|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | | **작성자** | **2018180033 이세철 2018180046 허재성 2018182009 김승환** | **팀명** | XD |
| **계획 시트** | <https://docs.google.com/spreadsheets/d/1rUOOKpcVfkma18YsvTZXZz44z6UhbGfOsrimVgs-mzE/edit#gid=0> | | | | |
| **주차** | **16~17** | **기간** | **2023.05.26~2023.06.08** | **지도교수** | **정 내 훈**(서명) |
| **이번주 한일** | * **이세철:** * **허재성: 서버-클라 동기화 버그 수정 협업, 애니메이션 제작 및 모델 수정 / 적용 및 개선작업, 충돌 모션 및 이펙트 구현중** * **김승환:**   **1. 클라이언트 스레드 분리**  **2. 플레이어 공격 및 NPC 피격 처리 (+ 총알 수 적용)**  **3. 시야 처리 (View List 적용)**  **4. 미션 구현**  **5. 점령 구현** | | | | |

**<상세 수행내용>**

1. **이세철**
2. **허재성**

**[1] 서버 – 클라 동기화 협업**

플레이어가 애니메이션이 들어간 사람모델로 변경전에는 보이지 않던 동기화 버그 현상이 생겨서 서버와 원인을 찾는데 많은 시간이 걸렸습니다.

사람모델로 변경하면서 인게임 입장시 많은 버벅임과 함께, 이동과 회전이 매우 끊기는 현상이 생겨서 원인을 찾기 시작했습니다.

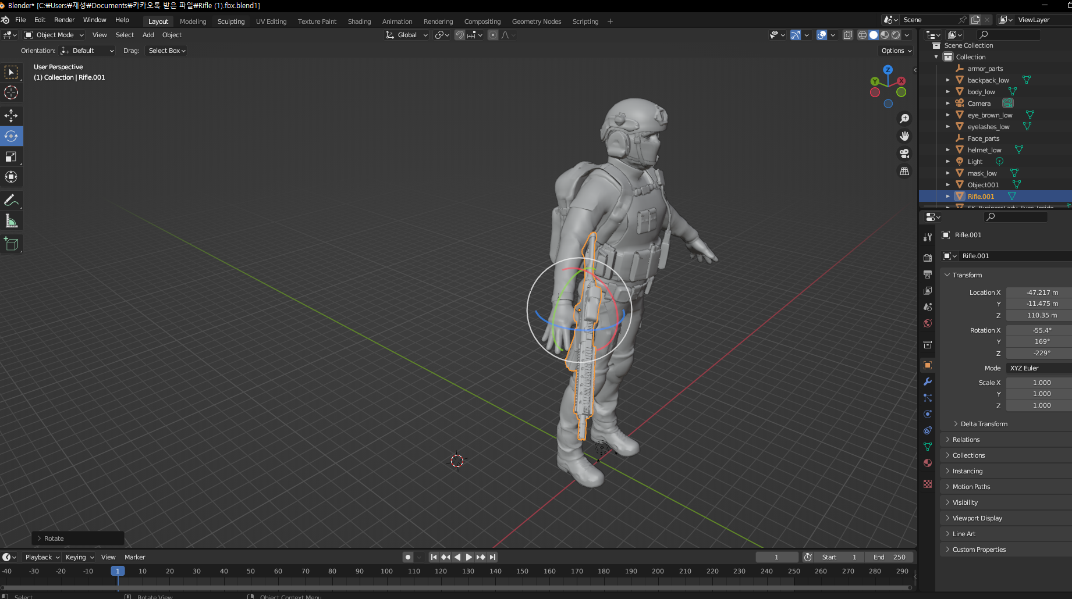
클라이언트 입장에서는 리깅이 들어가고 이동간의 헬리콥터보다 많은 연산이 요구되어 패킷이 밀리는 현상이 생겼다고 판단하여, 메모리 릭이 발생하는 부분을 계속 찾고 있기도 했고, 승환이와 이런저런 예상되는 모든 문제점을 찾아본결과, 클라이언트 메인 스레드와 패킷을 보내는 통신 스레드를 하나의 Winmain에서 처리하고 있던 부분을 나누기로 하였고, 서버에서 스레드를 분리 시켜주어 렌더링 파이프라인과 패킷이 송수신하는 부분을 그 전보다 원활하게 진행할 수 있게 되어 끊김 현상을 최소화 하였습니다.

분리 방식은 승환이의 작업일지에서 마저 설명 드리겠습니다.

**[2] 애니메이션 삽입 및 모델 수정 & 애니메이션 적용**

Unity에서 게임배경에 맞는 새로운 모델을 가져와서, 믹사모에서 애니메이션을 구해 적용을 했습니다.

총과 사람이 떨어져 있는 상태로 모델이 와서 “Blender”에서 모델을 수정하고 Export하여 애니메이션을 적용하려 했으나, 총까지 리깅이 작용되어 MAX 에서 리깅모델링을 통해 작업을 해주어서 다소 시간이 걸렸습니다.



이후에 애니메이션을 Unity에서 적용시키고 저희 프로젝트에 Import 시켰습니다.

 동작은 현재 **앞/뒤/오른쪽/왼쪽 걷기 모션**이 따로 따로 되어있고, **총을 쏘는 모션 장전과 점프, 죽는 모션**까지 되어있습니다.

앞으로 추가할 동작들은 부상을 입었을 때 **절뚝거리는 동작과**, 헬기를 쏴야하기에 **상공을 보고 쏘는 모션**을 추가할 예정입니다.

동작들의 연결이 어색한 부분이 있기에, 다음면담때까지 수정가능단계에 있으며, 준비된 피격 이펙트도 서버에서 판단되는 충돌테스트가 거의 끝나가기에 이펙트와 모션을 자연스럽게 적용하고 맵충돌을 서버와 같이 작업하는 것이 다음주까지의 작업이 되겠습니다.

컨텐츠와 동기화를 하는데 시간을 많이 투자하고 시험기간도 겹친 부분도 있어서, 종강 시즌이 다가왔기에, 남은 6월은 빠르게 컨텐츠를 마무리 해가면서 앞으로의 남은 시간을 많이 쏟아서 비주얼적인 개선도 더 힘쓰는 한달이 되겠습니다.

1. **김승환**

**1. 클라이언트 스레드 분리**

플레이어 객체를 헬리콥터에서 인간으로 바꾼 뒤로 객체들의 움직임이나 UI 업데이트 속도가 상당히 지연되는 문제가 확인되었습니다. 측정결과 키 입력 시간으로부터 패킷을 서버로 보내는 시간까지 대략 20초 정도의 딜레이가 발생하는 것을 확인하였습니다.

이러한 문제를 해결하고자 가장 처음 접근했던 방식은 네이글 알고리즘을 끄는 것이었습니다. 클라이언트에서 키가 입력되었을 때 패킷이 바로바로 전송되지 않고, 한참 지연되었다가 한 번에 몰아서 전송되는 것이 네이글 알고리즘을 연상케 하였기에 혹시 원인이 네이글 알고리즘에 있지 않을까 해서 setsockopt를 사용하여 네이글 알고리즘을 꺼보았습니다. 하지만 네이글 알고리즘을 껐음에도 패킷 전송 지연 문제는 그대로였습니다.

클라이언트에서 리깅을 도입하면서부터 패킷 전송이 지연되었던 것이기에 클라이언트 파이프라인이 리깅으로 인해 복잡해지면서 패킷 전송 및 수신이 늦어지는 것이라 판단하였고, 클라이언트의 렌더링 파이프라인과 네트워킹 기능을 병렬적으로 돌아가게 하기 위해 클라이언트 메인 스레드로부터 네트워크 스레드를 따로 분리하게 되었습니다. 그 결과 패킷 전송 및 수신이 지연되지 않고 정상 작동하는 것을 확인하였습니다.

서버로부터 패킷을 받아 UI를 업데이트 하는 것에도 상당 시간 지연되는 문제가 있었기에 UI 업데이트를 담당하는 스레드도 하나를 만들게 되었습니다.

현재는 네트워킹과 UI 업데이트 모두 지연이 생기지 않고 있습니다만, 이상한 문제가 하나 생겼습니다. 이전에는 클라이언트의 오프닝 씬에서 1번 키를 입력하면 서버로 스테이지 전환 요청을 보내고 서버가 스테이지1로의 진입을 허가해주면 클라이언트에서는 허가 패킷을 받자마자 바로 스테이지1 진입을 위한 작업이 진행되었습니다.

텍스트, 스크린샷, 소프트웨어, 멀티미디어 소프트웨어이(가) 표시된 사진

자동 생성된 설명

하지만 “Client[0] Stage1로 전환” 이라는 서버 메시지를 출력하였을 때 분명 서버가 클라이언트로 스테이지 입장 허가 패킷을 전송해주었고, 클라이언트도 그 패킷을 바로 받은 것이 확인되었는데 정작 클라이언트에서 씬 전환이 이뤄지지 않는 문제입니다.

텍스트, 스크린샷, 소프트웨어, 멀티미디어 소프트웨어이(가) 표시된 사진

자동 생성된 설명

위 이미지는 클라이언트에서 1번 키를 입력하고 서버에서 “Client[0] Stage1로 전환” 서버 메시지를 출력하였을 때 클라이언트의 디버깅용 콘솔창을 클릭하면 그제서야 모델들을 불러오면서 씬 전환 준비를 하는 모습입니다.

텍스트, 폰트, 스크린샷, 라인이(가) 표시된 사진

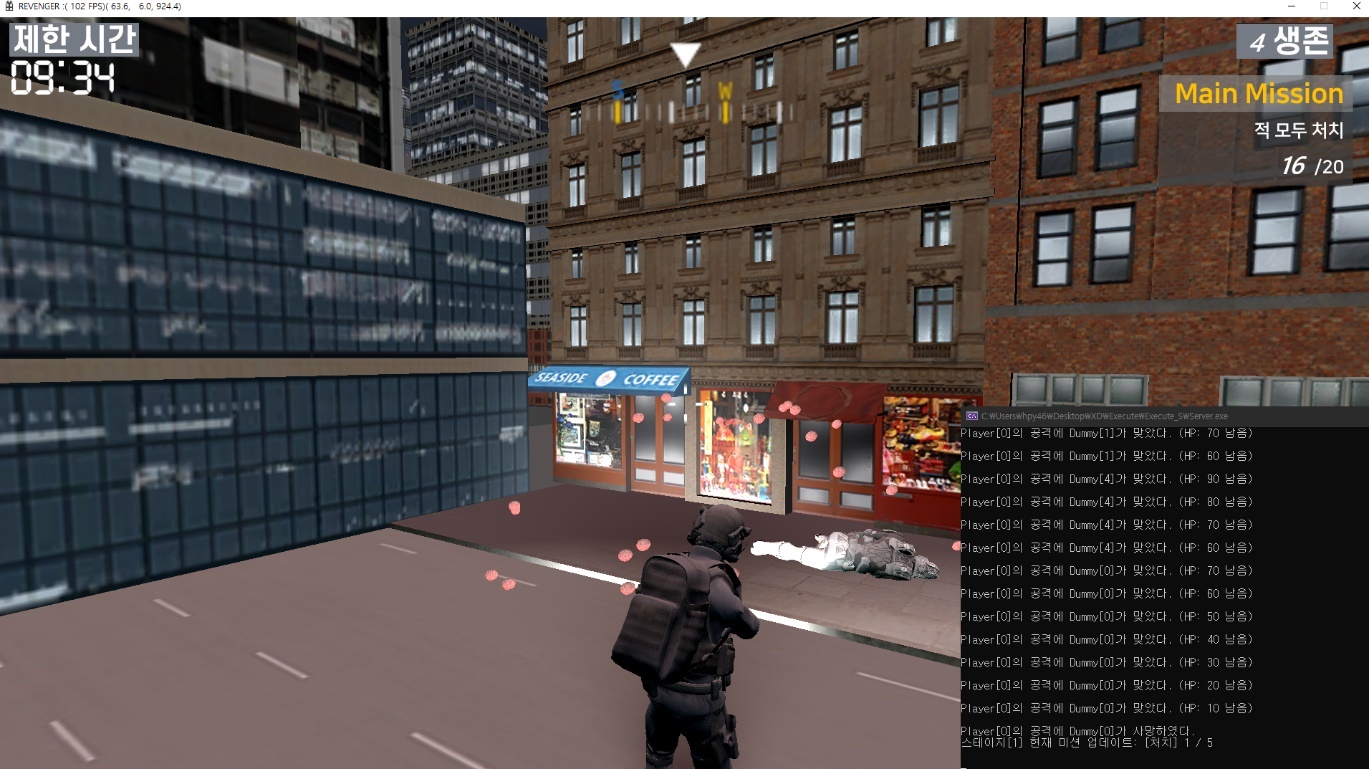
자동 생성된 설명텍스트, 폰트, 스크린샷, 블랙이(가) 표시된 사진

자동 생성된 설명

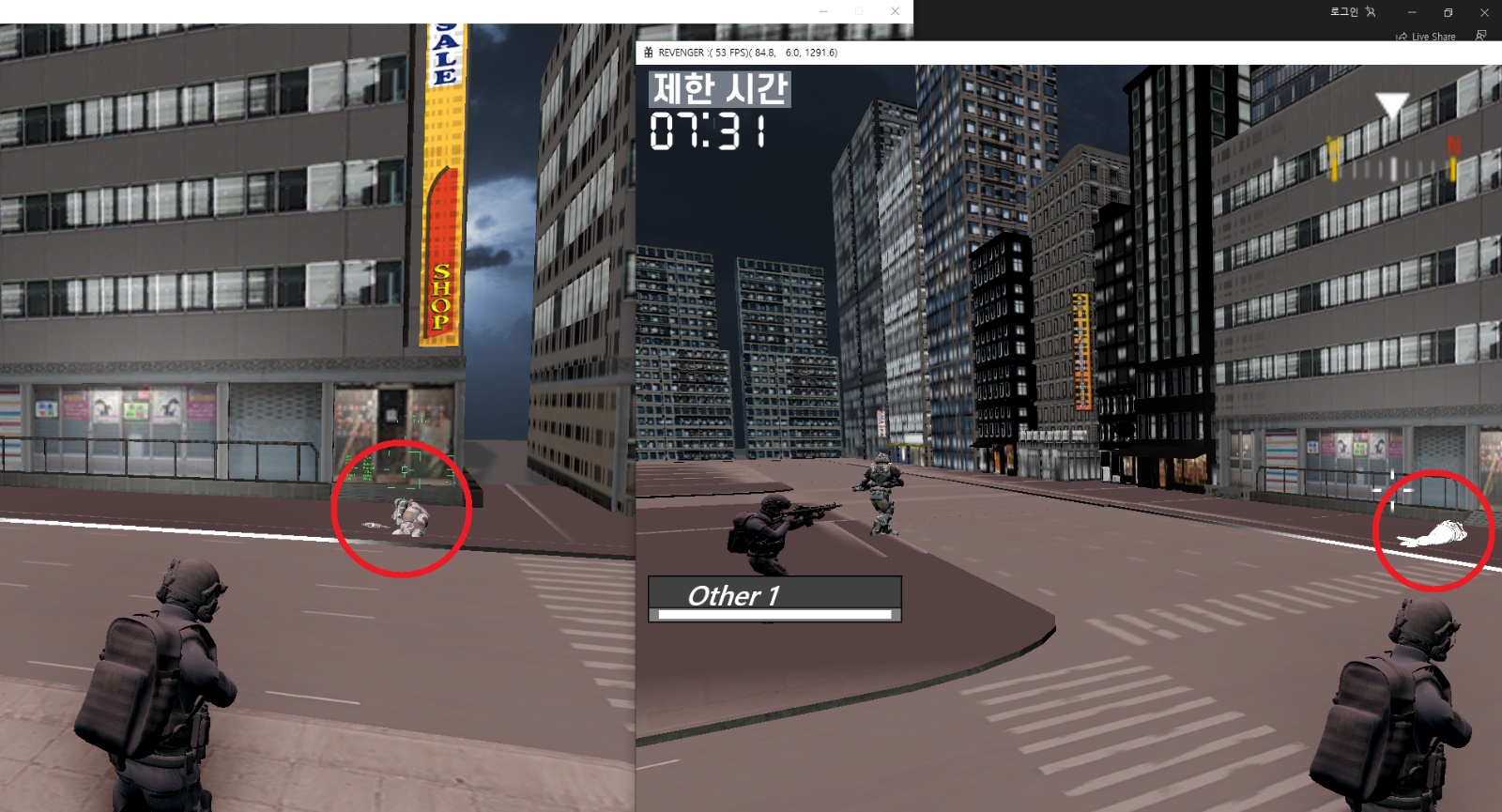
처음에는 이것이 네트워크 스레드나 UI 스레드에서 메인 스레드로의 스레드 컨텍스트 스위치가 이뤄지지 않아 생기는 문제라고 생각하여 네트워크, UI 스레드에서 메인 스레드로 양보하도록 yield를 써보았으나 효과가 전혀 없었습니다.

이러한 문제를 아직까지도 원인을 찾지 못하여 해결하지 못하고 있는 상황입니다.

**2. 플레이어 공격 및 NPC 피격 처리 (+ 총알 수 적용)**



원래 계획은 제가 만든 Ray Cast 함수를 사용하여 충돌체크를 하는 것이었지만, 이론상으로 충돌이 되어야 하는데 실제로 적용해보니 충돌이 전혀 되지 않았고 결국 Ray Cast를 사용한 충돌체크는 잠시 보류하기로 하였습니다.

그래도 전투 구현은 해야 하니까, 클라이언트가 공격키를 누르면 서버에서 점만큼이나 크기가 작은 가상의 총알 객체를 만들고 이를 플레이어의 Look Vector 방향으로 정해진 사거리까지 이동시키면서 총알의 바운딩박스와 NPC의 바운딩박스를 갖고 DirectX의 Intersect( )를 통해 충돌검사를 하여 충돌하면 NPC의 HP를 일정량 깎고, HP가 0이되면 사망하도록 구현을 하였습니다.

이 때 NPC들의 상태나 HP는 동기화 되어있어 모든 클라이언트에서 NPC들의 동일한 상태를 볼 수 있게 하였습니다.

스크린샷, PC 게임, 3D 모델링, 비디오 게임 소프트웨어이(가) 표시된 사진

자동 생성된 설명

총알 수도 서버와 연동하여 공격할 때마다 탄이 하나씩 줄어들게 하였습니다.

(재장전은 아직 미구현 상태여서 임시로 탄을 다 쓰면 다시 꽉 차도록 해놓았습니다.)

**3. 시야 처리 (View List 적용)**

스크린샷, PC 게임, 비디오 게임 소프트웨어, 건물이(가) 표시된 사진

자동 생성된 설명

시야처리는 당연히 처음부터 했어야 했는데 그 동안 못하고 있었고, 늦게나마 시야처리를 하게 되었습니다.

세션의 필드로 View List를 추가하여 플레이어가 이동을 할 때에 자신의 시야 안에 객체가 새로 들어왔는지 혹은 범위 밖으로 벗어난 객체가 있는지를 검사하여 시야 리스트에 추가 혹은 시야 리스트에서 제거하도록 하였습니다.

그리고 그 플레이어와 관련된 패킷 (ex. 이동패킷, 회전패킷 등)을 클라이언트로 전송할 때에는 시야 리스트에 있는 클라이언트에게만 전송하도록 수정하였고, 충돌 검사를 할 때에도 시야 안에 있는 객체(ex. Npc 등)들만 검사를 하도록 수정하였습니다.

**4. 미션 구현**

**5. 점령 구현**

|  |  |  |  |
| --- | --- | --- | --- |
| **문제점 정리** |  | **해결 방안** |  |
| **다음 주차** | **17** | **다음 기간** | **2023.06.09~2023.06.22** |
| **다음주 할 일** | 이세철:  허재성: 맵충돌, 애니메이션 믹싱, 피격 이펙트 삽입  김승환: | | |
| **지도교수**  **Comment** |  | | |