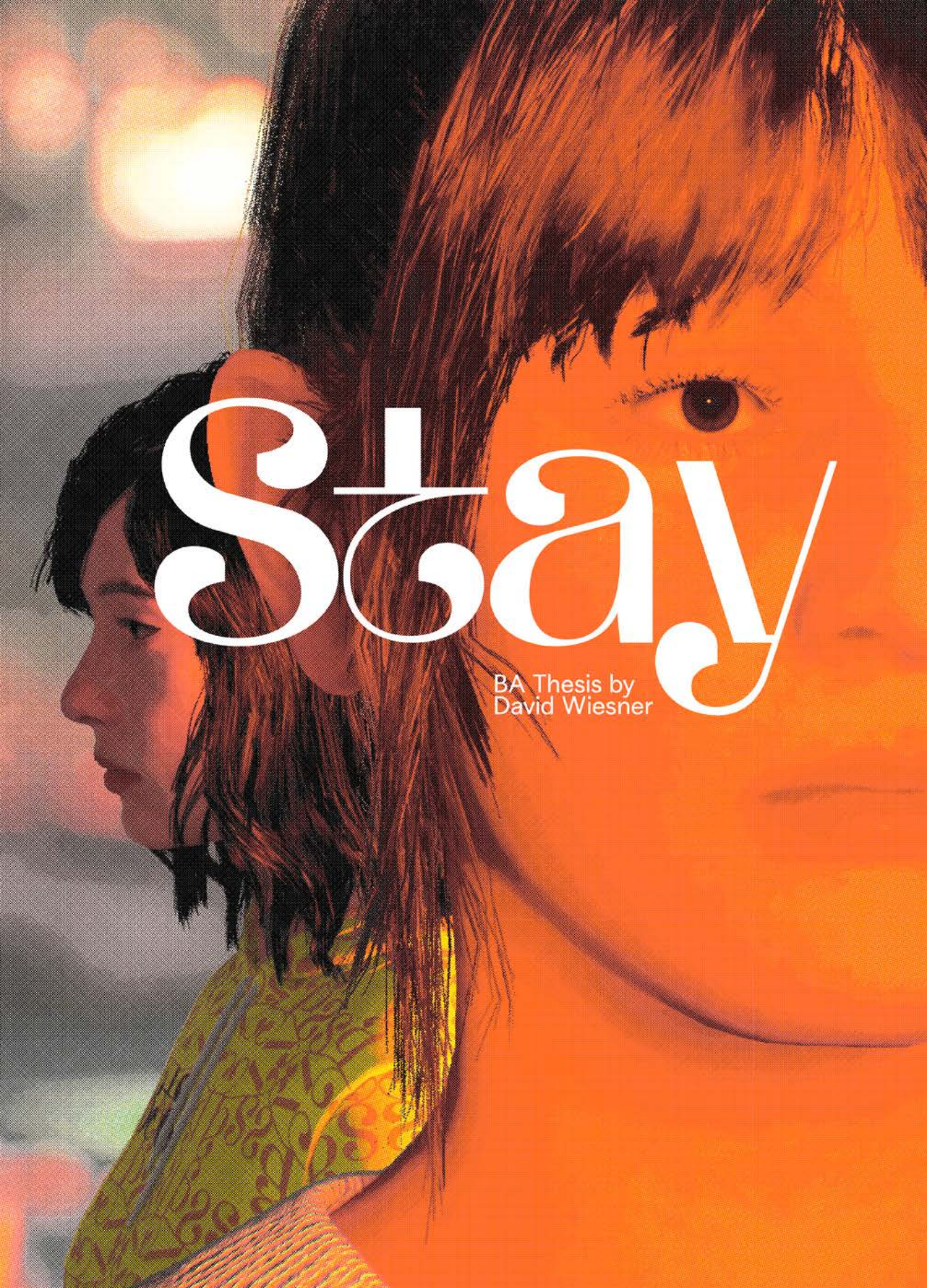

Stay
BA Thesis by
David Wiesner

Stay

An exploration on spacial-expressive typography
for dialog in 3D games

Final Thesis (Integrated Design B. A.)

Technology Arts Sciences TH Köln,
Faculty of Cultural Sciences,
Köln International School of Design

Supervised by Prof. Björn Bartholdy (CGL) &
Prof. Michael Gais (KISD)

Written by David Wiesner

Submitted on 19. January 2023

Hereby I declare that the Thesis submitted is my own unaided work. All direct or indirect sources used are acknowledged as references.

Cologne, 10. January 2023

Table of Contents

Research Goal

Video games still rely heavily on text to communicate speech. Even though many games nowadays are fully voice acted text is still used as an accessibility fallback, and in dialog choice-based scenarios. But voice acting every speaker is not a standard for games. Omitting some speakers from voice acting or the whole cast isn't uncommon, as voice-acting takes a considerable part of the budget. In these cases text is the only way speech is conveyed. A study I did on dialog interfaces in video games from the recent decade showed that text is displayed almost exclusively as a screen-overlay. [Wiesner, 2022 Layers, p. 41–42] In some cases text was placed similar to billboards inside the 3D game world. [Notably the Tomb Raider reboot as well as Death Loop.] But overall this approach remained an oddity.

> Placing text representation of text non-diagetical on screen has a lot going for it. There is a more control over infringing elements, that diagetical text otherwise has to deal with. [Diagetical in the sense it is used here is technically not correct, as it only fulfills the requirement that it is placed in the world instead of being rendered directly on top of the camera's lens. But the text would appear like a magic speech bubble—which is not really diagetical, except for games with magic systems.] As a trade-off for the clearer readability lay-over text stays disconnected from the 3D world. It always appears as a meta commentary to what was happening on screen. And while that is fine for menus; in dialog situations it keeps the player at an arms length. It often also results in the player having to jump between text and in-game action. Conversations for these reasons tend to be rather stiff or at least more passive compared to the otherwise free-form movement in 3D games.

In my initial study I also noticed that text was used mostly in a serviceable manner. The inspiration for this mode of display is closer to that of novels (where the design takes a step back for the story to unfold) than magazines or poems (where the design highlights and emphasizes the meaning of the text). This meant the expressive capabilities of text were often not explored.

> In this Bachelor thesis I therefore want to explore the possibilities of dialog placed inside a 3D world as a more active element, as well as the expressiveness of typefaces to convey the speaker's emotion.

Tools used

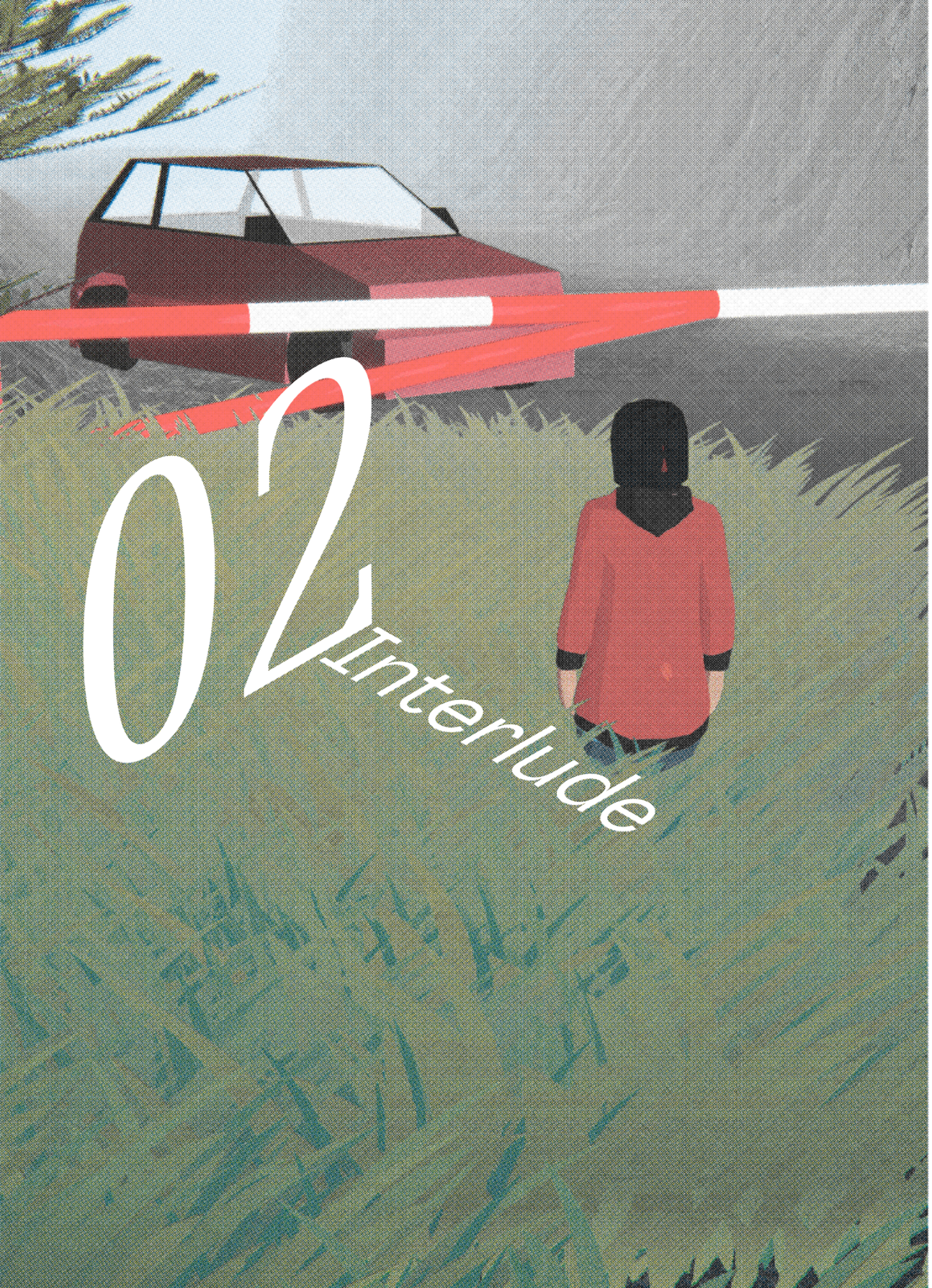
A side-goal of mine was to gather experience in both of the major game engines: Unity and Unreal Engine. The reason for that being that I wanted to find out how much expression through text in each engine was possible, and how accessible these options were for designers to implement. Up to this point I had been working with Unity for most of my projects. I already knew the capabilities of Unity's TextMeshPro, but I also knew how difficult it was to get some of the finer typographical details to work. I therefore developed the prototype in this engine to gather my first impressions, on how well positioning text in the 3D environment would work. For my main project I wanted to work with Unreal Engine (UE). I had worked briefly with UE4 for a collaborative project. But since my role in that project was that of a sound designer my knowledge of the engine was very limited. In this short amount of time I noticed that at least the implementation of fonts was easier than in Unity. I went with UE5 because I was interested in trying out Lumen and Nanite for my project. UE5 by that time had been released for more than two years, so I was positive the engine would be running without hiccups, and enough tutorials could be found online.

> Another reason why I opted for Unreal was the exclusive availability of Megascans and MetaHumans. I was hoping I could use those so I wouldn't need to model anything myself———as my focus was to implement typography in a 3D world———while also keeping a consistent look. During the creation of the prototype in Unity I noticed how difficult it was to get a good player controller running. This was another reason for me to try out UE as it already comes with a third person controller set that could be activated from the get go. There were similar projects available for Unity, but since Unity is mostly based on scripting those were more difficult for me to adjust to my needs. In the end I resorted still resorted to a third party plugin for the character controller in UE: Advanced Locomotion System v4. It had smoother transitions between movement states and a more elaborate move set.

For the audio implementation of the prototype I used FMOD. I had already worked with the plugin in several projects and liked how well it handled complex song structures and transitions. Due to some technical issues I encountered during my implementation in UE I decided to try out the MetaSound tools that came with the engine for the thesis project.

> For the music production I went with GarageBand. I don't find this to be a particularly good audio work station, but it is the one I have already installed and know my way around. Since the editing options in this program are severely limited it would also keep me from fiddling and fine-tune the songs too much. Additionally I used Reaper as it allowed to export event markers, which I used to mark transition points within music loops. For audio clean-up I used Adobe Audition.

Finally for the creation of the fonts I used GlyphsApp. Next to the creation of the letters this allowed me to create the different weights/variations of the typefaces with relative ease. For the kerning of the normal weights I used the plugin KernOn.



Going through a Phase

The prototype “Going through a Phase” was an already existing project, that I worked on for quite some time at that point. This meant the basic story, character design, and player controller were already implemented. What I added then was more complex sound design, music, and of course the presentation of the text.

Typography

The role of the player in this game is that of a child on vacation. The character wakes up early and explores the nearby area. After discovering a certain spot she returns back to her mother to share her findings. I wanted the text to represent the child’s thoughts. I wanted to attach the canvas, on which the text gets projected, to the player——a form of three dimensional speech bubble so to speak. After several tries I decided against this. The reason being that I wasn’t able to attach the canvas to the movement of the camera. Instead I was only able to connect it to the direction the character faced. This resulted in jarring jumps as soon as the character strafed left or right. Instead I used an UI canvas, but made it look similar to the attached version from before. Using the UI canvas helped me to avoid getting the text obstructed by trees or terrain. In turn I lost the ability to make the text emissive——a feature that I thought would be fitting to illustrate the characters thoughts.

When testing out placements of the text I opted to align the text to the center. In my research, as well as my own gaming experience I noticed a disconnect between what was happening in the world and the subtitles or text boxes when they are aligned to the borders and corners of the screen. I wanted the player to be able to navigate and perceive the world, without having to jump back and forth between text and action. For subtitles in film as well as games this is often an issue. This is mostly due to the text being an accessibility addition. I wanted to test the capabilities of text to be more integrated into the play experience. For this I need to adjust the font size accordingly. Compared to what’s used for subtitles normally the text in the prototype appeared quite big. It almost behaves like a title screen. The increased size was also necessary because with the text sharing the same screen space with the environment there was a high risk of the text becoming difficult to read. In my proposal study I found the BBC guidelines for subtitles the ones most helpful for this scenario. [Wiesner, 2022 Layers, p. 18] Unfortunately with game engines and varying screen sizes taking a specific font size as my guideline was not really an option. The BBC recommended a line height of 8% of the screen height. But since the text was not just meant as subtitle I opted to pick a font size slightly larger than that. With the font size in place the next issue was to come up with a good

line length. The BBC recommended 40 letters per line for mono-spaced text. In general this was a good direction to work with. However, with the increased font size I wasn't able to use the full width of 40 letters, as I still wanted the player to be able to read the text without moving their head too much. This in turn resulted in more lines overall. That meant that I had to keep the sentences short. Otherwise the player would lose focus of the screen center. I therefore reworked the text and split up longer thoughts into several shorter ones. I noticed that this display method forced a certain style of writing. Long trains of thoughts were not really suited and neither were detailed explanations. With less then 40 letters per line and three lines at max no displayed message could exceed 120 letters———which is less than half of what twitter allows. Of course it is possible to split up sentences. But since only one message could be displayed at the same time it doesn't make for a great reading flow. I still ended up splitting a few sentences for dramatic effect.

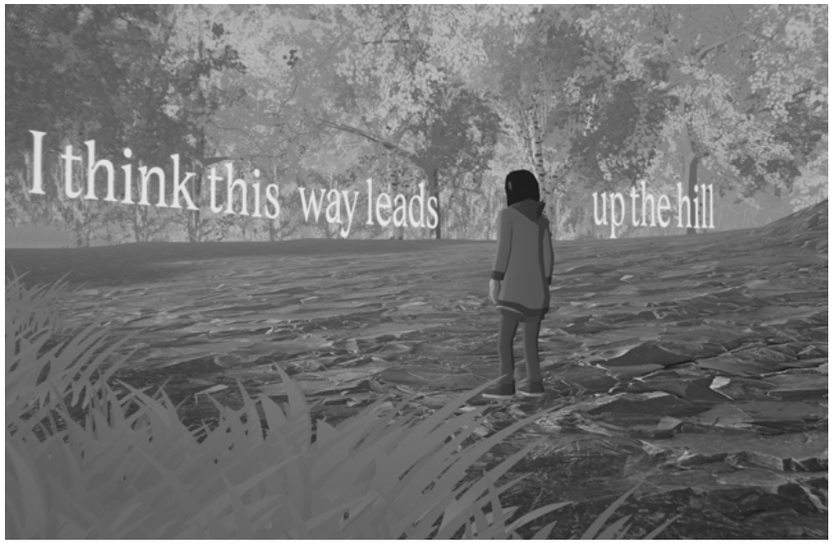
Text lenght of about 40 letters per line.

Another mode of display that I wanted to explore in this prototype was placing text inside the world. I used this technic in two instances where the character is talking about a house in the one instance and a car in the other. The words 'house' and 'car' are rendered inside the 3D world right next to the objects. In a way the typography in these cases acted like a gesture, drawing the players attention towards these objects. In another situation in the game the player comes across a forking path. The character's comments about the two directions are also rendered inside the world. One path leads up a mountain and the text is placed on that path. After a short while the text moves up the path indicating

"Our Car!" is pointing towards the car in world space.

the direction that from the players perspective might not be obvious from observation only. The other direction leads towards a lake that———at that point———is still obstructed behind a small hill. The text is rendered in such a way that it teases what is to come, should the player head in that direction. The last experiment I did in this prototype was to split up a sentence (like mentioned before) to give each word more gravitas. After running around for a while the player will enter a field and the character's excitement is shown by words that appear inside the field in increasingly larger font size. Additionally the text "smashes" into the ground to hammer home the statement.

"This" "is" "AWESOME"— text relating to the space the character is celebrating.



"up the hill" is animated to wander up the way that leads up the hill.

Although I was overall very satisfied with the outcome of these experiments I also became aware of their difficulties. For example if the player was facing the wrong direction they could miss the text animation that was playing the moment they entered the trigger zone. In case I wanted to use depth of field for the camera I needed to set the focus area very generously otherwise I ran into the danger of blurring the text, making it indecipherable in the process. Another issue was that I gave the player no control over the text speed. The text was activated through various trigger zones. Once triggered the text animation would play and vanish for good. I decided against allowing to re-trigger text as it would break immersion in some way. That meant on the other hand that I also needed to space out text in a way that it wouldn't overlap. Meaning trigger zones needed to be far enough apart that each animation could play out till the end [That meant around four seconds for each animation.] In the end I think I was able to find a good rhythm for the text. On the flip side I also had to accept that players might miss a message on their journey.

Visuals

I made it into a challenge for myself to create as many assets and sounds as I could myself. For the modeling I used "Blender," the environmental sounds I recorded with a "Zoom H6" and an "iPhone 6s," the music was recorded and mixed in "GarageBand." Additional assets were taken from the "Unity Asset Store." In order to make all the assets fit together I decided on using a cartoon cel-shader. I edited the albedo maps of the pre-made assets into more monochrome color schemes and used the cel-shader for the shadows. For the landscape materials I started off with single color materials and an attached normal map. But this made the world seem too flat and difficult to navigate. I therefore started to use "Quixel Megascan" textures. The limitation of these textures was that they mostly stemmed from environments other than German forests. Since I only used an albedo, normal and ambient occlusion map I figured I could add textures of my own. So I started taking photographs of forest grounds whenever I went hiking. I later edited them in "Photoshop" to have seamless borders and used the normal map creator (also in Photoshop) to give them depth. The results mixed well with the "Megascan" textures, so I used both———resorting to use the "Megascan" ones as a fallback in case I wasn't able to take a picture of something fitting myself.

The approach to the prototype was to create an atmosphere that encouraged the player to role-play as a child, eager to explore the surrounding nature. For the environment this meant I needed to place visual cues on the way to guide to player through the level. Additionally the player character mentions these points of interest as something she finds interesting. This way the player gets the feeling they are unraveling the forest the same time as the character does.

Sound

Next to the visual cues the sonic atmosphere played an important role in immersing the player in this forest———making the world feel plausible and real. I collected various bits and pieces of sounds over the course of months while hiking. The sounds were recorded sometimes by using my phone (though it only records in mono) other times using a borrowed "Zoom" field recording device from TH Köln. Since I didn't go "hunting" for sounds at fringe times during the day I also caught a lot of undesired noise like driveways in the distance or wind. In the prototype I was rather strict with filtering out these noises. I cut off most of the lower frequencies and high hiss. This resulted in cleaner audio, but it also felt a bit lifeless at times. Additionally it left some of the remaining unfiltered sounds in a strange state where it seemed something was missing. This can be heard most prominent with the birdsong. Before and after the chirping the natural reverb trail cuts off or the frequency sounds too mid-focused. In my initial attempt I also tried to get rid of the wind for the ambient tracks. Later I realized that wind is an important part of the natural ambience during

walks in the forests. I resorted to adding fake winds that I recorded by breathing onto the microphone. All in all the sound in the prototype works, but I learned that for the thesis project I shouldn't be as strict with the frequency cleanup.

> When I implemented the audio into the game world I used a mix of 2D tracks and 3D sounds placed at specific locations. The reason for this was to keep the ambience varied at all times. The 2D environment track consisted of one to two minute loops of ambience. They can be seen as the foundation. The 3D sounds are scattered through-out the map. 3D sounds change their stereo panning and volume depending on the direction the player looks. They also loop, but fade when the player leaves proximity to the audio source. This way the player gets the impression a bird they hear inside a tree is really located in a specific tree. Because when they get closer to the tree the bird gets louder, if they turn away they hear the chirping behind them. I adopted this technique from a talk by Anne-Sophie Mongeau where she explained how she implemented the audio design for Shadow of the Tomb Raider. [Mongeau, 2019]

Next to the nature sounds the sounds coming from the character play a huge role in creating believability. The character's sounds are also important to give feedback to the player. Most important for rooting the character in the world are the footstep sounds. Again I used my own recordings. I tinkered a lot with the sounds. First cleaning the low hum frequencies and cutting the steps into lots of single wave files. I lost a bit of impact by doing this so I EQ'd some of the lower frequencies back in. I also created an array of sounds, that the game picks from at random———sometimes over ten files per surface type. The different surfaces like grass, leaves, gravel, and concrete are triggered according to the landscape material the player walks on. Additionally I recorded sounds of rubbing fabric to also account for the kind of clothing the character wears. I wanted to record breathing as well, but noticed that even in breath the voice of the person breathing comes through. So I wasn't able to use my breath to believably imply that the breath comes from the girl in the game. I decided to leave it out, as it was more of a "cherry-on-top" thing.

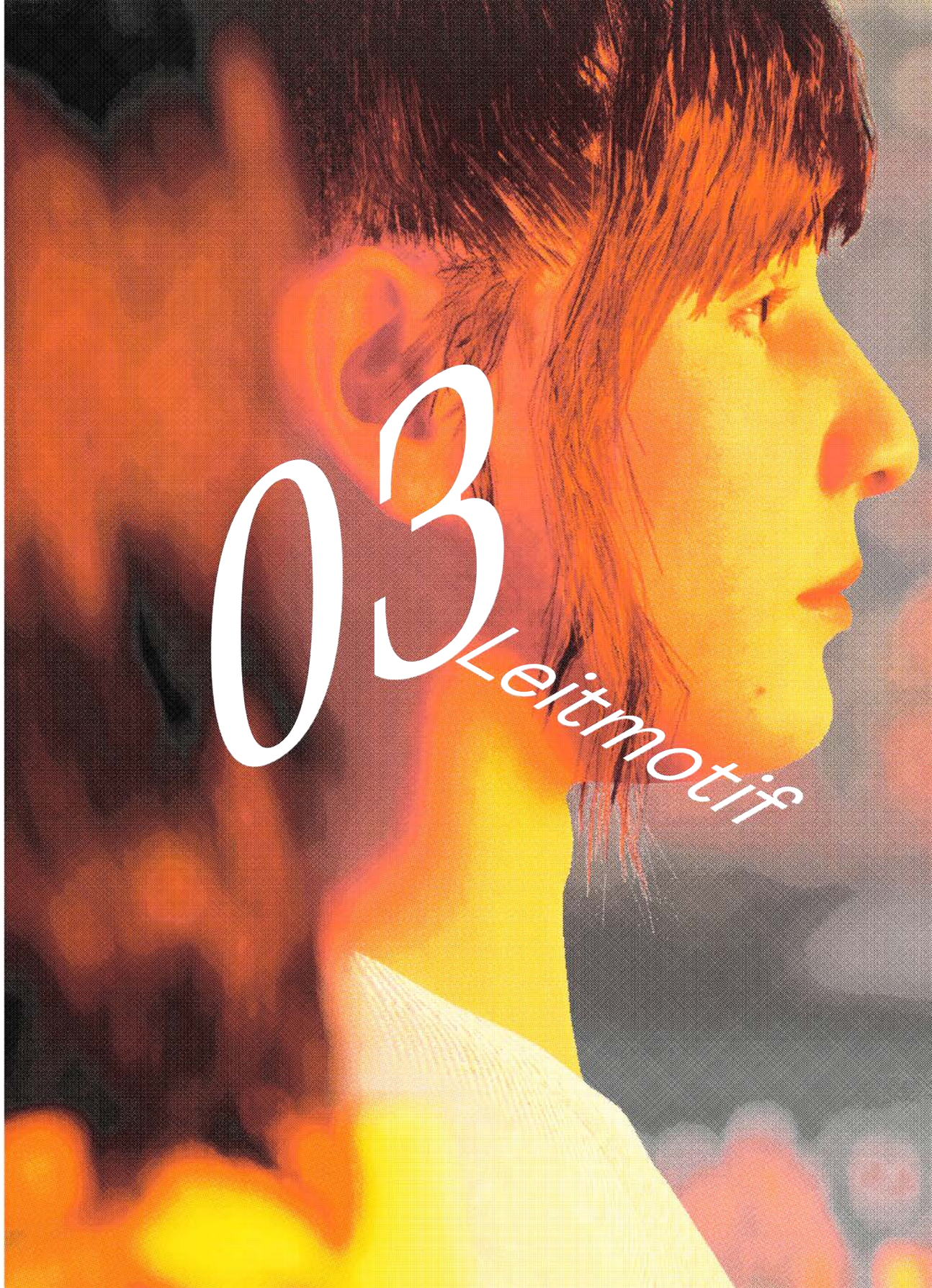
> By this point the various audio elements sounded pretty dry. So I added different reverb zones that give an additional sense of space to the actions. The reverb changes in each zone. So when the player crosses a ravine the sound becomes more closed compared to when they walk in the fields. When mixing the various audio sources I decided to make the character sounds comparably louder to the environment. Not by a large margin, but still distinct. It is a trick often done in games to convey a sense of loneliness or introspection.

Finally I composed an original piece of music for the game. My goal with the music was to create a feeling of mystery and wonder in the opening and a floating feeling for the rest of the game. As electric guitar is my main instrument I wanted to use it for the soundtrack as well. Instruments in soundtracks often have established tropes to transport meaning. Electric guitars in game soundtracks are mostly associated with drive, fury, and a sense of power. I know as a matter of fact that this instrument is capable of conveying many more emotions, so I made it my challenge to make it the lead instrument for the soundtrack. Though the guitar takes a huge part of the composition I also added strings, horns, and drums. In the beginning a wall of sonic textures rise to convey the wonder of a new place. I iterated on this first movement several times trying to find a good balance between mystery and comfort. In the beginning the music was too dark and disorienting, so I got rid of some of the dissonance until I was happy with the result. In other projects I worked on before music was playing at all times of the game. I wanted to try how a few moments of silence work for this game. So after the opening the music fades and the player is surrounded by only the nature ambience. About one third into the level the music kicks back in. For this moment I wanted to convey on the one hand that the character is alone, but also confident. For the player I wanted to create this aforementioned feeling of floating. My reasoning behind this was that with a straighter melody the player would recognize the loop and might get annoyed by the music faster. I also wanted to communicate to the player that they can explore at their own rhythm———there is no urgency. The way I achieved this feeling was by picking a relatively slow tempo and syncopating the melody very heavily. Most of the notes are placed on an off-beat and with varying lengths. That way the listener can't anticipate what is coming next. Still, I made sure that the melody can be hummed along. For the third movement the character is overcome by the urge to include someone into their own discoveries———she wants to share what she has found. The third movement is based on the same melody as before but this time drums and strings accompany the guitar. Additionally the horns feature some of the melodic highlights. All the while the music keeps the floating feeling, but now gains more direction.

Findings

Overall I was very satisfied with my prototype. It was rich in atmosphere——— something various play-testers pointed out———, the typography was well readable, and my experiments with text-placement were met with positive regard. I wanted to include a house for the small lake at the end, but was never able to create or find one. Till this time the space is still empty. This is even more sad as the character specifically points out that it is "such a cool house." I decided to leave the remark in, because the Animation was so nice. Another issue was that it is possible to get out of bounds. As a one person operation I focused my priority on making the intended path work. Therefore I was fine with

unintended paths not being accounted for. For the music I had the problem that due to the slow tempo and long loops it was difficult to time changes. They happened when the loop was done and that meant the musical shift was not always traceable to what happened in game. For the final project I also wanted to commit to more “imperfections” in the audio design. It is very rarely that you can’t hear reminders of civilization when hiking in Germany. And even though it is nice to create a more idealized version of reality in games I think this level of rigor when treating the audio wasn’t necessary. Something I didn’t go into detail before was getting the third-person character controller to work smoothly. For the final project I was hoping for an easier implementation, that also controls less stiffly.



Stay

While “Going through a Phase” was focusing on how to make text work in a monolog, for the final project “Stay” I wanted to explore this prominent, involved text in a dialog situation.

As engine I picked Unreal Engine 5.03 and switched to 5.1 later in the process. Since up to this point the only thing I did in UE was implementing audio I basically had to start from scratch. Solving technical issues was a large part of what I had to struggle with during the production. Solo-development always puts a lot of strain on the developer, as it means they can’t solely focus on what they are actually good at, and instead have to become knowledgeable in areas they are not apt in. I was very fortunate that I get help along the way by Lewin Hustedt, who helped me program the dialog system that the whole game hinges on. Still I lost a lot of time with solving issues outside of my area of expertise. For this documentation I will mostly omit the programming side of this project. My focus will be on typography, audio implementation, and story.

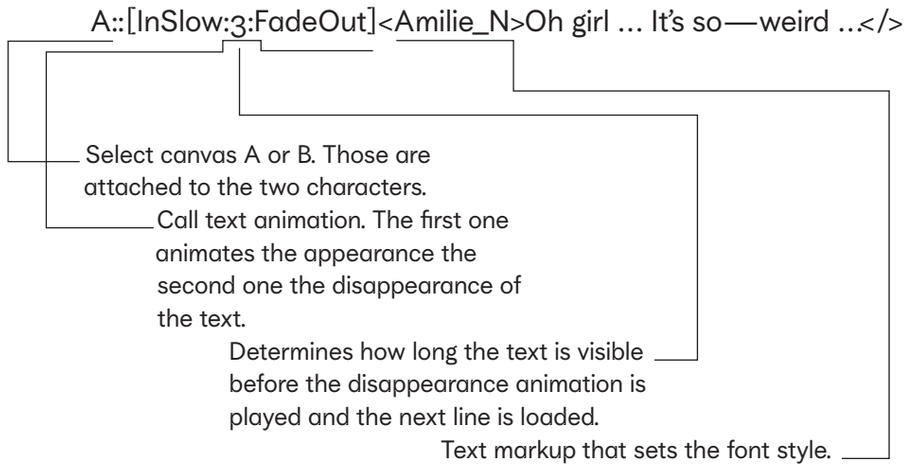
Typography

The usage of typography is the main research focus of this thesis. And though the game is a fully functional experience, the surplus value lies within the somewhat novel approach to displaying dialog as text in a 3D environment. In order to untangle the typographic aspects of the game I will observe them under the premise of gameplay first and expression second.

Typography: Gameplay Aspects

In UE5 there are two different text widgets: the “UMG Text Block” and the “UMG Rich Text Block.” While the first one gives you full control over how to stylize your text it is somewhat limited as the style you select is static. The Rich Text Block allows dynamic text stylings through markup text. In the created markup style font family, size, letter spacing, and color can be selected. Though it comes with the caveat that these elements can’t be controlled via the animation tool. Rich Text Block only allows to manipulate the text container, not the content. For my purposes the Rich Text Block still was the better solution, as I wanted to use different fonts for each speaker, and a variety of font weights all in the same text box. This left me with a more rudimentary animation toolset, that nonetheless sufficed for my application.

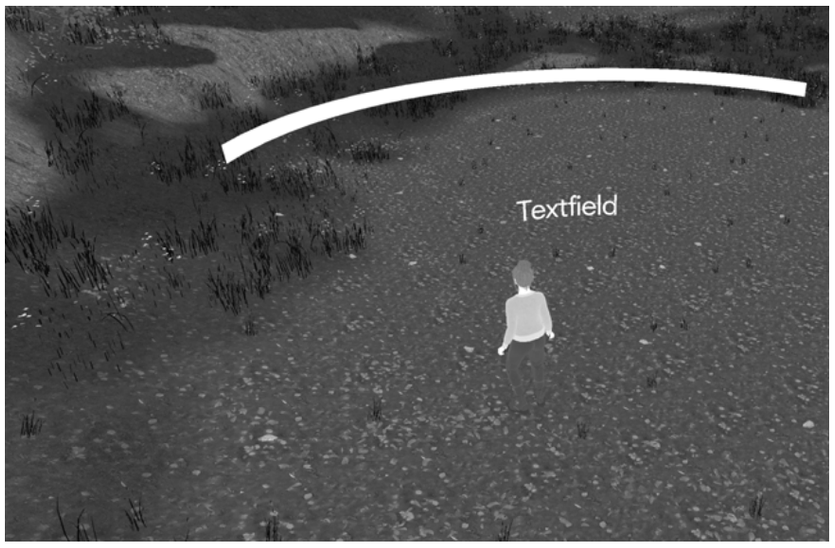
With the markup text covering the assignment of different fonts to the two speakers, I needed an additional script that would cover loading the next line of dialog, assign animations, and project them to the corresponding canvas of the character. As mentioned the credit for this blueprint goes to Lewin Hustedt. His blueprint allowed me to implement the dialog in a scaleable manner. Meaning I was able to change text pretty easily, edit times, and also trigger different scenes. The blueprint is based on a rich text table. The syntax inside this table is comprised of a number of selectors that are working as follows.



Inside the text animations I also included an event trigger that was able to change the music and activate the credit roll. Additionally I used trigger volumes in the 3D world that get activated when one of the characters steps inside.

The script loaded the next line automatically. No player input was needed. This also means the player had no control over how fast the text appeared or how long it was visible. It is closer to a real conversation where it isn’t possible to pause either. Contrary to a real conversation raising a query isn’t possible———this would need a more complex script and lots more writing. It has to be seen as a compromise. But pressing a button to continue a conversation isn’t that engaging a form of gameplay to begin with———at least in my book. It is a legitimate form of accessibility though. So, for a released game allowing to call the next line via button press would be an option worth considering in the accessibility menu.

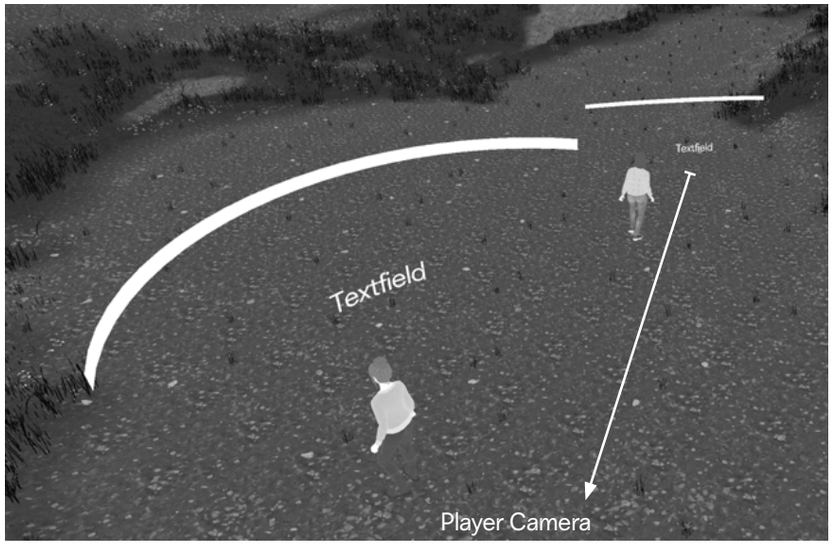
With the dialog running and projecting to the two characters, the next thing that I needed to figure out was how to present the text inside the game world. I knew already that I wanted to stick the text to characters, and that I didn't want to project directly onto the camera lens. I found a way to place an UI canvas inside the 3d environment, as can often be seen when placing videos or interfaces inside a level. This also allowed me to use an emissive material———giving the text a light glow. This feature is not available when projecting on the UI layer, as the UI is exempt from post-processing effects (Where the bloom is located that gives light effects its shimmer). I attached a canvas to each of the characters and also used a feature to bend the canvas in a concave shape. This way the text is not just flat but features a small curve———giving it a more three-dimensional feel. This approach came with the problem of aligning the text correctly. The issue here was that instead of giving the canvas only a slight bend this option changed the flat plane into a half-barrel. This meant that the text on edges of the canvas took a too steep angle to be readable anymore. I therefore had to move the text somewhat close to the center, but not so close that a bend in two directions appeared. All of this resulted in the canvas being very unintuitive to position on the screen. The anchor of the canvas was positioned on the on the left side of the "half-barrel" but my text was in the center that was deformed backwards. Meaning instead of just placing the canvas where I expected it to go I had to position it via trial and error.



"half-barrel"-shaped text canvas marked by a line on top. After adjusting the size I also reduced the amount of curvature on the canvas to make text placement easier.

And while all of that was manageable for the player character the second character came with its own set of challenges. For the player character the position of the canvas was bound to the camera position. Meaning the text always took the same screen space no matter the direction the camera was facing. For the second character I wanted the position of the canvas also to align to the player camera. But that meant I had to set the canvas up in an inverted style. To illustrate what I mean one could think of the canvas as a newspaper. The camera is showing what the player character is reading by filming the paper in a looking-over-the-shoulder kind of way. If I wanted to use the same approach with the second character this would mean that the body of the character would obstruct the view on the paper most of the time, as the camera was not stuck to her shoulder but that of the player character. To get around that issue I could position the canvas higher———like holding the newspaper up so to speak. But this meant the player would either need to keep a lot of distance to the conversation partner or look up for the better part of the game. Another approach was to face the canvas to the player instead of the second character. To stay in the metaphor the second character would turn the newspaper, so they can't read it anymore to show it to the other person. This was the option I went with in the end. It meant I had to mirror the text on the canvas and get it to align with the player camera. Both things were possible, but getting the text to display in the right distance and position to the character again was a lot of trial and error. For future applications this approach works but it is less than pixel-perfect and fine-tuning it is difficult.

The next challenge was the text size for the text of the second character. As the canvas of the player character was always positioned in the same distance to the camera controlling the size was easy. With the second character the difficulty was that, while the distance between it and its canvas was always static, the same was not true for the distance of the player to the second character canvas. This meant the text size seemed larger or smaller depending on the distance to that character. In the beginning I tried to find a middle ground, but was ultimately dissatisfied with the result. In essence going with a static size kept the player at distance during the conversation. In case I made the font size small enough to work at close range it would become difficult to read as soon as the player would fall further behind. If I made the font size larger the size would be too large to fit the screen space at close range, resulting in the player moving further away to read everything. In the end I found a passable solution by scaling the second character's canvas based on the distance to the player. At certain break points the canvas jumps to a larger or smaller scale. Ideally the size would scale smoothly, but I lacked the programming skill to implement this. The result I ended up with is acceptable to me, as it allows to read the text at an appropriate size no matter how close the player is to the second character———which was the goal of this undertaking.



Eline's canvas orients towards the player camera.

For text in games it is common to have some sort of animation tied to its appearance. Very often a form of typewriter stylization is used wherein every letter appears like they are typed in at that moment———though the writer in question would need to be exceptionally fast with typing. It serves to indicate to players that something new happens so that they deploy their attention towards that new information. I personally am not too big of a fan of this mode as I think it makes reading the sentence as a whole difficult. Because as long as new letters appear the reader's eye is drawn towards them. Effectively this means you watch the whole animation play out before starting to read. And since the effect is linear longer sentences take longer before they can be read, slowing down the process unnecessarily. With the Rich Text Block I only had a limited amount of animation potential at my disposal. Making the letters appear in this way would certainly have been a possibility, but like I said I'm not that big a fan of this style to begin with. Another way could have been working with a fancy dissolving shader, that makes the letters appear like they are etched into existence. While that would certainly look interesting I didn't find any justification story-wise for such an effect. The setting was a grounded conversation in a forest between two old friends, so this mystical effect wouldn't seem warranted. I opted to work with simple fade animations, as I felt those would represent speaking best. Additionally I changed the scale of the text in the animation. A quick scale up to give a "pop-up" effect together with the fade-in and a slower scale-up to create the effect of the text moving towards the player afterwards. I created three different animations that resembled different intensities of verbalization: A quick animation with a larger text scaling as a form of yelling or speaking in excitement, a medium-tempo animation for regular speech, and a slightly slower animation for the more pensive sentences during the conversation. The fade-out animations I kept simple: just a quick fade out with a scale down to the original size to prevent jumping in text size.

As the tempo in which the text appeared couldn't be manipulated by the player finding a good rhythm was necessary. I wanted the text to emulate the flow of a conversation, while still being readable for a first-time reader. Finding said rhythm was complicated by the fact that I was very familiar with the text, due to me reading it many times as well as being the author of the lines. In a video editing job I did some time ago, someone taught me the trick that when you are able to speak the lines instead of reading them in your head you get a good estimate how long the text needs to be present on screen. I applied this knowledge and simply started with some estimates and then played through the game while speaking the text that was shown. That way I found the moments where a sentence needed more wait-time and the ones that could do with less. At the end I developed a good feeling for how much time a sentence of a certain length needed, so I was able to deduce a fitting time without speak-testing the whole conversation every time. Of course I still had to check, but it made the process more scalable. Otherwise I would lose a lot of time just running the whole game to adjust the text in the latter half.

A process that would take about ten minutes for each run.

Finding a good font size for the text on screen is a difficult prospect. Not that reading on screen works completely different to reading on paper. The issue lays more with how difficult it is to use relatable metrics. Yes, there are many factors that influence the printed result, like different base glyph size or x-height. But the result of a 12pt-sized font is somewhat predictable on paper. Inside a game engine font size is not a reliable metric what so ever, as it is not the only determining factor. Even more so when the text is not just rendered on the UI layer but inside the world. The text size in engine is influenced by the font size, the

text box scaling, the canvas size and scaling, and finally the distance of the player in the 3D world from the text. In my proposal I explored different guidelines for figuring out an appropriate font size on screen. [Wiesner, 2022 Layers, p. 17–20] I found the guideline for subtitles by the BBC to be the most applicable in general. But for the 3D situation that I created in the game it only would only go so far. While the resulting font size for a screen coverage of 68% of screen width would work well enough for subtitles, in the game world the text would struggle too much with all the surrounding assets———it would simply be too light in weight. The font weight of the typefaces I created was a little bit thicker than a normal regular, but still far from semibold. This meant I had to pick a larger font size as a result. I still stuck with the around 68% screen width for the player character's text, as it was well perceivable in one glance, no matter if displayed on 13" or 27" screens. Due to the larger font size I had to use shorter line lengths. Instead of the 40 letters that I calculated in "Layers of Text" I ended up with roughly half of the amount of letters per line. [Wiesner, 2022 Layers, p. 30.] Even though this way the reader can't gather as many letters as possible per eye fixation, the text is still well readable due to the way the text is sliced up. With the short lines there was a limit on how many letters could be shown at once. With five lines á 20 letters I had to adjust my writing to accommodate for these limitations. I shortened sentences a lot or separated a single sentence into various thought units. With this shorter sentence structure getting overwhelmed or tiered by the text was less likely. This also worked because the kind of reading this created couldn't be compared to reading a book or subtitles: It neither had to create a reading flow similar to a book, nor did it force the reader to jump between action and text like is often the case with subtitles. Instead it created a rhythm of navigating/taking in the game environment and reading the text without ever taking the eyes off the road———so to speak. As such the reading experience came closest to reading road signs or billboards. The text size of the second character was a concern though. In comparison it was a lot smaller than the player character's speech. Inside the setting of the game I think it still works: It connects the text to the speaker and creates an impression of locality. Similar to 3D audio where a sound source closer to the listener is perceived louder.

All in all I had to make a compromise here. On the one hand the reading situation was not ideal and would probably need accessibility options for a shipped game. On the other hand the approach allowed for a very immersive reading, that allowed the player to stay on the action at all times while still being able to read the dialog.

Another issue with the 3D-placed canvases was font resolution. For a very long time I had pixel artifacts especially on the curves of the font. After several attempts to fix this I figured out that the anti-aliasing that smoothed the letters was only applied to the 2D canvas. The 3D canvas didn't receive another pass but displayed the 2D canvas "as is." This was an issue as I used a relatively small font size on the 2D canvas in order to control the how curved the text appeared on the "barrel-shaped" 3D canvas. After I tried several different scaling options I ended up simply increasing the font size on the 2D canvas and adjusted the size of the 3D canvas accordingly. My issue before was simply that I used a very zoomed in image of the font, and displayed that image up close———revealing all the artifacts common to zooming in.

Excourse: Importing Fonts

Importing fonts doesn't seem too complicated in either game engine———at least at first glance. An issue that I had in both engines is that .otf-fonts only work in a very rudimentary fashion or not at all. The working standard are .ttf-fonts. This meant that many of the opentype features were not accessible. But this wasn't too much of an issue as opentype features weren't supported by the engines anyways. As I mentioned in my proposal getting kerning to work requires a deprecated kerning table format. The issue with the kerning table is that the number of entries is limited. This can result in having to omit some diacritics from exporting. For the basic latin it should work and would still allow for a respectable amount of kerning combinations. I didn't know of any other way to write the kern table but opening the font with a font creation software like "Glyphs." Under user parameters one can add the line "Write Kern Table." This command doesn't show up in the suggestions and I only found out about it by searching through very specialized forums. The inaccessibility of this knowledge explains why so few games sport font kerning at all.

Another difficulty in importing fonts is to get ligatures to work. Fortunately Unity and Unreal Engine both allow to activate ligatures by now. Unfortunately the process is not that obvious. In UE this feature can be found in the Text/Rich Text Block widget. Near the bottom of the list under "Localization" is a box named "Text Shaping Method" with a dropdown menu. This box needs to be ticked and inside the dropdown menu the option "Full Shaping" needs to be selected. Not only does this activate the kerning it also enables basic ligatures. Fancy custom ligatures are not included though. But it allows to access the standard ligatures. In Unity the process is different. As long as the .ttf-file has a written kern table included you can import it at the creation of the font atlas. In order to

activate ligatures you need “TextMeshPro 3.2.0.” At the time of writing this version can’t be installed from the established sources but needed to be searched for and downloaded via a GitHub link. This version then allows to create ligature pairs. In that regard it is more capable than UE because the amount of glyph combinations is not limited to the standard ligatures. The issue is finding the right glyph index entry for the ligature glyph. [Stephan_B, 2022] I personally had trouble finding the right glyph index. In Glyphs the index didn’t match with what was displayed in Unity. The author of TextMeshPro used FontCreator (windows only) and mentioned that FontForge might also be able to display the glyph index. I wasn’t able to confirm this, due to a lack of trying as I got the three ligatures I needed to work by trial and erroring index numbers. After that I moved on to work in UE. I was able to open my fonts in FontForge at a later timer and saw some sort of index entry, but can’t confirm it was the correct number. In summary it can be said that the functionality in Unity is better than in UE. The same cannot be said for the ease of use. Regardless, in order for the ligatures to display correctly in either engine thresholds for letter spacing needed to be considered. If the spacing was manipulated over a certain amount the ligatures would also dissolve. This was also the case in programs like “InDesign” and made sense in the context, as otherwise the letter balance would look off.

——Typography: Expressive Aspects——

Based on the research and experiments I did in my proposal “A Voice through Type” I wanted the characters of the two friends reflected in the typography. Additionally I wanted to express certain moods and vocal inflections through variations of the two typefaces. In order to draw a meaningful connection between the characters and their assigned typefaces a few words about their backgrounds are due. The player characters name is Amilia and her friend is named Eline. They both come from the same rural German small town. While Eline stayed in this town Amilia was drawn towards an unnamed major city. Amilia can be characterized as sophisticated, jumpy in mood, and with an air of an art student (Think of a stereotypical Berlin-based student in mix and match second-hand clothing). Eline on the other side is a bit more settled, but still retains an edge. She embodies the type of person that kept a very open mind towards the wider world, while still being firmly rooted and satisfied with living in only a small part of this wider world.

In order to translate these traits into typography I decided for Amilia to go with a sans serif. Especially through the “International Style” established during the 50s–60s of the last century sans typefaces (in particular Helvetica) tend to be ascribed a universal and international appeal. The “Global Style”——a term coined by Jeffrey Keedy [Keedy, 2001, p. 37–42] ——revived that grid- and Helvetica-based style in the last decade and rededicated it from global business into a signature suijet of the art and cultural industry. In that regard I thought it fitting to use these associations with Amilia. Though it has to be stated that for the contemporary art scene deconstructed typefaces opposed to the clear letters start to become more prevalent. Still, the global style left its mark on the culture sector and can be seen in posters to this day. With that being said, Helvetica with its static and balanced shapes seemed too cold for Amilia. That’s why I opted to go with a type design based on the American form principle or gothic sans as described by Lilo Schäfer. [Schäfer, 2019, p. 68] I personally found gothic sans often harbored a certain quirkiness that relieved them of the stiffness that is inherent to grotesque sans typefaces. Other than that Gothic Sans are characterized by their high stroke contrast, large x-height, as well as the wider uppercase letters and the angled——and thus more open——counters. I picked this form principle because the high contrast would mirror the somewhat jumpy temperament of Amilia. Additionally I thought a form principle that is inherently American without the imperial undertones of homogenized global business would underline her interest in the wider world.

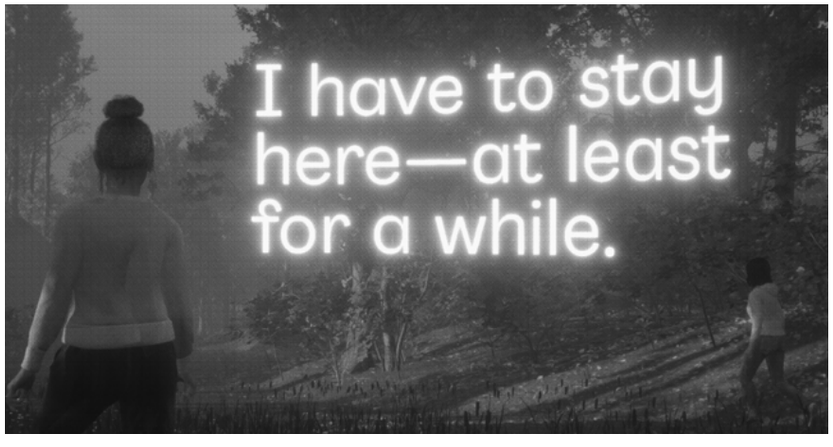
For Eline I wanted to design a serif typeface. Serifs are often associated with tradition. This is especially true for serif typefaces not based on the modern form principle. [A typical typeface based on the modern form principle would be Didot. For a full listing of traits refer to Lily Schäfer’s Model as laid out in: Wiesner, 2022 Voice, p. 26–27] The problem with many modern form principle based fonts is that they are better suited for headlines, as their high stroke contrasts become tiering to read in bulk text. I still went with a modern serif design and adjusted the contrast to be more easy on the eye. This reduced contrast also fit well with Eline as she is not a high-fashion minded person——an association modern serifs often create. I also wanted to show that while Eline is traditional in regards to staying in the town she grew up in, she also retains an edge that bares traditionalism. I decided to show this in the shape of the serifs. Different to the “bowled” serifs of the transitional or humanist form principle I added a kind of wedge-shape. I also retained the flat-stroked serifs typical of the modern serif, but combined them with the aforementioned wedge angles.

Next I wanted to create variations of the typefaces that would show the timbre or inflection of the spoken sentence. The normal weight would be used for the most part of the dialog. But during the conversation the two also become angry, joke around, and are grasping for words. These three additional states I wanted to reflect in the typography.

For the angry timbre I developed an idea pretty quick. For Eline I wanted to work with overemphasized spiky serifs. I saw this feature used in another font and thought it would translate ultimately pretty simple to implement. The only difficulty being the spacing completely being thrown off. But as it

Amilia Regular

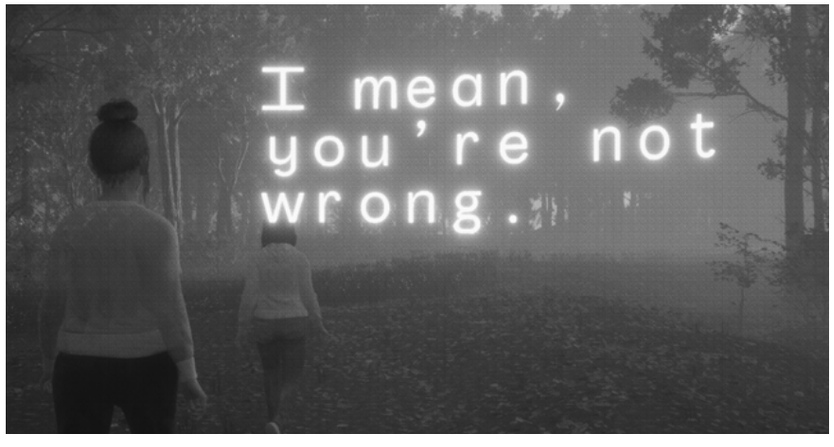
A a B b C c D d E e
F f G g H h I i J j K
k L l M m N n O o
P p Q q R r S s T t
U u V v W w X x Y
y Z z



Amilia Regular used for neutral speech.

Amilia Mono

A a B b C c
D d E e F f
G g H h I i
J j K k L l
M m N n O o
P p Q q R r
S s T t U u
V v W w X x
Y y Z z



Amilia Mono used to show speaking while thinking.

Amilia Distorted

A a B b C c D d E
e F f G g H h I i J j
K k L l M m N n O
o P p Q q R r S s T
t U u V v W w X x
Y y Z z

Amilia Distorted used to show anger.

Amilia Wonky

A a B b C c D d E
e F f G g H h I i
J j K k L l M m N
n O o P p Q q R r
S s T t U u V v W
w X x Y y Z z

Amilia Wonky used to show irony.

Eline Regular

A a B b C c D d E e F
f G g H h I i J j K k L l
M m N n O o P p Q q
R r S s T t U u V v W w
X x Y y Z z

Eline Regular used for neutral speech.

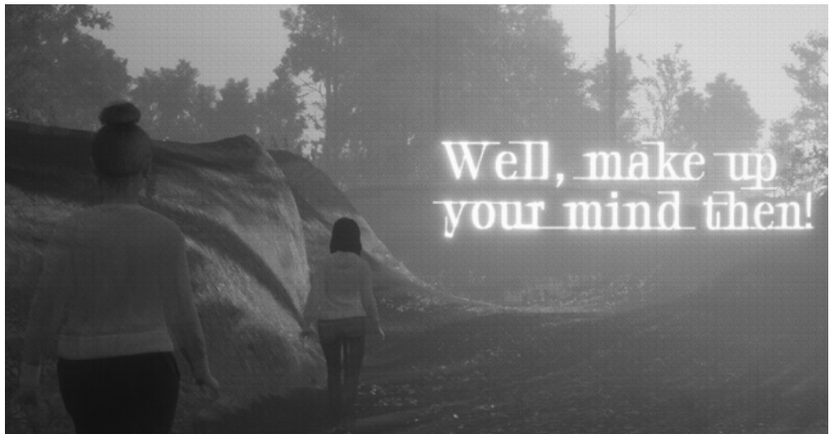
Eline Mono

A a B b C c
D d E e F f
G g H h I i
J j K k L l
M m N n O o
P p Q q R r
S s T t U u
V v W w X x
Y y Z z

Eline Mono used to show speaking while thinking.

Eline Spiky

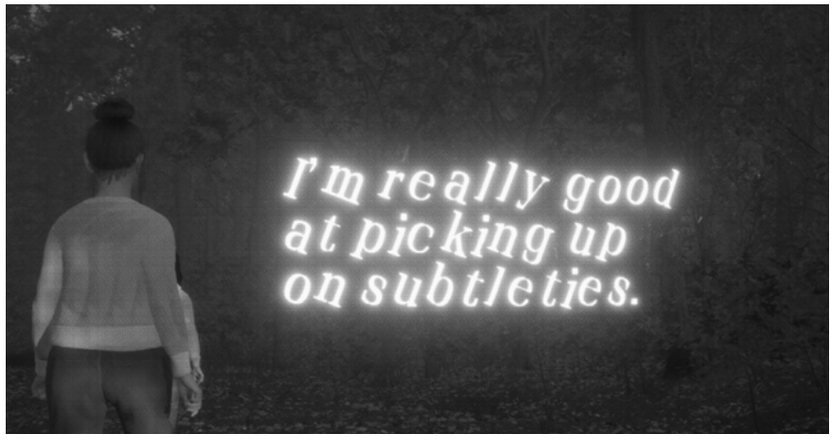
A a B b C c D d E e
F f G g H h I i J j
K k L l M m N n O
o P p Q q R r S s T
t U u V v W w X x
Y y Z z



Eline Spiky used to show anger.

Eline Wonky

A a B b C c D d E e F
f G g H h I i J j K k L
l M m N n O o P p Q
q R r S s T t U u V v
W w X x Y y Z z



Amilia Wonky used to show irony.

was meant to show anger———not a smooth emotion at all———these added dents wouldn't hurt. For Amilia I wanted to work with deformed curves that would create overlap and negative contours. At first I tried to work with adding curves to the stem of the letters and moving them across the opposite line of the stem to create the negative contour effect. Soon I found this effect to be too soft to express anger. I tried again, but this time instead of adding curves I kept the added artifacts sharp and angled. This gave the letters a sort of shattered glass look. To me this seemed very fitting to express anger. I tried to make sure that despite the distortion the letters would remain legible. I also omitted some letters from the distortion to keep some contrast.

The next emotion that I wanted to show was the pause or grasping for words that often happens in a conversation——— especially if someone reflects on their own feelings. In spoken conversations this results in some syllables and letters being drawn out, accentuation placed too late, and additional pauses added to the sentence. I thought working with an overemphasized monospace setting could replicate this effect well. Especially if not all letters are placed at the optical center of the glyph space. This would create an uneven rhythm while reading. After a few tries I figured out that I needed to avoid moving letters to the end of the available space, in order to avoid gaps at the beginning of the line. Apart from that the font variation created the effect that I was looking for. I still had to do a few iterations in order to find a good monospace glyph width. Too small and the effect wouldn't translate well. Making it too wide and the words would fall apart. I also decided that I wanted some variation of that effect with Amilia's font. Initially I tried a version with extended wide letter shapes. But this lacked the effect of the off-kilter rhythm that speaking while thinking creates. It was simply too harmonious. I then decided to also use the monospace approach but mix it up a little bit. While I kept all the letters in Eline's font the same for the mono weight I stretched the uppercase letters with straight bars to fill out more of the glyph space. I also kept the letters overall a bit more centered in comparison.

At some point in the conversation the two speak in a joking tone. I thought a good way to emphasize this emotion would be to simply rotate the letters. This would create a silly look that would fit with the lighting of the mood the joke would have in the spoken conversation. The difficulty here was to find a good compromise in the alignment. Overall this worked out better with Eline's font than with Amilia's. In the latter the letters sometimes form a wonky baseline. Which isn't too bad and still translates the emotion. But from a type design perspective it would need to be reworked, so it would be useable as a standalone typeface.

In play tests I did with a few of my fellow students they noticed the different font alterations and tied them back to reflecting different emotions. Though several testers asked me specifically if they represent emotions. My guessing for these questions comes down to a) being unsure if it is just a "design play thing" without deeper meaning or b) typography being used to display inflections in voice not being an established shape language. When I explained my reasoning for the different choices they agreed. So as far as I can tell, their associations overlapped with my intention: The emotions I wanted to show are perceivable in the typeface variations.

Story

Amilia is back in the small town where she grew up in. But not on her own volition. For reasons undisclosed she was forced to come back. Now she is confronted with living in the countryside again—a prospect she deliberately tried to avoid by moving to the city. Eline—her longtime friend, who still lives in that town—asks her to go for a walk in the nearby forest. A place that they often visited in their youth. With Eline's help Amilia tries to untangle her negative feelings towards her hometown.

Initially, I was on the lookout for an existing story to use in the game. I searched for a small theatre piece or short story with a length of about five to ten minutes. As I wasn't too experienced with either medium I reached out to different writers and acting students. Unfortunately none of the people I contacted were able to point me towards a piece I could use. I also read a number of stories in order to scout something. But this didn't yield useable results either. Thought was able to find found one piece that I really like the style of writing in. I contacted the author if she would have more stories like this———as the story in question only featured a very small amount of lines of dialog. Unfortunately this also turned out to be a dead end———though the author was generally interested. At that point I my search was already taking more than a month so I finally decided to write something on my own.

I took a visit with my parents as the opportunity to write a first draft. As the basis I took my own feelings of discomfort with being back in the small town world of my family. The first draft was basically a conversation with myself that I split into two characters. After reevaluating it I found it lacked in tension and also didn't feel very natural. For the rewrite I included more fighting as that would allow me to apply the typographic variations that I had in mind. I also made the topic more about city and countryside. After sending the draft to several other people for feedback I made another edit. The conflict I planted came out of nowhere and made the characters seem dislikable. I also wanted to use the conversation to reconcile some of my own ambivalence towards the countryside. In the edits I received it was pointed out to me, that the second character (the arguing for the positive sides of the countryside) felt too wise in her arguments. So, I tuned these parts down a notch. To balance the situation a bit more I also made the player character react in a way that I found more realistic: She listens to the arguments, maybe even agrees to some of them, but doesn't immediately change her mind about her hometown. Overall I wanted to draw an arch with this story that goes from discomfort to gaining a new perspective.

Transcript

(Amilia) (Eline)

Oh girl … It's so—weird …
Umm ... What is?
Being here.
Being BACK here—it's weird.
Huh? Why?
Because …
I was so DONE with this place.
You sure don't seem to be done!
...
I thought you would enjoy coming back here.
We always used to go for walks in these woods—
at least back then.
That was then and now is now …
Okay, fine. Seems bringing you here was a bad idea.
I'm …
not sure.
Well, make up your mind then!
Easier said than done!
Maybe we should just go home.
I don't want that either …
What DO you want then?
Let's …
just keep walking for now.
Fine.
Look, I'm sorry.
I didn't mean to spoil the mood.
It's just—I don't want to be back HERE.
Apology accepted.
I for one am really happy you're HERE.
Really?
Well, I'm glad that at least I get to be with you again.
Makes this situation a lot more bearable.
These woods also help.
I forgot how soothing they have always been.
Yeah. They are, aren't they?
So, why did you leave all of a sudden?
Well, to be frank:
This town—

I've just always felt so suffocated.
Nothing new ever happened,
nothing interesting,
and everyone seemed so complacent.
Sounds like someone's still holding a grudge.
What gave it away?
I'm really good at picking up on subtleties.
Haha. Very funny.
Ditto.
...
Anyway, I don't think this town is the problem.
Uh, what makes you think that?
Places are only a canvas.
They just offer different opportunities.
Seriously?
What kind of opportunities could a place like this offer
... compared to living in the city?
I think they can't be compared.
They are simply different.
Yes, and cities are better!
For some things ...
But, for how much city folk claim to love city life they sure complain a lot about it.
What do you mean?
You know—parents wanting to move outside the city ...
... people wanting to get back to nature ...
These kind of things.
I mean, you're not wrong.
And living in the city is expensive.
If you don't have the money you can't enjoy most things.
I guess you have a point there ...
And then there is the loneliness ...
Hey, loneliness is also an issue in the country side.
Sure.
Though I think it's easier to cope with it here.
It is easier to get to know the people around you.
...

I think what I wanted to say is:
Give our town another chance.
You can have good experiences no matter where you live.
I'm still scared this place will swallow me whole if I stay.
I don't want to become so ... boring?
I don't think most people here are boring.
They just have a different idea of a happy life.
Well, I guess so.
Besides, being happy doesn't mean you can't have goals anymore.
You've got a point there.
...
Anyways, not like I have a choice.
I have to stay here—at least for a while.
But honestly, thanks for trying to warm me up to it.
You were a piece of work.
You're exaggerating.
Regardless ...
I think I will try to make the best out of my Stay

Sound

As I need to pick my battles in order to keep the scope of this project manageable, I decided agains recording my own sound effects. Instead I used the audio library "Soundsnap," that I was able to get an education code for. I used around 90 different sound files that I downloaded from the library for the game.

The majority being character related sounds, like footsteps.

I tried a few different settings for the time of year and day but in the end settled for a sunset in late summer/early autumn. I consulted a bird atlas to figure out what birds would sing at that time. The sites hinted at larks and nightingales. When testing this in game it sounded too much like the mid of day during spring. So I decided to leave out the nightingales and instead added pigeons. I noticed their song a year ago when I did field recordings in autumn, where I mistook it for owl hoots. Later I learned that the cooing of pigeons was a sound that some people that grew up in small towns connect to the countryside. I also searched for sound files that depicted a general forest

ambience in Germany. As is the case with most stock-sites, they mostly feature settings in North America. I was still able to find a few recordings that I mixed and matched. This time I also didn't filter out the base hum in the recording, as I noticed the hum caused by civilization to be present at all times when I went hiking. I also decided to include cicadas, as I heard them still being active pretty late into September of 2022.

Initially I worked with FMOD to implement the sounds. But after I encountered a bug that was caused by some unzipping artifacts created on MacOS I ditched the plugin and went with the integrated MetaSounds. Even though I was able to resolve the FMOD issue a week later, I was at that point already invested in learning the other workflow. In summary I have to say MetaSound isn't as user-friendly———especially as it lacks the DAW interface that many musicians are familiar with———but it is very versatile. I was able to replicate almost everything I prepared in FMOD before. One of the things I wasn't able to replicate immediately was the usage of scatter instruments. Those play sounds in random intervals, volumes, and distances. The reason I wanted to use this feature was to create randomized sounds of birds that wouldn't become repetitive and be less work to implement. I now know that MetaSound provides this feature as well through MetaSound Colors, but since at that point I had already scattered sound sources by hand the work reducing aspect wasn't relevant anymore. In essence the setup I used here was the same I used in my prototype: A 2D background track as the base acoustic layer, 3D sounds placed in the world to create a sense of direction for the environment, and the noises the characters emit.

As there was no voice acting it fell upon the music to do the heavy lifting in aligning the player's mood with that of the characters. I opted again to use the electric guitar as the main instrument. This time I decided agains using orchestral instruments as texture but instead heavily processed the guitar signal. Additionally I asked a friend of mine to record and program some percussions for the movements.

The cold opening was comprised of a cacophony of sounds that was followed up by a driving but discomforting musical motive. As the player is thrown into the situation without any context I felt this tense instrumental rise was really fitting. After the rise faded out the conversation immediately started. Since Amilia wasn't comfortable with being back in her hometown she was seething. She communicated her anger quite unfiltered resulting in the conversation having a really tense atmosphere. After the two of them agreed to walk on in silence for a while, the music transitions into a very reduced musical movement. A bit of the tension still remained in the harmony, but overall the mood became calm. During this calm section Amilia apologized to her friend and started to reflect on why she reacted so fierce. The next musical movement is triggered when Eline tried to warm Amilia up to the thought that not everything is bad with living in the countryside. The music becomes more active again the harmony gets an open and more hopeful feel. At the climax of the conversation Amilia decided to give up on her anger for now and tried to take ownership of her forced time in her hometown. At this moment the music transitioned into the outro: A driving pop-rock song with full drums. At the same time the credits roll, creating a form of relieve of tension and conclusion to the game experience.

In order to make the transitions between the different movements reactive while retaining musicality I worked with a mix of various transitions. I took the idea of in-between transitions from a talk by Sho Iwamoto on the music design for "Final Fantasy XV." [Iwamoto, 2017] There he described that in order to move from the high action of a battle theme to the outro he added a looping pre-outro to the track. I included this idea in the transition between the first and the second movement. Additionally I worked with bar markers that would allow me to transition out of a musical movement before the full loop would finish. I added these in "Reaper" and called them in MetaSounds. I was quite happy that I was able to create this functionality with MetaSounds, as I previously thought this only possible with FMOD. [The blueprint was based on a tutorial of the UE sound engineer Dan Reynolds. Raynolds, 2022]

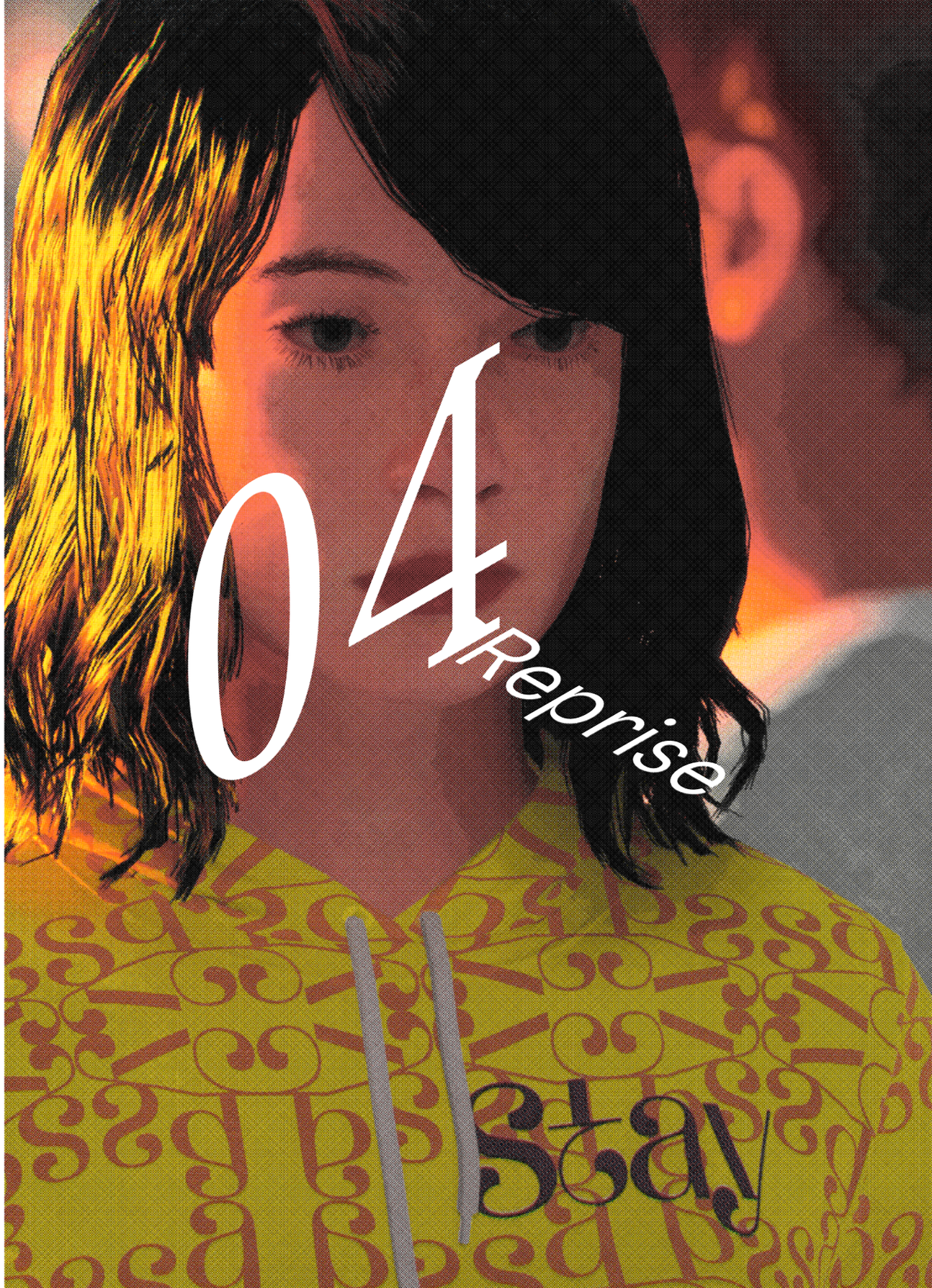
Art Direction

As the focus of this thesis was to experiment with typography I decided to use the "Quixel" library in conjunction with the "MetaHumans" creator tool. That way I didn't have to bother with creating 3D assets. The assets all have a realistic look to them and fit together in style and quality. I only had to make sure that I matched them together in a way that would be convincing in creating an environment, while avoiding to display anything that I had to create by myself———like houses, lanterns or park benches. The lack of lanterns was also the reason why I opted for sunset instead of night. It would simply be too dark. Also I felt it wouldn't make for a relaxed atmosphere for two people to have a conversation in a forest in the middle of night.

With photorealistic assets crossing into the uncanny valley is an immanent danger. If everything looks real but behaves differently then expected, it can break the immersion of the player pretty fast. With the Advanced Locomotion System v4 I had a pretty solid movement set, but there was no gesturing of the characters while talking. Also I wasn't sure in what quality I could export the game and still have it be playable, as I already dipped really low in fps while working in engine. I came up with the idea to connect the game to this documentation. With the recent addition of a "Riso" printer at KISD I wanted to make use of the unique color and look it brought to images. In order to print pictures they need to be rasterized. Also the colors——— despite being very

vibrant———have a nostalgic look to them. The more so due to its half-analog mode of operation. It works similar to silk screen printing in regards to working with a perforated foil that only allows color to be applied through the tiny holes. The rasterized optic amplifies the association with silkscreen printing and creates the out-of-date printing quality. For small publications Riso printing is the only way to integrate special colors outside of the CMYK-scale. Here is also where the fasciation for a lot of designers and artists lies. For me I wanted to incorporate the printing technique to ground my otherwise purely digital work with something tangible. In order to connect the printing of the documentation to the video game I tried to bring that look into the game. I decided on a selection of colors I wanted to use for printing. I used the program “Spectrolite” to prepare the images for printing with the Risograph. I then processed a color lookup table (LUT) through the program and split up the colors. The resulting LUT I then added in the post-processing volume of UE5 to change the color scheme of the game. I suppose it is possible to rasterize the textures in UE in a similar fashion as “Spectrolite” does, but I didn’t know how to do it. Instead I took a grey square image and also treated it to “Spectrolite.” Then I used the resulting image as the texture for the film grain in the post-processing. Though the result is not one hundred percent there, the effect evokes the association with a Riso-printed image close enough.

I also called back the small imperfections that appear when printing text with two colors with a Riso printer. To create that effect I simply used a drop shadow. I use this effect also to communicate what option is selected in the menu, by tilting the drop shadow ever so slightly. I also used this effect in the intro and outro.



When I was searching for a topic for my thesis I wanted to combine the various designs fields that I love. Making a short game allowed me to do exactly that. With the simple idea to decouple typography from the overlay-UI I added a form of interactivity to the narrative experience that I didn't foresee. When walking along Eline the player has to face her in order to read the text. Just like in a real conversation where you would also face the person you're talking to. Admittedly, a similar effect can be achieved with 3D audio and voice acting. When you turn your back to the speaker or walk away their voice becomes less audible. While the setting is similar I think that using typography instead of just audio in this "spacial" way adds a layer of interactivity. Very recently I played "Star Wars Jedi: Fallen Order." In the situations where you can talk to a character outside of a cutscene, I found myself just standing there, a bit bored while waiting for the dialog to play out. In "Stay" the waiting felt more involved. I had to pay attention where the text would appear and position my camera in a way that would frame Eline's conversation parts while still walking on and staying on the path. Since the text is only active for a set amount of time it also heightened my focus so I could read everything in time. All of that without the issue of having to jump between text and environment———or forgetting about the environment completely because I read the text (Thinking about games like "Disco Elysium" where during the reading parts one tends to forget about the game world to the left of the text box.).

In order to make this approach more useable for other games it would be necessary to add a lot more programming. For example if the player is not facing the character they are talking to, there should be a sort of notification to encourage the player to turn towards the conversation partner. This could go as far as projecting text that calls the player out for being rude. In role-play games this could then be integrated into relationship systems that determine the stance of other characters towards the player. Additionally a more elegant solution would be needed to display the text in an appropriate size to the players position than the sudden scale adjustment I used. Also in some cases the text would get blocked by the environment. This infringes the readability a lot, so a different solution would be necessary. Maybe the text could always be rendered on top of every other element in an x-ray-like fashion. Another issue is this method doesn't work well with close-ups at the moment. Placing the text directly in front of the speakers face, while their face covers 90 percent of the screen would certainly be to Neville Brody's liking, but it would also cover some of the facial expressions that contemporary games have gotten really good with. Overall this way to display dialog is more suited to slower paced conversation-focused games. In a more action-focused game it would create too much clutter. As I mentioned already in the typography chapter, accessibility options for this feature would be needed, as the tempo can become an issue with slower readers. For some of the more extreme font variations either a tamer version could be added or the intended inflection could be spelled out in brackets while falling back to the regular font style.

Regarding the font variations I think they work quite well. With more time it could have been interesting to work on more elaborate variations. Creating these variations would probably be a good opportunity to collaborate with other type designers. Because at some point I felt a bit drained and having difficulties to come up with new connections between emotion and typeface. I would also find it interesting to do show more inflection personality in the general typeface. For example if a speaker pronounces a certain letter in a distinct way, to make this letter stand out from the rest. Or simply using the IPA to highlight deviations from the expected pronunciation. There was also an MA thesis by Filip Despotovic that explored the capabilities of variable fonts in order to represent volume variations of speakers in subtitles by changing the font weight. [Despotovic, 2020] This could be an interesting addition for games as well. Unfortunately variable font support———to my knowledge———is non-existing in Unity and Unreal Engine. From a practical standpoint the volume changes would probably be derived with a wave-file as source otherwise every thing would need to be assigned by hand. But if one can provide a wave-file for every sentence that probably means there is voice acting for the whole game already———rendering the need for an expressive text representation less necessary.

The line breaks at the time of writing were generated automatically based on screen width. I wasn't able to get the programming to work that would enable me to set manual line breaks. This resulted in some cases where I had only one word in the last line, while the line before that was crammed. With a dedicated programmer this might have been possible, but I couldn't justify asking the programmer that had set up the text script for even more of his precious time.

In the prototype I experimented briefly with text that animated in the environment. I had plans to illustrate the inner thoughts of Amilia after the first fight with Eline. I thought about making words like "anger" or thoughts of doubt like "maybe too harsh" appear in the brief pause between the conversation as billboard like representations. But I think it would have been too overblown———the short moment of silence is well placed in my regard. As a concept I wouldn't throw it overboard, as I think it could be useful when a character stands at a crossroads and ponders which way to go.

With the sound design I was overall happy. It created a convincing atmosphere, but I would have liked to make the ambience layers a bit more complex and varied. There are a few other approaches by Dan Reynolds that I would have liked to check

out. Also building my own library of audio recordings could be a nice future project. I think the recordings I did for the prototype had a bit more personality to them or at least tied back to the actual flora and fauna I was referring to.

Regarding the text animations I think they could have been a bit more involved and playful. In the projects that I worked at up to that point the UI has always been a bit understated and reduced. So, I was lacking experience with designing complex UIs. For this setting the UI animations were fitting, but I'm sure it would have been possible to add a bit more of a wow-effect———without being obnoxious.

All in all I was glad how the game turned out and that I was able to explore this relatively underutilized method to display text-based conversations. As with so many things in the games industry the quality with typography varies a lot: From the expressive UI text in Persona 5 and the well crafted text logs in Control, to the often neglected typographic punctuation and non-existing letter kerning in many titles. I hope that with this thesis I was able to contribute a small part to future improvements and experimentations in the realm of video games.

Sources

Anne-Sophie Mongeau	[Audiokinetic] (2019, February 11). DESIGNING THE SOUND OF REALITY IN SHADOW OF THE TOMB RAIDER [Video]. Youtube. https://www.youtube.com/watch?v=Y0jr8uUUdNY
Dan Raynolds	(2022, 05. May). Soundscape Update Color Points and Conditional Spawning [Video]. Youtube. https://www.youtube.com/watch?v=ZHPZoNFNIY8&t=1536s [Reviewed 05. January 2023]
David Wiesner	(2022). A Voice through Type.
David Wiesner	(2022). Layers of Text.
Filip Despotovic	(2020). Subtype. ZHdK Diplom 2020. https://visualcommunication.zhdk.ch/diplom-2020/subtype/ [Reviewed on 07. January 2023]
Jeffrey Keedy	(2001). The Global Style: Modernist Typography after Post-Modernism. In: Total Armageddon. A Slanted Reader on Design. Edited by Ian Lynam. Slanted Publishers.
Lilo Schäfer	(2019). Das Trainieren eines künstlichen neuronalen Netzes zur Erkennung und Klassifizierung von Schriften.
Sho Iwamoto	(2017). Epic AND Interactive Music in 'Final Fantasy XV' [Video]. GDC Vault. https://www.gdcvault.com/play/1023971/Epic-AND-Interactive-Music-in [Reviewed at 05. January 2023]
Stephan_B	(2022, 9. March). *** Preview 3 Release of Version 2.2.0 and 3.2.0 Now Available! ***. [Forum Post]. Unity Forum. https://forum.unity.com/threads/preview-3-release-of-version-2-2-0-and-3-2-0-now-available.1250764/ [reviewed on 02. January 2023]

Links

Stay	david-wiesner.itch.io/stay
Going through a Phase	david-wiesner.itch.io/going-through-a-phase
Stay OST	davidwiesner.bandcamp.com/album/stay

Fonts in Use

David Wiesner	Amilia Sans Eline Serif Fritzi Sans Seeboh_v2

Thanks

Anne Wiesner	for being a though critic of my writing—it elevated the text a lot—and for being so encouraging, loving, and awesome all the way.
Benjamin Paska	for making the music so much better. It was a joy collaboration with you on this game.
Prof. Björn Bartholdy	for supervising me as a non-CGL student. I'm very grateful that you made things so easy for me with the interdisciplinary collaboration. I hope we will see more and better typography in games in the future ;)
Cory Schmitz	for taking time in your schedule, for being such an awesome person to talk to, and inspiring me to try out riso printing. I'm sorry the interview didn't make it into this thesis, but I haven't given up on it yet.
Florian Köhne	for taking me into the business lounge at GamesCom. It was such a pleasure talking to you and getting to know you.
Jana Lehning	for dishing out "soundsnap" codes whenever I needed one, and for the super detailed and insightful writing feedback.
Katinka Rosenfeld	for the perspective on my writing and for dying my hair—the color stayed vibrant for soooooooo long!
Kosan Aziz	for helping setup UE and for all the help with asset optimization
Lewin Hustedt	for taking so much time off your schedule to help guide me through setting up the dialog blueprints.
Prof. Michael Gais	for the positive feedback, and helping me to narrow down the scope of this thesis.
Melanie Eckmann	for being such a cool writing companion.
Moritz von Laufenberg	for setting up the riso printer for me and answering all my riso-related questions so patiently.