VG Assignment – Personal Contribution for GP3

Stevan Likusic, FG23GP

Project Brief and Purpose – Blueprint samples

In this section I shall be presenting some of the more meaningful work I have done during the Game Project 3.

To begin with, I had a lot of tasks which included working closely with designers and helping them visualize the game in the first place, as well as its core mechanics. I would work on certain prototypes that would let them decide whether they wanted that feature to be a part of the game.

After that I started picking up certain, actual tasks, that would be some of those decided mechanics and features.

Before I begin talking about the bigger tasks I dealt with, some of the ones that are worth noting are: Fixing/Adjusting the flashlight trigger, adjusting the furniture dragging feature, adjusting the prompts system, adjusting the clipboard so it updates the tasks based on the progress the player makes, and finally putting the end game screen when player finishes the game.

With my “smaller” tasks out of the way I will begin by saying that the two of my biggest assignments during this GP were:

* Making **a proper movement system**
* As well as being **the sound programmer.**

I have encountered a lot of challenges with everything that I have been assigned to do, and I don’t think that any task that I had was particularly easy, but I’m glad it turned out that way because it let me learn a lot during those 7 weeks. So, without further ado, I will get into the most important parts of my programming journey.

One last thing to note is that I will explain everything that has been done regarding the code itself in this word file, however, for the sake of keeping this doc as readable as possible, I will restrain from posting any of the code snippets/screenshots here and will instead leave them in a separate folder so whoever is reading this can check that folder out as well.

Movement system

The movement system consisted of a couple of core mechanics such as walking, sprinting and crouching. Now I know that none of these sound like anything special, and they precisely aren’t, but back then this used to be my first task and I wasn’t nearly as experienced as I am now, so even that was quite challenging.

Before even beginning with it I needed to do proper research because there were a lot of ways that I could do it, but in the end, I simply decided to go with input actions and build up on what the engine had made in the first place (which was just some simple code to move around). It is worth noting that even though I had taken that part from the engine itself, I did need to adjust it properly later, so I understood everything that was going on with the code. The example of this was the fact that I made a separate part of the movement which was the backwards movement which was changing the speed of the player if he was going backwards.

Sprinting and crouching, on the other hand, were both fully developed by me, and they were somewhat similar (you will be able to see it in the pictures). Their conditions, as well as the executions, were quite similar, and the thought process behind making both of those wasn’t all that different.

Sound Implementation

Dealing with sound was the bigger part of my project contribution. As I mentioned earlier, I had a lot of different asks since I was working closely with designers to make certain prototypes and changes that would, simply put, suit them or the game itself better, but sound implementation was by far the task I put most of my time and effort into.

Going back to it and thinking about it, there were so many different sounds that I had to implement that I doubt I can even mention all of them, so I will simply categorize them based on their executions or action that was turning them on.

First things first I had to implement were some “background sounds” that would play constantly during certain periods of the game, and the background sounds would change as the player progressed further (stepped into another room, started getting chased by a monster, etc.).

Next were the sounds that would sort of track the players position or state and play based on that, in this case I will refer to them as “conditional sounds” and the best examples of those are the sound of player walking/running around the game, or the player getting close enough to the exit/entrance of a room to hear a sound that would produce a jump scare.

And lastly something that I call “action sounds” which were a response to player interacting with the world around him, things such as turning the flashlight/UV light on and off, trying to open a locked door, turning off some switches, dragging/moving the furniture around the game, etc.

All of these were extremely important and were key pieces to making a proper horror game that would let you get immersed in it while playing.

Since I have already talked about the most important part on my side, I will keep the upcoming things quite brief.

Workflow – Planning and development process

When it comes to planning and development, I do not have a lot to talk about overall. Most of the project planning was not done by me, I was someone who wanted to have the role of dealing with tasks that were necessary and trying to help my team out in any way possible. My own planning and development went well since I could take my time to learn new things, try them out, see how they work, often even fail, but lastly overcome the problem and make a cool feature. All our work was being tracked in our Jira board and all the ideas that we had were coming from Miro board. They weren’t set up by me, but I did end up using them quite a lot since they were a great tool to help me navigate around the tasks and the project itself. They were keeping everything in check and plenty clean, I got quite used to using them and I was satisfied with the results.

Workflow – Technical Challenges

Now lastly when it comes to technical challenges I wouldn’t even know where to begin with. We all had them, of course, I’m no exception. They are quite a normal thing to stumble upon, but during this project there were a lot of them at the start due to how inexperienced I was with the engine and the coding language itself.

I didn’t use anything that helped me in developing, outside of YouTube and certain forums/websites online. I tried talking with people who were more experienced and relied mostly on myself in the first place. It was hard to start since I wanted to do everything in blueprints rather than code, there wasn’t really a step-by-step guide on how to deal with blueprints or how to properly use them. My road was a long and tough one, but I’m glad that my team and I managed to get through it.

As for some details that I could get into, I can only talk about my own experiences and examples, perhaps those would let whoever is reading this get a clearer picture of my journey.

When I started developing, I started with the movement system. At that starting point I didn’t know how to do a lot, and so I had to begin by researching the things that I directly needed to do and work with. One of the first issues was getting to understand all the terminology and trying to keep the code as optimized and as clean as possible.

In my example: don’t have your blueprint nodes connected everywhere around without knowing where is the starting or the ending point, and don’t make all actions start from “Event Tick” because that would shut down the program.

One of the bigger issues that I had to deal with was having to differentiate walking from crouching from sprinting so that they would all happen separately. This was quite challenging because all of them had their own speed, sound, and the sort of different camera effect that was supposed to put them apart from one another. In the end, of course, I overcame the issue and fixed the code.

When the time came to pickup the sound implementation task, I was somewhat aware of how to deal with nodes and the engine, so on that side it was easier. However, that was the time when I needed to modify A LOT of other codes that I didn’t do since they needed to, for example, produce a sound after an action would happen.

That part of the journey really let me understand the project to its core and see all the different ways of people dealing with their tasks and how they handle their code. It wasn’t easy at all since I would spend hours just trying to understand how someone else’s code works before I could even do my own, but I the end I pulled through!

Conclusion

In summary, this project has been a valuable learning opportunity, offering insights into the dynamics of teamwork, effective communication, and personal contribution. I have gained a better understanding of how to maintain a constructive attitude and the importance of striving for continuous improvement. While I am satisfied with my contributions to the project, I am also aware of areas where further growth is possible. Moving forward, I am dedicated to applying these lessons to future projects, aiming to enhance my effectiveness as a team member and improve my collaborative skills. This experience has laid a strong foundation for my ongoing personal and professional development, and I look forward to applying these insights in future endeavors.