Cross-Platform Multi-User Interaction Tools in Unreal Engine

CMGT Graduation Project

Student:
Mihai Constantinescu
Saxion University of Applied Sciences CMGT
Student number: 446505
mihaiadrianconstantinescu10@gmail.com

Company:
Zwartlicht
Company Coach: Danny Kroonen
dkroonen@zwartlicht.nl

Graduation Coach:
Yvens Rebouças Serpa
y.reboucasserpa@saxion.nl

Date:14/06/2021

Abstract

This report aims to provide details of the creation of a cross-platform multi-user solution for the company Zwartlicht. It contains all the steps that have been used over the duration of the whole graduation project, as well as the findings acquired from exploring new functionalities of the game engine Unreal. It also emphasizes and explains the procedures used to reach the goals of the project. The topics that are discussed in this paper are: the usage of the Collab Viewer plug-in that is provided by Unreal, a solution that has the basic functionality of multiplayer set up already for the developers. Furthermore, it talks about the external plug-in VaRest which allows the communication between an Unreal Engine application and a database such as MySql tables. Another important topic present in the document is the creation of a dedicated server for an Unreal Engine application with the help of Amazon Web Services solutions, that allows people around the world to join the same session. The final topic consists of the export methods used to put the application on mobile devices. The platforms used for this were Android and IOS. The product managed to solve the communication problems and allowed the designers to create changes without the need of rebuilding the applications. The cross-platform functionality increased the accessibility and allowed the people involved in the project to work from multiple types of devices such as computers, VR sets and mobile devices.

Table of Contents

List of figures

1. Introduction

1.1 Graduation Company

Zwartlicht is a company located in the city centre of Utrecht that specializes in architectural visualizations (Zwartlicht, 2021). It contains a technical branch called Lichtstorm (Lichtstorm, 2021). This branch works with Augmented Reality and Virtual Reality in order to create practical applications and find more interactive ways of presenting the company's products. These products are created with the help of the environment designers of the company, who are creating 3D assets. They are responsible for the layout designs of the scenes and the quality of the models created by them.

1.2 Objectives of The Client

The goal of the company is to research new ways of giving the designers the ability to place and move 3D objects in an Unreal Engine scene. Unreal Engine is a game engine that can be used to create multiple types of applications that use real-time rendering (Unreal Engine, 2021). The company uses this engine in order to create real-time architectural visualisations applications in which the users can explore the environment. These applications that are in production, are deployed and they can only be changed by making modifications in the engine's editor, which is followed by a redeployment of the application. They want to use Unreal Engine without the need of accessing the editor or to redeploy their projects. This would allow them to work on PCs and then test the environments or do presentations on other devices such as VR and mobile phones.

Unreal Engine's multiplayer system called The Replication System (Actor Replication, 2021), can enable the interaction between workers which will edit the environment inside the scenes. This will allow them to make changes faster by working together in the same virtual space.

The application will allow both designers and their clients to share an online real-time environment, with persistent changes, which will improve communication. The application will be available on computers, Virtual Reality sets and mobile devices.

The solution must present a basic user interface which will be easy to understand and use and the functionality of the tools must be intuitive. The target audience for this application consists of professional designers that use Unreal Engine alongside the clients that need to review their work or make changes to it.

2. Problem Definition

The environment designers from the company Zwartlicht spend too much time making changes on their Unreal Engine products. This negatively affects the speed at which they can modify the already deployed applications. Moreover, they do not have the ability to interact with their colleagues in the same 3D space. This leads to misunderstandings and errors that can cost more time to be corrected in a later stage of production. With the use of a new system they expect to overcome these issues. They need a simple solution that does not require technical knowledge in order to be used.

3. Research

3.1 Main Question

How to allow the designers from the company Zwartlicht to modify Unreal Engine environments in already built applications and enable interaction between the designers and the clients?

3.2 Sub-questions

Sub-question 1:
How to keep track of changes made in Unreal applications that are in production?

Sub-question 2:
How to make the application work on different types of devices such as PC, VR and mobile phones?

Sub-question 3:
What tools are necessary for editing environments inside built Unreal Engine projects?

Sub-question 4:
What Unreal Engine features can be used in order to enhance interaction between the company and the clients?

3.3 Methodology

To answer the sub-questions, desk research will be conducted in order to look into possible solutions for the current problems. These solutions will be evaluated and the corresponding systems will be used for the application. In addition to this, at least five designers and volunteers will test the application and will also be interviewed and asked about the ease of interaction with the application. This will be followed by an observation session in which the participants will use multiple versions of the application using different interaction methods.

3.4 Project Scope

The architecture of the application will allow a maximum of ten people to use the application on the same session simultaneously without any performance problems. The limit was set by the company and the clients. They believe it is a reasonable limit and that they do not have more than ten people working on the same project simultaneously. The final version of the product will use a dedicated server where people will be able to meet but not create additional sessions. Joining the session will be made by entering the IP of the server, or the application will make it join the server based on an IP address present on the database. The project will allow voice communication for all of the platforms.
The testing sessions will use the three categories of devices: PC, VR and mobile phones at the same time, Android and IOS representing the focus platforms for mobile devices. The movement system will be different for each platform but the main functionality will remain the same. Due to previous experiences, the VR version will use teleportation for movements in order to avoid motion-sickness.

The application aims to be simple to use. The main goal is to achieve a comprehensive layout created with placeholders, the creation of assets such as images for buttons or theme colors are not the focus of the project.

External volunteers will be asked to test and give feedback on the application. The volunteers consist of the clients of the company and experienced programmers that work with similar systems.

The product will be the first step in gaining full control over the scenes used in the applications. Both the designers and the clients will be able to edit and save their versions independently and there will be no limitation regarding technical knowledge.

The limitations of the project are represented mainly by the amount of interactive scenarios that the application will contain. For the final demonstration, a single interactive scenario will be made in order to create an achievable time frame. The design and production stages of the user interface will be kept on a minimum in order to place the focus on the functionality of the application.

4. Theory

The solutions below represent the systems and external tools that were selected for the realization of the application and explains why they have been chosen over other existing solutions, if applicable.

Sub-question 1: How to keep track of changes made in Unreal applications that are in production?

In order to keep track of the changes, the application needs an online environment that will store the information of the objects such as position and rotation. The company preferred a database solution on a server, similar to what they are currently using. The company required the server to be inside of the Netherlands to ensure the smallest delay time. The company is currently using an Azure server (Microsoft Azure, 2021), but they wanted to explore other possibilities. They have recommended the website “hostnet.nl”. The company used to have a physical Linux server which contained the database and suggested that it would be another possibility to rent a Linux server and create the database on it.
The systems of the company are running using PHP and MySql and they want both to be used for the server. The developers of the company want to be able to make the changes on the application in the future without the need to learn new frameworks.

For the hosting of the servers there were three options available that were recommended by the company. Using Microsoft Azure on the company’s account, using a HostNet server and renting a Linux server. The selection criteria will be based on the development time required and price.

The Azure server presents the option to use MySql databases directly within the hosting package. It takes a considerate amount of time to set up the server with and start the MySql database. The website is hard to use and the features are not intuitive. For the basic package the Azure server costs 50 euros per month. The advantages of using this site consists of it’s scalability, the databases can increase in size and the components of the servers can be selected. In case the server requires more power to keep up with the data transfer, it will give the possibility to be upgraded. Azure also has other services that can be used by the company.

The linux server would require the most amount of time to set up out of the three options. According to the MySql documentation (Getting Started with MySQL, 2021), the database needs to be manually set up. The biggest disadvantage of this solution is the time required to have it set up. Without previous knowledge of Linux, the server is considered to be a big risk for the project. Regardless of the risk, these are advantages that it could provide. It can enable the ability to fully manage the database. Moreover, it allows for a big amount of features that can be added, for example, using another type of database for the server. All of the functionality that is added to the server does not require more money in order to be maintained. All of the features presented by Azure can be achieved on a Linux server without extra costs.

HostNet presents fewer solutions for their servers. But these solutions come already prepared and ready to use, the launching of the databases is done in less than thirty minutes and then it is ready for production. The package with a MySql database costs 6 euros per month. The biggest advantage of this hosting solution is represented by the fast deployment of the server and the intuitive user interface.

It was decided that HostNet will be used for the server of the application. The website is easy to use and allows for easy navigation. Because the application will be used to send only a small amount of data, it will always perform correctly and will not need to be upgraded to a better or bigger server. The company used Azure for downloading large amounts of data and in that case the server needs to be powerful enough. But because the HostNet server can provide a much cheaper alternative and it’s limited power accomplishes the goal of the project, it will be used as the host website for the application.

HostNet also represents the best financial decision for the future of the company. Because of its price, the company will manage to increase the budget for the development of other products without worrying about the costs of hosting the database. Another financial advantage is represented by the ease of deployment for the database. The developers will require less time to set up the environment, which in the end will save money and resources such as time, which is the most important resource for any project.

In order to create the connection between the server and Unreal Engine, there needs to be a system that will allow the transfer of data between them.

The first solution was represented by a system created by MySql (MySQL Connector/C++, 2021). Because Unreal Engine uses C++, the Engine can use C++ libraries. This library communicates to the databases and runs the queries directly.

Another solution is represented by VaRest (Vladimir Alyamkin, 2021) a solution that uses the blueprint system in order to send the requests to the database. It is fully compatible with Unreal Engine, their community is active and the documentation is up to date.

The most direct approach is represented by using C++ HTTP calls in the code in order to run the scripts on the server. As presented in the article “Making HTTP Calls from Unreal Engine 4” (David Y., 2016), Unreal Engine uses C++, and it can directly make HTTP requests to the database.

Out of these three solutions, VaRest was selected for the project. The biggest advantage is that the plug-in uses the blueprint system. Because the majority of the project is done in the blueprint system, writing the scripts in C++ would need more work to make them compatible with the other blueprints. The plug-in also has already prepared functionalities, this would increase the production speed of the project. Furthermore, the active community and the amount of documentation will allow for an easier implementation.

Sub-question 2: How to make the application work on different types of devices such as PC, VR and mobile phones?

For VR applications, Unreal engine uses a built-in plug-in (Unreal Engine, 2021). The plug-in allows all of the VR sets to be connected through SteamVR (Steam, 2021), with the help of this plug-in, all of the inputs generated from various devices can be translated into commun inputs and it allows for an easier development of the application.

For utilizing the mobile devices, Unreal Engine will require external tools for building on Android (Android Studio, 2021) and IOS (Xcode, 2021) in order to export the application for the desired platforms. The mobile version of the project will need to have different movement options compared to the computer variant.

For exporting on Android devices, Android Studio will be used. It is a program that provides tools for exporting applications in a format suitable for Android devices. It will allow the creation of APK files which is the standard format for the Android projects.

For exporting the project on IOS, Xcode will be used. It is a program that allows the developers to export the projects on Apple products such as IPhones and IPads. Because Xcode can only run on an Apple device, an external computer will be used to build the application. After all the licences are set for the project, the target devices must be plugged in, in order to start the installation.

For the mobile versions of the applications, the user interface of the mobile devices presented two digital joysticks, one for movement and one for camera rotation. The camera rotation had the Y axis inverted and during the test sessions, people complained that they found it difficult to use and hard to adapt to. This required research in order to understand why the joystick is inverted by default and if it needs to be changed.

According to the article “When up means down: why do so many video game players invert their controls?” (Keith Stuart , 2020), the old gaming consoles used to have the Y axis reversed, this created the setup to be standard for future consoles, and this is why Unreal Engine implemented the same layout. Knowing this, the application will still have to change the Y axis and make it so that when the joystick faces the up direction, the camera will also look up. Even if the reversed Y axis is standard in the gaming industry, the people that did not use these consoles will get confused and believe that the controls are wrong. It was decided to change the current setup and remove the reversed controls.

Sub-question 3: What tools are necessary for editing environments inside built Unreal Engine projects?

For changing the position and rotation of the object, a system needs to be made that will allow the users to interact with the objects from different devices such as PC, VR and mobile phones. The tools must be intuitive in order to enable the users to make modifications to the scenes without any problem. The interaction will be similar on all of the devices allowing the people to switch between platforms without learning a new layout.

According to the article “How to Create an Intuitive Design.” (Ditte H., 2021), an intuitive design is based on the daily experiences of the users. Their daily actions influence their perception of how an action needs to be performed, creating a sense of familiarity. In the case of the environmental designers, their daily work is mostly spent in 3D modelling applications. All of these applications have a similar system for moving 3D objects in space called Gizmos (AutoDesk, 2020). They are 3D representations of axis that the designers can use to manipulate the position, rotation and size of objects. A good example is represented by the documentation on Gizmos for AutoCAD “About Using 3D Gizmos.” (AutoDesk, 2020). Due to its usage in all of the modeling programs, the Gizmos will be used in the project for the representation of object manipulation.

The addition of Gizmos required the creation of the manipulation system. It allowed to be built in the blueprint system or a plug-in could have brought the functionality and the application would only need adaptation. All of the plug-ins that were available would have classes that would modify the player class in C++. An example of a plug-in is the “Runtime Transformer Tool (Gizmo)” (Juan Marcelo, 2020), the transformations were completely created in C++ and it did not allow for the use of a separated player class. These plug-ins want the developer to use their custom player classes which limits the potential of the application.

Out of these two, the creation of a new system was selected. A system created in the blueprint system that would be easy to integrate in the project and it would allow the application to use the desired player classes. It would also improve control over how the transformations that are being executed.

Sub-question 4: What Unreal Engine features can be used in order to enhance interaction between the company and the clients?

In order to enable the interaction of users over the network, a multiplayer system needs to be created in order to ensure communication between the applications. The system can use a separated C++ library or the Actor Replication (Unreal Engine, 2021) system that is already integrated in the Engine.

A solution would be represented by creating a custom networking system using the protocols UDP and TCP. UDP is a protocol used for sending and receiving information over the internet. It is fast but the protocol is not reliable. It can lose packages of information, receive them duplicated or even lose them (Cloudflare, 2021). In the application it can be used for sending information such as movement of users and objects, as precision is not required (Cloudflare, 2021). TCP is a protocol that requires a connection (Cloudflare, 2021). It is slower than UDP but it is reliable. It makes sure that the packages arrive at the destination in the right condition without problems (Cloudflare, 2021). The protocol can be used in the application for events such as a user joining the server or the interaction with the objects, as these events must take place and cannot be ignored. The main disadvantage of this solution is the amount of time required for its realization. An advantage of this system would be the amount of control that can be achieved over the multiplayer behaviours.

The Replication System present in Unreal Engine, is a system already integrated that allows the developers to create networking systems for the applications. The system allows functions and commands to be sent to other devices in order to synchronize the sessions of the users.

ENet (ENet, 2021) is one of the C++ libraries that are used for multiplayer systems. It uses the protocol UDP in order to send the data over the internet. Because the solution only contains the UDP protocol, it can only be used for actions such as movement and rotation. The system is not reliable in the situation of sending important data such as executing events on the other devices.

Another library is Libevent (Libevent, 2021). It is based on event notification and it is more reliable than ENet. A possible disadvantage would be that the system would not keep up with the amount of information sent from the movement and rotation of objects. In this case ENet will perform better due to its UDP protocol.

The Replication System was selected for the networking system of the application. The system already contains blueprints that can be used to fire events over the network or send data such as positions and rotations. Another big advantage is the amount of time that the system requires to be implemented. Furthermore, the Unreal Engine community is active and this can ensure the successful implementation of the networking system.

The Multi-User plug-in (Collab View Template, 2021) will be used in the project, because it provides a basic scene with the controls already set in the project for PC, VR and mobile devices and it will speed up the production process. The biggest advantage is that it uses the Replication System present in Unreal Engine. It represents a very good starting point that would allow the application to grow faster. It also contains a session system already implemented. With this the users can join the sessions of other people from multiple types of devices.

For the dedicated server, a hosting website will be needed in order to make the application available over the world.

Amazon Web Servers (Amazon, 2021), is a hosting solution that allows the developers to place their applications on the web. Its user interface is intuitive and easy to navigate. The pricing of the services is done by how much the server was used instead of a monthly subscription. The server is also scalable and allows for modification of components.

Azure Gaming Server (Microsoft, 2021) is another service provided by Microsoft Azure. It provides a scalable server that can be modified by the developers. This service has the same disadvantage as the database hosting, being the most expensive solution out of the ones selected for this project. The implementation of the application on the server will be harder than the rest because of the hard to navigate user interface.

Transip (Transip, 2021) is a cheaper alternative to the Microsoft Azure servers. The hosting packages provide fixed amounts of resources on a monthly subscription. The hosting website has a more intuitive user interface than Microsoft Azure. Due to its small community, the possible problems that can appear may put the project at risk.

Out of the three solutions, Amazon Web Servers was selected for the project. The user interface allows for an easy implementation of the application and the server can be deployed in less than ten minutes. The financial aspect of the service is also an advantage. If the application is not used, the servers are not going to charge money like the other two web hosting solutions that have a monthly subscription and require the customer to pay for it even if it is not used.

5. Development

5.1 Synchronized Sessions In Unreal Engine

For the synchronized sessions, as mentioned in Unreal Engine's Documentation (Collab Viewer Template, 2021), it contains a multi-user solution that allows developers to start creating the application using networking directly. It has a session system already set up that allows the other users to join a session on a different device using a premade interface. The way that the developers can connect to other sessions without any graphical tools is by using the console. They need to type "open" followed by the IP address of the hosted session.

The template contains a base class that is used for the three types of platforms supported by the system: PC, VR and mobile devices. Because they all inherit the same object, shared functionality can be programmed and then used by all of the platforms. The application created for the project takes advantage of this and uses the same functionality for the editing tools. This section is explained in more detail in 5.3.

Other functionalities present in the template consist of the ability to draw lines in the scene, which are synchronized by default for the other users present in the session. Another feature is represented by the ability to move objects. Due to it's inaccurate functionality, it has been decided to be removed from the project and to use the tools that were developed for more advanced editing capabilities.

As presented in Figure 1, the template allows for users to join sessions locally. The functionality is only created to make the export of the application available for Windows, for exporting on other platforms such as Android and IOS, the developers need to set it up themselves. The application was able to create a session and the other Windows computer could join the user if they shared the same internet connection.

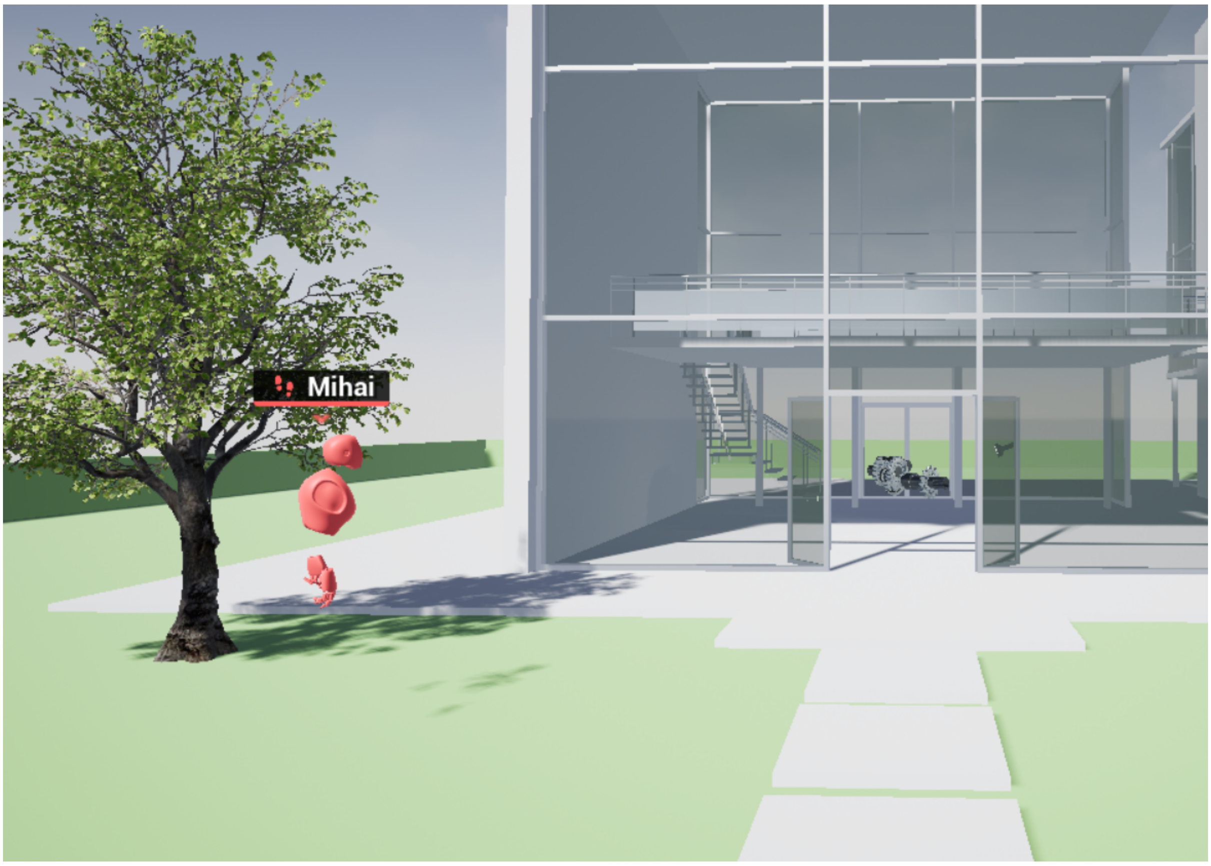


Figure 1. Screenshot presenting the base of the project using the Collab View Template.

Joining other sessions over the internet was not possible at this stage. In order to allow this, the users needed to open a port on their router, which meant that they had to make additional configuration inside their internet routers. Because this would not be possible for the majority of the users, it was decided to switch for a dedicated server. This topic is addressed in section 5.4.

The basic functionality of the template also allows the users to move in the scene at a very basic level, each platform requiring their own series of modifications that were applied to conform to the feedback of the testers.

5.2 Communication between Unreal Engine And Databases

The website HostNet was used in order to acquire a server that would contain the database used for the application. With the help of the site's documentation (Hostnet Academy, 2021) the server was created and basic functionality could be tested on it. On this server the database was represented by MySql tables. The tables were integrated inside of the hosting package. For each file saved there would be a table that contains the information of the stored items. Another table is used for keeping track of the existing save files. In this way, the code would be able to tell if a save file with the same name already existed on the server.

The plug-in VaRest packages the information that needs to be sent using json, and it also receives the information in the same format. Because the engine was sending the data as json packages, more functionality had to be added to the server. According to PHP's documentation (JSON Functions, 2021). The package must be opened with the function "json_decode" before being able to interact with the provided information.

Unreal Engine development can be done through its blueprint system (Blueprint Visual Scripting, 2021). This allows the developers to program the functions of the applications. The blueprint system is a powerful tool and it is possible to program complete applications with it. C++ is a programming language supported by Unreal Engine, which allows the developers to create content with it. This represents an alternative to the blueprint system and it can be used to create functionality that the blueprint system does not provide.

The functionality that was missing from the blueprint system for the project, was represented by the ability to split string entries similarly to C#. Because of this, the functionality was made in C++ (Programming with C++, 2021). As the json object was received by Unreal as a long string, the engine did not contain the functionality to split the string into information and to load objects based on the coordinates. The C++ scripts that were created for the project, managed to fix this problem.

As presented in Figure 2, the application is capable of loading and saving object layouts to a database. New entries can be created and old ones can be overwritten.

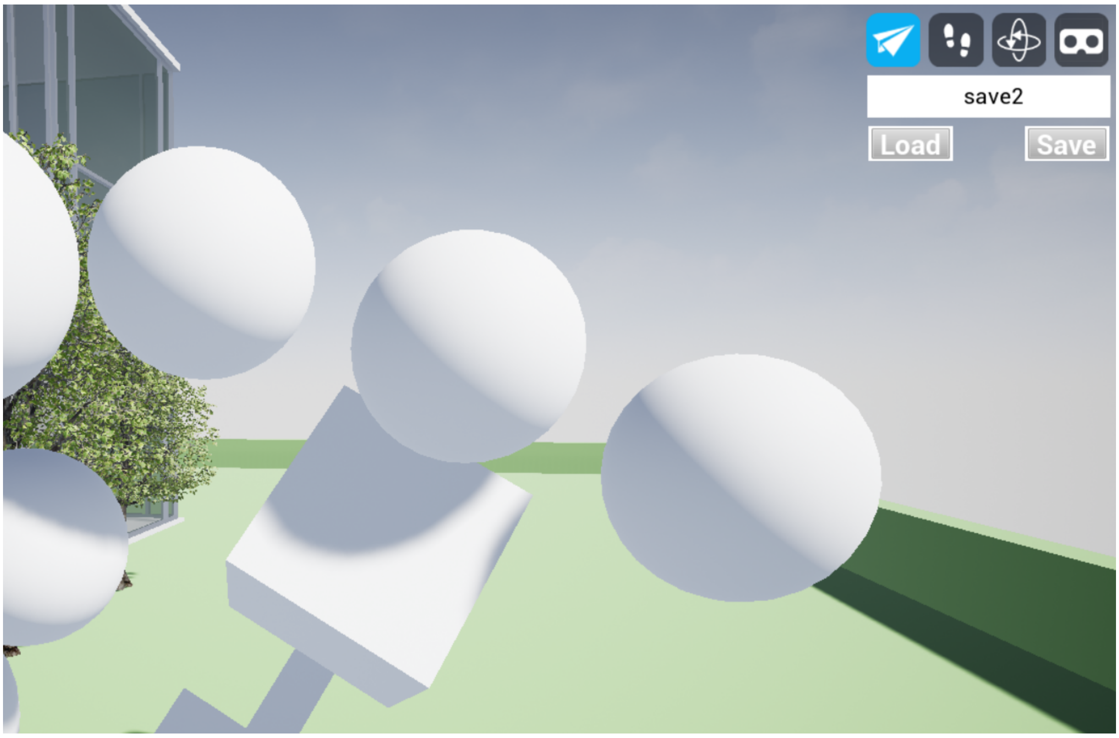


Figure 2. Screenshot presenting the process of loading a layout of objects.

5.3 Editing Tools

The users required a way to manipulate the objects that were loaded from the database in order to create the changes they desired. The application contains three types of interactions: movement, rotation and scale.

The movement of the objects was done on three axis, each represented by an arrow. They represent the world position of the objects and they do not rotate with it. They keep their orientation in order to make the movement of the assets easier. On the intersection of each arrow, they have a commun plane that, if used, will allow the object to move on both axis. This allows the users to move the objects in their desired direction, for example, they can move the objects on ground level without adjusting the height.

The rotation tool is similar to the arrow and allows the users to rotate the objects in three directions. For the horizontal rotation, the current object will follow the pointer of the user, while for the other two axis, the rotation is additive. For example, if the user wants to rotate the object on the X axis, they aim their pointer upwards. The object will continue to rotate as long as the user is moving the pointer.

For the scale of the object, it was decided that only the complete scale of the object should be changed. If the object were to scale on a single axis, it would look stretched. For this tool the same additive movement is used for changing the size of the object. If the pointer goes up, the object will grow depending on the distance from the origin point, and it decreases if the pointer goes down.

As presented in Figure 3, the manipulation system uses a Gizmos systems in order to allow an intuitive interaction with the application

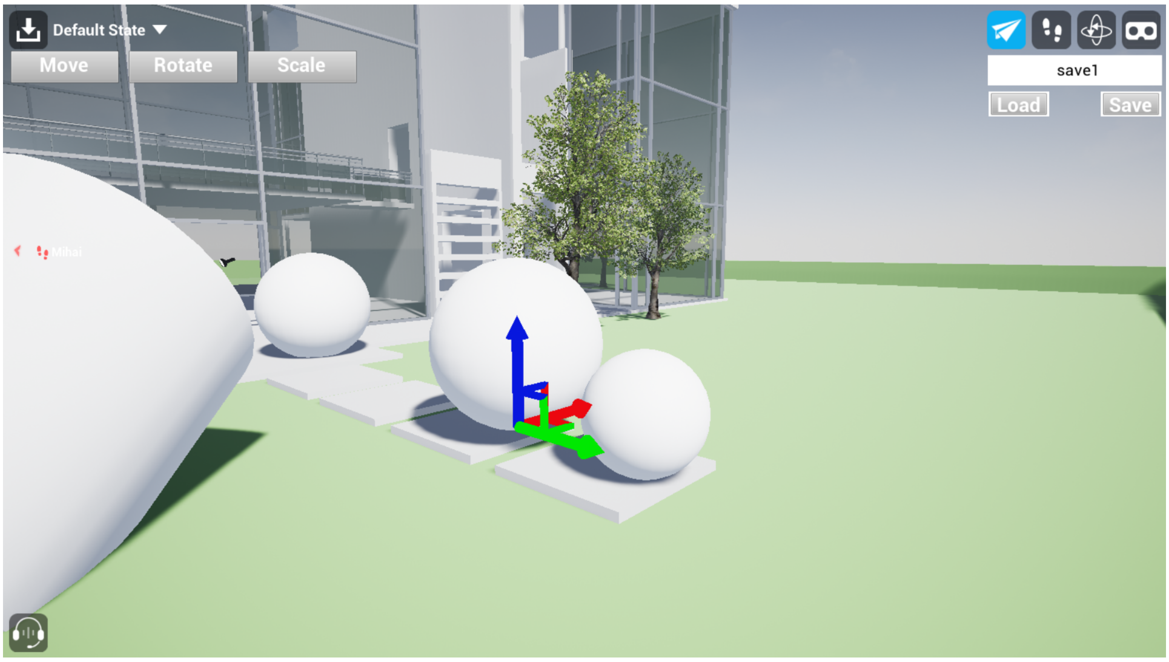


Figure 3. Screenshot of editing tools that resemble Gizmos.

For the PC and mobile versions of the application, the user interface has similar functionalities and both versions are represented in the same way on the screen. For the VR version, the user can change the editing mode by using the grab button of their VR controller.

An interaction mode was added in order to activate animations or use scenarios. For example It can be used to open the doors of the buildings present in the scenes or to play scenario animations for presentation purposes.

5.4 Dedicated Server For The Unreal Engine Application

Unreal engine allows the developers to create dedicated servers for their applications. According to their documentation (Setting Up Dedicated Servers, 2021), the developers need the source build of the editor in order to create the application console which can be found on their GitHub page (Epic Games Repository, 2021).

With the source build of the engine, another version of Unreal Engine was created. This was the source code for the editor version 4.26.2. After the new version was compiled, It had to convert the current project in order to prepare it to be exported as a server build. Both processes took a considerable amount of time. A total of twelve hours were required to compile the editor and the server build. For future projects it is recommended to compile them both before starting the production phase.

For the server provider, an Amazon server was used (Amazon Web Servers, 2021). The most important reason for using a dedicated server was the need of having a machine that has an open port that can allow other devices to connect to it over the internet.

After the application was placed on the server. Other devices were able to join the session over the network.

With this the application was ready to be tested by the company. The test had both people that were using the same internet connection as well as participants that used the application from their home over the network.
The server was very responsive and the network delay was not noticeable. As presented in Figure 4, the test allowed the people to join the same sessions and interact with the surrounding environment.

Because all of the participants had the authority to move objects and load new layouts, it allowed for a stress test on the networking capabilities of the server. The application worked without problems despite the amount of data that was exchanged.

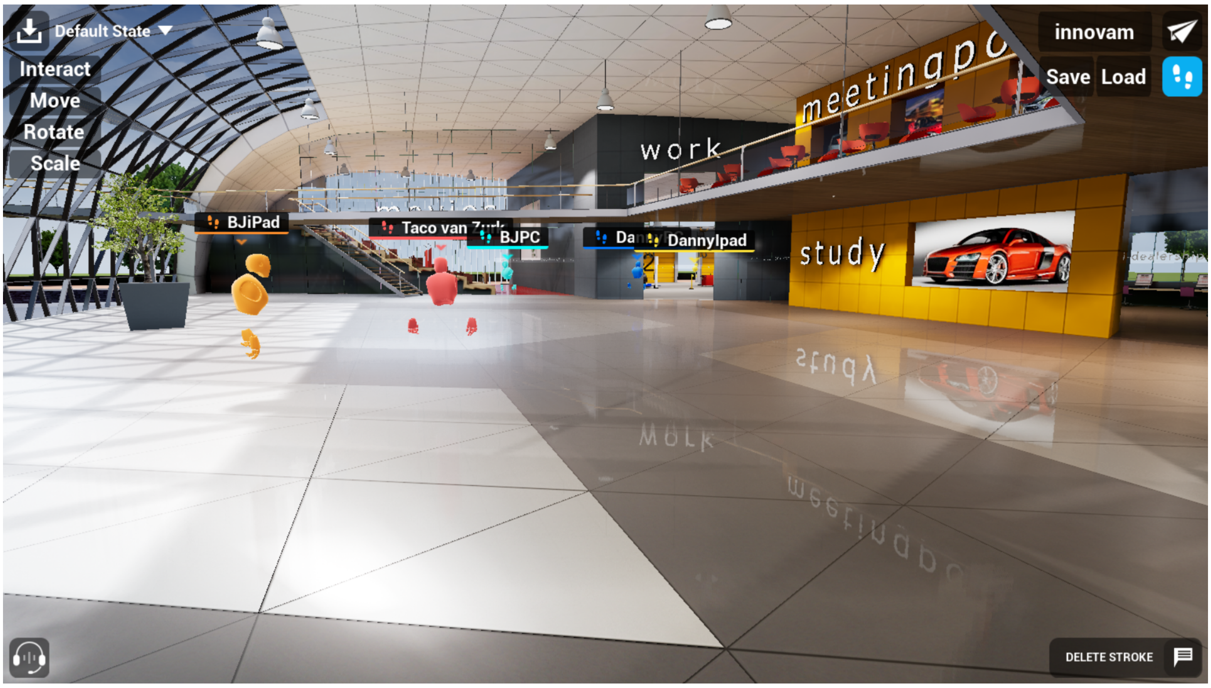


Figure 4. Screenshot taken during the internal stress test of the application.

5.5 Exporting To Android

Unreal Engine provides a detailed guide (Android Quick start, 2021) on how to export the application for Android.

The first step was to install Android Studio (Android Studio Developers, 2021) in order to install the SDK and NDK required by the application. The tools can package the project and generate an APK file that can be used to install the application on Android devices.

In order to allow the Unreal Engine application to be built on the Android devices, a key had to be generated (Unreal Engine, 2021). The Android key contains information about the company and about the application. The information is entered by the developers and it can be customized to their requirements.

A difficult task was represented by the installation of the OBB file on the mobile devices (Android Studio, 2021). This file contains additional information that is required for the installation. If the application is installed using only the apk, it will not be able to run and will ask for the OBB file.

The installation of the file can only be done if the mobile device is connected to the computer through a cable. When packaging an application for Android, Unreal Engine generates three main files: the APK, the OBB and the installation executable. By starting the installation using the executable, the program will install the APK and OBB on all of the Android devices connected to the computer.

After the installation on multiple Android devices was completed, the mobile devices were able to create sessions over the local network and connect to each other. The problem that appeared during this stage is that the computers could not connect to the mobile devices over the local network. According to “UE4: How to set up LAN-Multiplayer for Windows and Android” (Jonathan H., 2019), the computer fails to send the correct local IP address over the network to the mobile devices.

As presented in Figure 5, during the test with the dedicated server, the mobile phones managed to connect to other computers with no problem, this allowed all of the devices to connect and it was decided that the local networking functionality was not needed anymore.

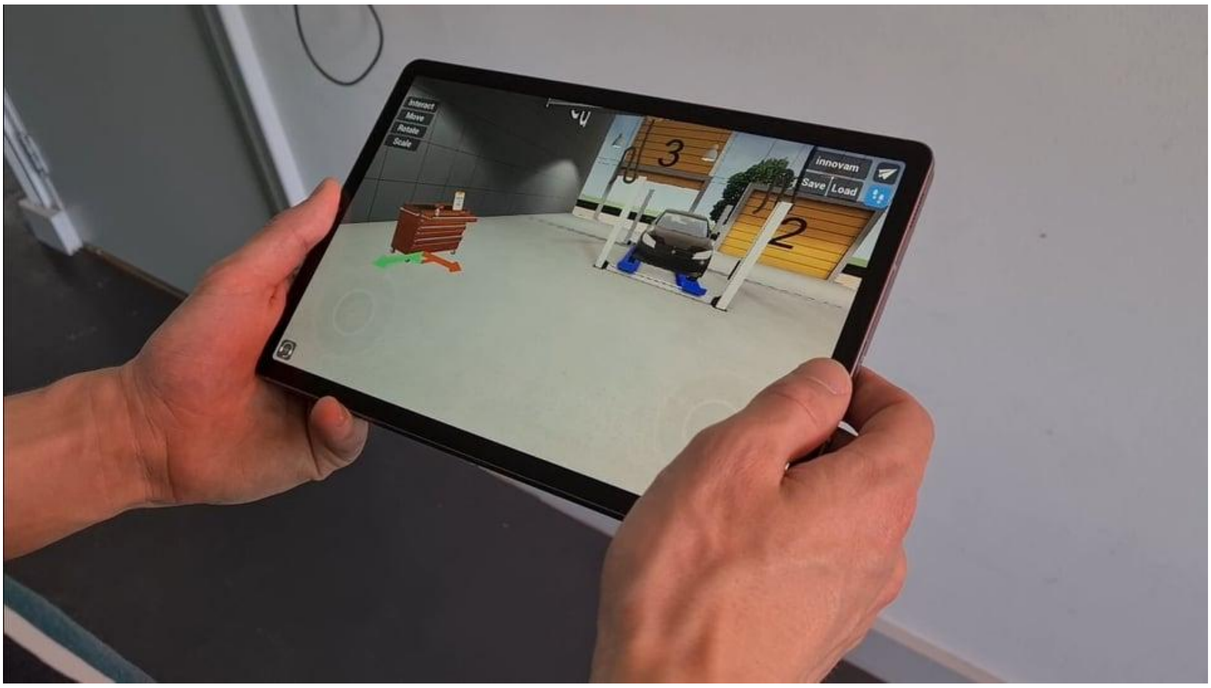


Figure 5. Screenshot presenting the interaction with external sessions using an Android device.

5.6 Exporting To IOS

Unreal Engine provides the guide ”Building for iOS on Windows” (Unreal Engine, 2021) for exporting to IOS in order for the developers to build on Apple mobile devices.

For this operation, a Mac computer has to install the application on the devices. It requires the latest version of Xcode (Apple Developer, 2021). Without a computer that is provided by Apple, the application will not be installed on these devices, due to security reasons. Both computers need to share the same local internet connection. The export for IOS devices was the most complex out of the platforms used for the project.

Unreal Engine requires two main files for the installation as presented in the article “IOS Provisioning” (Unreal Engine, 2021). The first one being the application certificate. The file contains an unique id that is given to the application by Apple. This certificate needs to be created using an Apple developer account.

The second file is represented by the provisioning file. This file is generated based on the certificate. It specifies on which devices the certificate can run. These devices must be manually registered by the developers on the Apple developer account.

Once the files are added to Unreal Engine, the computer will need to have a shared local network connection with the Apple computer. The installation will be done by making a copy of the project on the Apple computer, which will be later used in order to install the application on the mobile devices.

During this stage, the company provided assets for the application, as presented in Figure 6. The scene is well optimized and performs well on mobile devices. The interaction tool was also added, and It allowed the users to interact with the objects, for example, Figure 6 captures the opening of the door of the car. This action is also replicated through the network to the other devices present in the session. All of the other platforms: PC, VR and Android were able to participate at the same time.

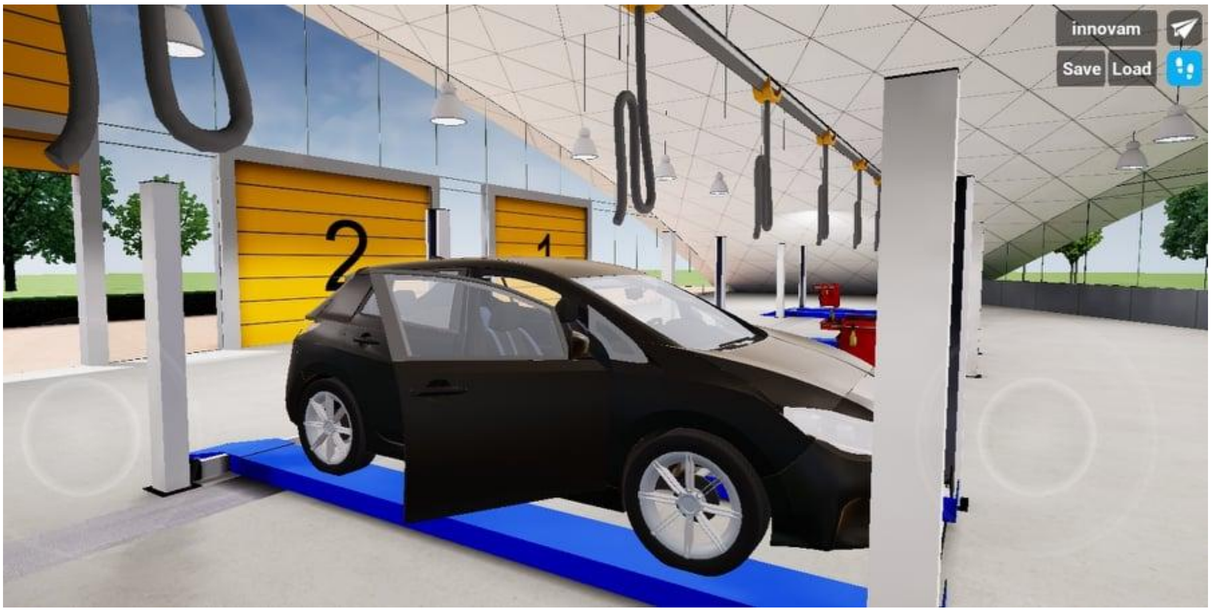


Figure 6. Screenshot presenting the integrated assets inside the application.

6. User Testing

6.1 Testing plan

The following goals have been set for the testing phase of the application. It contains tasks that the users need to complete and then give feedback based on their experience.

Goals	Tasks
Joining the session on the dedicated server	Downloading the application, entering a username, manually entering the IP of the dedicated server and entering the session
Moving inside the scene	Using the platform specific layouts in order to move around the scene
Loading objects layouts from the database	Entering the name of the scene through the user interface and loading a layout
Using the tools to move, rotate and scale objects	Using the tools in order to manipulate the objects in the scene depending on the used platform
Using the interaction tool in order to complete the presented scenario	Using the interaction functionality of the available platform in order to complete scenarios with the other participants

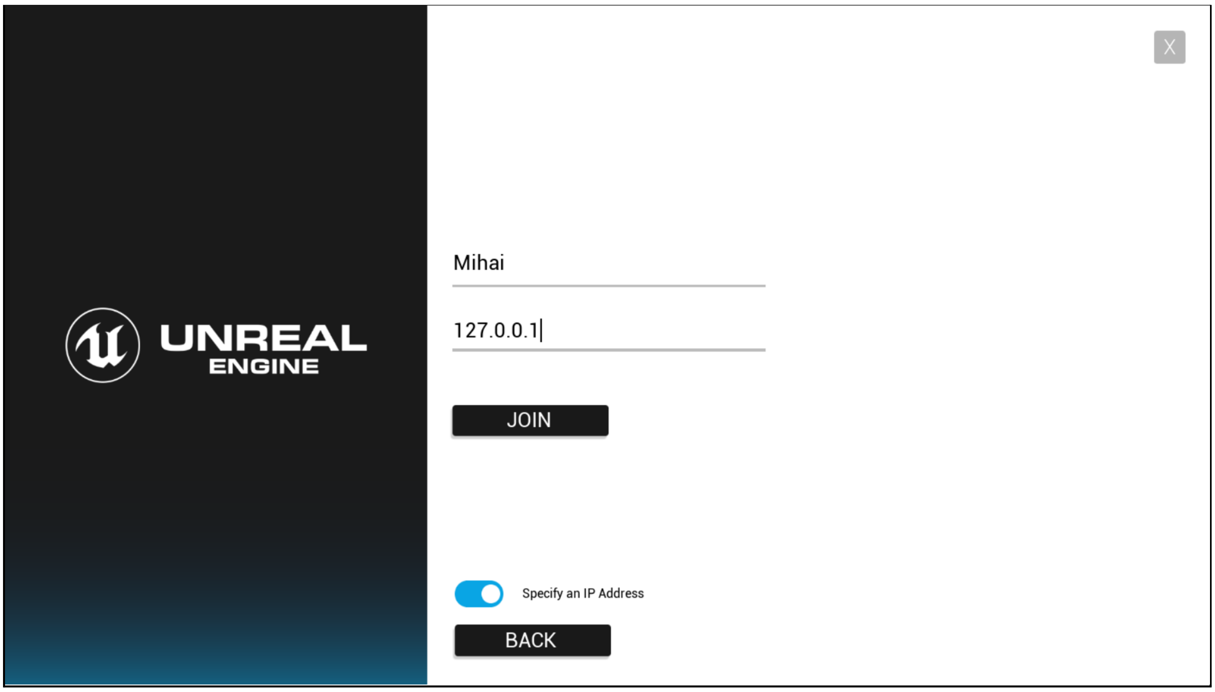
The test also consists of a survey in which the participants can grade their experiences based on various aspects of the application. They can also give written feedback and explain their opinions more in depth. The most important aspect of the test will be the observation of the participants. This will show more results on how comprehensive the tools are and what design choices need to be taken in order to improve the application.

The test had a total of nine participants. Seven of them were using the application on the dedicated server. Out of these participants, one used a VR set, two used an Android tablet and an IPad and the rest of participants used computers from their homes.

6.2 Testing results

Goal 1: Joining the session on the dedicated server

	Test Results
Overview of feedback	All of the participants managed to enter the session using the IP of the dedicated server as presented in Figure 7. Most of them requested help in order to enter the online session.
Problems Encountered	Without previous explanation and without a provided IP address, all of the participants would have not been able to join the dedicated server. The application could not be used independently.
Suggested features	The participants suggested an option that would automatically connect them to the server and allow them to enter the session without problems.


Mihai

127.0.0.1

Figure 7. Screenshot representing the user interface used for joining the session.

Conclusion

As presented in Figure 8, the participants believed that joining the server was efficient, but this was because they had been assisted and they managed to enter the session rapidly. By observing the testers, it was decided that the problem was more important than they believed. Not being able to join the server without guidance represented the biggest problem of the application. While a few were able to join without assistance, the application still had to be used by people without technical background.

Regarding the questionnaire, Figure 8 presents the results received after the test session. It can be observed that the application can use improvement in the areas such as audio quality and object manipulation. While the questionnaire received high grades for each category, the observation and direct communication with the testers managed to gather more information about what aspects of the application that had to be worked on.

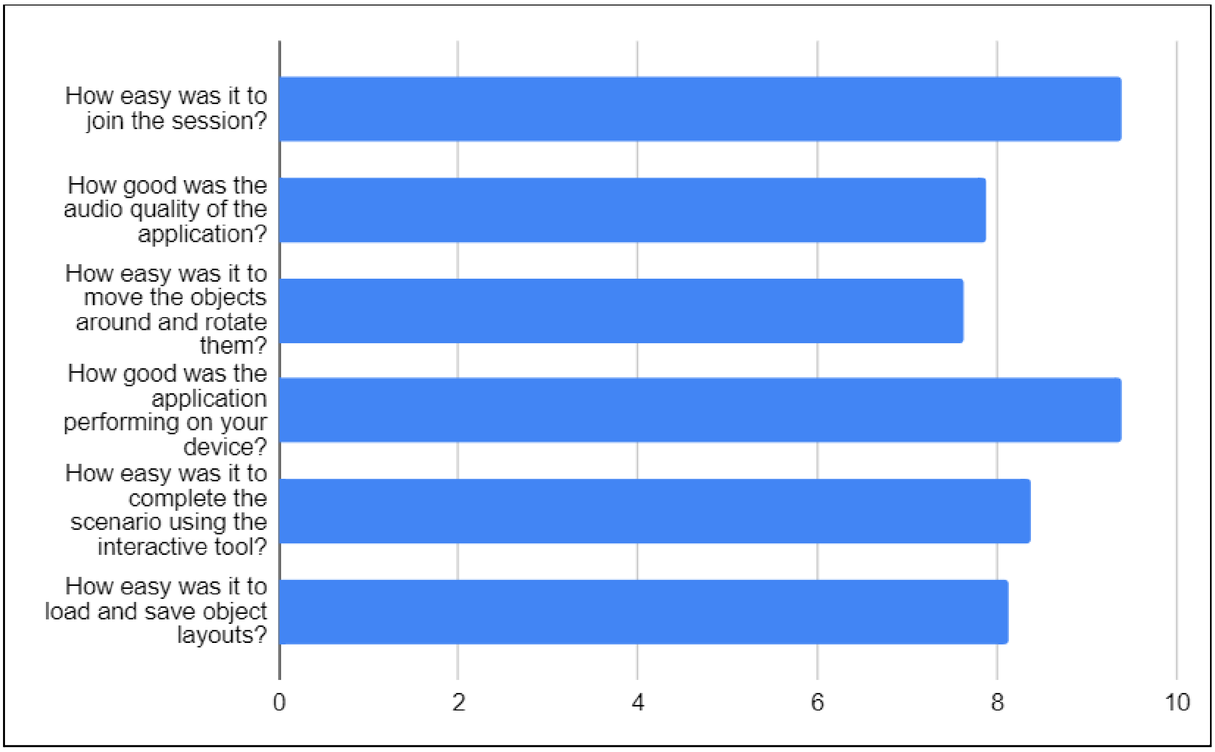


Figure 8. Table representing the averaged grades filled in the questionnaire by the participants.

Goal 2: Moving inside the scene

	Test Results
Overview of feedback	All of the participants managed to move through the scene and explore the surroundings. The user interface allows them to move between a flying mode, a walking mode and a VR mode. Two of the testers were from Romania and Scotland and they experienced a fast internet connection. They were impressed by the small networking delay and they were pleased with the speed of the server.
Problems Encountered	The mobile version of the application had the vertical axis reversed for looking up and down. With the basic controls the user could look up by sliding the right controller down, confusing the participants. People that did not have a VR set and entered the VR mode would get stuck and would have to restart the application. People would jump from the map and would continue to fall.

Suggested features	The mobile testers wanted the axis to be reversed as they could not get used to it. The majority of people that tested the application on the computer wanted the ability to jump, stating that they believe their mobility is limited. The testers also requested a way to go back to the PC mode after entering VR.

Conclusion

The mobile testers said that the reversed axis is the biggest problem of the application for them. After some research, the reversed axis is a standard layout for the old gaming consoles. People that grew up using them would have a hard time using a new type of axis. It was decided to use the normal axis in order to make the application more intuitive for people that did not use joysticks before.

The mobile testers did not like the ability to enter VR mode. This would make them unable to exit the mode and they requested the removal of the functionality.

Goal 3: Loading objects layouts from the database

	Test Results
Overview of feedback	All of the testers were able to load layouts form their devices. They enjoyed the response time of the server and how easy it was to save new layouts.
Problems Encountered	When people tried to load saves that did not exist on the database, It would still delete the objects in the scene.
Suggested features	All of the testers agreed that there should be a list of available save files that they can load, so that they do not have to type the names.

Conclusion

The testers were pleased with the speed of the system and the ease of operation, but they did not like that all of the participants had the same authority. While they were trying to use the objects in the scene, if another user tried to test the loading feature, all of the objects would get deleted. They would have preferred an authority system that would give the rights to only a few people.

Goal 4: Using the tools to move, rotate and scale objects

	Test Results
Overview of feedback	All of the participants managed to manipulate the objects on all of the devices tested. They were impressed by the speed at which it synchronized their actions over the network.
Problems Encountered	During the test session some participants that did not have any technical knowledge of the 3D programs did not manage to use the tools directly but they adapted easily.

Suggested features	The majority of the testers were able to use the tools directly, they said that it was very natural for them to use them. But at the same time they all stated that people outside the field would get confused by them.

Conclusion

The Gizmos system present in the application was well received as it presented familiar axis representation systems that they are used to. But at the same time the application should also give an alternative for the people that do not have technical knowledge.

Goal 5: Using the interaction tool in order to complete the presented scenario

	Test Results
Overview of feedback	In the scenario, the participants had to put the car in the parking mode, turn off the car, take the key out and put it in a box and mark the perimeter around the car in order to ensure safety. The people that used mobile devices found it difficult to use the tool compared to the PC version. They expected it and agreed that it will always be more difficult to operate from a mobile device. The VR participant was very pleased with the ease of interaction. It was stated that it was easier to use than the PC version because of the extra mobility provided by the VR set. Figure 9 presents the VR version of the application and shows how the participant used the interaction tool.
Problems Encountered	In the scenario, the participants had to put the key of the car in a box. They found it difficult especially on mobile devices. The objects had to overlap in order to complete the task.
Suggested features	The participants suggested that the collision perimeter for the box in scenario with the key, should be increased. They believe that it would allow the users to complete the task more easily and it would not break the realism of the scenario if it is modified to the right dimensions.

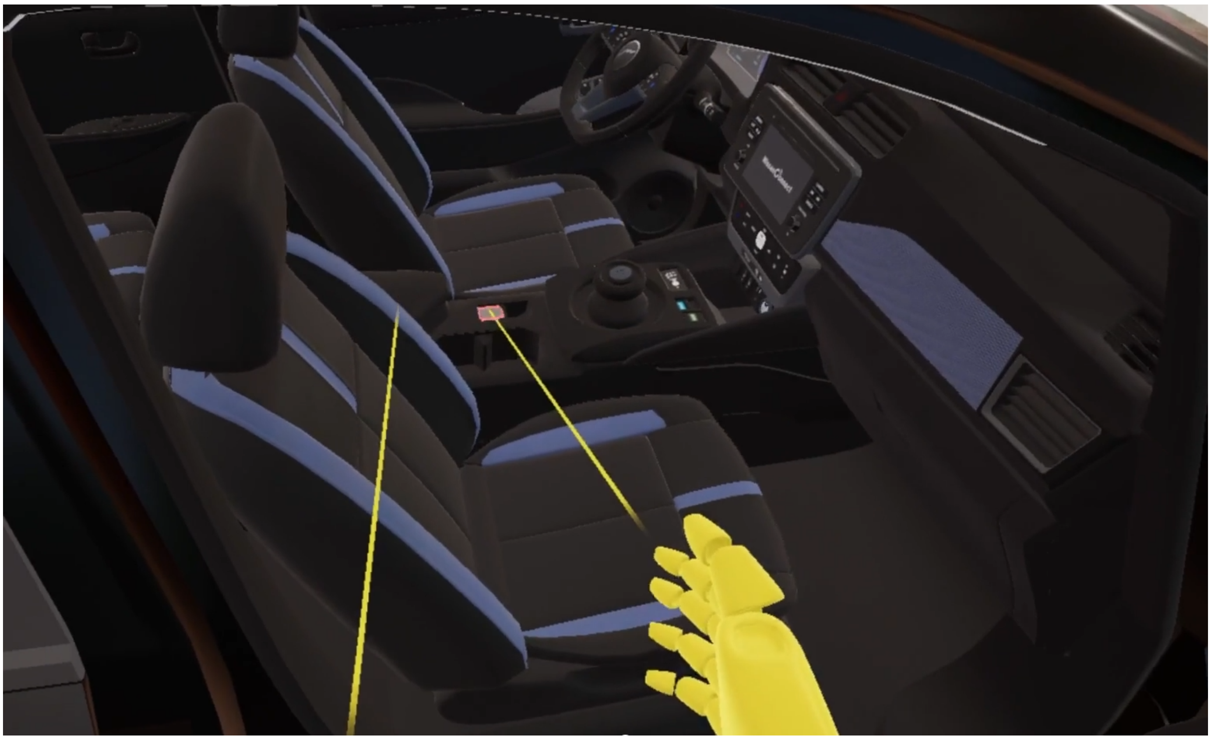


Figure 9. Screenshot representing the interaction tool used by the VR devices.

Conclusion
The participants enjoyed clearing the scenario and stated that they see the potential of the application. The mobile devices users found it more difficult to operate the scenario because of the movement system, due to using joysticks. They were still impressed by the ability to complete the scenario from a mobile device.

6.3 Implemented Feedback

After the feedback, the application received a number of changes. The most important change being the ability to join the dedicated server without manually entering the IP address. This ensures that people will be able to enter the session directly without requiring new additional information.

Another problem that was addressed was the reversed axis on the mobile devices. Because all of the mobile testers wanted this feature to be implemented, it was placed on the priority list. The movement system for mobile devices became easier to use with this change. Alongside this, if the users would fall from the map, they would be teleported to a safe spot in order to avoid the continuous fall.

The application received the ability to change between the movement modes using the keyboard. If a user would press the VR button, and they did not have a VR set, the application would get stuck and they would not be able to go back to the normal walking mode. After the change, the users can freely change between all of the modes using the keyboard. For the mobile devices, the VR button was removed because it provided no way to return to the previous walking mode.

7. Conclusion

The application was received with positivity according to the test sessions. The project represents a base on which scenarios can be built on. It can be extended into a more advanced architectural editing tool, event room furniture presentation and even a custom online training application that can be built in order to explain how specific elements work such as mechanical parts and electronic systems.

During the testing session, the participants would always point out that the synchronization of the sessions was unexpectedly fast and that the delay of actions was not noticeable. The furthest distance that the application was used was from Scotland to Romania, the response time was good and the participants did not experience any performance issues on their devices.

Computers can create local sessions and join other computers. Android devices can also create and join other Android devices. The computers cannot join Android devices and the other way around. The Apple devices cannot join any device over the local network. But all of the devices can join a dedicated server. Because of this, it was decided to focus the work on the dedicated server.

A video was created for the project in order to help promote the application (Mihai C., 2021). The video consists of all of the functionality present that can be used by the clients. It serves as a short demonstration of the application and it can help the company to show the potential of the product and attract more clients.

The application is not perfect and it can still be improved, more systems can be built in order to make the communication with the database easier. The mobile devices can receive a VR mode in which the users do not need to use the touch controls for navigation and just look around in their room. The user interface would benefit from a redesign to allow for a much better experience in the application.

All of the sub-questions have been answered and the selected solutions have been used for the project.

Sub-question 1: How to keep track of changes made in Unreal applications that are in production?

For keeping track of the changes created in the application, a database was used. The users can make their own layouts of objects and save them on a server. These layouts can later be loaded or overwritten depending on the preferences of the users.

Regarding the financial aspect of the application, the solution that benefited the company the most was chosen. The hosting website HostNet was chosen for the project. It decreases the costs by 90% compared to the current solution. Microsoft Azure costs the company 50 euros per month while HostNet costs only 5 euros per month. Because the project does not require the extra functionalities of Microsoft Azure, Hostnet is a better option as it provides the required functionality and costs less.

Sub-question 2: How to make the application work on different types of devices such as PC, VR and mobile phones?

As the PC version of the application is already used by Unreal Engine by default, the other platforms had to be added. For the VR version the plug-in for SteamVR was added. This allows multiple VR sets to be directly recognized by Unreal Engine and automatically assign the controls to them.

For the Android version, Android Studio was used in order to put the application on the mobile devices. After creating a store key for the application, Unreal Engine creates an installation package that can deploy the project on the Android devices connected by cable to the computer.

For Apple products such as IPads and IPhones, Xcode was used to create the installation packages for the devices. Unreal Engine requires the certificate of the application and the provisioning in order to deploy the project on the mobile devices.

Sub-question 3: What tools are necessary for editing environments inside built Unreal Engine projects?

The tools required for the applications were the ability to move, rotate and scale objects. The representation of these tools were created with the help of Gizmos. These are 3D representations of axis that people can use in order to manipulate objects. They have been received positively by the testers and they were able to use all of the tools. Another tool that was added was the interaction tool. With this, the users could complete scenarios that were present in the application and were able to interact with the environment.

Sub-question 4: What Unreal Engine features can be used in order to enhance interaction between the company and the clients?

A dedicated server was used in order to enhance the interaction between users. It allows for all the platforms such as PC, VR, Android and IOS to join in a synchronized session and interact with the environment. The response time of the server was positively received by the testers and the networking delay was not noticeable.

An important financial aspect of the application was represented by choosing the right dedicated server hosting website. Amazon Web Services presented the best option for the project. The pricing is done based on the usage of the server. If the server is not used, the website does not charge the company. This allows the company to save money and only spend it when needed. Amazon servers are also easy to use and this would ensure that the future developers will be able to continue the project without problems.

8. Recommendations

A feature that was requested by many testers was the ability to only hear the people that are present in a specific room. Because the current versions allow all of the users to hear each other regardless of the room that they are in, they would hear other people talking from other parts of the building. A simple system that would automatically mute the other users that are not in the room would enhance the experience provided by the application.

Currently, the application has a drawing mode that allows the users to paint in 3D space and drawing would be synchronized to the other devices. The current problem is that the painting is done in relation to a perpendicular plane relative to the direction to the camera of the user. In order to perfectly simulate a blackboard, a solution would be to use the normal of the collision point in order to orientate the drawing canvas in the right direction.

A login system would help the application give modification rights to the right people. The person with administrator rights should be able to give the same rights to other users that are present in the scene.

The users should be able to join multiple sessions instead of the current single session system that is used in the application. Multiple sessions should be started on different computers and the users should be able to join the desired one.

9. Appendix

9.1 References

Zwartlicht (n.d.) Zwartlicht Home Page, Retrieved from: https://zwartlicht.nl/

Lichtstorm (n.d.) Lichtstorm Home Page, Retrieved form: https://lichtstorm.com/

Unreal Engine (n.d.) Unreal Engine Home Page. Retrieved from: https://www.unrealengine.com/en-US/

Unreal Engine (n.d.) Actor Replication. Retrieved from: https://docs.unrealengine.com/en-US/InteractiveExperiences/Networking/Actors/index.html

Keith Stuart (2020, February 28) When up means down: why do so many video game players invert their controls?. Retrieved from: https://www.theguardian.com/games/2020/feb/28/why-do-video-game-players-invert-the-controls

Microsoft (n.d.) Microsoft Azure. Retrieved from: https://azure.microsoft.com/en-us/

MySql (n.d.) Getting Started with MySQL. Retrieved from: https://dev.mysql.com/doc/mysql-getting-started/en/

MySql (n.d.) MySQL Connector/C++. Retrieved from: https://dev.mysql.com/doc/dev/connector-cpp/8.0/

David Y.(2016, November 23) Making HTTP Calls from Unreal Engine 4. Retrieved form: https://www.davidykay.com/UE4-Hello-Http/

Vladimir Alyamkin (n.d.) VaRest. Retrieved from: https://alyamkin.com/projects/nWE2E

Unreal Engine (n.d.) Developing for SteamVR. Retrieved from: https://docs.unrealengine.com/en-US/SharingAndReleasing/XRDevelopment/VR/SteamVR/index.html

Steam (n.d.) SteamVR. Retrieved from: https://store.steampowered.com/app/250820/SteamVR/

Juan Marcelo (2020, May 15) Runtime Transformer Tool (Gizmo). Retrieved from: https://www.unrealengine.com/marketplace/en-US/product/runtime-transformer-tool-gizmo

Ditte H. (n.d.) How to Create an Intuitive Design. Retrieved from: https://www.interaction-design.org/literature/article/how-to-create-an-intuitive-design

AutoDesk (2020, Mar 30) About Using 3D Gizmos. Retrieved from: https://knowledge.autodesk.com/support/autocad/learn-explore/caas/CloudHelp/cloudhelp/2019/ENU/AutoCAD-Core/files/GUID-7BD066C9-31BA-4D47-8064-2F9CF268FA15-htm.html

Cloudflare (n.d.) What is UDP?. Retrieved from: https://www.cloudflare.com/learning/ddos/glossary/user-datagram-protocol-udp/

Cloudflare (n.d.) What is TCP/IP?. Retrieved from: https://www.cloudflare.com/learning/ddos/glossary/tcp-ip

ENet (n.d.) Enet Home Page. Retrieved from: http://enet.bespin.org/

Libevent (n.d.) Libevent Home Page. Retrieved from: https://libevent.org/

Unreal Engine (n.d.) Collab Viewer Template. Retrieved from: https://docs.unrealengine.com/en-US/Resources/Templates/CollabViewer/index.html

Unreal Engine (n.d.) Blueprint Visual Scripting retrieved from: https://docs.unrealengine.com/en-US/ProgrammingAndScripting/Blueprints/index.html

Unreal Engine (n.d.) Programming with C++. Retrieved from: https://docs.unrealengine.com/en-US/ProgrammingAndScripting/ProgrammingWithCPP/index.html

Unreal Engine (n.d.) Setting Up Dedicated Servers. Retrieved from: https://docs.unrealengine.com/en-US/InteractiveExperiences/Networking/HowTo/DedicatedServers/index.html

Amazon (n.d.) Amazon Web Servers. Retrieved from: https://aws.amazon.com/

Microsoft (n.d.) Azure Gaming Retrieved form: https://azure.microsoft.com/en-us/solutions/gaming/

Transip (n.d.) Transip VPS. Retrieved from: https://www.transip.nl/vps/

GitHub (n.d.) Epic Games Repository. Retrieved form: https://github.com/EpicGames

PHP (n.d.) JSON Functions. Retrieved from: https://www.php.net/manual/en/function.json-decode

Hostnet (n.d.) Hostnet Academy. Retrieved from: https://academy.hostnet.nl/

Jonathan H. (2019, April 4) UE4: How to set up LAN-Multiplayer for Windows and Android. Retrieved from: https://blog.squareys.de/ue4-android-windows-lan-multiplayer/

Unreal Engine (n.d.) Android Quick Start. Retrieved form: https://docs.unrealengine.com/en-US/SharingAndReleasing/Mobile/Android/GettingStarted/index.html

Unreal Engine (n.d.) Setting Up Android SDK and NDK for Unreal. Retrieved from: https://docs.unrealengine.com/en-US/SharingAndReleasing/Mobile/Android/Setup/AndroidStudio/index.html

Android Studio (n.d.) Android Studio Developers. Retrieved from: https://developer.android.com/studio

Unreal Engine (n.d.) Signing Projects For Release, Retrieved from: https://docs.unrealengine.com/en-US/SharingAndReleasing/Mobile/Android/DistributionSigning/index.html

Android (n.d.) APK Expansion Files. Retrieved from: https://developer.android.com/google/play/expansion-files

Unreal Engine (n.d.) Building for iOS on Windows. Retrieved from: https://docs.unrealengine.com/en-US/SharingAndReleasing/Mobile/iOS/Windows/index.html

Unreal Engine (n.d.) IOS Provisioning. Retrieved from: https://docs.unrealengine.com/en-US/SharingAndReleasing/Mobile/iOS/Provisioning/index.html

Xcode (n.d.) Apple Developer. Retrieved from: https://developer.apple.com/xcode/

Mihai C. (2021, June 9) Graduation Video. Retrieved from: https://video.saxion.nl/media/446505MihaiConstantinescu/1_dn92be04

9.2 Final Product Demonstration

https://video.saxion.nl/media/446505MihaiConstantinescu/1_dn92be04

9.3 Reflection

I. Technological competences
1. Technical research and analysis
From the start of the project I had to research a lot of materials for the systems that I wanted to use for the creation of the application. The documentation of Unreal Engine covers very few aspects of the program. This made me look for information in a lot of places ranging from articles to tutorials on various websites. A lot of research was needed for the creation of the database communication present in the application. I had to make the right decision and pick the most suitable plug-in for the application. For the editing of the scenes I had to research intuitive ways in which the users can interact with the environment. Another important part of the research is represented by learning how to export the application to mobile devices. This proved to be the most difficult part of the project, but the research materials have helped me reach my goals.
2. Designing, prototyping and realizing
During the creation of the application I have designed and planned how the users would be able to interact with the application without the need of technical knowledge, making the tools as intuitive as possible. For any major feature that was added, I made sure to integrate it in the user interface and make it easy to understand and use. For every iteration I would make a prototype and test it on all of the platforms in order to search for problems, which were later fixed.
3. Testing and rolling out
With the addition of the tools required for editing the scene, I have managed to test the application and check for problems that were not visible before. This allowed me to make sure that the application works properly and does not present problems. After the dedicated server for the application was added, the testing became easier. The application could be used by multiple people around the world, making the problems easy to spot. With all of the feedback gathered I have applied the requested features and made sure that the issues were solved.

II. Designing competences

4. Investigating and analyzing

An important decision that helped the production of the application was the decision on the communication system that can be used for sending and receiving data from the database. VaRest, the plug-in that was used in the application, provided the best solution for the project.

Another example was the number of tutorials present for exporting the project to Android. The most important factor for finding the right source was the relevance of the content. Because the new versions of Unreal Engine can change a lot from one iteration to another, I had to make sure that the materials were not outdated.

5. Conceptualizing

After deciding on the features of the application, I have integrated them so that they are easy to interact with. All of the versions of the application: PC, VR and mobile, have a similar user interface and functionality, making switching between platforms intuitive. For the editing tools I chose to use Gizmos in order to represent the axis in 3D space. Because the majority of engines and 3D programs use this in their applications, I have decided to apply the same system to my project. The Gizmos for movement, rotation and scaling are easy to recognize and the users can interact with the environment directly without the need to learn a new editing style.

6. Designing

After the decisions were made, I came up with a user interface design that would be similar for computers and mobile devices. Because the touchscreen functionality on the mobile devices acts similarly to a click on computers, this allowed me to create shared functionality for both platforms. After feedback was received from the company members and clients regarding the user interface, the issues were solved and new versions were created.

III. Organizational competences

7. Enterprising attitude

With the addition of each feature I made sure that the functionality is intuitive. All of the versions of the application: PC, VR and mobile devices, use similar functionality that makes commuting between platforms easy. For the user interface I have kept it as simple as possible. The navigation of the user interface is done through the smallest amounts of clicks. I believe that an application that allows the audience to reach a tool very fast, improves the experience of the user and their efficiency in finishing their task.

For the financial aspect of the project, I have chosen the option that will benefit the company the most. For example, by choosing to use the hosting website HostNet, the costs for maintaining the database were reduced by 90%. The hosting server that the company used was from Microsoft Azure, costing 50 euros per month, while the HostNet server costs only 5 euros per month and it is easier to operate.

I have created a video in order to help promote the project. With this video the company can use it in presentations with their clients and to attract more people that are interested in this technology.

8. Enterprising skills

For the majority of the features of the application I have come up with suggestions that can make the experience of the users better. I have also initiated conversation in order to clarify the direction in which the project was going. After talking about the new ideas or features, we made decisions regarding the application. Later I would work on the project corresponding to the planification.

9. Working in a project-based way

During the creation of the application I have discussed weekly with the members of the company and made sure that the project is going in the right direction. All of the points of the project were discussed together. During these meetings I received a lot of feedback and recommendations that I would later be applied to the project.

10. Communication

After managing to make the application run on a dedicated server, I presented it to the company and gathered a lot of feedback and recommendations that were later added to the final product. After more features were added, I held a test session with the clients and made sure that the application progressed in the right direction by making the changes they requested to the application.

IV. Professional competences

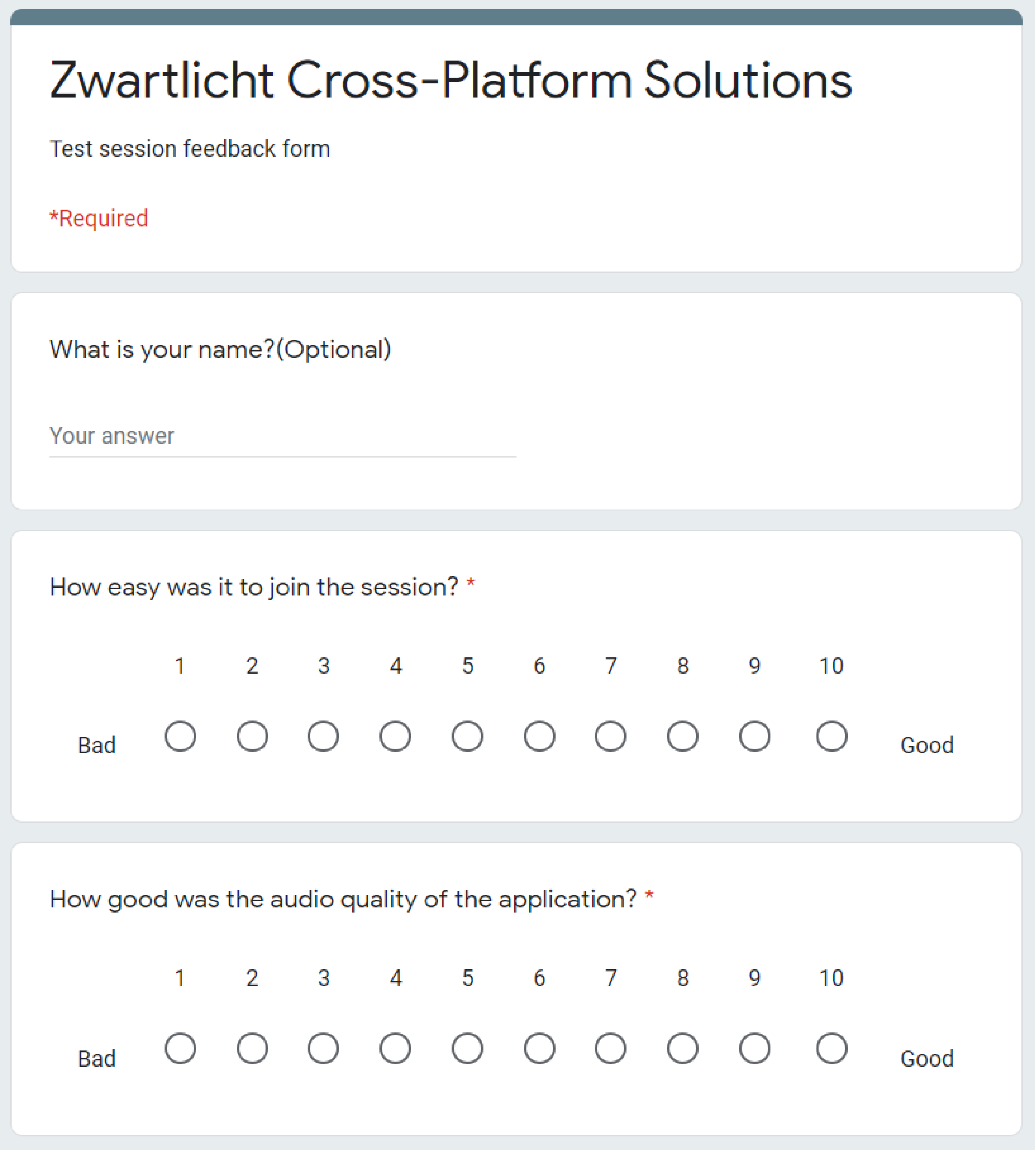
11. Learning ability and reflectivity

After managing to export the application on Android, this allowed me to hold a large test session with the members of the company. Out of all the platforms, I have received the most feedback for the mobile devices. The controls had to be changed and some more functionality had to be added. With all of the information gathered I have selected the most important aspects that need to be addressed and then I have applied them.

12. Responsibility

The most important goal of the project was to make the project easy to work with for future iteration of the application. As I was responsible for the architectural choices I made sure to make the structure of the application easy to understand and follow. For each major feature that was added, I have integrated it in a way that the content can be extended on and modified.

9.4 Questionnaire



Zwartlicht Cross-Platform Solutions

Test session feedback form

*Required

What is your name?(Optional)

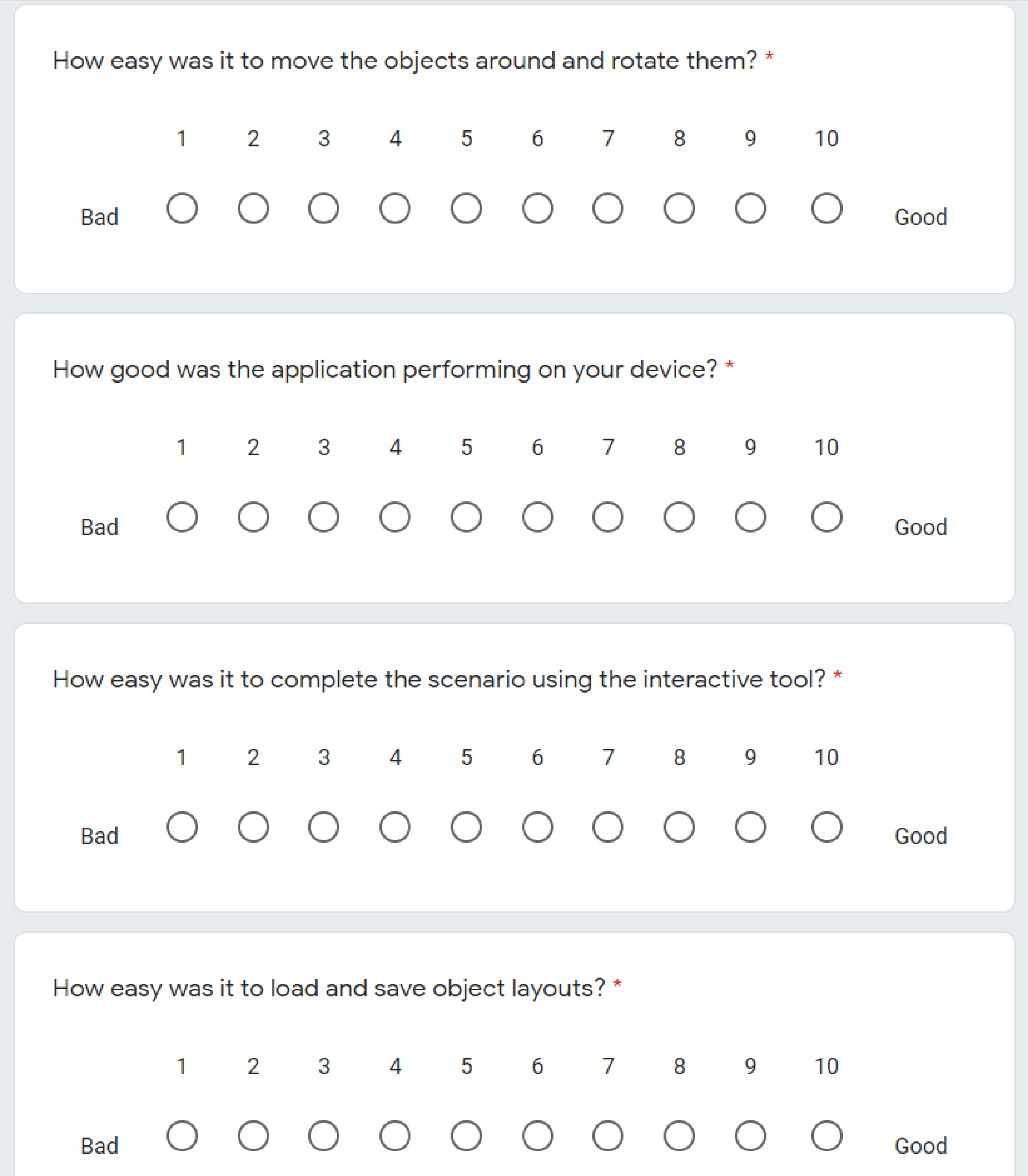
Your answer

How easy was it to join the session? *

	1	2	3	4	5	6	7	8	9	10	
Bad	○	○	○	○	○	○	○	○	○	○	Good

How good was the audio quality of the application? *

	1	2	3	4	5	6	7	8	9	10	
Bad	○	○	○	○	○	○	○	○	○	○	Good



How easy was it to move the objects around and rotate them? *

	1	2	3	4	5	6	7	8	9	10	
Bad	○	○	○	○	○	○	○	○	○	○	Good

How good was the application performing on your device? *

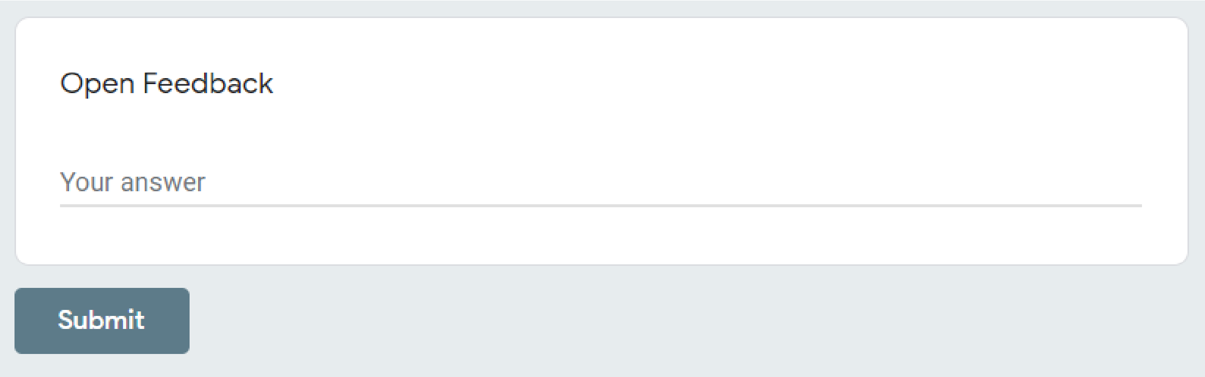
	1	2	3	4	5	6	7	8	9	10	
Bad	○	○	○	○	○	○	○	○	○	○	Good

How easy was it to complete the scenario using the interactive tool? *

	1	2	3	4	5	6	7	8	9	10	
Bad	○	○	○	○	○	○	○	○	○	○	Good

How easy was it to load and save object layouts? *

	1	2	3	4	5	6	7	8	9	10	
Bad	○	○	○	○	○	○	○	○	○	○	Good



Open Feedback

Your answer