|  |  |  |  |
| --- | --- | --- | --- |
| 도둑들 | | | |
| **7월 1주** | **2023. 7. 2 ~ 2023. 7. 8** | **작성자** | **정극훈** |
| **이번 주**  **목표** | **이도영: 네비게이션 매쉬 Astar 구현, 충돌처리 오류 수정**  **정극훈: 게임 UI 배치, 파티클 효과 제작**  **김혁동: 게임 로직 디버깅** | | |
| **7/2**  **일** | **이도영:** 선배를 만나 collision 관련 코드들을 종합적으로 다시 살펴보았음.  Collision이 잘못되었기에 잘못된 값이 할당되는 것일 수도 있기에  OBB SAT 축 정렬 방식을 이용하여 충돌 검사를 시행하는데 이 부분에는 문제가 없었음.  Collide 자체는 잘 되어지고 있다는 결론을 냄  면 기준 법선벡터가 잘 들어갔는지 디버깅하며 확인함.  Collide 자체는 문제가 없으니 계산하는 수식만 조금 바꾸면 될 것이라 말씀하심 | | |
| **7/3**  **월** | **이도영:** 다른 다이렉트 x의 코드를 보며 비교를 해봄  텍스트, 스크린샷, 디스플레이이(가) 표시된 사진  자동 생성된 설명    다렉 코드를 보며 구조는 다르지만 짠 코드는 큰 차이는 없다 생각함.  수식이 어디서 잘못된 지 다렉 함수들을 하나하나 까보았으나 특별한 코드는 없었고 그냥 계산하는 코드들뿐이었다. 벡터를 구하는 공식 그대로 되어있으며 vec3에 선언된 함수와 차이점은 없음. 이를 이용해서 코드를 작성해 봄. 다렉을 제외하고 코드를 작성하여 코드를 대입함.  드드드드 하며 캐릭터가 충돌시 반발을 받아 튕겨 나오게 됨.  이는 충돌면이 제대로 나오는 것이라 생각하며 충돌이후 처리가 문제가 있어서 그런 것이라 생각 | | |
| **7/4**  **화** | **이도영:** 충돌면이 제대로 나왔는지 다시 확인 면 기준 바깥으로 나온 벡터가 아니라 안쪽으로 가나?  그건 아님  맵이 x y z 축으로 거의 되어 있어 노말 벡터를 고정시켜 충돌을 몇 번째 면이랑만 했는지 확인  충돌 면 구할 때 거리기준으로 구했는지 확인  이래도 잘 안 되면 충돌면이 제대로 나왔는지 확인하며 다시 계산 코드 작성  박스 범위에 따라서 center 지점이 다른데 뜬금없이 윗면하고 충돌하여 판정하는 경우도 있음.  면과면 충돌할 때 바운딩 박스 center좌표 혹은 임의로 캐릭터 좌표 하나 잡아서 충돌면을 계산하거나 벽이면 바운딩 박스에 extents값이 들어가서 곱해져 있는데 1로 정규화 시켜 거리 구해보려 함. | | |
| **7/5**  **수** | **김혁동:** 룸 및 로비에서의 속도 저하 문제에 대하여 디버깅을 해 보았음.  클라이언트를 멀티 스레드 환경으로 변경하는 과정에서 mutex를 추가해 lock을 이유 없이 프레임 단위로 지속적으로 거는 부분을 다수 발견하였음.  현재 로비나 룸 등에서 프레임 마다 전부 lock을 걸고 있다면 상당한 오버헤드가 있을 것으로 예상되어 이 부분을 변경하였음. 이후에는 속도가 정상적으로 변한 것을 확인함.    **<프레임이 안정적으로 변한 것을 확인함> (수정됨)**  **정극훈:** particle 작업을 시작했다.  마지막으로 파티클 코드를 만진지 오래 돼서 그런지 파티클 코드가 기억이 잘 안났다.  그래서 파티클을 생성하는 코드를 다시 공부했다.  인강을 보면서 코드를 확인하며 복습한 결과 파티클 효과를 변경하려면 파티클 쉐이더의 코드를 바꿔야 한다는 점을 확인했다.  도둑들에서 사용할 여러가지 파티클 효과 중 상자를 열었을 때 파티클이 위로 올라갔다가 내려오는 폭죽과 같은 효과가 있다.  처음 목표는 파티클 쉐이더의 코드를 수정할 때 이러한 효과가 나오게끔 할 생각이다. | | |
| **7/6**  **목** | **정극훈:** 오늘은 파티클 쉐이더 코드를 이리저리 만져봤다.  우선 지금 쉐이더는 파티클이 사방으로 퍼지게끔 한다. 이 코드를 수정해서 파티클이 위로만 퍼지게끔 바꾸어 봤다. 이제 올라간 파티클이 다시 아래로 떨어져야 하는데 현재 파티클 코드는 alive로 파티클의 생애주기를 결정한다. 그런데 여기에 따로 alive를 추가할려니 쉐이더 오류가 계속 발생했다.  그래서 deltaTime으로 바꿀려했으나 쉐이더코드에 델타타임을 넣을 방법이 없었다.  아무래도 기존의 alive 방식의 오류를 수정해서 파티클 효과를 만들어야 할 것 같다.  **이도영:** Extents 값 생각을 하지 않으면 다른 곳 박았다고 처리될 수도 있음.  아랫면에 충돌하면 아랫면에 충돌했다고 나와야 하는데 extent값이 차이가 크면 충돌면이 이상하게 될 때가 생김 충돌면을 계산할 때 바운딩 박스마다 center좌표 거리차이로 계산해서 가장 가까웠나 멀었나 한 걸로 충돌면을 알려주는데 동그라미 영역이 조건에 부합할거기에 extents 값이 적용된 수식이 있나 확인 먼저하고 코드를 수정 해봄. | | |
| **7/7**  **금** | **김혁동:** 게임에 접속하였을 때, 플레이어 한 명을 제외한 나머지 플레이어가 움직이지 않는 현상이 다시 한번 발생하였음. 이전에는 단순한 네트워크 ID의 설정 문제였으나, 이번에는 네트워크 ID는 정상적으로 할당되어 있는 것을 확인하였음. 또한, 룸에 접속한 뒤 정상적인 종료가 아닌 방식으로 클라이언트를 종료하였을 경우 서버 측의 룸 데이터가 이상해지는 것을 확인하였음. 이 부분은 여전히 Disconnect 처리가 제대로 이루어 지지 않는 것으로 보이므로 다시 한번 확인할 필요성을 느낌.  **이도영:** 굳이 x y z 축 밖에 없는 맵에 SAT를 활용하여 슬라이딩 벡터를 계산할 필요성이 있는가에 대해 생각해 봄. 간단하게 충돌하였을 때 자연스러운 움직임을 원하는 것일 뿐이라면 면에 따라서의 대한 반응만 정하면 됨.  AABB와 SAT를 접목시켜서 같이 충돌처리와 함께 충돌 이후의 움직임까지 시행하게끔 하였음. 맵에 대한 부드러운 움직임을 시행하려 해봄 가장 부드러운 움직임이 되어 이를 채택하여 충돌처리를 완료함.  아이템을 주는 박스 관련 모든 데이터 뽑아 메모장에 저장 이와 충돌처리를 활용하여 게임로직에 적용할 예정 | | |
| **7/8**  **토** | **김혁동:** 플레이어가 움직이지 않는 현상에 대해 서버 측에서 움직이지 않는 플레이어들이 INGAME 상태로 변경되지 않은 것을 확인하였음. 위의 문제를 해결함.  플레이어들의 스테이트가 변경되지 않았던 이유는 게임 스타트가 된 이후 유저의 스테이트를 변경할 때에 모든 룸 안의 유저의 데이터를 루프를 돌며 하나씩 변경해 주어야 하는데, 이 부분에서 룸 안의 유저의 id를 받아 놓은 뒤 해당 id의 유저가 아닌 게임 스타트 패킷을 보내온 유저의 데이터만 계속해서 변경하고 있었음.  해당 부분을 제대로 룸 안의 유저 id를 바탕으로 변경하도록 고쳐 제대로 실행되도록 하였음.    **<player는 패킷을 보낸 클라이언트의 ID임. 해당 부분을 변경하였음.>**  **도둑들 주간회의**   * **충돌처리 거의 완료된 것을 확인** * **깃 머지 작업** * **로비 작업 확인** | | |
| **이번 주 문제점 및 해결방안** |  | | |
| **다음 주 계획** | **이도영: 다음주 계획 2차원 A\* 및 맵 데이터 확인 후 수정**  **정극훈: 게임 UI 배치, 파티클 효과 제작**  **김혁동: 인 게임 로직, 맵 모델 데이터 나누기, 캐릭터 애니메이션 확인** | | |
| **비고** | Github 도둑들 주소: <https://github.com/rmrgns/Thieves.git>  Github 졸업작품 회의록 주소: <https://github.com/rmrgns/gameproject_proceedings.git> | | |