**네트워크 게임 프로그래밍**

**Term Project**

**2015182004 김하윤**

**2016180011 김연지**

|  |  |
| --- | --- |
| **김하윤** | **김연지** |

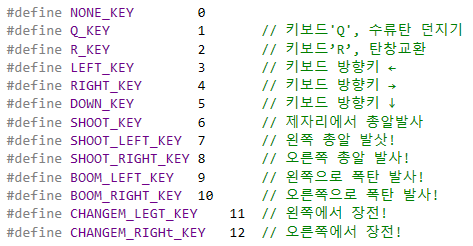
1. **개발일정**

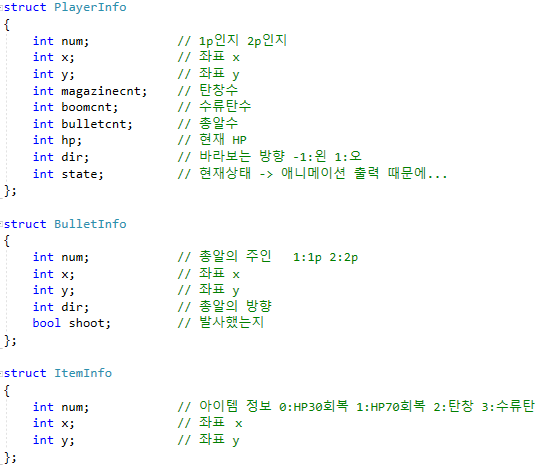
|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | |  | |  | |  | |
| 클라이언트 및 서버 프로토콜 디자인 및 구현  O | 기본적인  네트워크  환경 구현  O | Class Timer  구현  O | 게임 UI,  플레이어,  아이템,  맵 디자인  O | ~~Timer 스레드 함수 구현~~  X(스레드 안만듬) | 게임 UI,  플레이어,  아이템,  맵 디자인  O | Class Scene,  Class  SceneManager  구현  O | 게임 UI,  플레이어,  아이템,  맵 디자인  O |
| 기획서 내 기획한 프로토콜 구현 | 네트워크 환경 구현  (클라이언트가 서버 접속확인) | 시험 | 시험 | Class CTimer 서버 및 클라구현 | 디자인 | Scene, SceneManager,  SceneScript  구현 | 시험 |
|  | |  | |  | |  | |
| Class Input  구현  O |  | Class Object  구현  O | Recv, Send  KeyData ()  구현  O | Class Object  구현  O |  | Class Player  구현  O | Timer  UI띄우기 |
| Input (클라)구현 | 디자인 | Object 구현  Scened에서 Object 동적할당것들은 관리하게끔 구현 | Recv, Send  KeyData() 구현  그에맞는 프토콜 새로 추가 | 휴가 ^0^…  열심히 달렸음.. | SceneState  Send, recv 함수 구현 및 확인 | CPlayer 구현 | Player 리소스때문에 Timer보류 |
|  | |  | |  | |  | |
| Class Player  구현  O |  | Player  보완  O | RecvData()  구현  O | Player, Key  동기화 확인  O | Class Item  구현  (HP)  X | Player, Key  동기화 확인  O | Class Item  구현  (탄창)  X |
| Player 관련  Send, Recv함수 구현 |  | Player 애니메이션이 끝까지 안도는 것 해결 | Player관련 send, recv 함수를 DATA send, recv로 수정 및 보완 | Player Key입력으로 좌우아래 키입력시 동기화 되는 것을 확인함 | Player 리소스가 애니메이션구현과 적합 하지않아 새로 수정 | 총알과 수류탄 프로토콜 수정 및 보완 | G-star 견학 |

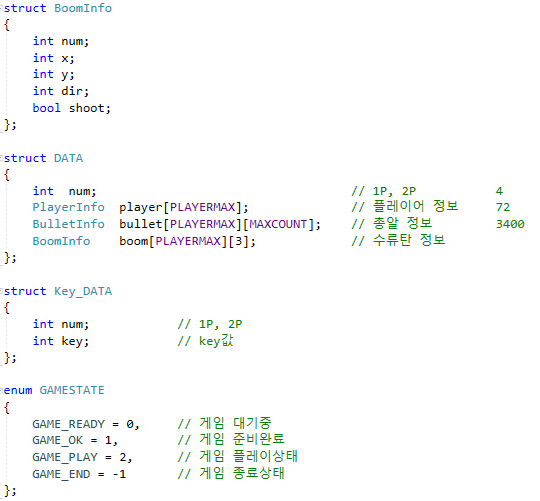
|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | |  | |  | |  | |
| Bullet 구현  O(서버에서 키 입력을 보내주면 총알 쏘는 것까지 했음)  (보류)1. 움직일 때 총알 쏘는 것 안됨 -> 움직일 때는 총알 쏘는 것 안되게 구성 | | | | | | | |
|  | Bullet을 클라이언트에서 구현 | 클라에서 구현한 bullet을 토대로 서버에서 Bullet구현 |  | Bullet의 데이터가 서버에서 클라이언트로 제대로 보내는지 확인 및 클라이언트가 Bullet정보를 받아 그 위치에 잘 그려지는지 확인,  Bullet 동기화 확인,  PlayerInfo내 데이터 동기화 확인 | |  |  |
|  | |  | |  | |  | |
| 총알과  플레이어  충돌 구현 | Class Item  구현  (수류탄)  O | 총알과  플레이어  충돌 구현 | Class Item  구현  (수류탄)  O | 총알과  플레이어  충돌 구현 | Class Item  구현  (수류탄)  O(클라) |  |  |
| Collision이라는 함수를 만들어 bool값으로 충돌이 됐는지 안됐는지 체크 |  | 만약 총알이 맵 밖으로 나가면 그 총알 데이터를 초기화 시킨다.(움직이고 그려지지 않게) |  |  |  |  |  |
|  | |  | |  | |  | |
| 아이템과  플레이어  충돌 구현 | Item관련  Send, Recv  함수 구현 |  | HP UI 구현  O |  | 수류탄 UI 구현 |  |  |
|  | |  | |  | |  | |
|  | 탄창 수 UI  구현 | 충돌 오류  체크 | 총알 수 UI  구현 | 탄창 수  동기화 확인  O |  | 총알 수  동기화 확인  O |  |
|  | |  | |  | |  | |
|  |  | 게임결과 UI  구현 |  | 강제종료  여부  체크 함수 | 게임 결과  send, recv  함수 구현 |  |  |
|  | |  | |  | |  | |
| 로비구현 | 로비구현 | 로비구현 | 로비구현 | 로비구현 | 로비구현 |  |  |

수정

1. 프로토콜





|  |  |  |
| --- | --- | --- |
| 11/5 | 김하윤 | 김연지 |
| 개발 목표 | 클라이언트 및 서버 프로토콜 디자인 및 구현 | 기본적인 네트워크 환경 구현 |
| 결과 | 기획한 프로토콜 구현 | 클라이언트가 서버에 접속하는 지 확인 |

|  |  |  |
| --- | --- | --- |
| 11/6 | 김하윤 | 김연지 |
| 개발 목표 | Class Timer 구현 | 게임 UI, 플레이어, 아이템, 맵 디자인 및 리소스 편집 |
| 결과 | 시험 | 시험 |

|  |  |  |
| --- | --- | --- |
| 11/7 | 김하윤 | 김연지 |
| 개발 목표 | ~~Timer스레드 함수 구현~~ | 게임 UI, 플레이어, 아이템, 맵 디자인 및 리소스 편집 |
| 결과 | Class CTimer 서버 및 클라이언트에 구현 | 리소스 편집 |

|  |  |  |
| --- | --- | --- |
| 11/8 | 김하윤 | 김연지 |
| 개발 목표 | Class Scene, SceneManager 구현 | 게임 UI, 플레이어, 아이템, 맵 디자인 및 리소스 편집 |
| 결과 | Class Scene, SceneManager, SceneScript 구현 | 시험 |

|  |  |  |
| --- | --- | --- |
| 11/9 | 김하윤 | 김연지 |
| 개발 목표 | Class Input 구현 |  |
| 결과 | Class Input(클라이언트)구현 | 리소스 편집 |

|  |  |  |
| --- | --- | --- |
| 11/10 | 김하윤 | 김연지 |
| 개발 목표 | Class Object 구현 | Recv, Send KeyData 함수 구현 |
| 결과 | Class Object 구현,  Class Scene에서 Object들을 관리하게 끔 수정  클라에서 SendKeyData() 함수 구현 | KeyData를 주고 받기위한 관련 프로토콜 수정 및 추가  void RecvKeyAndDataUpdate(); 함수 구현(서버) |

|  |  |  |
| --- | --- | --- |
| 11/11 | 김하윤 | 김연지 |
| 개발 목표 | Class Object 구현 | Recv, Send KeyData 함수 구현 |
| 결과 | 휴식 | GameState  Send, recv 함수 구현 및 확인 |

|  |  |  |
| --- | --- | --- |
| 11/12 | 김하윤 | 김연지 |
| 개발 목표 | Class Player구현 | 게임 제한시간 UI 구현 |
| 결과 | Class Player 구현, Animation 기능 구현  키 입력 데이터 변경이 되는지 확인 및 수정  (send, recv KeyData() 함수 수정) | Player 리소스 수정 |
| 11/13 | 김하윤 | 김연지 |
| 개발 목표 | Send, recv PlayerData() 함수 구현 |  |
| 결과 | Send, recv PlayerData() 함수 구현 |  |

|  |  |  |
| --- | --- | --- |
| 11/14 | 김하윤 | 김연지 |
| 개발 목표 | Class Player보완 | Recv Data() 구현 |
| 결과 | 총을 쏘는 모션을 할 때 애니메이션이 끝까지 실행이 안되는 버그를 발견하여 해결 | Send, recv PlayerData를 총알과 플레이어 데이터를 다 들고 있는 DATA구조체의 send, recv로 수정 |

|  |  |  |
| --- | --- | --- |
| 11/15 | 김하윤 | 김연지 |
| 개발 목표 | Player, Key 동기화 확인 | Class Item(HP) 구현 |
| 결과 | Class Network 구현으로 Network::Input()에서는 KeyData를 send하고 Network::Update()에서는 GameState와 Data를 recv함  PlayerData프로토콜에 state(애니메이션 상태)를 추가 | Player 리소스가 수정된 애니메이션과 맞지 않아 리소스 수정 |

|  |  |  |
| --- | --- | --- |
| 11/16 | 김하윤 | 김연지 |
| 개발 목표 | Player, Key 동기화 확인 | Class Item(탄창) 구현 |
| 결과 | 총알과 수류탄 프로토콜 수정 및 보완 | G-star 견학 |

|  |  |  |
| --- | --- | --- |
| 11/17 | 김하윤 | 김연지 |
| 개발 목표 | Bullet 구현 | Bullet 구현 |
| 결과 |  | 클라이언트에서 Bullet 구현 |

|  |  |  |
| --- | --- | --- |
| 11/18 | 김하윤 | 김연지 |
| 개발 목표 | Bullet 구현 | Bullet 구현 |
| 결과 | 클라에서 구현된 Bullet을 토대로 서버에서 구현 | Class Item 구현 |

|  |  |  |
| --- | --- | --- |
| 11/19 | 김하윤 | 김연지 |
| 개발 목표 | Bullet 구현 | Bullet 구현 |
| 결과 | Bullet의 데이터가 서버에서 클라이언트로 제대로 보내는지 확인 및 클라이언트가 Bullet정보를 받아 그 위치에 잘 그려지는지 확인  PlayerInfo내 데이터 동기화 확인 | |

|  |  |  |
| --- | --- | --- |
| 11/20 | 김하윤 | 김연지 |
| 개발 목표 |  |  |
| 결과 | 서버와 클라간의 통신을 초당 30번으로 수정  (서버) Class Timer 사용 X | 인공지능 시험 준비 |

|  |  |  |
| --- | --- | --- |
| 11/21 | 김하윤 | 김연지 |
| 개발 목표 | 총알과 플레이어 충돌 구현 | 수류탄 구현 |
| 결과 | Collision이라는 함수를 만들어 bool값으로 충돌이 됐는지 안됐는지 체크 | Class Item 구현 및 드랍 시키기 |

|  |  |  |
| --- | --- | --- |
| 11/22 | 김하윤 | 김연지 |
| 개발 목표 | 총알과 플레이어 충돌 구현 | Class Item 구현 (수류탄) |
| 결과 | Collision이라는 함수를 만들어 bool값으로 충돌이 됐는지 안됐는지 체크 | Class Item 기반으로 Class Healing 구현 |

|  |  |  |
| --- | --- | --- |
| 11/23 | 김하윤 | 김연지 |
| 개발 목표 | ~~총알과 플레이어 충돌 구현~~  (서버)스레드를 방 하나당 하나씩 생성하는 것으로 변경 | Class Item 구현 (수류탄) |
| 결과 | g\_socket[]로 접속한 클라이언트 소켓을 관리하여 임계영역으로 g\_socket[]에 소켓정보를 담을 때 마다 동기화를 시킴 | 아이템 drop을 위한 Class Heli 구조와 아이템 간의 flow 및 세부내용 재조정 |

|  |  |  |
| --- | --- | --- |
| 11/24 | 김하윤 | 김연지 |
| 개발 목표 | 제한시간 Timer UI 띄우기 |  |
| 결과 |  |  |

|  |  |  |
| --- | --- | --- |
| 11/25 | 김하윤 | 김연지 |
| 개발 목표 | 아이템과 플레이어 충돌 | Item 관련 Send, Recv 함수 구현 |
| 결과 |  | 아픔 |

|  |  |  |
| --- | --- | --- |
| 11/26 | 김하윤 | 김연지 |
| 개발 목표 |  | HP UI 구현 |
| 결과 | 수류탄 구현 및 데이터 송수신 | 다른 수업 과제 |

|  |  |  |
| --- | --- | --- |
| 11/27 | 김하윤 | 김연지 |
| 개발 목표 |  | 수류탄 UI 구현 |
| 결과 | 수류탄 버그 확인(수정 X) | 다른 수업 과제 마무리 |

|  |  |  |
| --- | --- | --- |
| 11/28 | 김하윤 | 김연지 |
| 개발 목표 |  |  |
| 결과 |  |  |

|  |  |  |
| --- | --- | --- |
| 11/29 | 김하윤 | 김연지 |
| 개발 목표 |  | 탄창 수 UI 구현 |
| 결과 |  | 서버로 item drop try 1  서버에 기존 timer 외의 아이템 드랍 시간 (30초 간격)을 세는 timer 추가하여 bool drop을 한 번만 전송하여 클라이언트에서 이를 받는다. 근데 송수신이 제대로 안 됨. |

|  |  |  |
| --- | --- | --- |
| 11/30 | 김하윤 | 김연지 |
| 개발 목표 | 충돌 오류 체크 | 총알 수 UI 구현 |
| 결과 | 플레이어와 총알간 충돌 오류는 없음 | 서버로 Heli 띄우기 try 2  서버에서 보내는 timer에 따라서 bool drop을 전송 시켜서 이를 클라이언트에서 받는다. 근데 송수신이 제대로 안 됨. |

|  |  |  |
| --- | --- | --- |
| 12/1 | 김하윤 | 김연지 |
| 개발 목표 | 탄창 수 동기화 확인 |  |
| 결과 | (UI미구현)탄창 수 동기화 확인 | Class Heli 구현.  일단 헬리콥터를 띄워서 헬리콥터가 중앙에 도착하면 아이템을 드롭하는 일련의 과정이 진행되게 만들었다. |

|  |  |  |
| --- | --- | --- |
| 12/2 | 김하윤 | 김연지 |
| 개발 목표 | 총알 수 동기화 확인 |  |
| 결과 | (UI미구현)총알 수 동기화 확인 |  |

|  |  |  |
| --- | --- | --- |
| 12/3 | 김하윤 | 김연지 |
| 개발 목표 |  |  |
| 결과 | 게임 승패 결과 UI 출력  승패 결과 send, recv함수 구현  <승패여부 조건>   * + - 1. 제한시간 전에 둘 중 hp가 0이되는 경우 -> hp가 0인 플레이어가 패       2. 제한시간이 끝난 후 -> 둘 중 hp가 높은 경우 승 | 헬리콥터&아이템 동작 클라이언트에서 구현해 놓은 것을 서버에서 보내준 Timer 에 맞춰 지정한 시간에 클라에서 띄우고, 아이템 드랍 까지 연속 과정 잘 된다. |