|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **주차** | 3주차 | **기간** | 2020.03.30~ 2020.04.05 | **지도교수** | (서명) |
| 이번주 한일 요약 | 김영완:  공복 게이지 구현, 캐릭터 이동 물리 적용  박건호 :  공룡 노말맵핑 완료, 맵툴제작, 애니메이션 오류수정  박정만:  캐릭터의 커브 및 드리프트의 로직 구현 | | | | |

<상세 수행내용>

김영완:

공복 게이지를 위한 ui 셰이더 추가, 이 셰이더에선 플레이어 객체의 guage값과 비교할 m\_guage를 변수로 갖고 있음.

m\_guage는 매 프레임의 update때 player의 게이지와 비교해서 더 작을 경우 1씩 증가하도록 하고 m\_guage를 셰이더에 보내서 셰이더에서는 그 길이값 만큼 노란색 막대를 그리도록 하였다. 플레이어가 고기를 먹을 시 게이지는 10~15가 상승하기 때문에 대략 10~15프레임 동안 셰이더의 변수 m\_gauge는 1씩 올라가면서 매번 셰이더에 그 값을 넘기기 때문에

게이지가 올라가는 애니메이션을 표현할 수 있게 된다.

캐릭터 이동에 관련 물리 구현(직선 운동)

ㄴ-> 저항은 공기저항, 회전 저항을 적용하도록 하였음.

플레이어가 up혹은 down키를 입력할 경우 플레이어에게 일정량의 힘(m\_fForce)이 들어가게 된다.

키를 누르고 있으면 m\_fForce는 계속해서 증가하게 됨. 매 update에서 Look벡터에 m\_fForce를 스칼라 곱하여 미는 힘을 구하고 현재 적용되고 있는 velocity의 반대방향과 총저항의 스칼라 곱을한 결과를 더하여 최종 적용될 힘 벡터를 구함. 그 힘 벡터에 질량을 나눔으로써 가속력을 구하고 velocity에 가속력과 시간변화량을 곱한 결과를 더해줌으로써 최종 속력을 구하여 이동시키도록 하였다.

플레이어로부터 아무 입력이 없을 경우에는 m\_fForce값에 0을 대입시켜 저항만 적용되어 조금씩 느려지게 하도록함으로써 물리적인 동작이 되도록 하였다.

노말 맵핑은 적용해봤으나, 적용안했을 때와 차이가 별로 안느껴져 적용할 지 안할 지 고민중.

박건호 :

캐릭터의 볼륨을 주기위해 노말맵핑을 만들기 위해 블렌더를 이용해서 하이폴리곤 모델을 새로만들어서 노말맵핑을 만들었다.

만들어진 맵에 오브젝트를 배치하기 위해 클라이언트에게 받은 맵툴을 이용해서 코스 주변에 나무 오브젝트를 배치하여 인게임맵을 제작하였다.

클라이언트에게 받은 애니메이션툴을 이용해 애니메이션을 계속 실행하면서 나오는 문제점들을 틈틈히 수정한다.박정만:

캐릭터의 커브하는 과정에서 적용되는 물리법칙을 생각하고 제안, 캐릭터가 드리프트 할때 적용되는 물리법칙을 생각하고 제안, 마찰력의 적용 방향에 대하여 회의

|  |  |  |  |
| --- | --- | --- | --- |
| **문제점 정리** |  | | |
| **해결방안** |  | | |
| **다음주차** | 4주차 | **다음기간** | 2020.04.06 ~ 2020.04.12 |
| **다음주 할일** | 김영완:  맵에 있는 오브젝트들과 캐릭터의 충돌 처리,  드리프트 구현  박건호:  애니메이션 틈틈히 오류들 수정 및 울타리 배치박정만:  드리프트에 대하여 마저 회의하고 적용 서버 프레임워크 완성도 높이높 | | |
| **지도 교수**  **Comment** |  | | |