|  |  |  |  |
| --- | --- | --- | --- |
| [Cranding(크랜딩)] | | | |
| **21주차** | **2024.5.12 ~ 2024.5.18** | **작성자** | **정경서** |
| **주간 회의** | | | |
| **회의 날짜** | | 5/16 | |
| 1. 보완해야 할 점  * 동기화 오류 * 씬 전환 오류 * 서버를 붙이며 생기는 오류도 있지만 시작하자마자 픽 메시지가 뜨며 생기는 오류도 있다. 해당 오류들을 먼저 잡는 것을 목표로 하자 * 인 게임 진입시 터지는 문제 * EVENT를 사용하면서 멈추는 것이 문제인 듯하여 빌드 오브젝트 관련 수정이 필요할 것 같다. * 공격 동기화시 애니메이션 딜레이가 생긴다 * 상 하체 분리에서 문제가 생기는 듯하여 애니메이션 수정 예정이다 * 시간 흐름 더 잘보이도록 스카이박스 수정하기 * 충돌 서버에서 관리하는 것으로 수정하기 * 맵 충돌박스 수정  1. 컨텐츠  * 한 맵에서 몬스터와 보스 등을 다 구현하고 다른 맵에 완성하는 것으로 하자 | | | |
| **이번주 한 일** | | | |
| 1. 신동엽(서버)   1. NPC Timer  기존에 클라이언트에서 시간을 체크하여 몬스터 젠 타이밍을 계산하여 패킷을 서버로 전송하면 서버에서 그 신호에 맞게 몬스터들을 생성하는 방식으로 구현하였지만, 게임 내의 시간을 클라이언트에서 재기보단 룸 단위로 서버에서 작동시키는게 맞는 것 같았다.  따라서 클라에서 시간채크하는 부분을 지우고, 서버에서 타이머 클래스를 수정하였다.  텍스트, 스크린샷, 소프트웨어, 멀티미디어 소프트웨어이(가) 표시된 사진  자동 생성된 설명  텍스트, 스크린샷, 소프트웨어, 디스플레이이(가) 표시된 사진  자동 생성된 설명  위 코드부분은 게임이 시작될 때 타이머 쓰레드에 시작타임과 몬스터 생성 타이밍을 맞춰서 집어넣어 정해진 시간마다 몬스터가 생성되고 생성된 몬스터들이 매 시간마다 우주선을 향해 이동하도록 설정하였다.  이에 따라 구조가 살짝 변경되었는데 몬스터의 행동들을 몬스터 Update로 묶어서 변화를 즉각 전송하도록 바꾸어 주었다.  텍스트, 스크린샷, 소프트웨어, 디스플레이이(가) 표시된 사진  자동 생성된 설명  몬스터의 속성과 행동들을 멤버함수와 멤버변수로 설정하였고  텍스트, 스크린샷, 소프트웨어, 멀티미디어 소프트웨어이(가) 표시된 사진  자동 생성된 설명  이를 다 관리하는 룸안에서 몬스터 업데이트 NPC를 불러 몬스터의 상태에 따라 이동할지 삭제할지 정해준다.  2. 충돌구현  충돌체크는 클라이언트 측에서 바운딩박스를 통해 급하게 구현하였다.  라인, 멀티미디어 소프트웨어, 스크린샷이(가) 표시된 사진  자동 생성된 설명  플레이어의 칼 바운딩 박스에 설정되어 있는 몬스터의 바운딩 박스와 충돌이 이루어 졌을 때 서버로 충돌이 되었다는 패킷을 전송하면 서버에서 그에 맞는 이벤트 처리를 해주게 변경하였다.  3. 몬스터 패킷 변경  기존에는 몬스터 array에서 각 객체마다 전송을하여 10마리의 몬스터를 전송하게되면 10번의 send를 루프를 통해 보냈어야 했다. 이는 송수신이 너무 많이 일어난다고 판단하여 몬스터 전체를 묶는 방식으로 패킷전송을 수정하였다.  텍스트, 스크린샷, 소프트웨어, 폰트이(가) 표시된 사진  자동 생성된 설명  이런식으로 밤에 출몰하는 몬스터들의 구조체를 한번에 묶어서 10번 보내는 것을 한번에 보내게 하였다. 패킷 사이즈가 커졌지만 255 사이즈가 넘어가지 않기에 아직 문제가 발생하진 않고있는데 추후에 성능을 비교해보아야 한다.  4. 중간발표 이후  중간발표때 실습실 컴퓨터 네트워크 환경이 집에서 테스트 하던 환경보다 좋지 않다는 것을 굉장히 간과하였다. 기존에 WaitforSingleObject를 통해 렌더링 쓰레드를 멈추고 전원이 ready패킷을 수신하면 이벤트를 넘겨 동시에 인게임을 그리게 하는 방식을 사용하였는데 자꾸 끊기는 원인이 이것으로 추정된다.  따라서 씬전환을 변경중에 있다.  소켓 옵션 설정.  중간발표 때 패킷이 밀리고 끊어지는 오류가 발생하였는데 김경철 교수님께서 소켓옵션NODELAY옵션을 추가해서 해보라고 하여 집에 돌아와 확인해보았다.  NODELAY 옵션은 네이글 알고리즘을 끄는 옵션이다. 네이글 알고리즘이란 네트워크 전송횟수는 줄이고 작은 데이터를 여러 번 전송하기 보단 데이터를 묶어서 한번에 보내는 방식을 사용하는 것이다. 네트워크 전송빈도가 큰 서버에게 설정하는 것이 바람직하다는 것을 알게되었다. 그렇다면 네이글 알고리즘을 끄는 것이 효율적일까?  정답은 없지만 트래픽에 민감한 서버에는 네이글 알고리즘을 활성화 시키고 응답시간이 중요한 게임 클라이언트에서는 네이글 알고리즘을 비활성화 시키는 것이 효율적일 것이라고 판단하여 소켓 설정을 변경하였다.  LINGER옵션 추가  끊어진 소켓에 대해 끝까지 송수신을 완료할 수 있도록 LINGER를 설정해주었다.  중간평가를 통과하긴 했지만 굉장히 아쉬웠던 느낌이 강했다. 서버에 대한 많은 개선이 필요할 것 같다. 더 열심히 해야할 듯  5. 다음주 할일  팀원들과 회의 결과 남은 5월안에 중간발표 때 생겼던 모든 버그들을 우선적으로 수정하기로 하였다.   1. 정경서(클라이언트) 2. UI 오류 관련   회의를 통해 오류 고치기를 최우선으로 하기로 하고 중간 발표 전에 UI 관련 오류를 고치기로 했다.  먼저 원래 코드의 문제는 CGameFramework::BuildObjects에서 각 씬에 맞는 UI를 생성해주었다. 이후 release를 해주려 했지만 생성한 이후 씬을 바꿔버리면서 어디가 꼬이는지 메모리 누수가 너무 심하게 나고 있었다.  그래서 따로 클래스에서 모두 관리를 해주는 방식으로 코드를 수정했다.  CGameFramework::BuildObjects에서 UILayer::Create를 해주고 초기화를 진행해 주었다.  초기화를 해주며 각 씬에서 필요한 모든 text, brush, textformat 그리고 D2D1\_RECT\_F까지 UI에 필요한 모든 부분을 초기화 해주고 릴리즈 해줄 때 초기화 해준 모든 변수들을 릴리즈 해주면서 오류를 잡았다.  Ui를 그릴때도 UILayer::Render에서 조금 더 편하게 그려줄 수 있게 되었다. 씬 넘버를 받아와서 그리는데 이후 씬매니저를 추가해서 더 쉽게 그릴 수 있도록 하고싶다.  씬을 수정해서 다리로 가면 이동 ui가 뜨도록 해두었다.    처음에는 이런식의 바운딩 박스를 만들었는데 왜인지 코드에서 위치를 못읽어오고 바운딩 박스가 없다고 하길래 살짝 꼼수를 써버렸다    이런식으로 돌의 바운딩 박스를 길게 늘려주었다.    돌과 충돌이 되면 이동 메시지가 뜨게 했고 이후 수정을 더 할 예정이다.   1. 화면 해상도 변경   중간발표 전에 전체 화면으로 전환하는 것이 좋겠다는 의견이 있어서 해상도를 변경하였다.    이에 맞게 UI도 모두 수정해주었다.  3. 수정 또 수정  원래는 ip주소를 코드에 입력하고 실행을 했는데 exe 파일을 추출하게 되면 매번 코드를 수정할 수 없어서 ip 주소를 입력받을수 있게 수정했다.     1. 홍예나(클라이언트) 2. 충돌처리   : 동엽님께서 npc 로직을 구현해주셔서 몬스터들이 우주선을 향해 걸어오기 시작했다  그래서 걸어오는 몬스터들이 플레이어가 공격할 때 플레이어의 칼에 맞으면 죽도록 구현하기 위해 몬스터의 m\_xmBoundingBox.Center를 npc의 pos로 계속해서 update 해줘야 했다    또한, 충돌 체크를 위해 아래 함수에 67번째 줄에 플레이어가 현재 공격 하는 중이며, npc와 플레이어의 칼이 충돌되는 경우 true를 return하도록 구현했다.  또한 충돌하게되면 충돌한 npc의 id를 서버로 보내서 해당 npc를 사라지게 하도록 구현했다.  Npc는 3~12번까지 존재하는 객체이고, id가 0부터 시작하기 때문에 i-3을 id로 생각할 수 있다.    이때, 충돌체크가 npc 중 0번째만 되고, 플레이어끼리도 한 플레이어와만 충돌체크가 되는 문제가 발생했다  이의 원인은 하나의 모델로 모든 플레이어와 npc 객체를 생성해주었기 때문이었다  이를 방지하기 위해    사진과 같이 전부 모델을 다르게 하여 생성해주었더니 충돌체크가 제대로 되기 시작했다   1. 애니메이션 속도 조정   : 2명만 로그인 되던 걸 3명으로 늘렸더니 다른 플레이어들의 속도가 내 화면에서 전부 2배 빠르게 나와서 전부 0.5배로 애니메이션 속도를 줄여주었다     1. 원소 생성 && 충돌시 사라지도록 구현   : 원소를 npc 다음 객체들로 생성해주어서 npc들이 플레이어가 공격할 때의 칼과 충돌하게 되면 해당 위치에 생성되고, 그 원소와 충돌하면 해당 원소가 사라지도록 구현하였다.      몬스터와 공격중인 플레이어의 칼이 충돌해서 몬스터가 죽으면 해당 몬스터의 position에 원소가 위치하도록 하였고, 원소의 바운딩 박스도 해당 pos로 옮겨지도록 하였다.    충돌 처리는 checkObjectbyObjectCollisions에 만들어주었다.  원소와 플레이어가 충돌하게 되면 원소를 0,0,-999위치로 옮기고, 바운딩 박스의 center.y 좌표도 -999로 옮겨서 게임 플레이에 방해가 되지 않도록 하였다. | | | |
| **다음주 할 일** | | | |
| 1. 신동엽(서버)  * 씬 전환 방식 변경 * 애니메이션 밀리는 오류 해결 * 동기화 테스트  1. 정경서(클라이언트)  * UI 수정 및 빌보드를 이용한 UI 추가 * 씬 전환 방식 변경 * 외곽선 추출  1. 홍예나(클라이언트)  * 공격 애니메이션 수정 * 그림자 공부 | | | |
| **비고** | | | |
|  | | | |