SAVE THE OCEAN

A Virtual Reality Experience

Human Computer Interaction

Group 25.1
Songha Ban

Irene Klop

Furkan Ceran

Laura Hagedorn

Contents

1. Introduction

For the course Human-Computer Interaction for CSAI, we created a simulation in a virtual reality environment with the intention to raise environmental awareness among students. Specifically, we want people to be more aware about the way they treat plastic. We chose this topic for its importance and urgency in the current world state. Virtual Reality simulation is an effective and promising approach, because it includes mental and physical interaction as well as embodiment in the shown scenes. For topics like these, people need to feel and see actions and consequences, instead of only hearing about it, to make a stand and take action.

1.2 Planning

Our first lesson consisted of us getting familiar with the use of Unity, which is the application we used to create our VR environment. We also conducted a planning including what needed to be finished every week and by whom. We considered everyone's strong points and based our task division on that. In addition, we decided to use online platforms such as Google Drive and Whatsapp to communicate and share our findings.

2. Theoretical Framework

For our basic conceptual design, we decided to follow the Double Diamond Design which includes the following steps:

a. Discover
b. Define
c. Develop
d. Deliver

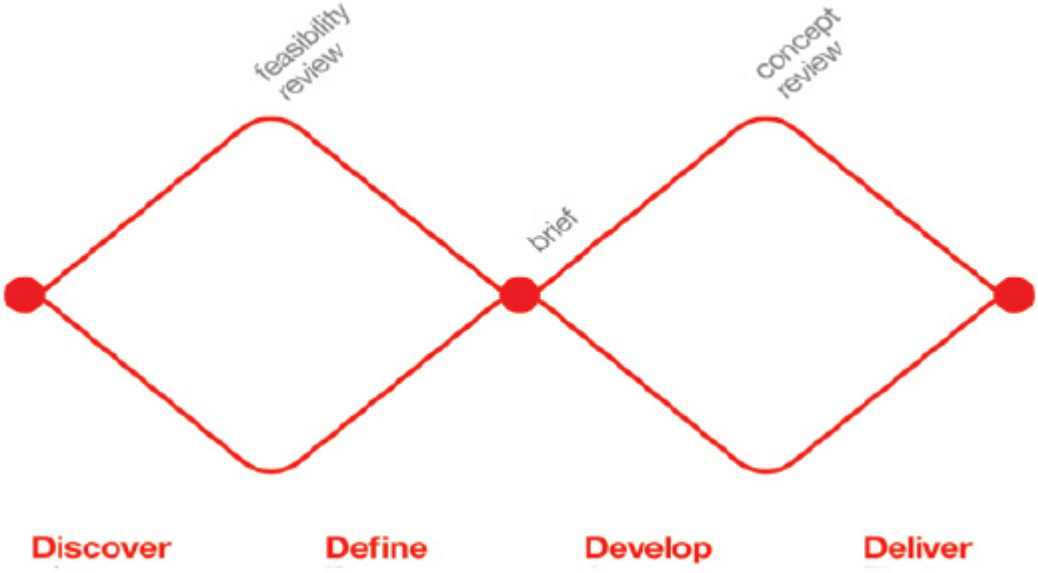


Figure 2.1: Double Diamond Design

Specifically, we will use the 'GOMS' task analysis which consists of the steps: Goals, Methods, Operators and Selection Rules. In order to conduct the user research, we used the 10 usability heuristics and the User Centered Design Circle, as well as a persona of the user.

3. Conceptual Design

3.1 'GOMS' Task Analysis

a. Goals:

In the first step, we identified our broad goals in order to subdivide them during the process. Our broad goal was to raise environmental awareness and more specifically, the impact of plastic waste in the ocean. To do so, we wanted to create an emotional experience for the user by showing the impact of careless handling of (plastic) trash. Our subgoals consists of seeing the consequences our actions have, to show that one careless piece of plastic or other trash thrown in the ocean has a bigger impact than might be suspected. We wanted to make people realise that the not-longer-than 1 minute task of throwing away trash in an actual trash can make an impact for years. Also, we wanted to create an unexpecting storyline, so that the message we want to bring across will only become clear while playing. We hope our application will change the way the user takes care of plastic waste and other trash, cleaning it up from then on.

b. Methods:

In order to create an emotional experience, we needed to make the experience as realistic as possible so that the player feels embodied in the simulation. A Virtual Reality experience offers these factors effectively. An important method in our game is the contrast between the two scenes; the boat scene with a pleasant atmosphere, and the confronting ocean scene, where all the plastic trash from the night before can be found. This should show that while people have a good time not worrying about their actions, they are polluting someone else's living environment.

c. Operators:

We kept the interaction with the environment as simplistic as possible, so that the focus lies on the perception of the environment and the user is not distracted by useless features. On the boat, the user is given clear instructions on what to do (such as drinking Cola and throwing the can into the ocean), using a simple pointer click. In the boat scene, the user is negatively influenced by the text "Can't find a trash can? Just throw it into the ocean!", which then follows with there only being one option; throwing it in the ocean. This forces the user to do the wrong thing, but at the same time shows the thought process of a lot of people; "Why should I make an effort to find a trash can?" Again, we want to show what consequences this thought process and these actions have on marine life.

d. Selection Rules:

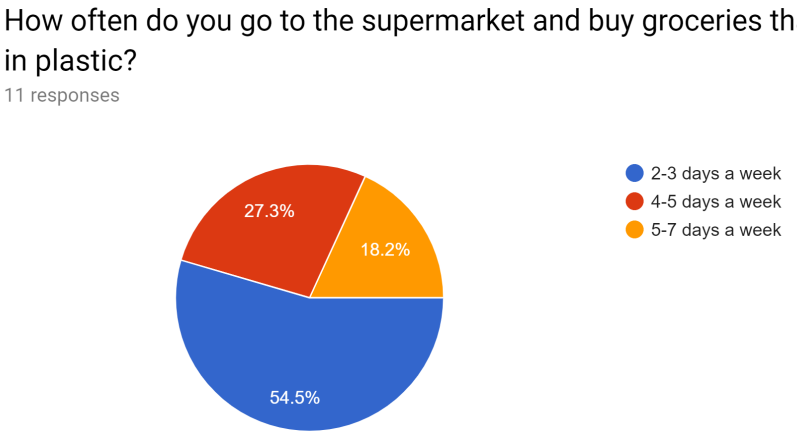
There are not really selection rules, as, as stated above, we force the user to undertake a specific action at the beginning. From drinking cola to throwing it away into the ocean, the user does not have multiple methods to reach the goal, but rather has to complete a series of missions. We do give the user a choice in the end and that is whether they want to "go back in time" to the party on the boat the night before and clean up their plastic waste and other trash, or if they don't. However, when choosing that they don't, they will not really reach our goal. They haven't changed their mindset on how to handle trash. Nevertheless, they might have gained more awareness of the negative influence of plastic and other trash in nature (the ocean), which is a huge part of our goal.

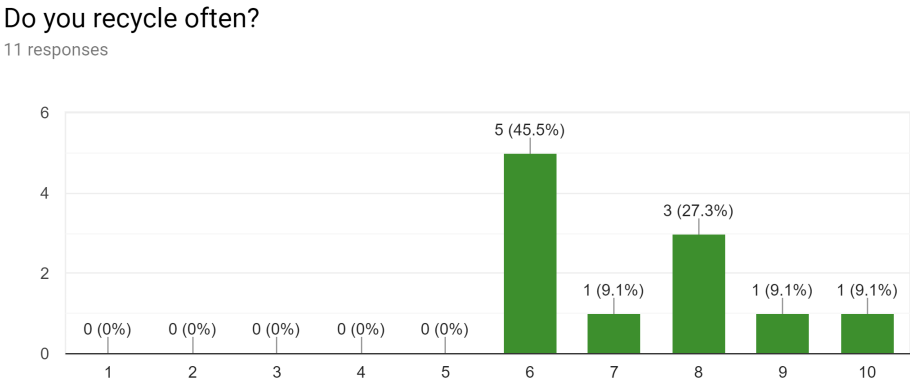
4. Users

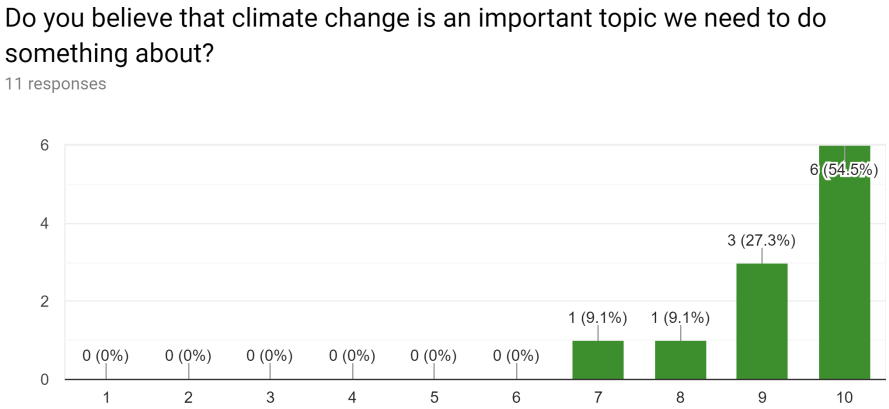
4.1 User Centered Design Circle

a. User research and analysis

In order to plan our game and understand how we can communicate our goal in the most effective way, we first needed to evaluate for whom we are creating the game. To be able to classify our user audience, we decided to focus on students from Tilburg University and more specifically, students who also study Cognitive Science and Artificial Intelligence, because of their expertise on the topic and their experience with VR simulations. We created a survey and used the answers to get an image of how our users treat plastic. As a result, we concluded that taking care of the environment regarding climate change, pollution in oceans, plastic, chopping down rainforests and so on are important topics for many students, and it is necessary to raise even more awareness to its urgency.







b. Concept design

To create an image of what our steps needed to be, we first created a storyboard as shown in Figure 4.1. Based on that, we created our low fidelity prototype and following that, started making our high fidelity prototype. Every step consisted of new insights, fixing mistakes and better creating our vision in a way that worked.

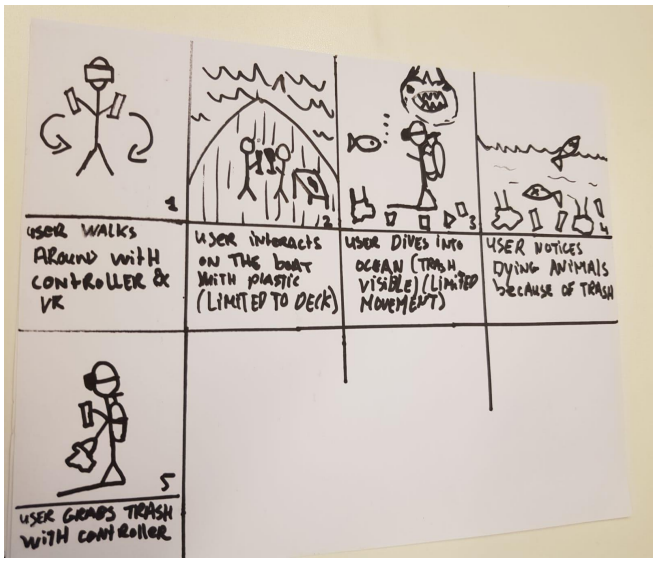


Figure 4.1: Storyboard

c. User testing prototypes

We started with a low fidelity prototype, for which we took two pieces of paper where on one we drew the basics of the boat scene and on the other we drew the basics of the ocean scene. We did this with only a pencil to make a great distinction between the fixed part of the scenes and the interactive and moving parts of our scenes. We added these dynamic parts with colourful printed images. You can see this in Figure 4.2 and Figure 4.3. We then asked our users to walk through this low fidelity prototype and give us feedback. During this investigation we realized that some tasks, which were straightforward for us, were not completely clear to the users. We realised we needed to make adjustments to the moving space, how to give instructions and how to know what to look for as the user.

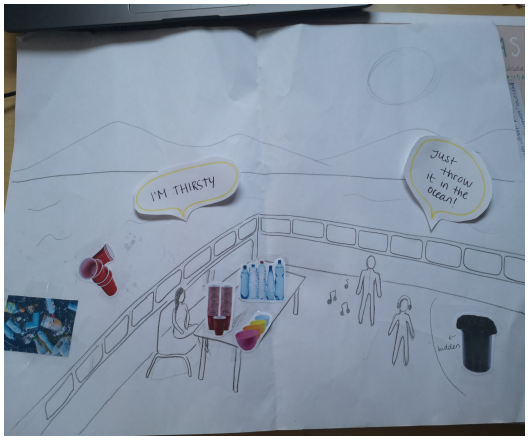


Figure 4.2; The Boat scene

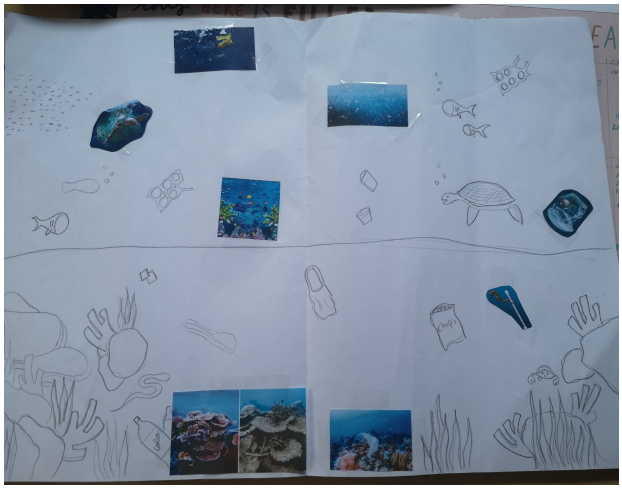


Figure 4.3; The ocean scene

d. Detail Design

After we received the feedback, we clarified the game concept immensely. Then, we started to build the environment for our game and implement our ideas and concepts as well as the feedback from the users. We started with the basics for the boat scene: the boat itself, lights, chairs, and a table with a lot of drinks, food and plastic bottles and containers. The basics for the ocean scene were also made and consisted of the terrain and ocean water. Later the humans were added on the boat and the fish were put in the ocean. After that we added the user interaction and movements. We then created the third scene, which consists of the ability to go back in time and fix your actions. Lastly, the pop-up messages were added as well as implementing the three different scenes into one application.

e. Measure and Develop

The creation of our simulation will be explained throughout the following chapter(s).

4.2 Persona

After interviewing our users (other students around our age), we found that they answered almost every question the same way we did, so instead of mapping our users, we combined our answers with some fictional things and made a persona[1].

5. Outline of the game

Our game starts at the front of a boat, overlooking the ocean as it sails. You are welcomed to the game and invited to a party. You then move to the back of the boat, where the party takes place. There is a pleasant and friendly partying atmosphere where the player can look around and interact with the environment. There are people dancing and a table with drinks and other party necessities. A message then pops up saying that you're thirsty; urging the user to drink something. A cola can is highlighted and you can pick it up with your controller. When clicking again, you "drink" the cola can, hearing a gulping sound. You're then left with an empty cola can and there's no trash can in sight. A message pops up, as if a friend is saying: "If you can't find a trash can, just throw it in the ocean!" You can then throw the can in the ocean. After that another message shows itself, saying that the party is over and the

[1] https://oceancleanup.xtnsio.com/hw88y9z1

next morning you're invited to a diving trip. You're then transported to our ocean scene, where you can look around and even swim around a little. You're surrounded by many colourful fish, but sadly also plastic waste and other trash. There's even your used trash from the night before. The cola can you drank from is highlighted again, and when you go and pick it up, you will be back to the boat scene with a message asking whether you want to clean up the trash on the boat from the night before. When you choose 'yes', you go back to the partying scene where you are now able to interact with more objects and even move around. You will be asked to find a trash can that is highlighted, and then clean up all the trash. If you choose 'no', the screen goes black with the following single message: "Plastic polluted the whole world and killed all marine life thanks to you."

We wanted to create a dramatic and obvious change between the two scenes to create a feeling of cognitive dissonance. The first scene on the boat is supposed to arouse positive emotions and a feeling of embodiment, as the player probably recognizes behaviour and characteristics of himself from the 'real world'. This scene shows how most people (and maybe the player) deal with the urgent topic of plastic waste: they don't. The consequences of careless behaviour are often unknown or unimportant to many citizens. Therefore, the player is given the ability to go "back in time" and change what was done previously.

We used this feature in order to create an emotional experience for the player. During our group discussion, we came to the conclusion that emotions are often more forceful and impressive than knowledge through e.g. reading articles.

Nowadays, when people create a lot of plastic trash and/or do not recycle it properly, they do not see the consequences of their careless behaviour, because all the trash is dumped somewhere in the ocean. In our simulation, the player immediately gets confronted with the consequences of his behaviour, creating (hopefully) a feeling of culpability.

6. Execution

In order to create an emotional experience and to warn the user of the consequences of careless plastic waste handling, we made sure that the user perceives the scenes as realistic as possible to feel embodiment, but keep the interactions simple so that the game is not too complicated.

6.1 Equipment

As our game requires basic button uses, we used Google Cardboard and a simple bluetooth controller. Because it only requires a pointer click interaction, only one button from the controller is used for input. Throughout the simulation the user receives feedback to the visual and auditory channel. The scenes are supported with fitting sounds and the interaction with objects create noises as well.

6.2 Implementation

As we are using Google Cardboard and building our game as an android app, we used Google VR SDK (Gvr) for Unity to make implementation of interactions easier. We created a game object called player and put the main camera inside the player object to make sure the camera view is always the player's first-person view. In addition, we added GvrReticlePointer and GvrPointerPhysicsRaycast, which enable a gaze-based reticle pointer at the center of the camera to click other objects with a button click.

To make the instructions and pop-up texts visible within the camera all the time, we created a separate UI camera inside the player object, which has the same position and orientation as the main camera but only displays the objects with the tag of UI. For the text which does not have to be within the view all the time such as invitation letter at the first scene, we set

the render mode of the canvas as *World Space*, so that no matter where the camera is oriented, the text stays there as shown in Figure 6.1. For the pop-up text, however, which always has to be within the view during the game, we set the render mode as *Screen Space - Camera* and set the event camera as UI Camera, not the main camera, so that the text is not hidden by the other game objects.

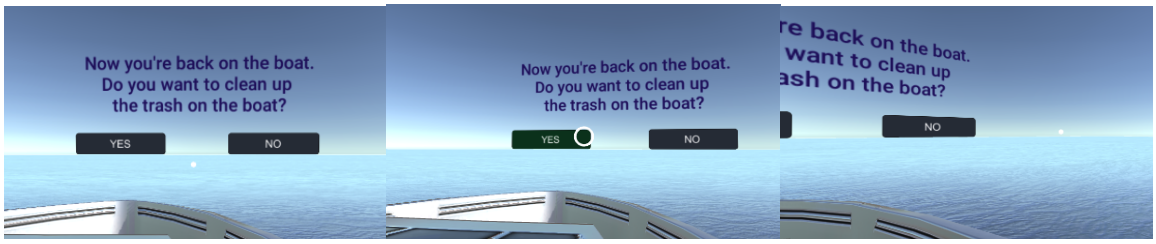


Figure 6.1: Text examples on the canvas with *World Space* rendering.

Figure 6.2: Text examples on the canvas with *Screen Space - Camera* rendering.

In order to connect different scenes together and make the whole game as one story, we used EventTrigger on several game objects along with the GvrEventSystem. For each object, we set the pointer click as a trigger and assigned corresponding events such as updating the pop-up text, playing an audio source, enabling the player to move around, disabling the glow of an object, destroying an object, and switching the scene. For the events that are not built-in features in Unity, we wrote C# scripts to update the states of the scene or modularize particular functions. For the events that should happen a few seconds later than the trigger, we used Coroutines to wait for a given amount of time and then trigger the event.

Even though we are not tracking the movements of the actual user, we wanted to add some level of freedom to move around, as the user will feel more realistic in the ocean that way. Thereby, while we do not let the user move at the initial boat partying scene, we let them move around in the ocean. We used a C# script that slowly moves the player forward if the player looks down with the angle of 30 degrees. The user is able to explore a broader range of the ocean without complicated controller, and because the user is not actually walking around, it feels more like swimming in the ocean. In addition to that, looking down causes the user to get confronted with the plastic waste which is laying on the bottom of the ocean. The boat scene coming after the ocean scene, where user can clean up the table on the boat, we enabled the player to move around in order to give a mission to find a trash can and also not to stifle them by limiting more freedom than the previous scene.

Furthermore, we needed some game objects to move automatically. For example, when the user accepts the invitation at the beginning, the camera moves backward automatically to take the player to the party. In the ocean, we had to make the fish animated, otherwise, the scene does not look realistic at all. We created some empty invisible game objects to set the

path that we want the objects to move along, and used the C# script to update the position of the objects following the path.

We do not give the user an option to exit the game in the middle of the story, because we have a certain message we want to bring across, which is only achieved if the user reaches the end of the game.

6.3 Build

Even after we made sure that the game runs smoothly on Unity, we had a problem building it into an android application because of the limited computing power. As our project was overloaded with too many objects, lighting, shades, and animations, we had to optimize the scenes by getting rid of some unimportant lights and shades and grouping the animated objects. After the optimization, the build was successful, and the game ran smoothly on android phones.[2]

7. Feedback

In our project design we followed the structure for gulf of execution and evaluation in order to ensure that the tasks that were given to the users in the game did not cause any errors or a hindrance in the progress in completing the task that they were given. During the design of our low-fidelity prototype, we did some user evaluation and testing to determine which functions we would require additionally that would help the user in completing their task. The feedback that we received from the evaluations gave us insight into our project needs and adjustments that were required to deliver the users a game with clear functions and feedback. Particularly, the remarks that were made about the feedback that the users would receive after performing a certain action were most helpful since at that stage we were not sure what kind of feedback the user should receive after every action. Thus, after implementing the changes we went through another cycle of testing to ensure that the process of each task was clear to the user.

At a later stage, we finished working on our high-fidelity prototype and did more user evaluation and testing. Continuing with our design, we ensured that the changes that were made in the low-fidelity prototype for our high-fidelity prototype were still following the gulf of execution and evaluation. The users struggled with picking up the right object, and therefore we added a more clear and concise instruction, including that the users can interact with 'glowing' objects. On the other hand, if we look at the evaluation part of this task we will see that the user receives visual and auditory feedback when they drink from the soda can. First of all, the user will notice that the neon-highlighted can that was on the table will disappear once they interact with it, indicating that it was picked up and providing visual feedback to the user this way. Secondly, when the user picks up the soda can they will receive auditory feedback in the form of the soda can being opened up and drunk from. This gives the user confirmation that they now have drunk the necessary liquid that they required to complete the task. And lastly, more visual feedback is provided immediately after the user finishes drinking from the soda can, providing another confirmation that the task was completed and explaining to them that they will now have to perform another task.

For our final feedback, we made sure our prototype was finished and we created two surveys, which were meant to be taken when someone was undergoing our VR project. The

[2] apk file: https://drive.google.com/file/d/1KLVUzUl26TnhtlCBLqbXz4x964JtnbxY/view?usp=sharing

surveys are found in the following links[3]. The first survey mainly includes questions about how students handle their plastic waste and their thoughts on climate change. The second survey asks for certain feedback on our game based on the user heuristics and it recites the questions from the first survey in order to see if the experience actually changed the user's point of view on plastic waste. The answers to these surveys helped us improve the game. To be specific, there were some feedbacks that the pop-up messages need to be at a better angle. This was before we had the separate UI Camera and the text was hidden by other objects. We fixed it to display the text better to reflect the feedback. In addition, one of our users thought it was unclear what they had to do or could do in the ocean scene and therefore we added a pop-up message within the ocean scene that gives instructions on how to find the glowing can.

8. Group work

As agreed, we all met up each week, bringing along our finished task. Each week, each one of us had a task and even though the details of the task changed often, we managed to finish our VR environment well on time. Table 8.1 shows the overview of what each member did, and all together, we created the storyline and took part in exercises given in class.

Songha	the storyboard, all forms of interaction and integration in the VR environment / the controls, the pop-up messages, making the fish move, reporting about the equipment and implementation, and making sure our VR project runs smoothly
Laura	the low fidelity prototype, the ocean scene, creating the survey, creating the outline for our report, and writing about the theoretical framework, conceptual design and user centered design circle
Furkan	the storyboard, the humans on the boat and making them move, writing about / evaluating the user feedback and limitations, creating the surveys
Irene	the low fidelity prototype, persona, the boat scene, the fish and corals in the ocean, and writing about the introduction, theoretical framework, conceptual design, user centered design circle, the story / outline of the game, group work, limitations and the conclusion

Table 8.1: Division of tasks.

9. Problems and limitations

Our biggest limitation was that we wanted to create and expand more than possible with our time, available equipment and materials. In addition, we needed to make sure we didn't waste time, so we couldn't just dive head first into Unity. Several sessions of discussions were needed to determine which textures and models we wanted to put in the game. The reason for this was that it would require us to spend more time working on the textures and

- [3] https://docs.google.com/forms/d/e/1FAIpQLSe8DpYnYTv4eJji_MpmLo8b061hJaIDUG07BWkOwo0d1ONb1g/viewform?usp=sf_link
- https://docs.google.com/forms/d/e/1FAIpQLSdANVnL3qR1a_2A1Rv-C9bLUD2B0-tqNRxtcBL937XOaSid-w/viewform?usp=sf_link

models which would clutter the game with a continuous stream of additions that might even go unnoticed rather than focusing on the actual functions.

Furthermore, it took us several trials to define and design the process in which our functions would be used. This was due to the fact that each function would require us to analyze and test the program from the beginning, not only in order to ensure that the function was working properly, but also to determine whether it was necessary to include that function in the program to perform the given action and no more.

Some simple things took more time than expected, for example, putting all the bottles, cans, chips etc on the table. Everything needed to be scaled perfectly and placed *on* the table, not floating above it or half in the table. The same went for placing the fish. Creating the humans also took more time than expected. We also ran into the problem of not saving properly, since you needed to upload the whole assets folder, not just the scene, when wanting to export it. The ocean terrain was also too big to export at first. Next, it was hard to put the pop-up messages at exactly the right angle. However, our biggest set-back was that our project was too big to build and therefore upload it to a phone to be able to actually play it. We needed to render some objects or factors down to finally have it running, which was frustrating.

10. Conclusion

In conclusion, we hope that we will reach our goal with our VR simulation and that the students who tested our game feel more conscious about their plastic waste. We have learned how to work in Unity to create our own VR program, how to put the user first and keep our project a learning progress. We learned how to overcome the limitations by spending our time wisely and asking others for help when needed. We learned about the Human-Computer interaction in an interactive and interesting way and created a project to be proud of.