|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **주차** | 여름방학 2주차 | **기간** | 2020.07.13 ~ 020.07.19 | **지도교수** | (서명) |
| 이번주 한일 요약 | 로딩 화면 추가  모션블러 기능 추가  패킷 처리 함수 추가 | | | | |

<상세 수행내용>

**-로딩 화면 추가-**

**게임씬으로 넘어가면서 리소스를 제작하는데 시간이 많이 걸려서**

**진행도를 보여주기 위해서 로딩씬을 하나 추가하였음**

**리소스 생성 진행량에 맞춰서 게이지가 증가하고**

**게이지가 다 찻을 경우 게임씬으로 넘어가게 된다.**

**-모션블러 기능 추가-**

**기존에 속도감을 나타내기위해서 블러링을 적용하였으나**

**흐림처리 픽셀 수가 많다보니 프레임 드랍에 문제가 심각하였음.**

**ㄴ> 하나의 픽셀에 256x256의 계산이 적용됐음.**

**그래서 모션블러를 추가하는 방법으로 해결하였다. 이전프레임 위치에서 현재프레임 위치로의**

**방향을 저장할 속도맵을 만들고 속도맵의 내용을 채우기위한 랜더링 패스를 하나 더**

**늘렸다. 속도맵이 완성이 되고나면 후처리 랜더링패스에서 그려진 랜더타겟의 각**

**픽셀이 속도맵에 저장된 방향으로 이동하면서 픽셀색상들을 결합하여 최종 색을 표현한다.**

**속도맵은 랜더타겟과 같은 픽셀 위치에 방향값들이 저장된다.**

**-패킷처리 함수 추가-**

**각 씬에서 서버로부터 패킷을 받으면 처리를 담당할 함수 추가.**

**아직 서버와 연동이 되지않아 완성되진않았음.**

-EventHandler 추가-

게임 씬내의 모든 이벤트를 처리하기위한 목적으로 만듣 클래스.

게임 내에서 이벤트 발생시(아이템 생성, 삭제, 오브젝트 비활성화 등)

이벤트에 대한 정보를 핸들러에게 넘겨주면 핸들러는 정보들을 큐에 담아두고

매 업데이트마다 큐에 쌓아둔 정보들을 명령대로 처리한다.

핸들러는 싱글톤으로 만들어져 있어서 게임 실행시 단 하나 만들어짐.

|  |  |  |  |
| --- | --- | --- | --- |
| **문제점 정리** | - | | |
| **해결방안** |  | | |
| **다음주차** | 여름방학 3주차 | **다음기간** | 2020.07.20  ~ 2020.07.26 |
| **다음주 할일** | 렌더링 최적화  게임씬& 서버 연동작업 | | |
| **지도 교수**  **Comment** |  | | |