This is composed of three elements, a movement system, a score collection system and a widget display. All logic is done on C++ layer with the exception of the creation of the widget, which is a child of our C++ widget but since it has more elements for easy access to design, it needs to be instantiated on the Blueprint layer.

I always find it easier to start with the end class so I can tinker on how it should be implemented and then build the logic from there. In this case it was the SkaterMovementComponent, and this technique proved fruitful because I decided to do an architectural change to my initial planned approach.

When building SkaterMovementComponent I started trying to make a physics movement, but it kept feeling weird, after much tinkering I decided to just use the regular approach of velocity, after that the programming was fluid, this starting point was the only "stumper" I had. Another issue I had was with the size of the recommended packages, it made the game too big with things that weren't going to be used, so I took some time to clean it up as well.

Planning: 1 hour.

SkaterMovementComponent: 6 hours.

Other Classes: 4 hours.

Cleaning up overweight and fixing LFS: 2 hours.

Level Design: 1 hour.