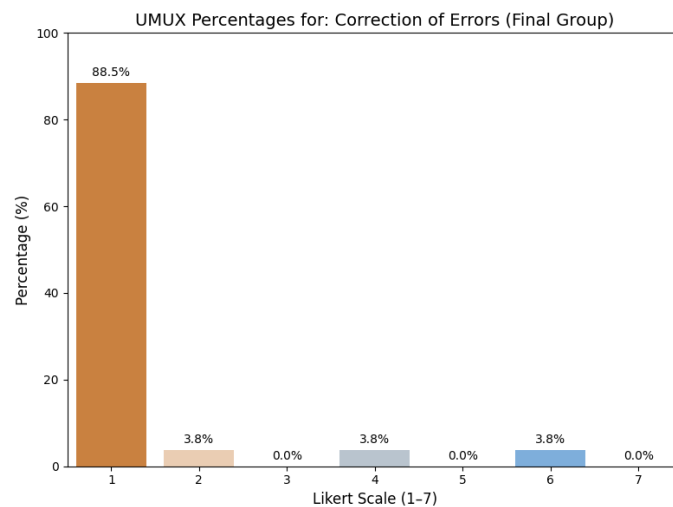


Figure 4.15: UMUX 4: Correction of Errors – Final group Likert scale evaluation.

Finally, error correction was assessed. The results show that 23 (88.5%) of 26 users rated their experience with correcting errors at the minimum level 1, as seen above in Figure 4.15, while 3 (11%) outliers rated it at higher levels, such as 2, 4, and 6.

Curiously, this graph contains some outliers, specifically ratings of 4 and 6 on the Likert scale. We hypothesize that the rating of 6 was likely a mistake made by the user, as their responses to the other UMUX items were consistently positive, indicating general satisfaction with the application.

The rating of 4, on the other hand, may have been influenced by one of the limitations discussed in Section 4.4. One user mentioned that the challenge of learning to effectively interact

with the tablet's keyboard, making multiple errors, was a significant factor in their decision to rate the application lower.

Overall, these findings suggest that the application is generally well-received in terms of usability. However, the presence of outliers in the frustration and error correction metrics indicates areas that still require targeted improvements to further enhance the user experience. Additionally, considering that 20 (76.9%) of the users analyzed do not use a tablet daily, the positive reception is particularly noteworthy, as it shows that the application is accessible and easy to use even for individuals unfamiliar with tablet interfaces. Moving forward, the next step in the development process will focus on addressing these specific pain points to ensure a more refined and universally positive user experience.

4.5.3 Comparison of Task Completion Times Between Users Aged Below and Above 35

One of the objectives of this analysis was to determine whether age influences task completion times and to verify whether the application is equally usable for individuals across different age groups. To address this, we divided the participants of the final user test into two groups: those aged below 35 and those aged over 35. The total number of participants was 26, with 69.2% under the age of 35 ($n = 18$) and 30.8% aged 35 or older ($n = 8$).

We then analyzed the time taken by each group to complete various tasks. For each task, we first examined the distribution of the data using histograms to assess whether the data met the assumptions for parametric testing. Additionally, we performed the Shapiro-Wilk test for normality on each dataset to evaluate whether the distribution significantly deviated from normality. Based on the results of this test, we determined the appropriate statistical method for analyzing the data.

The following sections detail the results for each task, starting with Task 1.

Task 1: "Power on the executable of the local server on the computer and initialize the application on the tablet."

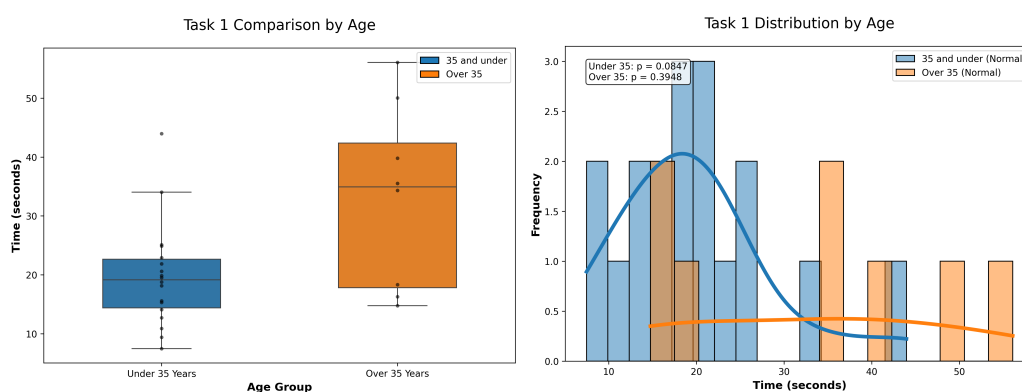


Figure 4.16: Task 1: Comparison of time taken by users below and above 35 years, with corresponding data distribution.

For Task 1, the histogram shown in Figure 4.16, indicates that the data for both groups was independent and met the assumptions for parametric testing. Therefore, we performed an independent t-test to compare the time taken by each group. The p-value for this task was 0.0540, which is close to the 0.05 threshold, suggesting that there is a near-significant difference between the two groups at the 95% confidence level. This result may indicate that older users experienced some delay due to their relative unfamiliarity with the two distinct devices (computer and tablet) required for the task.

Task 2: "Configure the XR IP addresses within the tablet interface."

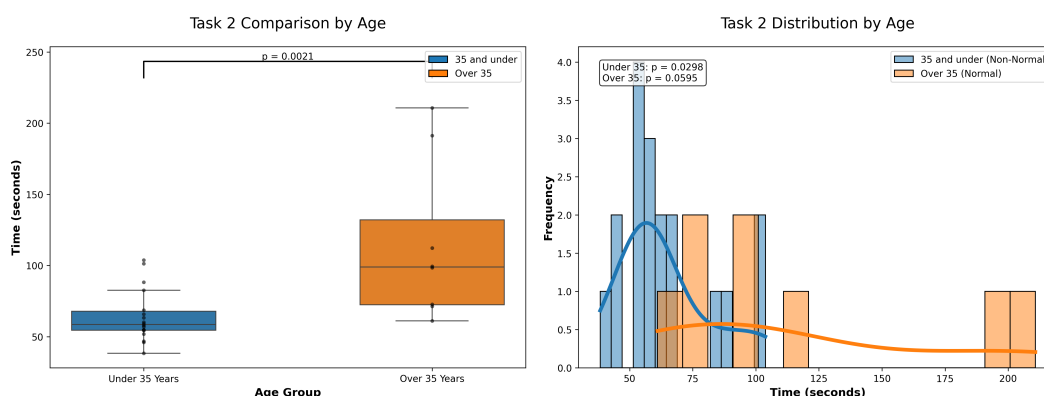


Figure 4.17: Task 2: Comparison of time taken by users below and above 35 years, with corresponding data distribution.

For Task 2, the histogram shown in Figure 4.17, suggests that the data was not normal, prompting us to perform a Mann-Whitney U test for comparison. The results revealed a significant difference between the two groups ($p < 0.05$). Specifically, users aged over 35 took longer to complete this task. This can be attributed to the several factors identified in Section 4.4, including difficulty with the tablet's keyboard input and the small size of the UI elements. Additionally, the lack of clear guidance and potential language barriers, also referred to in Section 4.4 may have contributed to the challenges faced by older users, resulting in longer task completion times.

Task 3: "Power on the XR headset (simulating two headsets with a single device) and verify their status via the tablet."

For Task 3, the histogram in 4.18 demonstrates that the data for both groups was normal. Accordingly, we performed an independent t-test. The results showed no significant difference between the two groups, indicating that both younger and older users were equally proficient at turning on the XR headset and verifying its status on the tablet. This suggests that both age groups were comfortable with the task, which primarily involved simple interactions with the tablet interface.

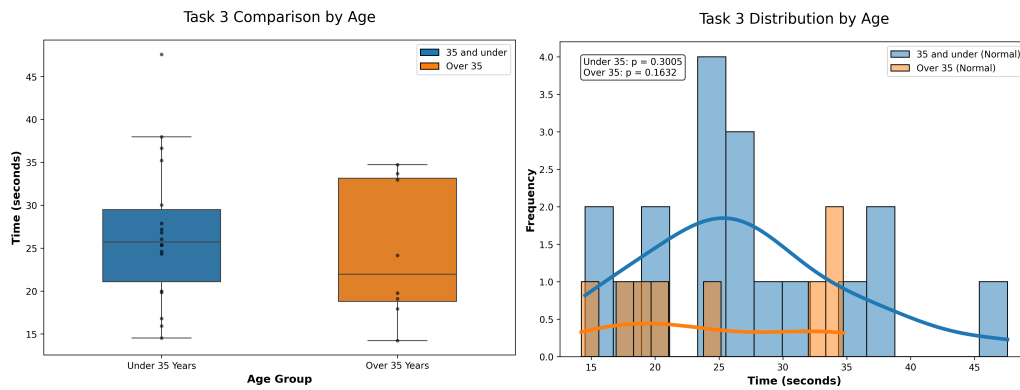


Figure 4.18: Task 3: Comparison of time taken by users below and above 35 years, with corresponding data distribution.

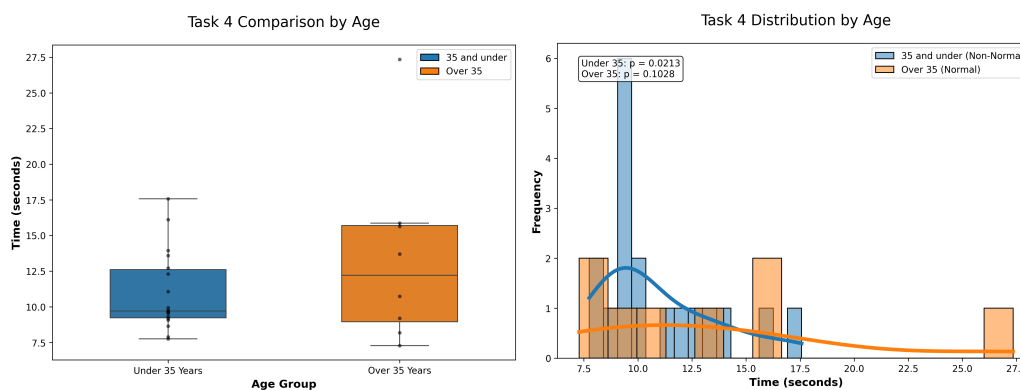


Figure 4.19: Task 4: Comparison of time taken by users below and above 35 years, with corresponding data distribution.

Task 4: "Initiate the XR experience for two hypothetical users."

For Task 4, the histogram in 4.19 indicates that the data was not normal. Therefore, a Mann-Whitney U test was performed. The results showed no statistical difference between the two groups, suggesting that both younger and older users were equally capable of initiating the XR experience for the two hypothetical users. This task involved interacting with the experience toggle buttons and did not present particular challenges for either group, as both were able to complete it with similar efficiency.

4.5.4 Comparison of UMUX Data Between Formative and Final User Tests

This section provides a comparative analysis of the UMUX questionnaire responses obtained during the Formative and Final User Tests. The primary objective of this analysis was to evaluate how modifications to the application influenced user experience across key usability metrics.

The study utilized ordinal data, necessitating the use of non-parametric statistical methods for analysis. Specifically, Mann-Whitney U tests were conducted for each comparison, as the data distributions were non-parametric and independent. It is essential to note the limitations of

the analysis, including the small sample size of the Formative group ($n = 6$), which constrains the statistical power and generalizability of the findings. Additionally, as the participants in the Formative and Final tests were distinct groups, the results should be interpreted as indicative rather than definitive. The primary focus is on identifying relative trends in usability improvements between the two groups.

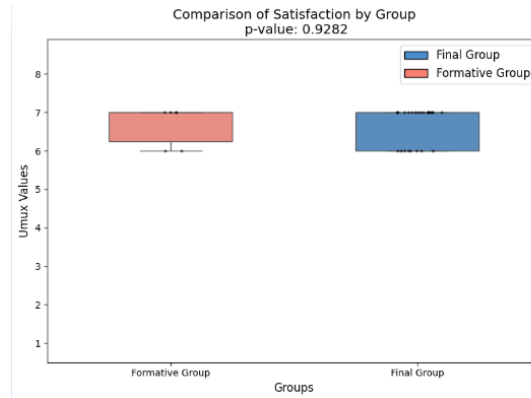


Figure 4.20: Comparison of UMUX1: Satisfaction Values for the formative and final group.

Satisfaction, as shown in Figure 4.20, was analyzed using a Mann-Whitney U test. The p-value of 0.9282 indicates no statistically significant difference between the Formative and Final groups. The mean satisfaction score increased 0.38% from 6.67 to 6.69, reflecting minimal impact from the application updates, likely due to the already high baseline satisfaction levels observed during the Formative tests.

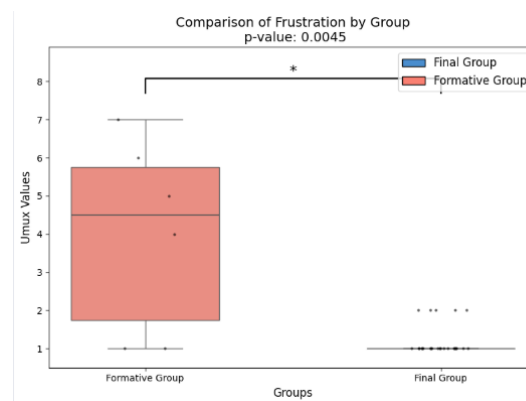


Figure 4.21: Comparison of UMUX2: Frustration Values for the formative and final group.

Frustration levels, depicted in Figure 4.21, showed a significant improvement in the Final group, with a mean score reduction of 235% from 4.00 to 1.19. Consequently, a Mann-Whitney U test was employed to analyze the data. The analysis produced a p-value of 0.0045, indicating a statistically significant reduction in frustration levels between the Formative and Final groups.

Ease of use, as shown in Figure 4.22, exhibited a statistically increase in the mean of 23% from 5.11 to 6.65. The Mann-Whitney U test returned a p-value of 0.0043, highlighting significant

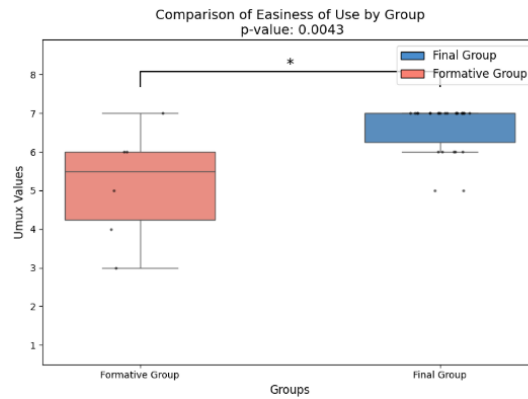


Figure 4.22: Comparison of UMUX3: Ease of Use Values for the formative and final group.

improvements in this usability metric. These results suggest that the adjustments implemented in the application contributed to a more intuitive and accessible user interface, particularly for participants with limited experience using tablets, who represented 76.9% of the Final group.

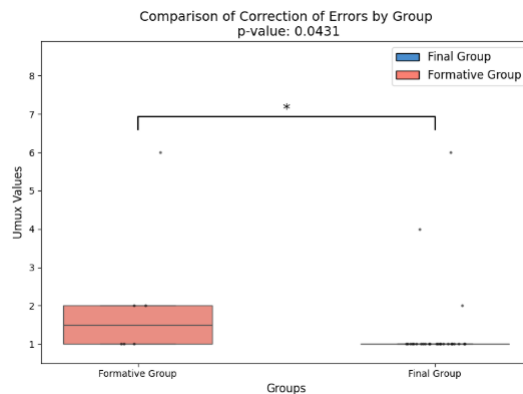


Figure 4.23: Comparison of UMUX4: Correction of Errors Values for the formative and final group.

Error correction scores, presented in Figure 4.23, improved significantly, with the mean score decreasing 61%, from 2.17 to 1.35. The Mann-Whitney U test yielded a p-value of 0.0431, confirming the statistical significance of this improvement. These findings suggest that refinements to the application reduced the perceived effort required to identify and rectify errors, thereby enhancing the overall usability of the system.

Outliers observed in the box plot for the final group were previously discussed in subsection 4.5.2, where their potential impact on the results and interpretations was addressed.

The comparative analysis of UMUX data reveals that iterative modifications to the application produced measurable improvements in usability metrics, particularly in reducing frustration, enhancing ease of use, and facilitating error correction. Satisfaction levels, while stable, remained consistently high across both groups. The results underscore the importance of an iterative design process in addressing specific usability challenges. However, the small sample size and independent nature of the participant groups necessitate cautious interpretation of these findings. Future

studies should aim to replicate these results with larger and more diverse participant samples to validate the observed trends.

Limitations

The study faced several limitations that may have influenced its results and interpretations. One significant issue was participant selection bias, as the user tests were conducted primarily with friends and acquaintances of the researcher. This selection method potentially introduced bias, as participants with personal connections to the researcher may not have provided entirely objective feedback. Their responses might have been influenced by a desire to offer more favorable ratings to support the researcher or avoid expressing criticism.

Another limitation was the influence of the researcher's presence during the testing sessions. The researcher being in the same room may have unintentionally created a sense of pressure among participants, particularly since they were aware of their personal relationship with the researcher. This dynamic could have led participants to provide more positive evaluations or refrain from offering negative feedback, ultimately impacting the reliability of the collected data.

To improve the reliability and generalizability of findings, future studies should address these limitations. Expanding the participant pool to include individuals with no personal connection to the researcher would help minimize selection bias. Recruitment through neutral platforms or organizations could ensure a more representative and diverse sample of participants. Additionally, implementing blind testing procedures, where participants are unaware of the researcher's identity, could reduce bias. Conducting remote or anonymous evaluations may further encourage participants to provide candid and unbiased feedback without feeling external pressure. Addressing these issues in subsequent research would enhance the objectivity and validity of the study's findings.

Chapter 5

Conclusions

This research set out to explore and develop an application for operating and directing multiple XR headsets within an immersive XR experience, specifically one involving interactions with an AI character. The project was framed within the context of the TransmiXR EU project, which brings together 7 European universities, 6 industrial partners, and 8 media collaborators to push forward the integration of XR technologies. The objective was to design a versatile, user-friendly system that could support live interactions, dynamic asset management, and multi-user XR sessions, with particular focus on improving usability and control over XR experiences.

The SatoroStudio application, developed as part of this thesis, was constructed on a modular Angular.js framework for the front-end, which communicated with an Express.js back-end. This platform offers a unique combination of real-time device management, intuitive controls, and flexibility for content customization, addressing several key limitations found in existing XR management platforms. By integrating this system with Unreal Engine, the application demonstrated its ability to deliver a more personalized and interactive experience compared to other tools in the domain, such as ArborXR, Protokol, and Meta Quest for Business, which were evaluated in terms of their functional capabilities and limitations. Unlike these existing solutions, the SatoroStudio application allowed for more dynamic scene management and offered a significantly lower cost and complexity for users, making it a compelling tool for various XR applications.

The iterative design process detailed in Chapter 3 highlights the development journey, from early-stage prototypes to a functional and polished final product. The initial prototypes lacked key features such as clear status indicators for headsets and user-friendly control interfaces. However, through successive refinements and user feedback, the final version of the application incorporated vital elements like real-time device status indicators, customizable interface elements, and improved visual design. These improvements culminated in a highly responsive and aesthetically refined system that offers efficient and easy-to-use modes for both operators and directors of XR experiences.

User testing results, as presented in Chapter 4, validate the improvements made during the development process. The usability evaluation demonstrated significant improvements in the "Ease of Use" and "Frustration" metrics, with users reporting a 23% improvement in ease of use and a 235% reduction in frustration using the system. The data from formative and final user groups

indicate a higher level of satisfaction and decreased frustration, suggesting that the application became more intuitive and user-friendly over time. Additionally, the study explored the usability differences between younger and older users, with a particular focus on the challenges faced by users over 35 years old. The results suggested that older users struggled with specific tasks, particularly those involving tablet keyboard interactions and the management of small UI elements. This insight into age-related usability differences emphasizes the need for further adjustments to ensure that the application is fully accessible to a wide range of users.

In conclusion, the research successfully developed and evaluated an application capable of managing multiple XR devices in a performance-driven environment, with a focus on ease of use and user satisfaction. The study highlights the importance of continuous user feedback and iterative design in improving application usability. Although the findings suggest positive outcomes, the limitations in statistical confidence, particularly due to small sample sizes, warrant further investigation with larger participant groups to draw more definitive conclusions.

Future work should focus on addressing the identified usability challenges for older users, expanding the application's capabilities to support a broader range of XR technologies, and refining the interface to optimize both performance and accessibility for diverse user groups.

Bibliography

- [1] Meta ®. Youtube: ' This is Meta Quest 3 '. https://www.youtube.com/watch?v=Exu7r2vZpcw&t=23s&ab_channel=MetaQuest. [Online - Accessed on 07-01-2024].
- [2] Apple ®. Youtube: 'Introducing Apple Vision Pro'. https://www.youtube.com/watch?v=TX9qSaGXFyg&t=196s&ab_channel=Apple. [Online - Accessed on 07-01-2024].
- [3] Samsung ®. Youtube: 'SAMSUNG AR and VR partnership with GOOGLE and QUALCOMM'. https://www.youtube.com/watch?v=6lJl3W-TA78&ab_channel=ARMRXR. [Online - Accessed on 07-01-2024].
- [4] Lex Fridman. Youtube: 'Mark Zuckerberg: First Interview in the Metaverse — Lex Fridman Podcast'. https://www.youtube.com/watch?v=MVYrJJNdrEg&t=1015s&ab_channel=LexFridman. [Online - Accessed on 28-12-2023].
- [5] TransmiXR ®. TransmiXR: Ignite the Immersive Media Sector by Enabling New Narrative Visions. <https://transmixr.eu/>. [Online - Accessed on 05-01-2024].
- [6] Polycam ®. Polycam: 3D capture for anyone. <https://poly.cam/>. [Online - Accessed on 05-01-2024].
- [7] Intel. Youtube: 'Intel® freeD Technology + NFL — Intel'. <https://www.youtube.com/watch?v=J7xIBoPr83A>. [Online - Accessed on 05-01-2024].
- [8] HumanRF: High-Fidelity Neural Radiance Fields for Humans in Motion . <https://synthesiaresearch.github.io/humanrf/>. [Online - Accessed on 18-12-2023].
- [9] Rokoko ®. Rokoko: intuitive and affordable motion capture tools for character animation. <https://www.rokoko.com/>. [Online - Accessed on 05-01-2024].
- [10] Move.AI ®. Move.AI: Move AI makes it easy to capture and create 3D animations. <https://www.move.ai/>. [Online - Accessed on 05-01-2024].
- [11] Volumetric capture: Wikipedia. https://en.wikipedia.org/wiki/Volumetric_capture. [Online - Accessed on 18-12-2023].

- [12] MetaHuman ®. MetaHuman: High-fidelity digital humans made easy. <https://www.unrealengine.com/en-US/metahuman>. [Online - Accessed on 05-01-2024].
- [13] InworldAI ®. InworldAI: NPC behavior powered by our Character Engine. <https://inworld.ai/>. [Online - Accessed on 05-01-2024].
- [14] Tim Siglin. AR, MR, XR and VR Streaming: Understanding the 4 R's. <https://www.wowza.com/blog/ar-mr-xr-and-vr-streaming-understanding-the-4-rs>. [Online - Accessed on 05-01-2024].

Appendix A

User Testing Data

27/11/24, 21:50

Teste com Utilizadores

Teste com Utilizadores

26 responses

[Publish analytics](#)



https://docs.google.com/forms/d/18RjSmHL37K5Z3OCnakidBSf_K_YidUZm6aof3FYxzY/viewanalytics

1/14

27/11/24, 21:50 Teste com Utilizadores

Tarefa 1

26 responses

- 14.09
- 25.11
- 16.29
- 34.53
- 56.08
- 00:00:38.790
- 00:00:18.140
- 00:00:18.330
- 00:00:17.780
- 00:00:34.350
- 00:00:09.410
- 00:00:13.750
- 00:00:14.560
- 00:00:18.840
- 00:00:15.300
- 00:00:11.730
- 00:00:42.960
- 00:00:07.470
- 00:00:09.890
- 00:00:34.050
- 00:00:23.890



https://docs.google.com/forms/d/18RjSmHL37K5Z3OCnakIdBSf_K_YidUZm6aof03FYxzY/viewanalytics 2/14

27/11/24, 21:50

Teste com Utilizadores

00:00:21.910

00:00:20.850

00:00:19.590

00:00:18.540

00:00:50.050



27/11/24, 21:50 Teste com Utilizadores

Tarefa 2

26 responses

- 1.41.31
- 1.28.29
- 1.38.38
- 3.11.29
- 3.29.77
- 00:01:11.310
- 00:01:42.780
- 00:01:11.680
- 00:01:03.270
- 00:01:01.050
- 00:00:45.800
- 00:01:39:410
- 00:00:53.860
- 00:01.00.160
- 00:00:38.330
- 00:01:21.530
- 00:00:59.070
- 00:00:57.070
- 00:00:46.020
- 00:01:04.820
- 00:00:50.840



https://docs.google.com/forms/d/18RjSmHL37K5Z3OCnakIdBSf_K_YidUZm6aof03FYxzY/viewanalytics 4/14

27/11/24, 21:50

Teste com Utilizadores

00:00:54.360

00:00:53.7801

00:00:56.980

00:01:07.600

00:01:52.280



27/11/24, 21:50 Teste com Utilizadores

Tarefa 3

26 responses

36.97
26.88
33.73
32.68
18.76
00:00:24.160
00:00:18.830
00:00:16.930
00:00:14.930
00:00:19.090
00:00:23.600
00:00:14.220
00:00:27.200
00:00:25.350
00:00:20.000
00:00:46.580
00:00:13.520
00:00:15.790
00:00:25.810
00:00:26.050
00:00:30.020



https://docs.google.com/forms/d/18RjSmHL37K5Z3OCnakIdBSf_K_YidUZm6aof03FYxzY/viewanalytics 6/14

27/11/24, 21:50

Teste com Utilizadores

00:00:25.370

00:00:35.650

00:00:35.200

00:00:24.340

00:00:31.950



27/11/24, 21:50 Teste com Utilizadores

Tarefa 4

26 responses

- 09.33
- 09.07
- 14.64
- 27.35
- 07.28
- 00:00:09.730
- 00:00:08.670
- 00:00:12.700
- 00:00:08.750
- 00:00:14.870
- 00:00:08.930
- 00:00:09.200
- 00:00:06.750
- 00:00:06.900
- 00:00:11.710
- 00:00:16.110
- 00:00:08.600
- 00:00:12.300
- 00:00:09.190
- 00:00:08.570
- 00:00:12.590



https://docs.google.com/forms/d/18RjSmHL37K5Z3OCnakIdBSf_K_YidUZm6aof03FYxzY/viewanalytics 8/14

27/11/24, 21:50 Teste com Utilizadores

00:00:12.940
00:00:07.640
00:00:11.080
00:00:16.570
00:00:08.190

Bom Trabalho!

As capacidades deste sistema satisfazem as minhas necessidades. Copy

26 responses

Rating	Count	Percentage
1	0	0%
2	0	0%
3	0	0%
4	0	0%
5	0	0%
6	8	30.8%
7	18	69.2%

A utilização deste sistema é uma experiência frustrante. Copy

26 responses

Rating	Count	Percentage
1	21	80.8%
2	5	19.2%
3	0	0%
4	0	0%
5	0	0%
6	0	0%
7	0	0%

https://docs.google.com/forms/d/18RjSmHL37K5Z3OCnakIdBSf_K_YidUZm6aof03FYxzY/viewanalytics 9/14

27/11/24, 21:50
Teste com Utilizadores

Este sistema é fácil de utilizar. Copy

26 responses

Rating	Count	Percentage
1	0	0%
2	0	0%
3	0	0%
4	0	0%
5	2	7.7%
6	5	19.2%
7	19	73.1%

Tenho de passar demasiado tempo a corrigir coisas com este sistema. Copy

26 responses

Rating	Count	Percentage
1	23	88.5%
2	1	3.8%
3	0	0%
4	1	3.8%
5	0	0%
6	1	3.8%
7	0	0%

Mais alguma observação sobre a aplicação?

5 responses

- Não
- Os oculos deviam ser reais.
- muito intuitiva
- Geralmente intimidada por tecnologia, super fácil de usar.
- É como andar, fácil e intuitivo.

Last Page

https://docs.google.com/forms/d/18RjSmHL37K5Z3OCnakIdBSf_K_YidUZm6aof03FYxzY/viewanalytics 10/14

27/11/24, 21:50
Teste com Utilizadores

Copy

Idade

26 responses

Idade	Respostas	Porcentagem
17	1	3.8%
20	5	19.2%
22	1	3.8%
23	3	11.5%
24	2	7.7%
25	2	7.7%
32	1	3.8%
35	2	7.7%
36	1	3.8%
37	2	7.7%
54	1	3.8%
57	1	3.8%
58	1	3.8%
60	1	3.8%
61	1	3.8%
69	1	3.8%

Gênero

26 responses

Gênero	Porcentagem
Male	46.2%
Female	53.8%
Prefer not to answer	0%


Em média, quanto tempo passa a utilizar o seu smartphone por dia?

26 responses

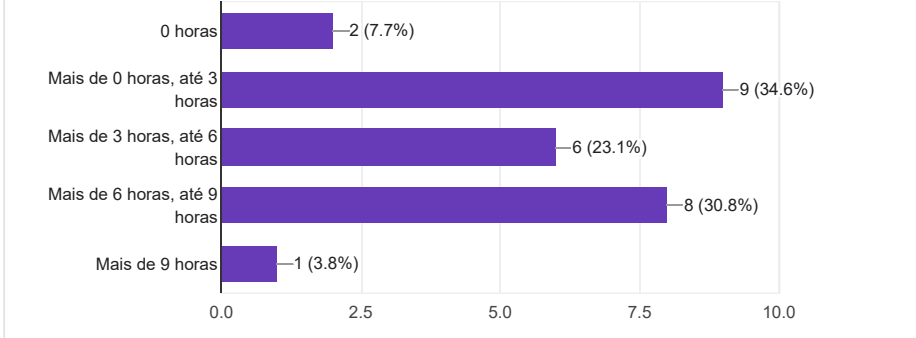
Tempo	Respostas	Porcentagem
0 horas	0	0%
Mais de 0 horas, até 3 horas	15	57.7%
Mais de 3 horas, até 6 horas	6	23.1%
Mais de 6 horas, até 9 horas	4	15.4%
Mais de 9 horas	1	3.8%

https://docs.google.com/forms/d/18RjSmHL37K5Z3OCnakIdBSf_K_YidUZm6afo3FYxzY/viewanalytics
11/14


27/11/24, 21:50
Teste com Utilizadores

Em média, quanto tempo passa a utilizar o seu computador por dia?  Copy

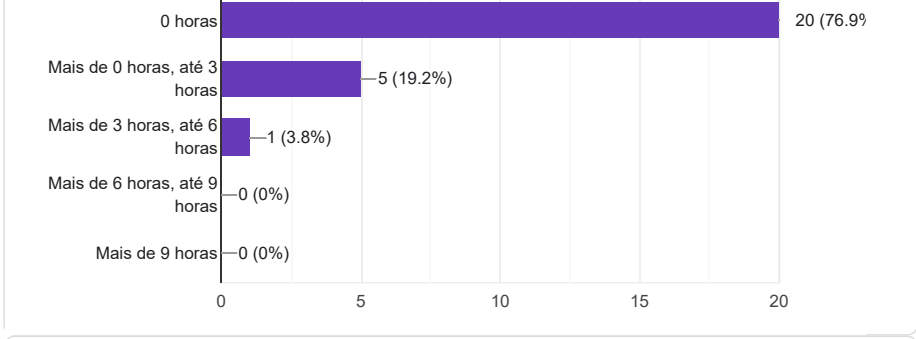
26 responses




Tempo	Respostas	Porcentagem
0 horas	2	7.7%
Mais de 0 horas, até 3 horas	9	34.6%
Mais de 3 horas, até 6 horas	6	23.1%
Mais de 6 horas, até 9 horas	8	30.8%
Mais de 9 horas	1	3.8%

Em média, quanto tempo passa a utilizar o seu tablet por dia?  Copy

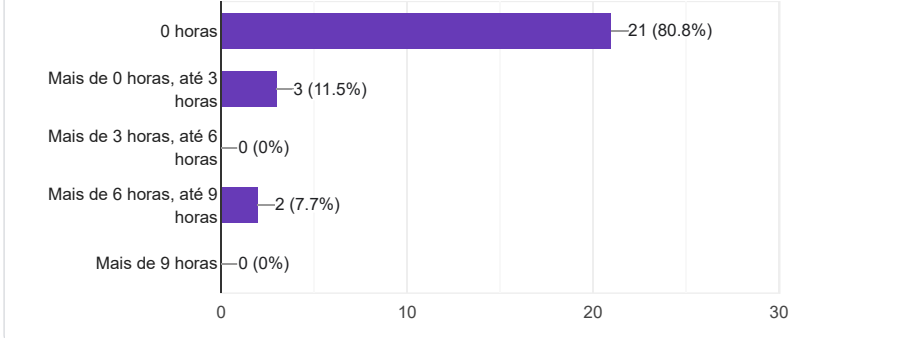
26 responses



Tempo	Respostas	Porcentagem
0 horas	20	76.9%
Mais de 0 horas, até 3 horas	5	19.2%
Mais de 3 horas, até 6 horas	1	3.8%
Mais de 6 horas, até 9 horas	0	0%
Mais de 9 horas	0	0%


Em média, quanto tempo passa a programar por dia?  Copy

26 responses



Tempo	Respostas	Porcentagem
0 horas	21	80.8%
Mais de 0 horas, até 3 horas	3	11.5%
Mais de 3 horas, até 6 horas	0	0%
Mais de 6 horas, até 9 horas	2	7.7%
Mais de 9 horas	0	0%

This content is neither created nor endorsed by Google. - [Terms of Service](#) - [Privacy Policy](#).

Does this form look suspicious? [Report](#) 

https://docs.google.com/forms/d/18RjSmHL37K5Z3OCnakIdBSf_K_YidUZm6aof03FYxzY/viewanalytics
12/14

27/11/24, 21:50

Teste com Utilizadores

Google Forms



https://docs.google.com/forms/d/18RjSmHL37K5Z3OCnakIdBSf_K_YidUZm6aof03FYxzY/viewanalytics

13/14

27/11/24, 21:50

Teste com Utilizadores



Appendix B

Procedure Document

28/11/2024, 22:55

Teste com Utilizadores

Teste com Utilizadores

Hoje, vai trabalhar como Guia para uma experiência de Realidade Virtual num Museu!
O seu objetivo é integrar dois visitantes (imaginários) na experiência de Realidade Virtual.

Para tal, terá acesso a um tablet para controlar o estado da experiência, isto é, começar ou parar a experiência que decorre em simultâneo para os 2 óculos de Realidade Virtual.

Aqui, indicamos as 4 tarefas a seguir e, de seguida, o questionário a preencher. Aguarde a confirmação antes de iniciar a tarefa.

ricboy20@gmail.com [Switch account](#)



Not shared

* Indicates required question

Tarefa 1

1. Para começar, deve executar a aplicação SatoreApp.exe no seu computador.
2. Em seguida, executa a aplicação Satore no tablet, para se ligar ao computador.

(Exemplo tempo: 00:25:11.000)

Your answer



https://docs.google.com/forms/d/e/1FAIpQLSc6cYDu6isH3TYnZ-Su-dRZOgHyM6RWgu11YZ3b2G_k6nAqw/viewform

1/5

28/11/2024, 22:55

Teste com Utilizadores

Tarefa 2

Agora, introduza a configuração dos 2 óculos no tablet, para verificar se encontram-se online ou offline.

Dispositivo 1:

Nome: VR1
Endereço IP: 127.0.0.1

Dispositivo 2:

Nome: VR2
Endereço IP: 127.0.0.2

(Exemplo tempo: 00:25:11.000)

Your answer

Tarefa 3

1. Ligue os óculos e verifique se os óculos estão online na aplicação.
2. Coloque os óculos nos visitantes e pergunte-lhes se conseguem ver o ecrã de lançamento da experiência de Realidade Virtual nos óculos.

(Exemplo tempo: 00:25:11.000)

Your answer

Tarefa 4

1. Finalmente, inicie a experiência de Realidade Virtual para os 2 visitantes .
2. Confirmar no ecrã do tablet se a experiência foi iniciada para os 2 visitantes.
3. Pergunte também se os visitantes já começaram a experiência de Realidade virtual.

(Exemplo tempo: 00:25:11.000)

Your answer

Bom Trabalho!

Agora que já terminaste, podes começar este teste:



28/11/2024, 22:55 Teste com Utilizadores

As capacidades deste sistema satisfazem as minhas necessidades. *

1 2 3 4 5 6 7

Discordo Fortemente Concordo Fortemente

A utilização deste sistema é uma experiência frustrante. *

1 2 3 4 5 6 7

Discordo Fortemente Concordo Fortemente

Este sistema é fácil de utilizar. *

1 2 3 4 5 6 7

Discordo Fortemente Concordo Fortemente

Tenho de passar demasiado tempo a corrigir coisas com este sistema. *

1 2 3 4 5 6 7

Discordo Fortemente Concordo Fortemente

Mais alguma observação sobre a aplicação?

Your answer

Next Clear form

Never submit passwords through Google Forms.

This content is neither created nor endorsed by Google. - [Terms of Service](#) - [Privacy Policy](#)

Does this form look suspicious? [Report](#)

Report

https://docs.google.com/forms/d/e/1FAIpQLSc6cYDu6isH3TIYnZ-Su-dRZOgHyM6RWgu1IYZ3b2G_k6nAqw/viewform 3/5

28/11/2024, 22:55

Teste com Utilizadores

Google Forms



https://docs.google.com/forms/d/e/1FAIpQLSc6cYDu6isH3TIYnZ-Su-dRZOgHyM6RWgu1IYZ3b2G_k6nAqw/viewform 4/5

28/11/2024, 22:55

Teste com Utilizadores

