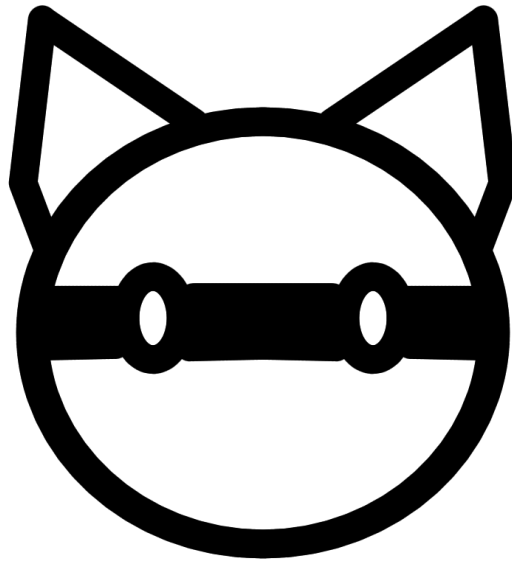# Final Report

DupliCat

DupliCat

08.02.2022

Brush n' Rush

Clemens Fromm

Georg Eickelpasch

Kim Simon

Klejdi Sinollari

Leonard Keil

Final Release & Project Recap

Final product
Over the course of the last semester Team Brush 'n' Rush invested a lot of time, sweat and effort into creating DupliCat - a VR drawing game in which you take on the role of DupliCat, a cat-criminal who breaks into meowseums, creates copies of paintings, stealing the original and leaving only the cheap counterfeit.

The gameplay loop plays out as follows: First, the player starts in a tutorial room, where they can already practise drawing, read the tutorial and later will also see their score and evaluation of their painting. The player can also change their preferred movement option, either a smooth translation or a teleport. Whenever they are ready, they can press the "Start Game" button. They will then have to enter the Miauseum through its roof. Once inside the painting begins. They have the option to swap around the painting that they wish to duplicate by pressing a button next to them. The player has access to a flashlight to light up the dark room, in which they will find the original drawing and an empty canvas. The task is to copy the original painting on the canvas, using four different coloured markers. During the process of copying the painting, the player has to look out for the guard. The guard patrols the Miauseum and periodically checks the player's room. If they see the player with their flashlight turned on, they will catch them and send them back to the tutorial room. Once the player is satisfied with the painting, they can end the round. A similarity algorithm will then evaluate the final painting. The player will get a score based on their similarity as well as a visualisation in form of a yellow overlay over their painting, showing the parts that are lacking.

Changes since Alpha
A couple of things were addressed between the alpha release and the final release now, especially given the feedback during the playtesting phase and beyond. These changes include a few gameplay features and mechanics as well as general improvements to the quality of life of the game itself.

We added several quality of life changes to make it easier for our players to play the game. The player now has a toggle in the form of a lever to decide what movement type they want to use. All the markers and the flashlight are now highlighted for better visibility and the markers themselves are now also colour-coded for more clarity. The player is now able to change pictures in the scene to increase replayability and adjust personal difficulty. The player can decide in the scene which painting to copy by pressing a button.

Some of the things we had planned to develop as part of our project were initially not done during the alpha and prototype phase but we managed to add them for the final release.

The guard now has basic animations and is equipped with a variety of different audio files to let the player know what is going on. These include footsteps, whistling, barking and puns. These are intended to give the player audio cues whenever the guard is nearby.

We also added the visual overlay to the evaluation based on our similarity detection algorithm to help our players understand what parts of the painting they could improve on. We kept developing our 3D sculpting process until it worked as desired. The sculpting level is similar to the main one, with the most notable difference being that there is a Reference and Duplicate sculpture, instead of painting. The player is expected to carve the Duplicate in a similar shape to the one we are stealing.

Another aspect that we further adjusted was the lighting in the scene. We received a lot of user-feedback to make the flashlight more crucial to use. The room itself is now a lot darker so that the player has to properly rely on the flashlight to see the painting.
We also added a small gameplay part to enter the miauseum as a more dynamic intro to the game to set the atmosphere and mood for the player.

Additionally, we further refined our game with feedback we got from more playtesting at Demo Day. And of course, we also fixed any bugs we encountered along the way :)

Experience during the class

Our team had a productive semester. Our initial design was changed a little bit to respond to new insights that we gained along the way. The biggest change would probably be that we decided not to add different settings as gamemodes, but instead included all of the different challenges at all times (darkness, guards). We followed our schedule pretty accurately until the alpha release after which the game should not be altered except for feedback implementing. However, everybody in our group fell in love with our game, so we still kept adding features and improvements in our free time. The different elements of the project structure all work hand in hand like a scheduler and a dispatcher. When doing a project without having to do all of these steps, we probably would not have done them, but after doing them we realised that we gained valuable insights. For example, we realised that we should mostly focus on the atmosphere of stress, anxiety and suspense for the player. The playtesting was a bit difficult and didn't add a lot of value compared to the effort but I think that is mostly due to the fact that we are using VR for our game which makes it a bit more difficult and playtesting is of course useful in general and especially if your game is not on the right track, which you would realise here. Luckily, our game was pretty good and fun to play! My (Georg)

impression of the course was pretty much as expected, since it is already my second time participating in it. This time around, I enjoyed it a lot more though, since our team was very productive and cooperative and our game actually turned out great. It's safe to say that everybody in our team is proud of the result and we definitely consider it a great success. Regarding the general time scope of the course we think that it was appropriate. Having a milestone presentation every two or three weeks never gives you the feeling that the next step is so far away that you can comfortably procrastinate. Therefore, everybody works on the project at least weekly, and our almost weekly meetings to assign tasks and update each other helped a lot with keeping everybody involved. The only milestone that we felt like was not planned carefully was the milestone over christmas. I think it was 3 weeks, but two of those weeks were official holidays, so you shouldn't expect us to spend our holiday working on this project. Because we didn't work on it for two weeks, we had to crunch the third week after the holidays. All in all, I think the time scope compared to the ECTS is fair. More time can always be used to build a better project or less time for a small project, but it should be reflected in the ECTS value.

The biggest technical difficulty was probably VR. The initial setup worked smoothly but every now and then it would stop working because of some setting or new hardware as it happened in one class presentation. The theme was a bit strict, since often the theme can also be used as a visual skin for any game which is not the case with duplication. (E.g. "fire" as a theme, you could make a game you want and just make the assets look fiery) Nevertheless, the theme still leaves a lot of possibilities as we can see from the variety of games created. I think the theme is good for people who don't have a team before the class and have no concrete plan of what they want to do. It gives a solid starting point for the brainstorming and conception phase. I would generally keep a theme for the class, but try to choose it as open as possible. What we would do differently in our next project is trying to practise our milestone presentations a bit in advance. Since we used hardware from different people merged together in the presentations, we faced some major problems there. This could have been prevented if we met up before the tech demo. The greatest success in our game is the integration of all the different parts. At first, we built the similarity detection, drawing and guard mechanics all separately and we didn't have a game loop. But after integrating all parts together, our initial concept was smoothly realised and the gameplay loop was fun. To answer all the questions I will mention once again, that our project definitely was a success and we have a fun game that we can be proud of. We mostly met our project plan and milestones, being open to change as you should be in state-of-the-art agile programming cycles. Finally, improvements for the course organisation is mostly that we should get one more week for the holiday milestone. Also, if possible getting feedback after each milestone could also help, especially with grading transparency.

I agree with the previous answers, especially with the holiday timing, we really could have used an extra week there. The biggest technical challenge for me(Klejdi) was the engine itself. To implement the similarity detection and sculpting mechanic, the code needed to be concurrent and parallel. Concurrent was not too hard to achieve using C# IEnumerable. However, parallelism was a complete nightmare, with non existing support from unity/c# for some parallel programming languages that can target the gpu, and very limited cpu parallelism. I think using a C++ based engine would have helped a lot in my tasks. I managed to reach my high goals with this project and learned a lot doing so, therefore I'm very happy and proud of the resulting game that we built as a team.

Most of the questions were already broadly answered when it comes to the general aspects of our game and the project itself but here are some additional answers to some of them. My (Clemens) biggest technical difficulty was not necessarily a specific feature or part of the game but the process of combining all the different parts into one cohesive unit and testing everything new or newly combined. We had originally used Unity's Resources and AssetDatabase to save and load things for several of our features. That caused several issues along the line as they are not necessarily intended to be used in a full build, outside of Unity's editor. Figuring these issues out, adjusting things along the way and making sure that everything works the way it is supposed to in a finished build was one of my biggest challenges. That and being one of the few people that had access to a VR device at all times as it sometimes was difficult for the team to test or troubleshoot features without being able to use VR. I would definitely make sure to have better ways to troubleshoot issues the next time I work on a project that heavily depends on a specific piece of hardware. I'm not sure if I could say that there was one great success on my end that stood out to me. There were just a lot of smaller parts that eventually came together.
I overall enjoyed the project, the theme and working with my team. It gave rise to a charming game that has the potential to be even better if we ever want to invest some more time in it. I think it was a success, without a doubt something to be proud of, and I learned quite a few things. We overall reached the goal of our game even if some parts had to be cut down or removed along the way like the scope of the story, the amount of game modifiers and the depth of the stealth mechanics.