|  |  |  |  |
| --- | --- | --- | --- |
| Magician War | | | |
| 18주 | 2021. 4. 25 ~ 2021. 5. 01 | 작성자 | 김덕규 |
| 주간 목표 | 홍범도   1. **서버 workThread 구현 ( 70% )**   김덕규   1. **충돌체크(PhysX 공부)** 2. **불기둥 (Dissolve 예시) 제작** | | |
| 이번주 한 일 | 홍범도   1. **서버 workThread 구현 & Mutex로 동시성 제어**  * 기존의 플레이어 1명당 1개의 playerThread를 할당하는 방식으로 그 안에서 recv(), 충돌체크, send()를 해주었는데 이런 방식으로는   충돌체크를 각 스레드 별로 해주어야 하고 그만큼 스레드 동기화도 각 스레드별 충돌체크마다 해줘야 해서 CPU에서의 부담과 mutex가 교착 상태가 되는 경우 디버깅이 어려웠다.  이를 workThread를 하나 둬서 충돌체크와 같은 일들을 해주고 send()를 하기로 하였다.  Mutex를 활용해서 내가 글로벌 변수들(공유 자원)을 활용하고 있다면 lock() 그게 아니라면 unlock()를 사용해서 다른 스레드에서의 공유 자원의 접근을 제어해 주는 작업중이다. ( 이 부분 하고 적용하면 중간 발표 전까지 서버 건들일 없을 듯 )  김덕규   1. 충돌체크 (PhysX 공부)   작년에 팀 제작했던 게임에서 피직스를 담당한 동기의 코드가 있어 공부할 수 있었다.  그 당시에 가장 큰 문제였던, 충돌 시 충돌했다고 반환하는 코드를 생각해야 한다고 했다.  지금 상황에서는 PxMeshOverlapUtil이라는 인터페이스 클래스 내부에 findOverlap으로 충돌체크를 판단하는 게 좋다는 생각이 든다.  주말에 충돌체크를 대입해보며 필요 시에 피직스 경험자인 동기에게 질문을 해서 붙일 예정이다.   1. 불기둥 (Dissolve 예시) 제작     3d Max를 통해 원통 메쉬를 제작하고, 디졸브 기법을 적용시켰다.  노이즈 텍스쳐를 통해 디졸브 되는 외곽선을 땄다. | | |
| 다음주 할 일 | 김덕규   1. **충돌 체크(피직스 대입)**   홍범도   1. **스킬 구현** | | |
| 문제점 | 홍범도  중간 발표까지 2주남았다. 시험도 끝났으니 집중해서 태우자  김덕규  중간 고사를 공부하느라 졸업작품에 제대로 집중하지 못했다. 더 열심히 해야겠다. | | |