

Summary

This project was developed with Unreal Engine 5.5.4 using only C++ (No blueprints code involved) as a challenge within 48 hours. It simulates a Skateboarding game leveraging a prototype that contains the following features:

- Skate Movement
- Jumping
- Pushing
- Slow Down
- Points System
- Playtest Level & UI

Development

Prototypes that need to be developed in a short period of time are not different from GameJams games. So I first started planning what I should do before actually starting to code. I immediately discarded ideas that I thought it would be cool to include in the project and started focusing on the gameplay. The plan was to develop the initial gameplay as quickly as possible, and then when I have something working I would polish the gameplay and refactor the code. This ended up being a double-edged sword.

The good thing was that I noticed that the gameplay wasn't really feeling like a Skateboarding game. The skate was basically glued to the ground, I was trying to jump off ramps and it was very difficult to get off the ground surface.

The bad thing was that since I already had developed the system, whatever I would do, I would need to adapt the code (which wasn't really big at the time), but still, I would lose some precious time. So I was very reluctant in the beginning because I knew I would need to expand the CharacterMovementComponent, but I wasn't really confident that it would improve the gameplay, and on top of that it would be very complicated to come up with something from scratch like that, but it didn't mean that the idea was totally discarded.

So after creating the initial gameplay I decided to move to the Points System. The points system was not complicated, actually, I think it was the easiest thing that I made in this project. And it was very good, because I had more free time to focus on what's important: The Gameplay.

So after finishing this system, I finally found courage and started extending the CharacterMovementComponent, and as I thought, it was VERY difficult. The good thing was that in a short period of time, I finally started to see the results.

I was able to get off the ground, YAY.

Since we have access to the CharacterMovementComponent.h and CharacterMovementComponent.cpp, I had a good reference to understand how to implement some things. And from there I continued to migrate the features that I developed on the character to my custom movement component (Slowdown and Jump).

After that, I finally had something that I was happy about, and I could say that it simulates a Skateboarding game.

Performance

Overall, I'm surprised with what I came up with. I've never extended the MovementComponent before, because all the games that I've worked so far relied on the default CharacterMovementComponent. So extending and polishing a custom one was really surprising and satisfying.

But there are some drawbacks that happened as well.

UI:

Even though it is not the main focus of the project, it ended up being SUPER simple and the code behind isn't really the ideal, but it works.

Jump:

This is something that I had to accept, otherwise I would not be able to deliver the project because I did not have enough time. The jump isn't polished, it feels weird sometimes, this is definitely something that I would focus more on if I had more time.

Camera:

The camera is also very basic, it just follows the controller yaw rotation and that's it. This sometimes feels weird when rotating the character to match the surface. But overall it's ok for the prototype.

Time

These are the approximated values that I spent on each task. This also considers the time that I spent adapting the code to the custom movement component

- Movement Functionality: 11h
- Level: 1h
- Jumping: 3h
- Pushing Functionality: 3h
- Slow Down Functionality: 2h
- Points system: 2h
- UI: 1h
- Animation: 1h
- Total: 24h