|  |  |  |  |
| --- | --- | --- | --- |
| [Cranding(크랜딩)] | | | |
| **20주차** | **2024.5.5 ~ 2024.5.11** | **작성자** | **정경서** |
| **주간 회의** | | | |
| **회의 날짜** | | 5/6 | |
| 1. 중간발표 ppt 최종 점검 및 수정   개발 내용  클라이언트(모델/맵 로드, animation, 디퍼드렌더링, 애니메이션 블렌딩, 상하체분리, UI)  서버(IOCP 기반 서버 구축)  문제점 수정하기 -> 월요일까지 개발하고 수정본 들고가기   1. 시간 관련 로직 및 기획 수정     서버 연동을 위해 낮->밤/밤->낮 인 경우 따로 패킷 보내기 | | | |
| **이번주 한 일** | | | |
| 1. 신동엽(서버)   계속해서 씬 전환 할 때, 게임에 접속할 때 빈번하게 프로그램이 죽는 상황이 발생하였다. 클라이언트 측 문제일 수도 있겠지만 클라이언트에서 서버에 로그인 하는 순간에 문제도 포함되어 있다고 생각해서 천천히 디버깅해보았다. 천천히 연결되는 과정을 따라가 보았다. 텍스트, 스크린샷, 소프트웨어이(가) 표시된 사진  자동 생성된 설명  이처럼 처음에 프로그램이 시작될 때 서버와 커넥트 준비를하고 커넥트가 이루어지고 그제서야 서버 쓰레드가 켜지게된다. 여기서는 문제 X  서버 스레드가 켜지면 send와 recv스레드가 실행된다.  이후로 내려가보면 처음에 로그인 씬, 으로 불리는 부분에서 게임시작 버튼을 누르게 되면 서버에게 클라이언트 로그인 정보를 보내주고 여기서 리시브 스레드가 정보를 받을 떄까지 WaitForSingleObject로 막아주어 순서를 보장해주었다.  텍스트, 스크린샷, 소프트웨어, 멀티미디어 소프트웨어이(가) 표시된 사진  자동 생성된 설명  기존에 ReleaseObject를 하고 send하는 부분이 엉켜있엇는데 이 때문에 오류가 발생하여 순서를 변경해주었다.  아이디 정보와 씬넘버를 넘겨받은 이후에 클라에서는 새로운 씬을 그리기 시작한다.  텍스트, 소프트웨어, 멀티미디어 소프트웨어, 그래픽 소프트웨어이(가) 표시된 사진  자동 생성된 설명  게임씬으로 넘어가게 되면서 마지막으로 패킷을 받은 클라이언트는 인원수를 카운트하여 게임 시작 을 알린다.  게임이 시작되면 서버측에서는 타이머 패킷을 보내주며 게임 시간을 체크한다.  텍스트, 스크린샷, 폰트이(가) 표시된 사진  자동 생성된 설명  타이머 패킷을 받으면 클라에서 시간을 체크하는 타이머 스레드가 켜지고  텍스트, 스크린샷, 소프트웨어, 멀티미디어 소프트웨어이(가) 표시된 사진  자동 생성된 설명  타이머 스레드 안에선 일정 시간이 지날 때 낮인지 밤인지를 서버에게 전달하여 시간을 맞춘다.  텍스트, 스크린샷, 소프트웨어, 멀티미디어 소프트웨어이(가) 표시된 사진  자동 생성된 설명  이때 시간을 재서 시간이 밤시간일 경우에 서버에 타이머스레드에게 NPC의 정보들을 넘겨주어 NPC AI를 실행시킨다.  ( 여기서 오류가 발생 제대로 타이머가 작동하지 않음 )  타이머 스레드를 살펴보면 텍스트, 스크린샷, 소프트웨어, 멀티미디어 소프트웨어이(가) 표시된 사진  자동 생성된 설명  이처럼 이벤트 타입과 이벤트 구조체를 만들어 두고 타이머 클래스에서  텍스트, 멀티미디어 소프트웨어, 폰트, 스크린샷이(가) 표시된 사진  자동 생성된 설명  텍스트, 스크린샷, 소프트웨어이(가) 표시된 사진  자동 생성된 설명  이를 전달받아 overlapped 구조체에 해당하는 룸 아이디를 넘긴 다음 workerthread에 넘겨주어 NPC를 처리한다.  텍스트, 멀티미디어 소프트웨어, 소프트웨어, 그래픽 소프트웨어이(가) 표시된 사진  자동 생성된 설명  넘어온 이벤트 들은 workerthread에서 처리되는데 여기서 문제가 발생한다. 제대로 시간마다 불려오지 않아 한번 이동하고 다시 이동하지 않는다.    릴리즈 실행 문제  디버그 모드에서 릴리즈모드로 실행한 결과 비쥬얼스튜디오가 알아서 대기하는 루프 코드를 스킵하여 로그인정보를 받기도전에 실행하여 동기화 문제가 발생하였는데 while문 안에 cout을 넣어 삭제를 방지했다. Volatile를 써서 해결하기도 했다.  뷰리스트 데드락 발생  텍스트, 스크린샷, 소프트웨어, 멀티미디어 소프트웨어이(가) 표시된 사진  자동 생성된 설명  기존에는 clients[pl]이 아닌 clients[id].unlock를 해줘 서로 다른 락에 접근하여 데드락 문제가 발생하였었다. 디버그할때 왜 통과되었는지 의문   1. 정경서(클라이언트) 2. 맵 및 모델 추출       맵과 모델의 바운딩 박스를 수정하여 다시 추출했다.  충돌체크 부분은 예나언니가 맡아서 하기로 했으니 언니와 소통을 통해 바운딩 박스를 가진 뼈나 객체의 이름 앞부분에는 bb를 붙여서 이후 처리를 편하게 할 수 있도록 해 두었다.  손목이 아파서 너무 힘든 작업이었다..   1. 카메라 수정   지금 카메라에 여러 문제가 있는데   1. 카메라 위치가 player의 position을 보고 있는 건지 우리가 원하는 화면처럼 나오지 않고 있다. 2. 우주선 맵에서 여러 환경이 잘 보이도록 카메라를 세팅해주고 이동시키고 있는데 이게 모든 맵에서 적용이 되고 있다.   먼저 이 두가지를 처리하기로 했다.  1번 문제 해결을 위해서 플레이어의 카메라 함수를 바꿔주었다.  카메라의 setlookat, update 함수의 인자로 들어가는 m\_xmf3Position 값 보다 조금 더 앞부분의 position을 넘겨주면 될 것 같았다.  룩벡터를 받아와서 ScalarProduct 함수를 사용하여 카메라가 플레이어보다 조금 더 앞부분을 바라보도록 업데이트와 셋룩을 해 주었다.  2번 문제 해결을 위해 플레이어 생성자 부분에서 scene Num을 인자로 추가해주었다.  if(sNum == 1)  m\_pCamera = ChangeCamera(INGAME\_SCENE\_CAMERA, 0.0f);  else  m\_pCamera = ChangeCamera(THIRD\_PERSON\_CAMERA, 0.0f);  이런식으로 필요할때만 INGAME\_SCENE\_CAMERA 로 설정을 해 주었다.  3. npc 그리기  저번과 같이 npc를 그리는데 isdraw라는 이름의 bool 변수를 추가해주어서 서버에서 그리라고 할 때만 그릴 수 있게 해주었다.  이후 myFunc\_SetMonPosition 함수를 통해 npc들의 위치를 그려주었다.  4. 시간 ui  처음 생각에는 다같이 들어가는거면 각자 알아서 시간을 체크해도 된다고 생각해서 인게임으로 들어갈 때 시간을 각자 재기 시작하고 ui로 그려주면 된다고 생각했다.  그래서 초반 코드는 매 프레임 불리는 함수에서 시간을 1씩 더해주고 1초가 되면 cursecond 변수에 1을 더하고 cursecond가 60이 되면 curminte에 1을 더해주는 방식으로 시간을 계산하고 이를 화면에 그려주었다.    동엽님과 회의를 통해서 서버가 관리해주는 것이 맞다고 판단하여 코드를 수정하였다.     1. 홍예나(클라이언트)   1. 이번주 한 것 :  1) 바운딩 박스  바운딩 박스 프로젝트의 어느 부분에서 m\_xmBoundingBox의 초기값이 들어가지는지 파악해 볼 것이다.  우선,    우리 프로젝트에서는 현재 AnimateObjects에서 걸렸을 때,    m\_xmBoundingBox 값이 아직 set 되지 않은 상태이다  그런데 AnimateObjects를 거치기 이전에 값이 set되어있어야 한다.  그 이유는    AnimateObjects 내부에서 충돌 체크를 해주는데  이전에 m\_xmBoundingBox 값이 세팅되어있지 않으면  사진과 같이 m\_xmBoundingBox의 값이 플레이어와 객체들 모두가 기본 값으로 되어있어서 무조건 충돌한다는 결과를 내게 된다.  그렇게 되면 플레이어가 이동을 할 수 없게 된다.    그런데 진짜 완전 이상한 점은 바운딩 박스 프로젝트에서도    사진과 같이 AnimateObjects에 걸렸을 때 여기도 m\_xmBoundingBox의 값이 제대로 세팅이 안된 기본 상태라는 것이다.  머지?  알아냈다    이 부분에서  214번 줄 거치기 이전 정보    214번 거친 후 정보    위 사진을 보면 214번 줄에서 m\_pMesh의 m\_xmBoundingBox가 가진 정보를 m\_xmBoundingBox에 넣어주고 있는 것을 볼 수 있다.  또한, 바운딩박스 프로젝트에서는    AnimateObjects를 거치기 이전에 이미 m\_pMesh 값이 초기화 되어있고,  m\_pMesh→m\_xmBoundingBox도 이미 값이 알맞게 들어가있다.  그런데 우리 프로젝트는    우선 m\_pMesh에 값이 들어있지 않다 허걱  m\_pMesh를 채우는 부분이 앞 과정에서 빠져있는 것 같다 이것만 찾으면 해결 될 것 같다!!        우선, 바운딩 박스 프로젝트에서는  CGameFrameWork::BuildObjects의 플레이어 생성 부분 → 플레이어 생성자  → 플레이어 bin 파일 읽는 부분에서 pMesh가 세팅된다.  그럼 우리 코드에는 저 부분이 어떻게 되어 있는지 한 번 볼까용    음 잘 set 하고 있는 것 같은디………..  설마 CStandardShader를 바운딩박스가 사용하지 않고 있어서 그런건 아니겠지?  뭔가 맞는 것 같다 아닌가?    디버깅 해보니까 값 잘 들어가고 있는 것 같은데 뭘까 대체???  543번째 줄에서 m\_pPlayer→m\_pMesh가 초기화되었어야 했는데 저 부분을 거쳤음에도 불구하고 m\_pMesh가 NULL임을 알 수 있다.    m\_pMesh가 플레이어 생성 과정에서 제대로 set 되지 않고 있다는 걸 의미하는데 다시 디버깅해보며 찾아보자      분명히 처음 값을 읽고, SetMesh를 한 시점에서는 분명히 m\_pMesh값은 물론,  m\_pMesh→m\_xmBoundingBox도 값이 잘 들어간 것을 확인할 수 있다.  setMesh를 벗어나고도    위와같이 값이 잘 들어가있는 것을 알 수 있다. 어디에서 값이 사라지는걸까  심지어 return하는 pGameObject에    값이 잘 들어가있  못찾겠어서 유림이한테 물어봤다  skinnedMesh 안에 있는 프레임 네임을 잘 보라고 햇다    mesh가 우주인 & 칼로 총 두 개 들어있었다.  파일이 상위 → 하위로 읽어가면서 pMesh를 업데이트하고, 그 pMesh를 GameObject의 m\_pMesh에 넣어준다.  그래서 우주인의 mesh가 들어가다가 칼 mesh로 m\_pMesh가 변경된다.  근데 그러다가 갑자기 null값이 들어감 → 이 부분을 찾아야 함  =⇒    여기였음  children에서 LoadFrameHierarchyFromFile을 한 번 더 타서 강제로 GameObject의 m\_pMesh가 빈 mesh로 채워진다. 그리고 children부터는 mesh값도 없어서 빈 값 그대로 넘어가게 되는 것임    m\_pMesh에 추가로 skin이 없는 부분의 메시 정보를 담는 m\_pNotSkinMesh, 자식의 mesh 정보를 담는 m\_pChildMesh를 만들어줬다.  우선, m\_xmBoundingBox의 데이터가 담기지 않은 m\_pMesh들을 Set하게 되는 것을 막기 위해 여러 방법들을 시도해보았다.  처음 시도한 방법은 아예 mesh 값을 읽어오는 부분에서 m\_pMesh가 NULL인 부분은 set하지 않는 방법이었다.  그런데 이 방법은 실패했는데 그 이유는 m\_pMesh의 데이터는 있지만 해당 m\_pMesh의 m\_xmBoundingBox의 데이터는 입력되지 않은(기본 생성 값 그대로) mesh 데이터들이 있었기 때문이다.  그래서 이 방법으로 구현하면 m\_pMesh의 m\_xmBoundingBox는 또 기본 값 그대로 들어가게 된다.    노란색 표시된 부분과 같이 구현해서 m\_pMesh와 m\_pSkinnedMesh가 NULL인 경우에만 set 하게 구현했었다.  두 번째로 시도한 방법은 mesh를 set하는 함수에서 m\_xmBoundingBox의 데이터가 있는 mesh만 set해주는 것이다.    아무리 생각해도 모르겠다  뭔가 내가 잘못 알고 있는 것 같다  pGameObject가 게임오브젝트 당 하나인거 아닌가?  뭔가 플레이어 안에 pGameObject가 여러 개이고, 가장 마지막에 만들어진 pGameObject의 값만 넘어가는 것 같은 느낌이다  그렇지 않으면 위에 사진 속 898번 줄에서 계속 m\_pMesh가 NULL일 수가 없는데……..  우선 처음 플레이어가 생성되는 과정은      이때, 이 부분에서 CGameObject가 생성되고, return되는 pGameObject(정보가 읽혀져 기록된)가 1번 사진 속 pLoadedModel→m\_pModelRootObject에 들어간다.    CGameObject 생성 개수  1번 : sk\_mesh\_astronaut\_sword가 프레임 네임으로 읽힘 → 메쉬 셋 x → 리턴x  2번 : astronaut\_mesh가 프레임 네임으로 읽힘 → 메쉬 셋 제대로 o → 리턴o  but 1함수로 안들어가고 children 읽으러 올라  3번 : RootJoint → 메쉬 셋x  4번 : hipjoint → 메쉬x  5번 : bellyjoint → 메쉬x, 리턴o but 1함수로 안들어감  6번 : leg\_l\_joint  플레이어 정보 전부 읽히고 난 후 저장된 정보    왜인지는 모르겠는데 가장 처음 반환된 프레임 네임이 들어가 있는데 정작 m\_pMesh는 NULL임  알아냈다  파일 읽어서 기록하는 구조를 몰랐던 내 무지가 원인이었다  우리가 사용하는 플레이어의 메쉬는 플레이어의 두 번째 자식 정보가 가지고 있었는데 내가 그걸 모르고 그냥 냅다 플레이어의 루트에 있는 정보만 가지고 충돌 체크를 하고 있었던 것이다…………………….    bin파일.txt와 bin파일 읽어오는 구조만 알았어도 더 빨리 찾을 수 있었을텐데라는 아쉬움이 남는다    충돌체크 내부 코드를 위와 같이 수정하였다  이제 클라와 클라간의 충돌처리는 된다  더 생각해봐야 할 부분은 맵과 클라들의 충돌체크이다  우리 맵은 루트의 자식이 아닌, 루트의 자식의 자식이 바운딩박스의 정보를 들고 있게 되어있어서 그부분은 좀 더 고민해봐야겠다  txt 파일을 보니 자식의 수를 읽어올 수 있다는걸 알게 되었다    그래서 오른쪽 사진과 같이 CGameObject의 멤버변수로 nChilds를 만들고 0으로 초기화 해준 후,  왼쪽과 같이 children의 정보를 읽어오는 부분에서  원래는 따로 int nChilds라고 만들어서 받아오던 것을 변수에 child의 수를 받고, 그 변수를 이용하도록 코드를 변경하였다    nChilds 수만큼 반복문을 돌리고,  pMapObject에 첫 번째 자식 객체의 주소를 넘기고  intersects함수가 true를 리턴하지 않으면 그 다음 자식(해당 자식의 형제)의 주소를 넘겨서 모든 맵의 객체들과 충돌체크를 할 수 있게 구현하였다  그런데 여기서 또 문제가 발생하는데 왜인지 모르겠지만 플레이어가 땅에 닿자마자 충돌체크가 된다 왜지??????????  이유를 알 수가 없다………..      알고보니 2번째 자식이 플레인이었다  플레인 바운딩박스는 설정해주지 않았는데 자동으로 들어가있는 것 같다  그래서    위와같이 FrameName이 Plane이면 그 다음 자식으로 넘어가도록 설정했고, 다음 자식이 없다면 for문을 빠져나가도록 구현하였다  오예 이렇게 충돌체크는 마무리됐다!!  2) 공격 동기화  동엽님이 플레이어가 공격키를 누르면    서버에서 누른 플레이어의 아이디의 isAttack을 true로 변경해준다  그래서 나는 그 정보를 받을 수 있도록 하기 위해    표시된 부분과 같이 인자의 마지막 부분에 getAttack으로 해당 순서의 플레이어가 공격키를 눌렀는지 안눌렀는지의 정보를 받아올 수 있도록 하였다.    위 사진과 같이 서버로부터 받아온 isAttack이 true면 클라 코드상에서의 해당 오브젝트의 m\_bIsAttack을 true로 변경하도록 했다    공격이 끝났을 때 클라 코드상에서의 m\_bIsAttack을 false로 돌려주는건 클라 코드에서 알아서 해줄 것이므로 안해도 괜찮다  이때, 공격이 이동보다 먼저 값을 입력받도록 해야 동기화했을 때의 딜레이가 없어질 것이라는 생각이 들어서    사진과 같이 다른 클라들의 공격을 그려주는 함수를 따로 만들었다    또한, 딜레이를 최대한 줄이기 위해 애니메이션이 실행되기 이전에 공격 여부를 받도록 순서를 정하였다.  이때, 다른 클라의 공격 애니메이션이 좀 더 빨리 실행되는 문제가 발생한다.  그리고 다른 클라 화면에서의 나와 내 화면에서의 나의 공격 애니메이션이 딜레이가 좀 있다.  3) 충돌 체크 동기화  씬전환 + UI + 서버 가 전부 들어가있는 브렌치에 collision을 머지했다  충돌 + 오류 나는 것도 경서랑 같이 고쳤다  우선, 0번 씬 - 로그인씬에서는 충돌체크가 되면 안되기 때문에    사진과 같이 충돌 처리 부분에 SceneNum이 0보다 큰 경우에만 충돌처리가 되도록 설정해주었다  여기서 문제가 발생한다  icemap은 씬 넘버가 3번이고, 로비는 씬 넘버가 1번이라  시작 후, 1번 누르고 3번으로 가면 충돌 처리가 안된다  정확히 말하면 충돌 체크도 되고, 충돌 처리 부분 코드도 타는데 충돌 처리만 안된다  왜지……………  혹시나 해서 바운딩 박스 내부 정보도 잘 들어가있는지 확인했는데 다 잘 들어가있다    이유를 알아냈다  서버와 동기화를 아직 안해서 그런 거였다 ㅎㅎ  그런데 서버에서 보내주는 포지션을 셋 하지 않아도 클라에서 해주던 move가 살아있어서 충돌 체크도 잘되고 클라가 충돌 체크 된 포지션을 서버에 보내주고, 그 포지션을 서버가 다른 클라들에게도 보내줘서 동기화를 하지 않아도 충돌 처리가 잘 보여지게 되었다!! 오예 | | | |
| **다음주 할 일** | | | |
| 1. 신동엽(서버)  * 씬 전환, 충돌, NPCAI 문제 해결 * 타이머 동기화  1. 정경서(클라이언트)  * 메모리 누수 잡기 * Ui 수정하기 * 타이머 동기화  1. 홍예나(클라이언트)  * 충돌 체크 & 충돌 처리 완성 * 충돌에 따른 공격 처리 완성 * 씬 전환을 위한 다리에서의 충돌 완성 | | | |
| **비고** | | | |
|  | | | |