|  |  |  |
| --- | --- | --- |
| EmberFall 개발 일지 | | |
| **10주차** | **2025.03.02 ~ 2025.03.08** | 회의 내용 |
| 방학 동안 작업한 결과를 Develop Branch 에 합치는 시간을 가졌다.  이후 각 팀원의 환경에서 실행하여, 서버, 클라이언트 모두 정상 작동하는 것을 확인하였다.  이후 함께 작업하다가 헤어졌다. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| EmberFall 개발 일지 | | 작성자 : 김승범 | |
| **10주차** | **2025.03.02 ~ 2025.03.08** | | 이번 주 진행 사항 |
| **03-02 ~ 03.05**  본격적인 애니메이션 렌더링을 시작하였다.  기본적인 스키닝 애니메이션 셰이더를 작성하고,  MeshRenderManager 에 BonedMesh 를 위한 과정을 추가하였다.  Animator 를 추가하여, AnimationClip 을 상수 참조하고, 이를 토대로 매 프레임마다 BoneTransform 을 계산하도록 만들었다.  Assimp 라이브러리에서는 행렬을 column-major 로 관리하여, 이를 Transpose 하여 연산을 진행 할 때에는  Row-major 로 계산하도록 하였다.  Assimp 라이브러리에서는 기본적으로 오른손 좌표계를 사용하여 모델을 파싱하므로, 이를 다시 왼손 좌표계로  바꾸는 과정을 추가하였다.  애니메이션을 인스턴싱으로 렌더링 하기 위해, AnimatedModelContext 의 BoneTransform 배열 시작 인덱스를  추가하여, 애니메이션 모델을 인스턴싱으로 그릴 수 있도록 만들었다.  좌표계 문제에서 헤메어 많은 시간을 낭비했던 것 같다.  **03-06**  전날 애니메이션 렌더링을 완료하고 문제를 수정하는데 있어 값을 확인하는데 사용된 코드를 정리하였다.  SkyBox 를 만들고 렌더링하였다. | | | |
| EmberFall 개발 일지 | | 작성자 : 김승범 | |
| **10주차** | **2025.03.02 ~ 2025.03.08** | | 이번 주 진행 사항 |
| **03-07**  Renderer 가 Deffered Rendering 을 하도록 수정하였다.  이를 위해 DefferedRenderer 를 만들었으며, 첫 번째 패스와 두 번째 패스의 렌더 타겟을 Texture 로 관리하여,  패스 사이의 전환이 쉽도록 만들었다.  기존 Texture 에 Clear Value 를 설정하는 옵션이 없어 이를 추가하였다.  **03-08**  기존 Animator 가 비효율적으로 계산하는 부분들이 있어 이를 수정하였다.  1. 애니메이션을 로딩 할 때, 이미 로딩되어 있는 importer 를 재활용하지 않고, 계속해서 새로운 importer 를  만드는 문제가 있어 이를 재활용하도록 수정하였다.  2. BoneTransform 을 계산 할 때, 한 번의 렌더링마다 vector 를 재할당하는 문제가 있었다. 이를 위해 최대  Bone 의 개수를 150 개로 제한하고, std::array 를 사용하여 계산하도록 변경하였다.  다른 팀원들의 환경에서 프로그램이 cpu 내장 어댑터를 사용하는 문제가 발생되어, 프로그램 초기화 시에  어댑터를 나열하고, 그 중 전용 메모리가 가장 높은 어댑터를 사용하도록 하는 기능을 만들었다.  애니메이션 로딩 시간을 개선하기 위해, 로딩한 애니메이션 정보를 직렬화/역직렬화 하는 모듈을 만들었다. | | | |
| EmberFall 개발 일지 | | 작성자 : 김성준 | |
| **10주차** | **2025.03.02 ~ 2025.03.08** | | 이번 주 진행 사항 |
| 03-02 게임 이벤트 처리 방식을 변경했다. 기존 구조에서는 이벤트 마다 모든 오브젝트를 돌면서 해당하는 오브젝트를 찾아 전달하는 방식이었으나, 이방법은 너무 비효율적으로 여겨져 이벤트 구조체에 ID를 하나 더 저장해서 송신/수신자를 기록하고 게임 씬에서 바로바로 수신자를 찾아 넘겨주도록 변경했다.  03-03 충돌에 대한 처리를 각 오브젝트가 담당할 수 있도록 구조를 변경하고,  전역에서 호출할 수 있는 타이머를 하나 더 만들었다. Update 함수에서만 시간을 구할 수 있다는 제약이 너무 커서 만들었다. 저번에 만든 GameUnit 에서 단위간 곱셈과 나눗셈에 오류가 있어 이를 수정하고 이에 맞춰서 단위를 사용하는 모든 클래스에 단위를 적용하도록 변경했다. (게임 맵, 오브젝트 크기 (미터단위), 속도, 등등)  03-04~03-08 몬스터가 특정 위치를 찾아 갈 수 있도록 길찾기 알고리즘을 작성했다. 길찾기 알고리즘은 인공지능 시간에 배운 A\* 알고리즘을 사용했으며, 이를 구현하기 위해 그래프 구조와 실외지형 맵에 대한 그래프를 만드는 코드를 작성했다.  이 작업에 생각보다 시간이 오래걸려 AI에 적용은 다음주부터 하기로 하였다. 중간에 테스트를 해보니 실외지형에 대해 그래프를 만드는 작업이 너무 오래걸리는 것을 확인했다. 실외지형 맵은 자주 바뀌지 않고 게임이 시작하면 바뀔일은 없으니, 일단 미리 만들어 두고, 파일에 저장한 후 시작시에 파일에 있는 데이터를 불러오는 방식으로 구현하기로 하였다.  03-08 토요일 – 추가로 이제 클라이언트 부분에서 진전이 생겨 프로젝트를 통합하였다. | | | |
| EmberFall 개발 일지 | | | 작성자 : 정영기 | |
| **10주차** | | **2025.03.02 ~ 2025.03.08** | | 이번 주 진행 사항 |
| 03-02  큰 실수가 있었다. 기존에 G버퍼에 기록이 잘 안되었던 것으로 알고 있어서, 셰이더 코드와 렌더링 코드에 문제점이 있을것이라 예상했지만, 월드변환 행렬에서 문제점을 발견하였고, 수정을 하였다. 잘 보인다.  03-03  기존에 스페큘러값도 G버퍼에 저장하였지만, 스페큘러값을 비롯한 조명 정보는 두번째 패스에서 계산하면 되므로, 굳이 저장 할 필요가 없기 때문에 삭제하였다.  그리고 깊이값을 저장하는 셰이더 코드가 잘못되어 수정하였다. 원근투영 나누기를 하지않아 깊이값이 아닌 실제z좌표가 저장되어 있었다.  03-04  G버퍼 저장도 해결했기 때문에 바로 두번째 패스를 렌더링을 시도했더니 처음으로 결과물이 나왔다.  하지만 조명의 이동에 따라 오브젝트가 없는 곳이 조명의 효과를 받기 시작했다.  그 이유를 생각해보니 G버퍼는 2D텍스쳐에 색상의 형태로 각종 정보를 저장하는 버퍼이기 때문에, 실제 오브젝트가 존재하지 않는 곳은 오브젝트의 정보가 아닌 G버퍼를 초기화 한 값이 텍스쳐에 남아있게 되었기 때문에 오브젝트가 없는 부분도 조명이 적용되는 것 이었다. 해결법은 간단하다. G버퍼에 저장한 깊이값으로 오브젝트의 여부를 판단하는 것 이다. 깊이값이 1.0인 상태는 초기화 된 상태에서 렌더링이 되지 않은 픽셀이기 때문이다. 각 픽셀의 깊이값이 1.0이면 픽셀 셰이더에서의 연산을 제외시키는 방식으로 해결하였다.  03-05  조명의 회전에서 문제가 발생하였지만, 해결하지 못했다. 당장은 해결 할 필요가 없다는 피드백 덕분에 디퍼드 렌더링을 마무리하였다.  03-06  두통과 허리통증으로 인한 해당없음  03-07  그림자 모듈화를 진행중이며, 김승범 팀원의 기존 프레임워크에 맞게 그림자기능을 모듈화 하기위해 위한 기존 코드 검토를 하였다. 김승범 팀원의 코드에는 아직 본인이 숙달되지 못하거나, 알지 못했던 STL이 상당수 쓰였기 때문에 시간이 더 걸릴 것 같다.  03-08  코드 검토를 진행중이며, 실제 프로젝트에 쓰일 그림자 클래스를 구상하는 과정이 있었다. | | | | |

|  |  |  |
| --- | --- | --- |
| EmberFall 개발 일지 | | |
| **10주차** | **2025.03.02 ~ 2025.03.08** | 다음 주 목표 |
| 1. 김승범 ( 클라이언트 )   지형/모델 Collider 구현 및, Bounding Box Rendering  DirectWrite 를 통해 글자 그리기 및 UI 추가   1. 김성준 ( 서버 )   1. 길찾기 이어서 구현, 2. 클라이언트에 네트워크 코드 삽입 3. 아이템, 공격 시스템 구현   1. 정영기 ( 클라이언트 )   그림자 모듈화 마무리, 계산셰이더를 이용한 블러링 작업 | | |
| **특이사항** | | |
|  | | |