|  |
| --- |
| **[ Catch Bear (캐치 베어) ]** |
| [ 3주차 ] 2022. 1. 9 ~ 2022. 1. 15 |
| 1/9 일) 파일 입출력: 텍스트 파일 -> 바이너리 파일 변경  1/10 월) FBX Exporter에서 발생한 msvcrtd.lib 링커 오류를 해결하는데 생각보다 많은 시간이 걸림.(링커->추가 라이브러리 디렉터리: VS\lib\onecore\x64) 성공적으로 바이너리 파일을 만든 후, 바이너리 파일을 불러오는 부분을 다른 프로젝트에 구현함(이 프로젝트는 아직 깃에 추가하지 않음.) 정적 메쉬를 렌더링하는 쉐이더 구현중이고, 내일 이어서 할 예정.  1/11 화) 정적 메쉬를 렌더링하는 쉐이더 구현을 하였다. 일단 FBX Exporter에서 구현을 완료한 메쉬의 정점 정점 정보들을 가지고 와이어 프레임으로 렌더링하는 것을 성공하였다. FBX Exporter에서 메쉬의 재질/텍스처 정보를 로드하는 부분을 구현하기 시작하였다.  1/12 수) FBX Exporter에 메쉬의 재질과 텍스처 정보를 로드하는 것을 추가로 구현하였다.  1/13 목) FBX Exporter에 메쉬의 애니메이션 정보를 로드하는 것을 추가로 구현하였고, 플레이어의 애니메이션 정보들이 잘 로드되는지 확인하였다. 텍스처와 재질 관련 정보들을 어떻게 추출해서 클라이언트에 어떻게 적용할지 내일 회의때 같이 의논해봐야 할 것 같다.(구현할 때 필요한 정보는 무엇인지)  1/14 금) 회의를 한 후, 메쉬에 텍스처를 입히는 것을 구현하기 시작하였다. 일단은 텍스처 좌표들의 정보를 읽어 들여서 버퍼 리소스로 만드는 것 까지 구현하였다.  1/15 토) 클라이언트 프로그램에서 바이너리 파일로부터 텍스처 매핑에 필요한 정보들을 모두  읽어 들여서 변수에 저장하였다. (텍스처, Ambient, Albedo, Emissive, Specular 등등) 내일 텍스처  와 재질, 조명을 적용하는 것을 시작해야겠다.(레지스터 번호 다시 확인해보기!) |
| [ 4주차 ] 2021. 1. 16 ~ 2021. 1. 22 |
| 1/16 일) 전날 텍스처링에 필요한 정보들을 모두 읽어온 것을 바탕으로 환경 오브젝트 메쉬 텍스처링을 구현하였고, 여기에 조명과 재질도 추가해서 구현하였다.(플레이어 메쉬는 아직 렌더링X) 근데 환경 오브젝트의 메쉬가 생각보다 너무 커서 일부러 scale값을 줄여서 렌더링했는데 로우 폴리곤 느낌이 많이 나지 않았다. 플레이어 메쉬까지 같이 렌더링해 본 다음, 환경 오브젝트 메쉬들이 많이 크고, 로우 폴리곤 느낌이 많이 나지 않을 경우, 환경 오브젝트 메쉬들을 다시 찾아봐야할 것 같다.(회의때 정하기로 함!!) 그리고 플레이어 애니메이션 정보들을 FBX Exporter에서 어떤 순서로 적는지 노트에 정리하였다.  1/17 월) 주말부터 목이 건조하더니 감기에 걸린 것 같아서 오후에 코로나 검사를 받고, 병원에 다녀와서 밥과 약을 먹고 쉬느라 작업을 거의 하지 못했다.  1/18 화) 어제에 비해 증상이 나아지기는 했지만 여전히 목이 아프고, 기침도 많이 나와서 작업을 많이 하지는 못했다. 플레이어 메쉬의 fbx파일을 추출해서 그것을 바탕으로 바이너리 파일을 만들어서 텍스처, 재질, 조명을 적용해서 렌더링을 해보았다. 다음으로 스킨 디포메이션 정보를 읽어들이는 부분을 디버깅으로 확인해보면서 구현해야겠다.  1/19 수) 기존 프로젝트의 플레이어에서 스키닝 정보(스킨 디포메이션)도 추가로 읽어들여서 렌더링하는 것을 구현하였다. 텍스처링을 위한 메쉬와 쉐이더(TexturedMesh, TexturedShader)와 스키닝을 위한 메쉬와 쉐이더(SkinnedMesh, SkinnedShader)를 합친 형태인  TexturingSkinnedMesh, TexturingSkinnedShader를 새로 만들어서 텍스처링을 하면서 스키닝 애니메이션을 할 수 있는 형태로 만들었다. (-> 이 3종류의 쉐이더는 나중에 상속관계로 더 간단하게 표현할 수 있을 것 같지만 지금은 공부를 위해 각각 따로 구현하였다.) |