

New Technologies



Size Matters.

Index

- Intro
- Team composition & tasks
- Subjects
- Used technology
- The 2 experiments
- The problems
- The solutions
- PMIs
- Links & Build

Introduction

Our goal is to create an experiment within 7 weeks where the player shall experience the feeling of growing and shrinking in real time while playing in a puzzle environment. One important aspect in this concept is to try and not making people feel nauseated while growing or shrinking.

We will do this to test out what players reactions are to shrinking and growing in a realistic manner, because of the virtual reality we will be using. A source of inspiration for us was Alice in Wonderland.

We will do this using the Oculus Rift & Touch inside Unreal Engine. Both are tools we are all completely new at.

The Team

Our team consists of 6 students:

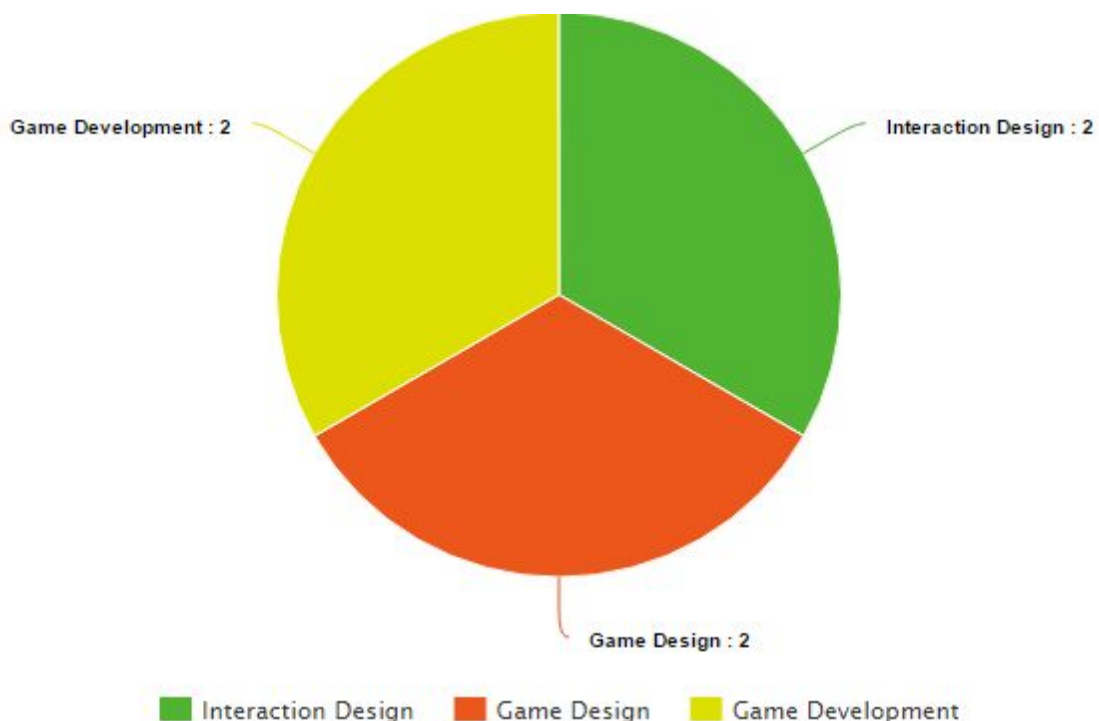
- Davey Verhoef
- Tim Hendriksen
- Nadia Groenewald
- Gijs Bakker
- Jesse van Vliet
- Rick Ruiten

Each individual has applied for the choice module: "New technologies".

Davey & Nadia will actively look for puzzle elements within our game to make the player interact in an environment completely set up for our experiment.

Tim & Gijs will design, shape and create the environment of our experiment - a futuristic style and implement this in Unreal Engine with beautiful lighting.

Jesse is the developer in the project who will create the game mechanics in Unreal Engine and the main operator for the Oculus Rift during the project.



Subjects

We have been looking into 2 different subjects for our assignment.

As main focus subject we're going to look into scaling a player in real time and doing tasks based on the player's height. The player will have to scale him/herself during a play session and we want to achieve the feeling of this being a "natural" thing towards the player.

The second subject we're going to look into is player movement, right now the current main way of navigation in Virtual Reality is by using a teleportation mechanic which quickly fades your screen to make sure you don't get nauseated but we're going to try to find a secondary way if possible.

Technology

For software to implement our game mechanics some members of our team have been learning the basics of Unreal Engine. We are totally new to the game engine and had to learn everything from scratch.

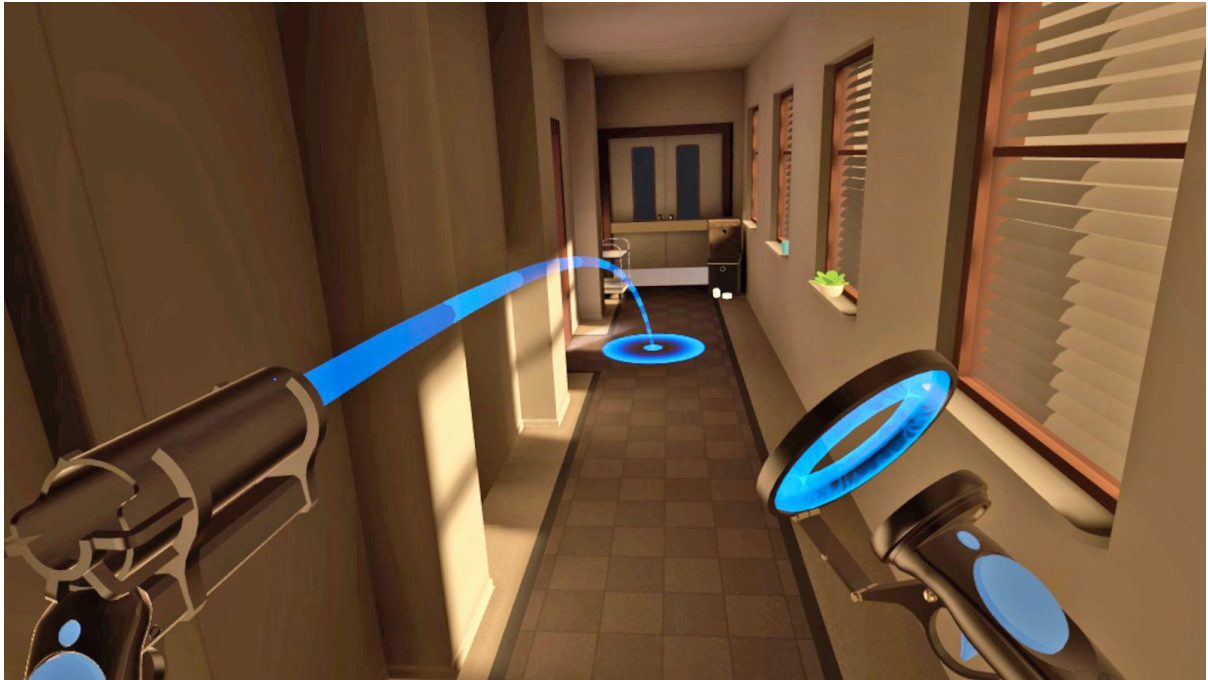
To program the features which were necessary to conduct our experiment most work has been done in the integrated blueprints workflow from UE rather than C++.

For the hardware side we're looking into the Oculus Rift with motion controllers as our virtual reality headset. This was the newest consumer edition of the OVR.

Experiments

Experiment 1: Finding a new way to navigate in Virtual Reality.

We wanted to search for a new method to move in the Virtual Reality space. Currently the only way of (long distance) movement in Virtual Reality is with a teleport system (as seen below)



We wanted to find a different, and maybe better way of moving around. Because we found no reason why our character would own such a teleporting device.

Experiment 2: Scaling the player in Virtual Reality

We wanted to find out how we can grow and shrink a player live while being in motion and make this feel natural to the player, not getting nauseated.

To do this we've created a simple puzzle the player has to complete. In this puzzle the player requires to grow and shrink to reach the end.

We've also made sure there is a distinct environment for the player so it will feel more lifelike and realistic to give the player a sense of proportion.

Challenges & Solutions

Conceptual Challenges:

Challenge:

We were very much searching for a concept.

Solution:

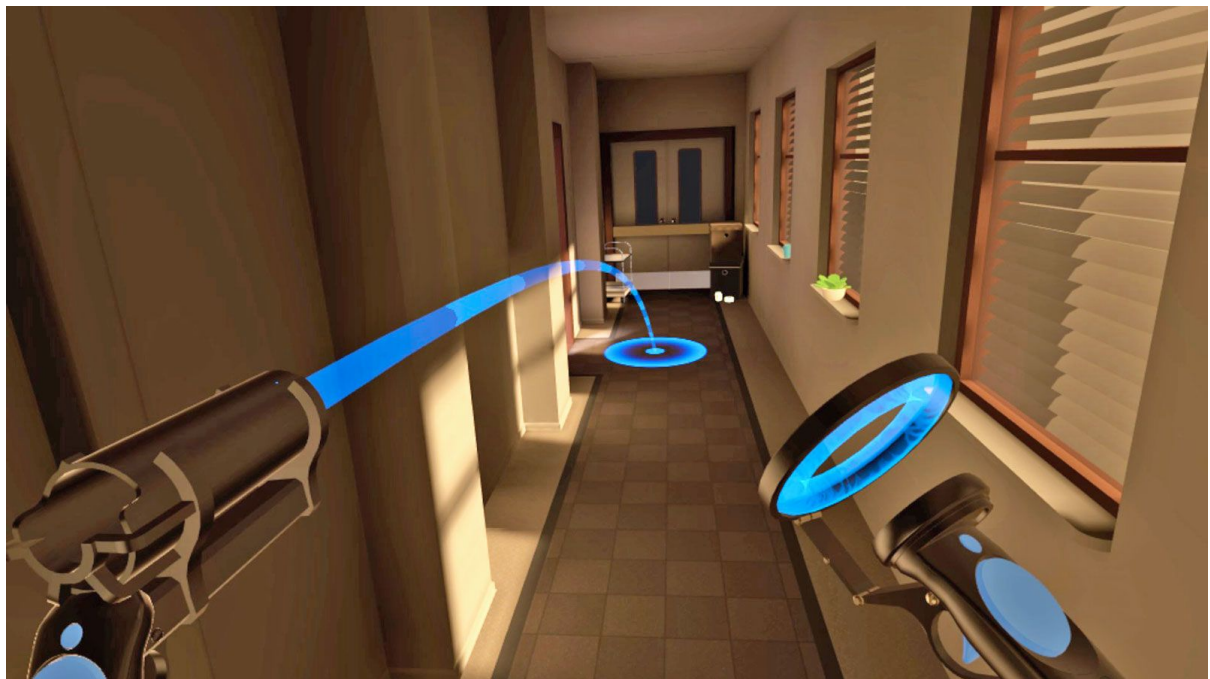
We started looking at what already was done and what would be possible in vr. This was a good start to work with. After that we had an idea that you would be a crane operator. We could play with height and make the player feel uncomfortable. This was already done several times tho. At last we ended up with scaling. Scaling would be our main mechanic and we would base our complete game on it.

Challenge:

If we would use the room scaling option, this would mean that you can walk around, then we would face the problem of physical room. If you would make yourself small end walk to the end of the physical room and become big again, you wouldn't be able to move to the rest of the digital room.

Solution:

Teleportation. A lot of vr games are already doing this and as far as we could think this was the best solution for our game.



Unreal Engine related Challenges:

Challenge:

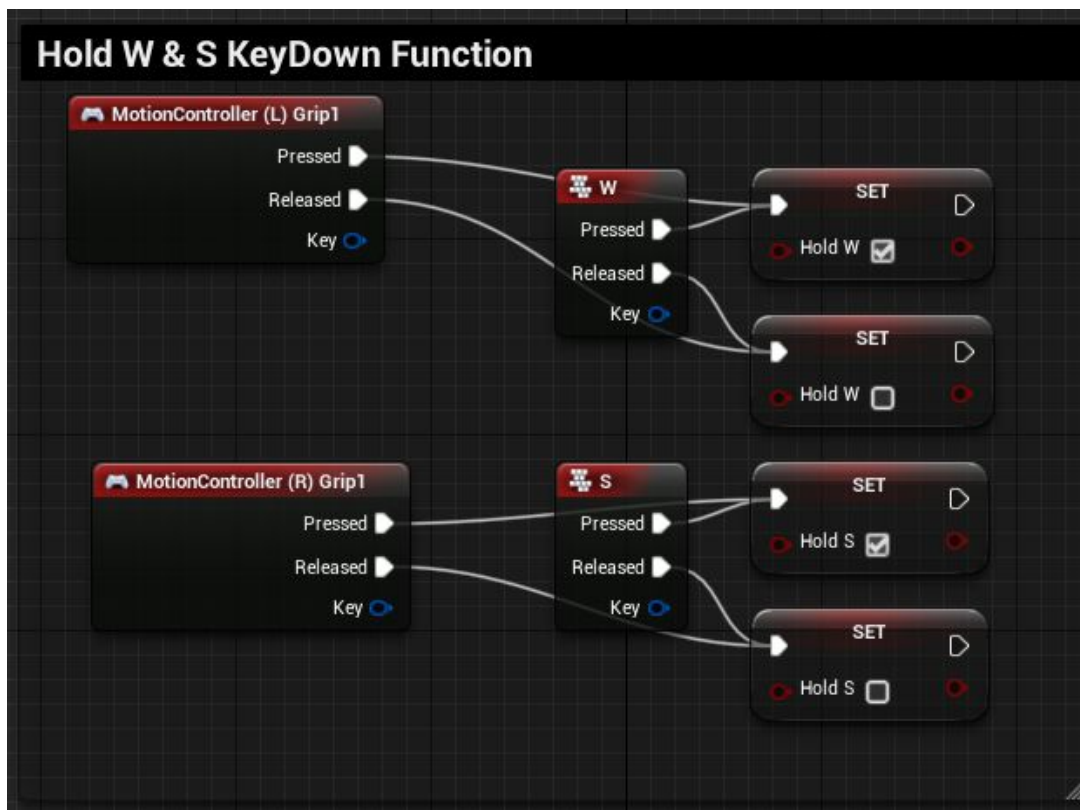
Unreal Engine has quite a complex editor, a lot of features but they are also very hidden every now and then.

Solution: A lot of time has to be spent in the Unreal Editor itself making the focus on our experiment take much longer than expected initially.

Challenge:

There are no “On Key Down” events in Unreal Engine.

Solution: Had to create a custom blueprint to set a bool on true on key press and once it was released return it to false.



Challenge:

Unreal Engine can't see a VR player during runtime unlike Unity where you'll see a camera in the "Scene" view. You can however see what happens in the object hierarchy so it gives you some form of information.

Solution:

There isn't really a solution to this issue as it is an unreal engine feature. You can however see what is happening to a player if you open the blueprint screens on another monitor or in a window on the side of your editor. The blueprints also have their own custom logs where you can follow everything.

Apart from that you can also semi see what is happening in the scene based on the object hierarchy. Such as a motion controller grabbing a child object.

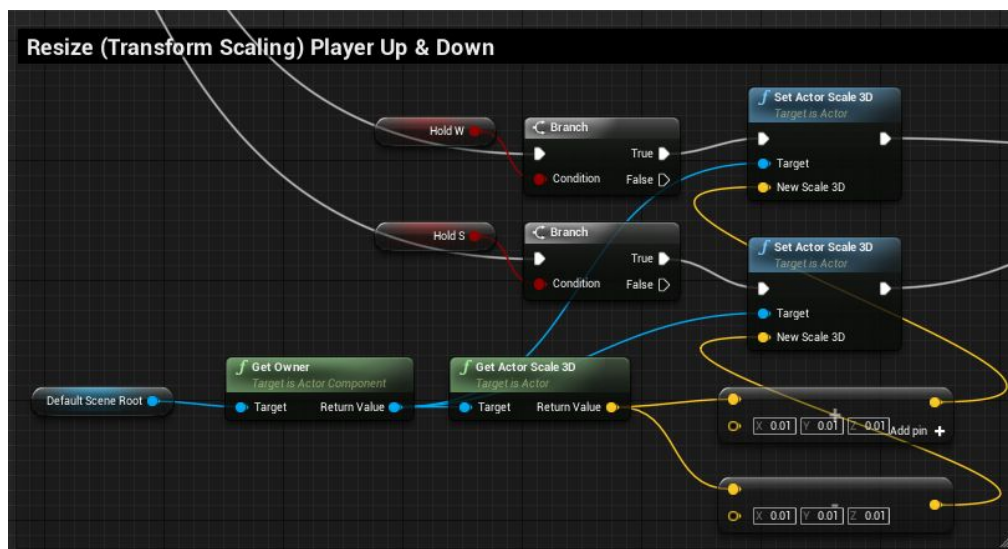
Challenge:

Scaling a VR player isn't just changing a scale slider.

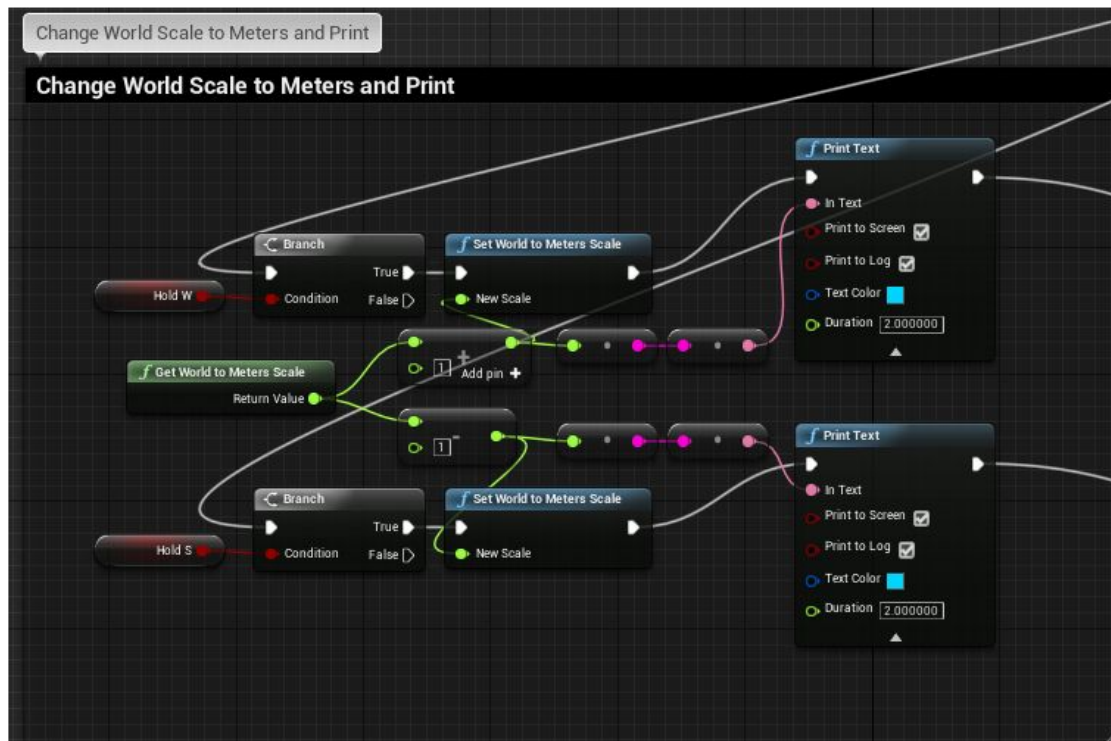
Solution:

To get to this point where we scale the VR player properly I went through changing the scale sliders which messed up the camera completely and made the player get enormous motion controllers/hitboxes. A second attempt was moving the camera up and down based on scale while also adjusting the FOV of the player. However this felt really unnatural and I'd have to make a custom blueprint to move the player hands accordingly with your body size too.

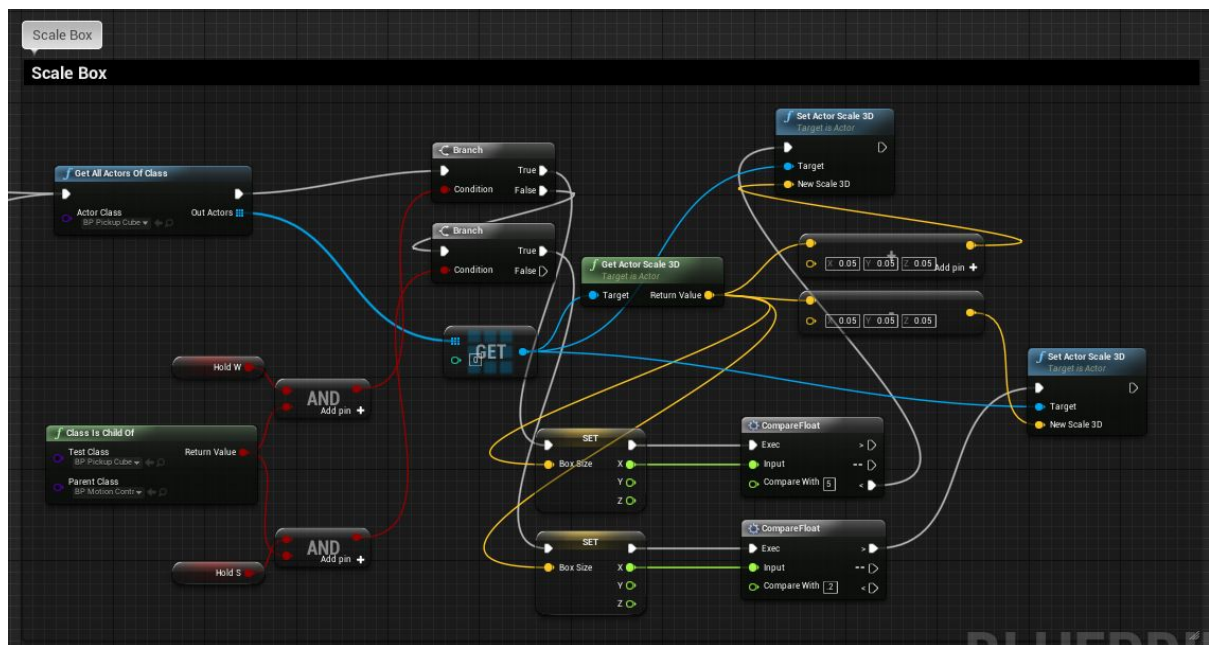
Finally after some attempts with different things there is a World to Meters scale in Unreal Engine which is basically changing your "OVR setup length" in real time, this could be adjusted and it felt super natural to scale the player suddenly.



This blueprint is used to scale the hands only right now.



This blueprint is used to set the world to meters scale smaller and larger.



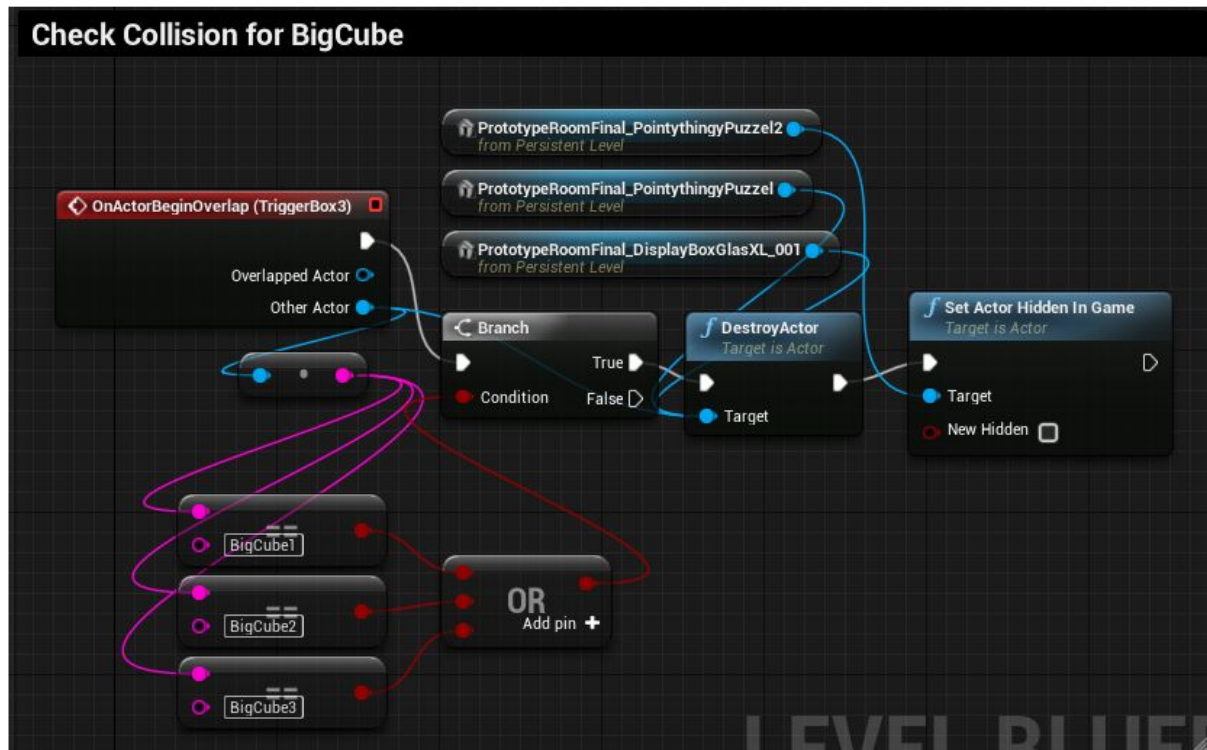
And as last this makes sure the object your motion controller has as child can't resize to be too small to interact with and not too large to interact with.

Challenge:

Triggers and checking the collider based on name string.

Solution:

So apparently if there are 3 objects with the exact same name and you check within a box trigger the "other actor" based on object name it will only work for one of the 3 objects. Which is why this required 3 different "BigCubes", I'll probably have to make an array of objects it should accept in the future - or perhaps based on tags.



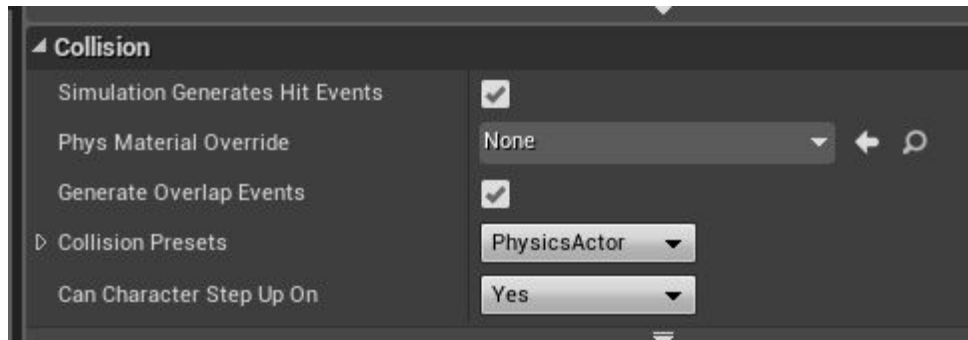
This was pretty *triggering*.

Challenge:

Triggers don't work.

Solution:

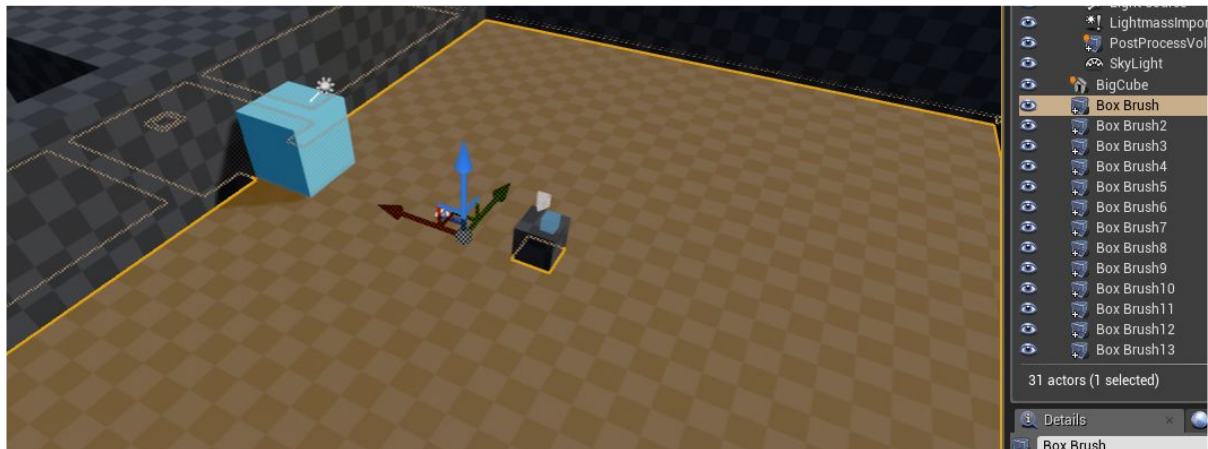
There are a million physics variables and things to set up in the unreal editor itself.



You have to make sure your Collision preset is ok and "Generate Overlap Events" is turned on.

Challenge:

Box Brush floors are used to teleport in Virtual Reality.



Everything else doesn't work.

Solution:

For now we've been using Box Brush floors to teleport onto but they are hidden within the game itself.

Challenge:

Collisions with custom meshes break nearly everything physics based in VR.

Solution:

Turning off the collisions of custom meshes and using Box Brushes for now as wall collisions.

Oculus Rift related Challenges:

Challenge:

Doing anything Oculus related after shutting down Unreal Engine is not really possible. It will mention something oculus related is still running.

Solution:

Restarting the computer always works but it's pretty time consuming of course, so far I have tried killing all unreal related processes and restarting the oculus app but unfortunately it doesn't work.

Challenge:

Setting up the Oculus rift is time consuming and necessary every single time because it keeps forgetting settings, maybe because it is a public oculus rift?

Solution:

The virtual reality room with a standard Oculus/IR camera's setup, making sure the camera's can't be moved.

Challenge:

Oculus their virtual reality application is always running and starting whenever you open Unreal Engine - even when it's not a VR related project.

Solution:

If you're not working with VR applications for a longer period of time you can just get rid of the application and download it again when you're going to work with VR.

Art Challenges:

Challenge:

We really had to think about the scaling of the rooms.

Solution:

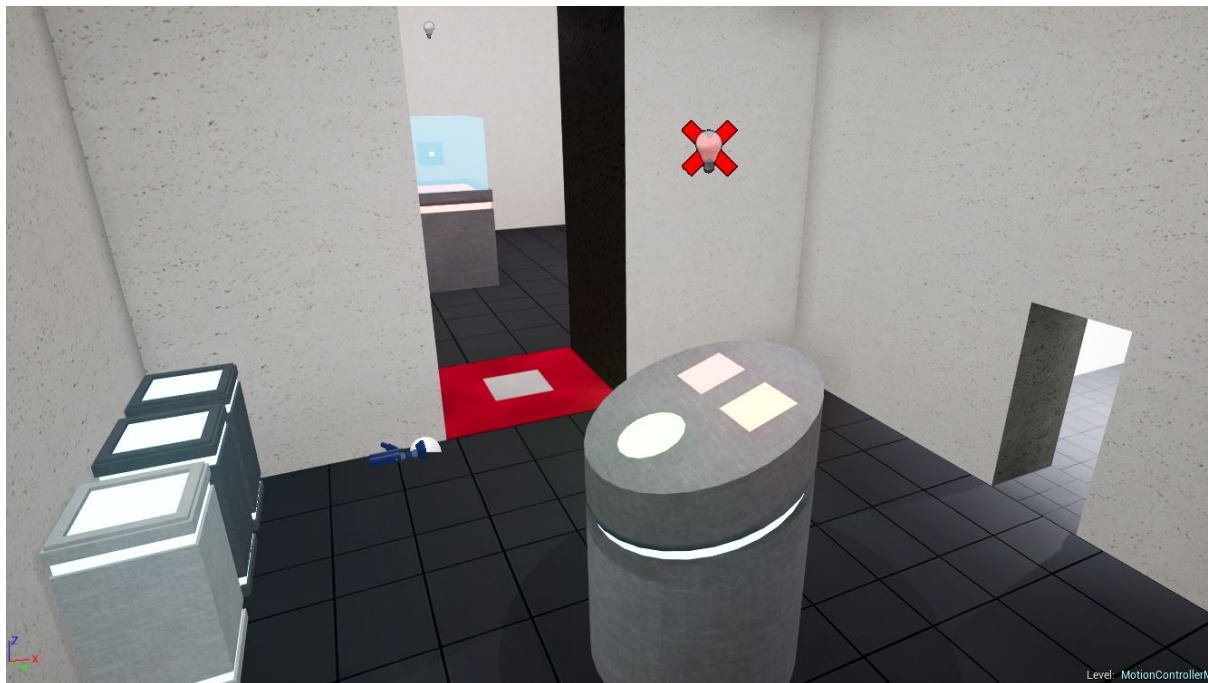
We decided to make every object at one scale at first. This was our base scale. After this we could easily scale the objects until they would fit to the right rooms. We could always use one object as starting position.

Challenge:

We wanted to give the rooms the look and feel of the game Mirror's Edge. This would mean much shiny floors and glass.

Solution:

Windows were not an option. This would mean that we had to make a scene outside our room. With lack of time and much other projects to do we had to make the decision that our room wouldn't have any windows. Instead of windows we decided to use glass in the objects we made.

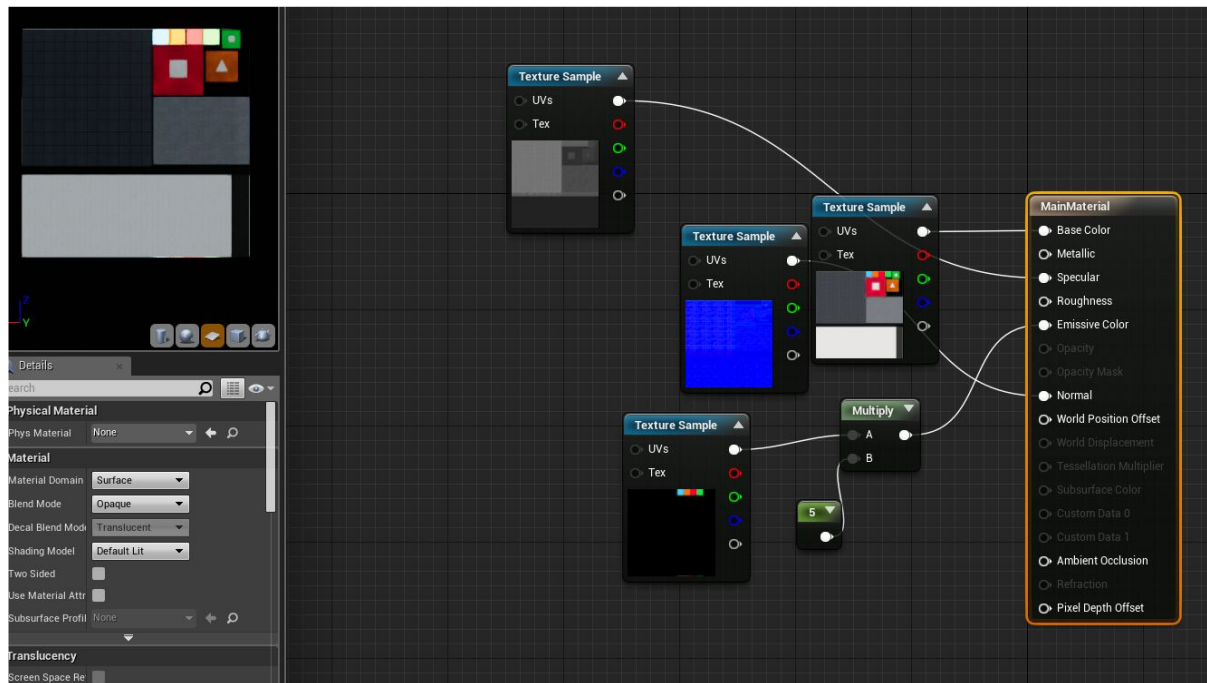


Challenge:

The texturing of the floor would be a big challenge. Because we are scaling a room we needed the textures to stay in the same resolution. Otherwise it would distort.

Solution:

As you can see we chose for the option that the floor has the same resolution in every room. This was the most aesthetically pleasing solution. To give the room some realism we used diffuse, normal, specular and emission maps.

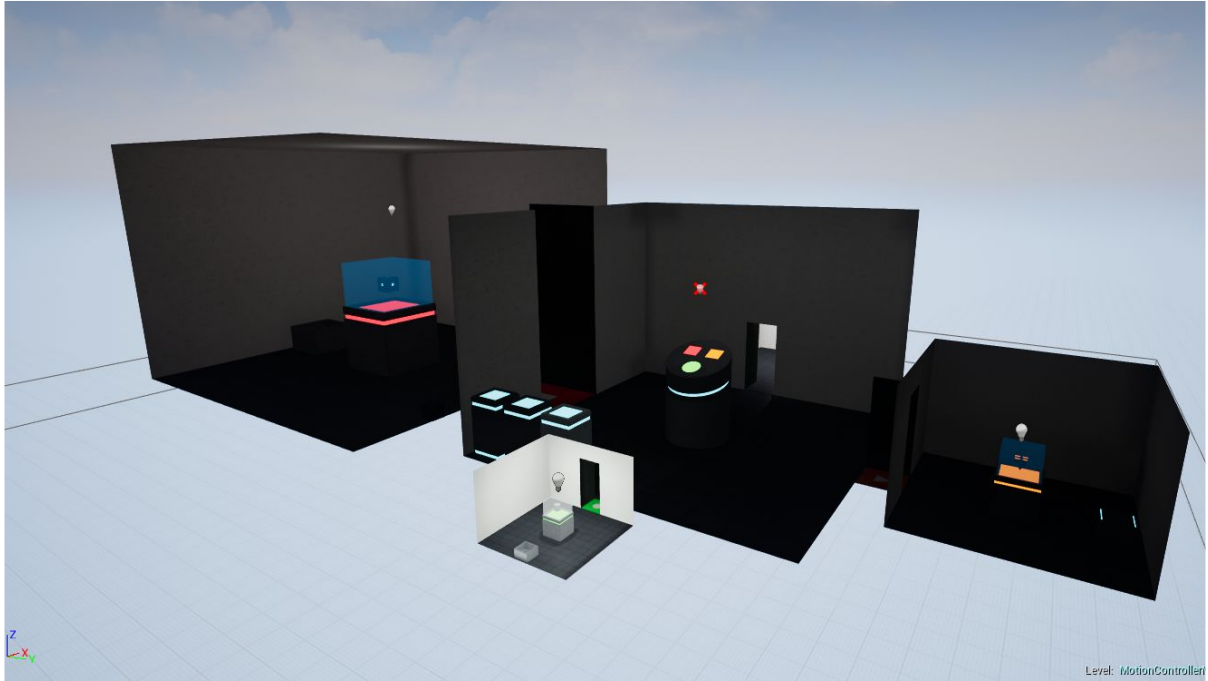


Challenge:

How can we make the goal of the game clear for players?

Solution:

We used color coding in our objects to make it clear that these objects needed to be used to solve the puzzles in our room.

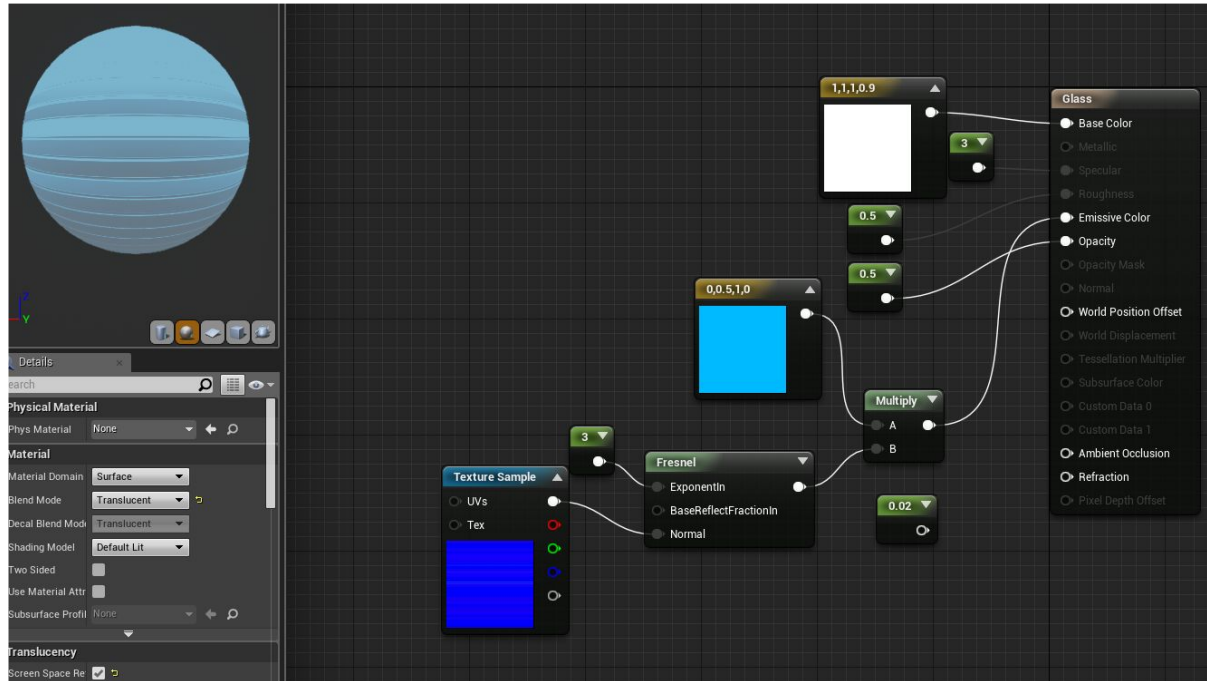


Challenge:

It was time. We needed to export our models to Unreal. First we needed to know how Unreal worked and after that we needed to know how to export objects to Unreal. We also needed to know how to implement and make materials.

Solution:

In terms of controls Unreal looked very much like Unity. Creating materials, complete different story. Unreal is node based. You have a lot of different nodes which you can all connect to the main node. After this you have to set values with value nodes. We used a lot of tutorials to get to know this system.



Challenge:

Pivot points change when you import models in Unreal.

Solution:

We just had to deal with this for the time being.

Challenge:

Gijs was very busy with his core module so Tim had to take over from this point.

Solution:

Gijs Was mainly involved with the concept phase and made the first version of the room. After that Tim had to do without him. This was doable but not much fun.

Experiment Answers:

Experiment 1: Finding a new way to navigate in Virtual Reality.

We have considered a lot of different ways to move around with a device. Usually one with a story behind it.

We have considered some kind of wheelchair where the player is bound to, and moving around would go with a few buttons that are integrated in the chair. The player would be stuck in a psychiatric hospital and would have to escape. However we felt like this would be pretty challenging to code and might also be a weird sensation for the player.

A different method we consider was lying or sitting in a mecha and moving around using the controllers in there.

The final method we actually tested out was when the player threw a certain device and could use a button to teleport to that place. However the player could actually see the surroundings while teleporting to the device. Which caused a lot of nausea for the players.

So in the end we just decided to use the teleportation method that is usually used by Virtual Reality.



Experiment 2: Scaling the player in Virtual Reality

Question:

We wanted to find out how we can grow and shrink a player live while being in motion and make this feel natural to the player, not getting nauseated.

Answer:

We did it using the triggers of the Oculus Rift controller. The right trigger would make you bigger while the left trigger would make you smaller. Speed was a very important variable to think of. You didn't really get nauseated but it felt kinda strange. Speed had impact on this. Too slow isn't good and too fast neither. We really had to find the right speed.

Other Challenges:

Challenge:

With our current concept it was quite the challenge to come up with elegant designs for our puzzles. The puzzles needed to be somewhat challenging but easy to understand. On a side note we also did not want the player to get confused about the rooms in relationship to the puzzles.

Solution:

The way we solved this challenge was by designing around our main mechanic, resizing. We came up with puzzles that you had to solve with the resize mechanic. We used a HUBS & SPOKES Level Design for our game. The artifacts/keys you had to collect and bring to the central room were gathered from the other rooms. There was a small doorway you had to get through, but the artifact you had to bring with you was bigger than the doorway. This puzzle was solved by picking up the artifact and shrink yourself together with the artifact, this way the artifact got smaller as well. To prevent the player from being confused where to go, we played around with shapes and colors, this way we wanted to guide the player through the different rooms.

Plus, Minus & Interesting points

Davey Verhoef

Plus

- Designing for Virtual Reality
- Fun group
- Oculus

Min

- Very little time due to other deadlines
- Nausea, I am not able to play every VR experience

Interesting

- Designing puzzles and general Game Design for VR
-

Nadia Groenewald

Plus

- Nice teammates
- Very cool to get the opportunity to work with Oculus Rift

Min

- A lot of deadlines at the same time so I didn't spend as much time as I would like on this project
- A big group, not everyone was quite sure what they were going to do

Interesting

You have to view game design in a different aspect thanks the the VR, this was quite interesting to work with.

Jesse van Vliet

Plus

- Virtual Reality is a very interesting and booming new technology coming up.
- Working in Unreal Engine proves difficult but very rewarding.
- Large group, a lot of ideas.

Min

- This part of the semester was one of the busiest all year so it was hard to find time for such time consuming subjects, both virtual reality and unreal engine.

Interesting

- Testing VR is very time consuming (also don't wear glasses).
- Was fun to experiment with different stuff in Unreal with another dev (cairan)

Tim Hendriksen

Plus

- This was my first time working in Unreal, so I now have a better understanding of the strong parts, the advantages of working in Unreal, and it's downsides. And I am certainly planning on coming back.

Min

- We weren't really communicating very well within the team. For example, I got to hear the latest concept for the puzzle, but when I made it and sent it to the developer he didn't know about it.
- The distribution of the work that had to be done amongst the team members wasn't ideal either.
- Everyone including me had to put a lot of time into Project Context. Because of that there was way less time or attention for this project.

Interesting

- The way materials have to be created in Unreal is node based, and totally new for me. But after using it for a bit now, I can really see it's potential.

Gijs Bakker

Plus

- We made a puzzle using a puzzle. VR has many restrictions which makes it really a challenge to find out what you are capable of. I also think VR could deliver a lot more undiscovered game mechanics

Min

- I was very busy with other school subjects which were more important to get to the next year. I unfortunately had to choose these before new technologies. This doesn't mean that i find new technologies less fun than those other subjects.
- Tim had to take over my part which felt really bad.

Interesting

- If i am looking at the end result i think it had become better than i had imagined.

Rick Ruiten

Plus

- At the start of the project we all had awesome ideas. We knew how to fix it and all of us started to think about how to tackle the small problems we could encounter. Together with the cool technologies we had to our disposal we figured a way out to make it even more awesome.
- Soon we figured out how to fix the puzzles and we managed to create a concept for the game.

Min

- Because the project was at the same time as other important projects and some personal issues I didn't put that much time in the project.
- To have 2 developers who don't see each other that often is not great
- To me the group didn't feel as a group, but as some people working on their part of the project. We also decided to work at different times which didn't help either.

Interesting

- Eventually the concept of the game worked out.
 - A group of 6 people is a lot harder to communicate to than a group of 4 - 5 , since we all have different schedules. However, it shows how important planning is in such a large group.
-

Links & Build

To get our **project data** you can download it at GitHub.com

<https://github.com/Thovex/Oculus-Player-Scaling/>

Playable **build** of our project (in .zip parts):

<https://github.com/Thovex/Oculus-Player-Scaling/tree/master/Executable>

Unreal Engine:

<https://www.unrealengine.com/>

The Oculus application:

<https://www3.oculus.com/en-us/setup/>

The Oculus Unreal SDK:

<https://developer.oculus.com/downloads/>

Our presentation:

https://docs.google.com/presentation/d/1lwN-2_mSMJlGr7M5MjB5kX0NA_DGet76RW4S0HXTYto/edit?usp=sharing

Made for the "New Technologies" choice module at the University of Arts Utrecht.

Lecturer: Valentijn Muijers

Date: 2 april 2017