2020182031 이서연 졸업작품 14주차 진행

1. Phong Lighting 제작

사소한 오류로 Lighting처리가 안되고 있다는 것을 깨 달았다.

두가지의 문제점이 있었는데 첫번째는 MAX\_LIGHT가 두 곳에 ( “stdafx.h”, “Light.hlsl” )저장 되어있는데 둘이 값이 달라서 서로 다른 크기의 Light를 만들고 있었던 것이다.

그래서 일단 캐릭터가 바라보는 방향으로 나가는 Spot Light와 Directional Light 이렇게 두개를 만들어 주기 위하여 2로 설정하였다.

또한 Game Scene에서 Light를 만들어 줄 때 Enable값을 False로 넣어주고 있었기 때문에 Light계산에서 제외되고 있었다.

이 문제들을 해결한 후 Lighting 처리를 MRT 이후로 넘겨주었다.

즉, 2-Pass로 넘겨주었는데 그 함수를 Multiple Render Target 내부에서 하도록 해주었다.

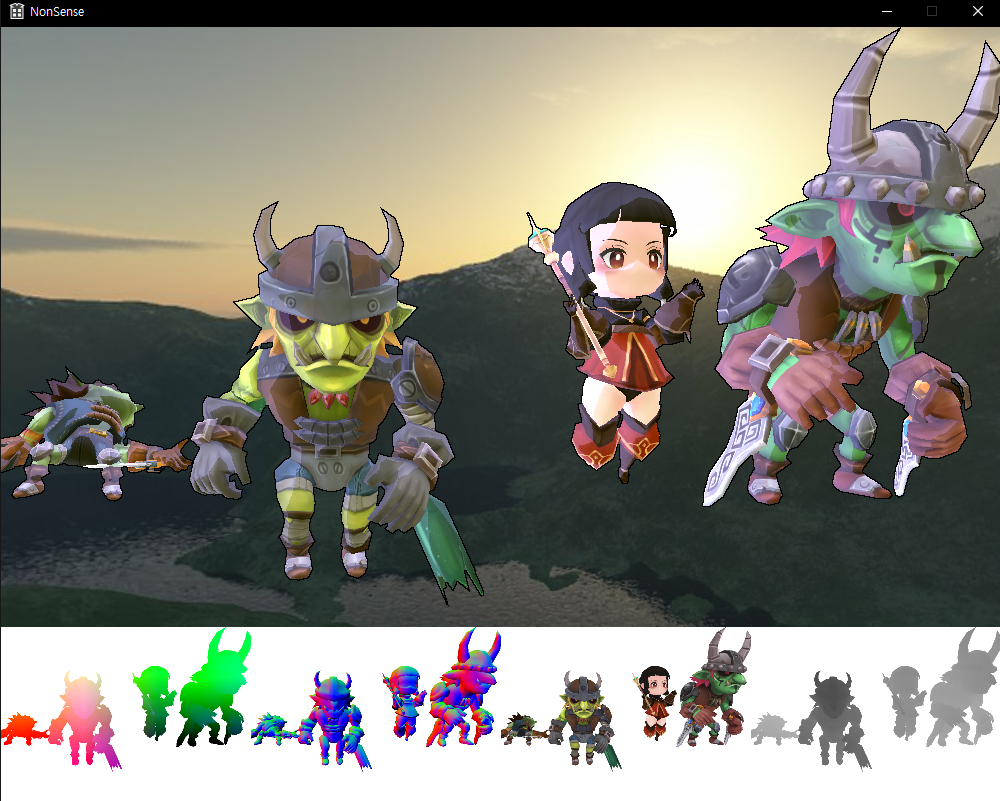
텍스트이(가) 표시된 사진

자동 생성된 설명

바로 이 부분인데 Texture부분에 Lighting처리를 MRT의 정보를 활용하여 더해준다.

이렇게 하면 모든 Object의 Pixel수 \* Light의 수에서 Screen Pixel수 \* Light의 수로 줄어들게 된다.

텍스트, 절지 동물, 게, 랍스터이(가) 표시된 사진

자동 생성된 설명

다음의 두 화면은 각각 Sky Box와 빛을 조절해서 만들어본 밤과 낮의 모습이다.

이렇게 내가 만들려던 Lighting Volume을 이용한 빛은 아니지만 임시적으로 Phong Lighting은 완성이 되었다.

1. In Game UI 제작하기

UI를 만드는 것은 너무나 쉬운데 그냥 Texture를 원하는 곳에 띄워 주기만 하면 된다.

UI는 모든 화면의 Render가 끝나고 그 위에 그려줘야 하기 때문에 한 번에 그려주기 위하여 UI Object를 만들어주어 상속하는 방식으로 다양한 UI들을 제작해 줄 생각이다.

웹사이트이(가) 표시된 사진

자동 생성된 설명

이것은 기획 발표 때 준비한 UI구상도이다.

인 게임 UI부터 만들어 볼 생각이고, 미니 맵은 뒤로 미루고 플레이어 상태 바, 스킬 창만 만들어 볼 생각이다.

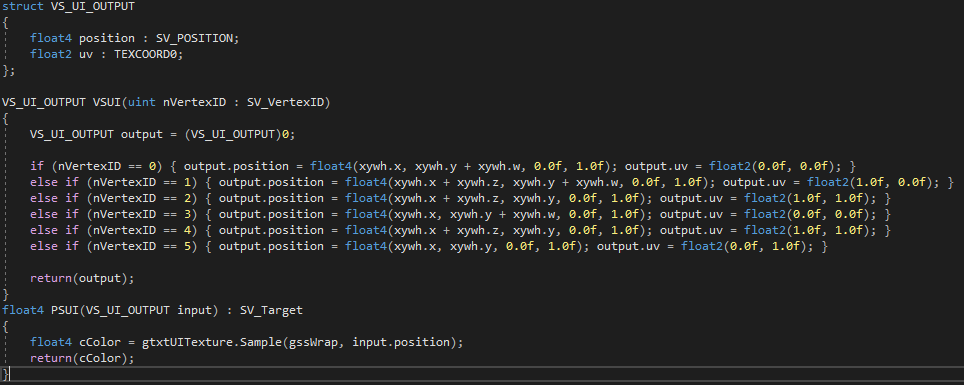
UI를 여러 Scene에서 자주 사용할 것 같기 때문에 Constant Buffer를 사용하여 시작 x, y 지점과 h, w까지 받아 그려주는 Shader를 만들 생각이다.

먼저 모든 UI의 베이스가 될 UI Object를 만들어줄 생각이다.

UI Object를 만들 때 XYWH라는변수를 만들어서 UI Constant Buffer에 넣어주어 Vertex Shader에서 사용해, 원하는 위치를 지정하고 Texture를 UI Object생성 시 지정하여 적용하고 그려주는 방법을 생각 중이다.

일단 Player State UI Texture가 아직 없기 때문에 Mage Player의 Skin DDS파일을 불러와서 확인해 주기로 하였다.

UI Shader코드는 간단하게 Vertex Shader, Pixel Shader를 설정해주고 아직 적용은 하지 않았지만 Blend Mode를 TRUE로 바꿔주어야 한다.



Vertex Shader와 Pixel Shader인데 UI\_CV에서 입력해준 XYWH를 이용하여 X,Y 부터 W,H의 크기만큼 늘려주는 것이다 ( -1.0 ~ 1.0 )