|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | | **작성자** | **2017180012 박주용**  **2017180029 이예준**  **2017182030 이수민** | **팀명** | FruitsPangPang |
| **주차** |  | **기간** | **2022.04.20~2022.04.26** | **지도교수** | **정 내 훈**(서명) |
| **이번주 한일** | * **박주용**   **중간발표ppt 제작, 라이트 관련 오류 수정, 병원 내부 수정, 백그라운드 제작**   * **이예준**   **중간발표ppt 제작, 근접무기 관련 버그 수정, 피격 애니메이션**   * **이수민 중간발표ppt 제작, 로비 서버 해결방안 생각, 치트키** | | | | |

**<상세 수행내용>**

* + **박주용**
* 라이트 관련 오류 수정(+병원 내부 수정)

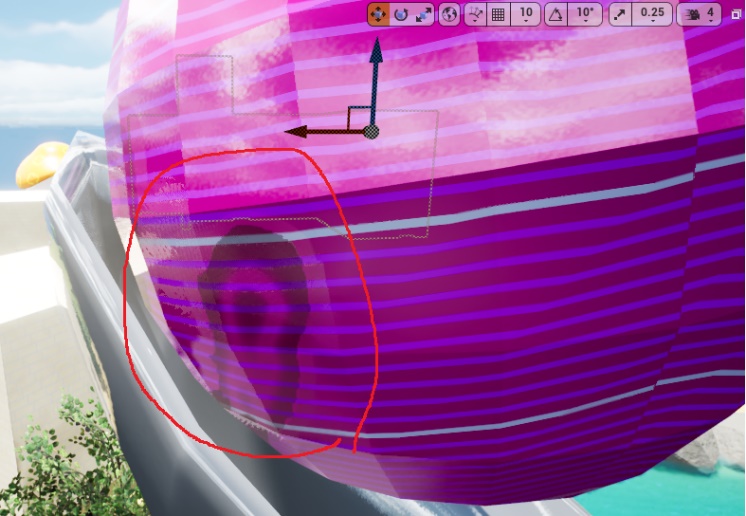
텍스트, 실내이(가) 표시된 사진

자동 생성된 설명<수정 전>

텍스트이(가) 표시된 사진

자동 생성된 설명<수정 후>

* 기존에 병원 및 다른 건물에서 알 수 없는 까만 선이 그려졌었다. 수많은 삽질 끝에 라이트 맵 관련, 언리얼 skylight의 DFAO(가짜 반사광) 문제라는 것을 알게 되었다. 올바른 라이트 맵을 갖도록 하고 라이트 맵 해상도 증가, 디렉셔널 라이트/스카이 라이트 수정 등의 작업을 통해 문제를 해결하였다. (+기존 병원 내부는 지저분한 느낌이라 깔끔한 흰색 원톤으로 맞췄다. 캐릭터와 상호작용하는 조명, 그리고 의자와 문틀도 넣어보았다.)
* 과일 상자, 세계수, 체리 라이트 블리딩 오류 수정.



겉 표면에 그림자(?)가 일그러지는 문제가 있었다.

역시 라이트 맵 관련 문제인 것 같고 기존의 스태틱 메쉬를 merge하여 새로운 스태틱 메쉬를 만들고 교체하여 해결하였다.

* 백그라운드 제작

하늘, 실외, 해안, 다채로운이(가) 표시된 사진

자동 생성된 설명

기존 맵의 WaterMeshActor를 이용하여 적정 거리까지 물 머터리얼을 깔고,

새로운 랜드스케이프를 하나 만들어서 넓게 깔고 끝 부분을 스컬프팅,

다운 받은 산 배경을 그 뒤에 배치해 연결감을 주었다.

+ 맵의 끝부분은 캐릭터가 떨어지지 않게 4면을 장벽을 두른 후 invisible 처리.

* + **이예준**

1. 근접무기 변경 시 당근이랑 대파가 동시에 들리는 버그 해결.
2. 근접무기 획득 시 근접무기가 가방에 바로 꽂히게 수정
3. 피격 시 피격 애니메이션 발동

텍스트이(가) 표시된 사진

자동 생성된 설명<피격된 상태>

후에 머리를 맞았을 때에만 피격 애니메이션을 하도록 수정할 것.

(몸통과 머리 타격감에 약간의 차이를 주기 위해)

* + **이수민**

- 게임 종료 10초전으로 만드는 치트키: P키

- 로비서버의 기본적인 설계

-> 8명이 매칭 중이면, win32api shellexecute함수를 통해 게임서버를 켜주고, 게임 한 사이클이 종료되면 게임서버는 로비서버에게 자신이 끝났음을 알리고 프로세스를 종료한다.

따라서 로비는 플레이어와 연결도 되어야 하고 게임 서버와도 연결이 되어야 한다.

* 현재 문제되는 점:

로비서버가 Accept를 받을 때 연결요청을 한 사람이 player인지, gameserver인지 판별이 안된다는 점

* 대안 1.

판별을 안 하기 위해 상속 구조를 사용한다고 하면, 현재 설계인

Object - Character - player

Server - gameserver

이 구조를 연결시키기 난해하므로 이 방법은 일단 보류한다.

* 대안 2.

CreateIoCompletionPort로 GameServer전용 IOCP 커널 객체를 하나 더 두어

GetQueuedCompletionStatus의 첫번째 인자로 전달하면 서버 전용 gqcs를 만들 수 있나?

-> 코딩을 직접 해본 결과, Accept를 받는 건 server socket이 하는 것인데, server소켓은 iocp커널 객체 1개만 두므로 이거 말이 안됨.

* 대안 3.

게임서버를 킬 때는 lock을 걸어서 accept에 게임서버만 들어올 수 있게 한다.

-> 코딩 결과 lock을 걸었을 때, 플레이어 or 게임서버 둘 중 누가 누구인지 모르는 건 여전하다. 실패.

* 대안 4.

게임서버를 미리 다 켜버린 후, 그 다음에 플레이어 accept를 받을 수 있도록 코드를 짠다.

-> 기존 계획은 게임서버가 수명이 다 하면(한 게임이 끝나게 되면) 서버를 종료하고 재부팅을 하려고 했으나

이 방법을 채택한다면, 게임서버를 종료, 재부팅이 아닌, 수명이 끝나면 재활용을 해야 한다(기획 수정).

* 대안 5.

로비는 그저 게임서버를 켤 뿐 상호간에 받는 데이터는 없도록 한다.

-> 이렇게 되면 굳이 서버간 연결을 할 필요가 전혀 없어진다.

-> 게임서버가 종료되고나서 바뀌는 플레이어들의 정보는 게임서버가 따로 db에 업데이트 하도록 한다.

-> 게임서버는 재활용되거나, 재부팅 되지 않고, 방이 필요할 때 마다 계속해서 켜지는 형식.

결과: 4번 또는 5번이 그나마 괜찮아 보이는데 어떤 걸 선택해야 할 지 잘 모르겠는 상황이다.

|  |  |  |  |
| --- | --- | --- | --- |
| **문제점 정리** | 로비서버 설계에 대한 난항. | **해결 방안** | 교수님께 조언 구하기 |
| **다음 주차** |  | **다음 기간** | **2022.04.20~2022.04.26** |
| **다음주 할 일** |  | | |
| **지도교수**  **Comment** |  | | |