|  |  |  |  |
| --- | --- | --- | --- |
| **[ The Client : desert (의뢰인:사막) ]** | | | |
| 5 주 | 2021. 1. 23 ~ 2021. 1. 29 | 작성자 | 최경훈 |
| 이번주  한 일 | **[0] 공동**  01. 17 회의   1. 내(경훈) 컴퓨터에선 프레임이 200이상 나오는데, 성능이 좀 더 좋은 성주 컴퓨터에선 프레임이 30정도 나와서 무엇이 문제인지 회의를 했다. 정확한 원인은 찾지 못했으나 성주 컴퓨터에 문제가 있다고 판단하고 보류하기로 했다. (다른 컴퓨터, 노트북에선 프레임 잘 나오는 거 확인) 2. 성주가 작업하는 플레이어 회전에 버그가 있어 같이 코드를 보면서 버그를 해결했다. 벡터 외적 순서와 정규화 문제였다. 3. 프레임을 유지하기 위해 최적화를 어떻게 할 것인지 회의했다. 일단 가장 간단한 절두체 컬링부터 해보고 다시 회의하기로 함   **[1] 윤성주**  **주간 목표**   1. **~~DX12 Input과 Timer, Material, Component, Scene 부분 공부~~** 2. **플레이어 idle -> run 애니메이션 블렌딩** 3. **플레이어 idle 구현 (공격 모드일 때의 idle)** 4. **~~플레이어 회전 보간 버그 수정~~**   **진행률: 50%**  **일별 공부 내용**  01. 23 (일)   * 플레이어 회전 보간 버그 수정 중. : 회전을 각도를 넣어서 하도록 했는데, 이를 각도가 아닌 벡터를 사용하도록 바꿨다. 하지만, 벡터를 사용했더니 방향이 시계방향으로만 이동하는 버그가 있었다.   01. 24 (월)   * Input과 Timer 부분 공부 * 플레이어 회전 보간 버그 수정 중. : 벡터의 내적을 이용하여 벡터의 사이각을 구하고 그 각도 만큼 회전하도록 했는데, 방향키를 빠르게 여러번 누르면 각도가 이상해지는 버그가 있었다.      * 플레이어가 방향키를 누르면 m\_xmVecNewRotate에 목표 벡터를 넣고 매 프레임마다 선형보간하여 m\_xmNowRotate를 갱신하도록 했다. 그리고 이 벡터와 이전 벡터와의 사이각을 구해서 up, look, right를 변환하도록 했는데 선형 보간된 만큼 실제 up, look, right 벡터가 변환되지 않아 이러한 문제가 발생한 것으로 보인다.     01. 25 (화)   * 플레이어 회전 보간 버그 수정 중. : look벡터에 직접 선형보간된 값을 대입하였더니 회전은 올바르게 잘되었다. 하지만, right, up 벡터가 look벡터에 맞게 변환되지 않아 플레이어가 일그러지는 버그가 있다. * ->정상 -> 일그러진 메쉬   01. 26 (수)   * Material 부분 공부 * 플레이어 회전 보간 버그 수정 중. : right, up 벡터만 회전한 만큼 회전하도록하면 되는데, up벡터는 회전하지 않아도되어서 up과 look을 외적해서 right를 구하면되겠다고 생각했는데 여전히 플레이어가 일그러지는 버그가 있었다....   01. 27 (목)   * Input과 Timer 부분 공부 * 플레이어 회전 보간 버그 수정 성공! : 외적 순서를 잘못 했었다... 궁극적 문제는 해결! 하지만 아직 180도 회전 하는 것은 문제 있음. 보간이 (0,0,1) -> (0,0,-1)로 될때, z값만 보간되므로 양수에서 음수로 바뀌는 순간 플레이어가 바로 회전하는 문제가 발생하는 듯하다. 벡터를 선형보간이 아닌 구형보간이 되도록하는 등 다른 방법을 생각해야겠다. * 카메라 마우스로 회전 구현 : 카메라 가까워지는 버그 있음   01. 28 (금)   * Component, Scene 부분 공부   01. 29 (토)   * 쉬었다.   **[2] 최경훈**  **개발 내용**   1. 맵 오브젝트를 절두체 컬링하기 위한 작업을 하였다. 아직 정확한 컬링은 안되고 유니티에서 매쉬마다 경계구 정보를 얻어와서 처리해야한다. Mesh Component의 Bounds라는 구조체안에 MaxExtentes정보가 있어서 이를 사용하면 될 듯하다. 2. 코드 효율을 높이기 위해 Component와 Prototype 디자인 패턴에 대해 공부하고 코드를 어느정도 작성했다. 여러 디자인 패턴의 이론도 대강 공부하였고 어떤 패턴을 쓰면 효율적일지 많이 생각했다. 3. Singleton패턴을 이용한 GameManager도 만들어서 자주 사용될 듯한 플레이어, 카메라, 씬 등을 쉽게 접근 가능하도록 하였다. | | |
| 다음주  할 일 | **[0] 공동**   1. 지금 작업중인 일이 끝나면 코드 합치기 2. 설 연휴에 푹 쉬었다가 오기.   **[1] 윤성주**  주간 목표   1. 카메라 방향으로 플레이어의 좌표축 변경 2. 카메라 가까워지는 버그 수정 3. 180 회전 시 보간 안되는 문제 해결 4. 플레이어 idle -> run 애니메이션 블렌딩 5. 플레이어 idle 구현 (공격 모드일 때의 idle)   **[2] 최경훈**  주간 목표   1. 디자인 패턴에 대해 공부하고 코드를 좀 더 효율적으로 작성해 볼 것. 2. 유니티에서 매쉬마다 경계구 정보를 가져와 클라에서 기록할 것. | | |
| 문제점 | **[1] 윤성주**   1. 플레이어 회전 버그를 해결하느라 많은 시간을 썼다. 버그가 오랫동안 해결이 안되서 계속 그 문제를 잡고 있으니 새로운 방법이 생각이 안나고 계속 같은 방법만 생각하고 있었다. 근데, 팀원과 회의할 때 이 문제에 대해 논의하면서 바로 해결했다. 앞으로 오랫동안 생각해도 해결되지 않는 버그가 있다면 꼭 팀원에게 문제점에 대해 논의하는 시간을 가져야겠다.   **[2] 최경훈**   1. 디자인 패턴들의 개념은 어느정도 알겠는데, 이거를 코드에 적용하기가 좀 까다롭다. 너무 구조를 이쁘게 짤려는 생각을 좀 줄여야겠다. | | |