

# Interim Demo:

# Beyond our Sight

May 2021

Team Two\*2

Maximilian Hess

Oliver Jung

Viktoria Kirchleitner

Moritz Schirra

## Task Progress

### Multiplayer Implementation

In our multiplayer implementation, we decided to use a client-server model, specifically a “listen server”, so we do not have to spend resources on the implementation of a dedicated server. This structure allows one player to play the game and also to provide it as a server that allows another player to connect as a client. Due to this structure, the client has more delay compared to the server player. However, this does not affect our game as it is designed as a cooperative multiplayer where both players work together. Because of that, the “listen server” structure is perfectly fitting for our project.

We integrated the core functionalities for this client-server structure very early to then immediately replicate each new action correctly. This also gave us a very early look on what the multiplayer will feel like in the final game, which helped us in the design process. The replication of actions in Unreal Engine means that we have to run most actions twice, once on the server and once on the client, thus essentially executing the same logic on both. If we do not do this, the server would act as a control instance for the client, so the actions of the client would be overridden by the server.

Right now, we have only successfully tested the multiplayer in Unreal Engine’s play mode. A real online connection has not been part of our test yet.

### Interactions and Puzzles

Both characters are able to move freely in three dimensions, both in normal walking speed but also by sneaking. The latter is done by holding “Shift” and will later allow the boy to pass by open doors, evading the eyes of the teachers. Jumping is also possible, pressing “Space” and is generally needed to climb smaller objects. However, as climbing is important to explore the full extent of the scenes, the players need to be able to climb larger objects as well. For this, we implemented crouching, which is activated by pressing “C”. A crouched character cannot move but allows the other player to jump on top of them and from there to even higher places.

Multiple forms of interaction with the world are also already implemented. Players can interact with books in the scene, read them and turn pages. These books are shown to the players as a large double page in the foreground of the screen. Each player sees a different set of pages based on their characteristics. In a similar fashion, a set of buttons on a puzzle

box can be shown to the players. They can then arbitrarily click on these and get a return value depending on the secret code behind the buttons, as they are primarily used for certain code based puzzle solutions. This puzzle box is supposed to contain a key which opens the door to the next room, however the spawning of the key is not yet implemented. Nevertheless, the key which later will be able to be picked up and used on the door is already a part of the interim demo but the visual part of the picking up process still causes errors so we excluded this from the demo. Furthermore, we are currently working on pushable and movable objects that the characters can use to reach higher up areas but this still needs a couple more hours of work before being fully functioning.

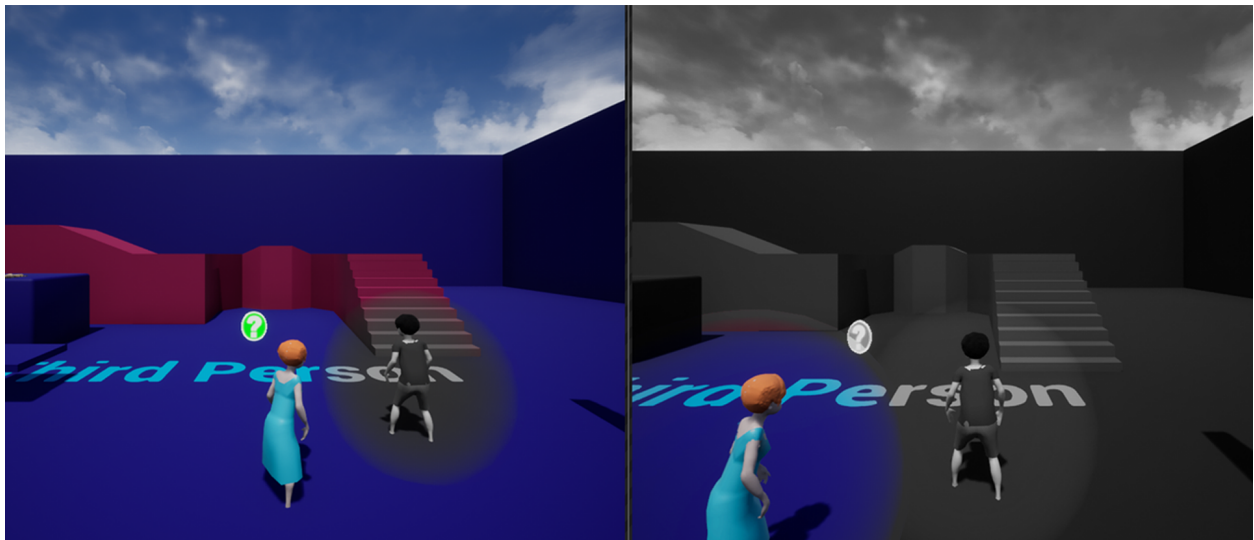
### Assets and Animations

We already integrated self-made models for our protagonists and assigned them animations for all possible actions and movements. They play correctly for both server and client and are fitting well for actions like crouching and picking up the key. For general assets in the scenes, we will work with a mixture of self-made and free downloaded assets from public stores. We already have chosen specific ones that fit our style and levels but have yet to integrate each of them into the Unreal project. We are taking great care to work with a coherent art style, so that none of the objects feel out of place. Furthermore, we are holding back with the model integration until the level design is finished and the core functionalities are implemented so that we have a full vision of how they should look and what objects we need in detail. Furthermore, we are hoping to add character specific animations as a nice to have.



## Shaders

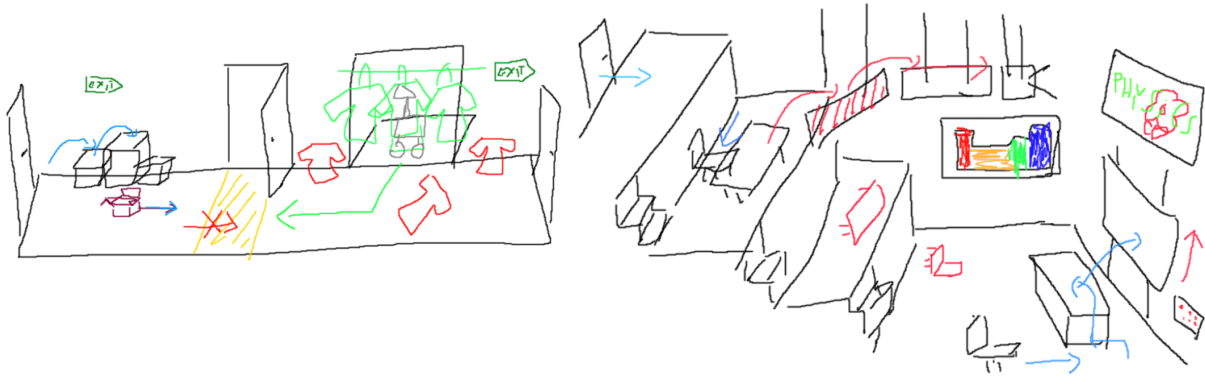
For creating the visual differences between the worlds of the female - and the male character we use two mesh shaders and one post processing shader. The latter is only present when the player controls our male character. It desaturates the whole view except a small area around the female character and thus creates the black and white world of the boy with a “window” to the colorful world of the girl. This “window” is defined by the mesh shader attached to the girl, which changes the value inside the stencil buffer wherever a fragment is rendered with it. Furthermore, it is responsible for manipulating the colors inside the “window” in order to smooth the edge between the gray world and the colorful area which occurs due to the use of the stencil buffer. Similar to this shader, the mesh shader of the boy is used for changing the colors inside a sphere around the character. However, instead of desaturating the edge of the characters area this shader smoothly desaturates the colors towards the middle.



## Level Design

The general level design and the design of puzzles is close to finished. Five completed rooms, each with very different and for the most part not repetitive puzzles exist on paper, which is the extent we planned for the desired target. They are also connected meaningfully, showing the escape from the school through different departments. The rooms differ greatly in layout and solution, but follow the same principle of progressing from left to right with the implementation of the camera in mind. The puzzles use the full extent of our implemented and planned interactions. Furthermore, they utilize the chaos

and order views of the different characters as well as the chaos and order properties of certain objects in the scenes. Having the level design completed allows us to fully focus on implementing them in the engine with the full amount needed interactions and content in mind. This prevents us from giving way to feature creep and implementing something that turns out to be unused for the desired target.



Our puzzle design documentation consists mostly of sketches and an additional, extensive text-based solution, which also captures possible thought processes of the players and how they would find their way through the room.



## Challenges

During the implementation, a few challenges occurred. First we had problems with liftable items like the key, which we need to open doors. Due to a not yet resolved bug, the player's movement is impaired while the figure is holding the key. Thus, we decided to let the pick-up animation play without actually grabbing the key model for now.

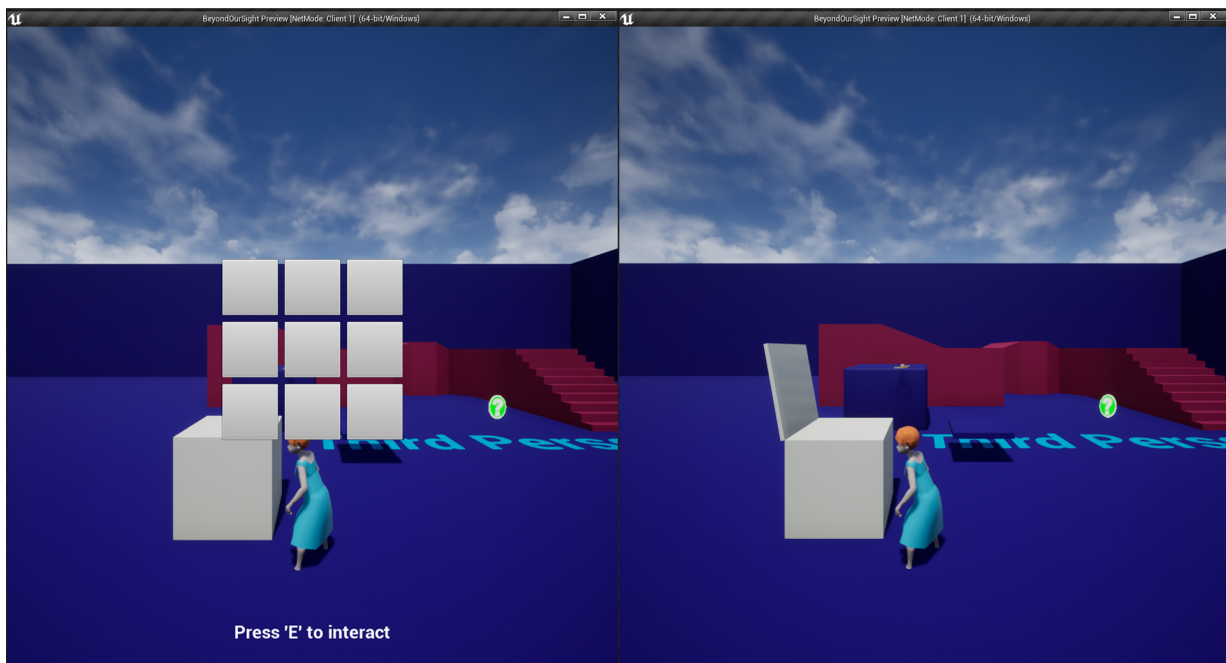
Furthermore, we faced some problems during the implementation of the multiplayer. The different types of replication on client and server side respectively lead to some inconsistencies regarding animation and object behaviour in the two perspectives. Most of the problems have been fixed so far but it also took more time than expected and it will again take some time in the alpha phase to achieve flawless replication.

Additionally, we have yet to implement the options menu to create and join multiplayer lobbies. While fundamental menus were originally intended for the interim demo, we prioritized them lower during the development since real online multiplayer is not yet implemented, and they thus were not created in time. Game menus are now set to be implemented as soon as we get closer to testing the actual online functionality of the game.

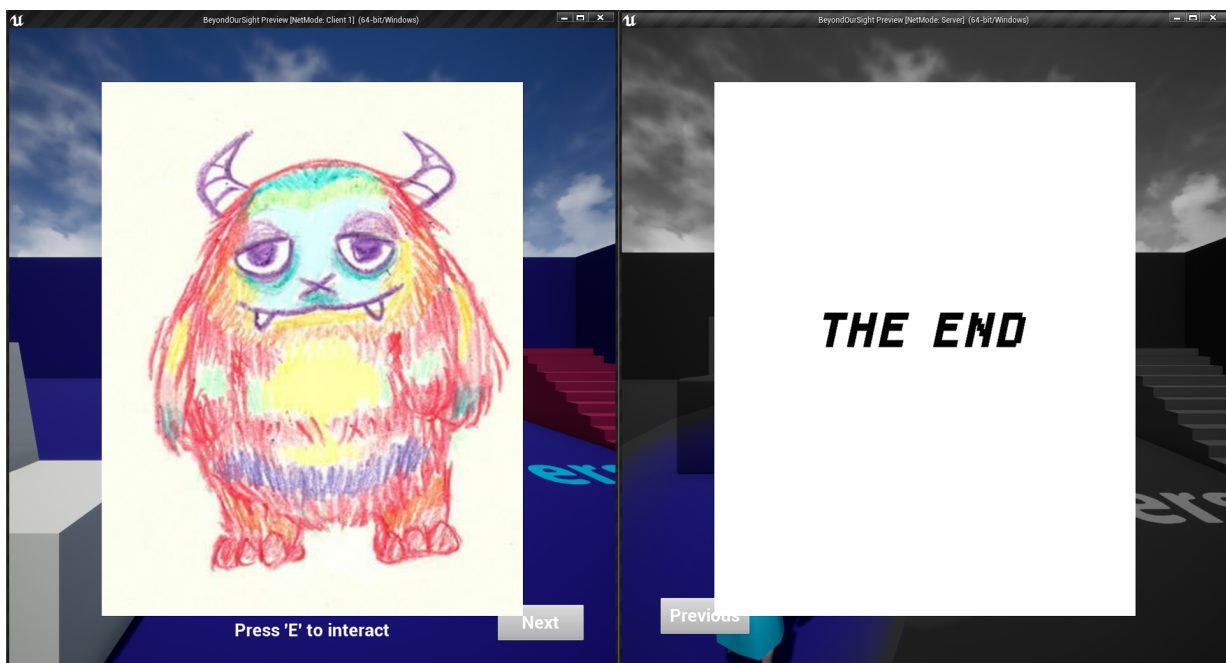
## Design Revisions

Currently, we are very content with the design decisions we made up to the prototype. Our original concepts could be implemented without considerable conflicts so far. Up until now, only one design revision had to be made. Due to serious problems with the camera assignment in our multiplayer, we changed the handling and angle of our camera to a better supported design. Right now, the camera is very similar to the default camera of the default Unreal 3D scene. However, our plan for the alpha is to show the scene from an upper left corner. Fortunately, this decision happened early enough for the level design team to pick up on it. They then planned the scenes in a way that placed all important content more to the back-right of the rooms, as not to miss it as a consequence of the new camera angle.

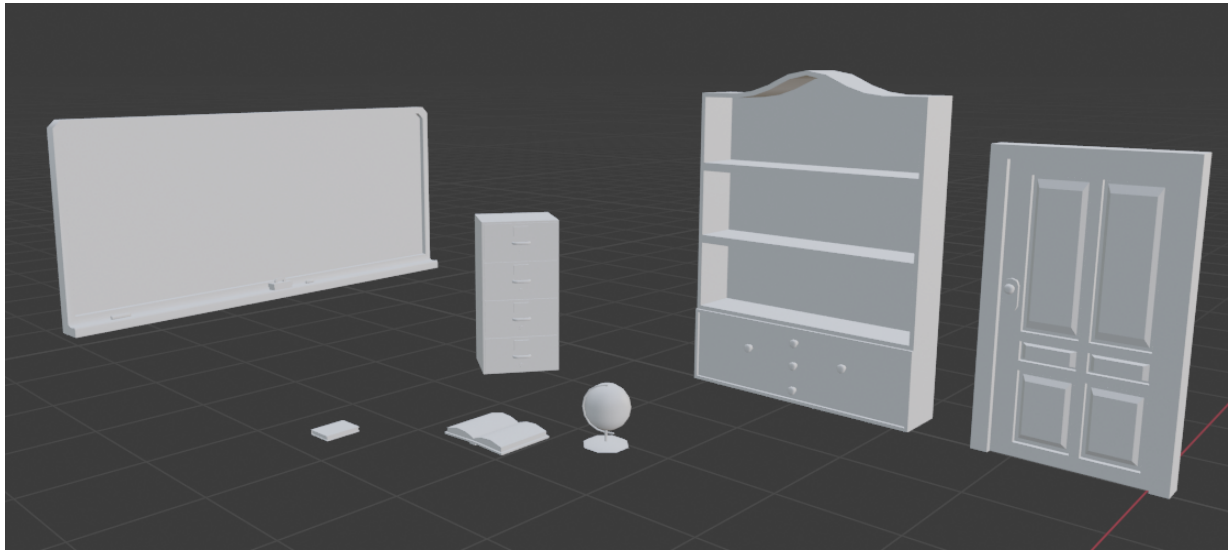
## Progress in Pictures



This temporary model of a box can be opened, if the correct code is entered on the buttons. The left picture depicts the code panel, the right shows the open box.



The book in the scene contains different text and images, depending on the played character. On the left side, the girl only sees colourful pictures while the boy on the right perceives text.



A few 3D assets to be used in our classrooms. We chose an art style that goes along well with our protagonists.