**일렉트릭 블루, 라인, 디자인이(가) 표시된 사진

자동 생성된 설명**

**네트워크 게임프로그래밍  
프로젝트 진행 보고서**

**2017114025 성현석  
2019182003 권순원  
2019182024 엄장헌**

엄장헌

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 일 | 월 | 화 | 수 | 목 | 금 | 토 |
| 10/29 | 30 | 30 | 11/1 | 2 | 3 | 4 |
|  |  |  |  |  | 시험 | LockQueue |
| 5 | 6 | 7 | 8 | 9 | 10 | 11 |
| Client  PacketManager  (연결부분만) | Client-Server  Initialize() | Client-Server  Send(), Recv() | Client-Server  Loop 구조 | Client  PacketManager |  | 졸작 회의 |
| 12 | 13 | 14 | 15 | 16 | 17 | 18 |
| ~~Client~~  ~~움직임 방식 변경~~ | 시험 | 시험 | 시험 | ~~Client~~  ~~타 플레이어 관리 추가~~ |  | 졸작 회의 |
| 19 | 20 | 21 | 22 | 23 | 24 | 25 |
| Client  움직임 방식 변경 | Client  타 플레이어 관리 추가 | Client  Stage 구조 변경 | InGame  Collision Manager | 플레이 가능 브렌치 생성 | 알파 테스트 | 졸작 회의 |
| 26 | 27 | 28 | 29 | 30 | 12/1 | 2 |
|  | 버그 픽스 |  |  |  | 베타 테스트 | 졸작 회의 |
| 3 | 4 | 5 | 6 | 7 | 8 | 9 |
|  | 문서 정리 | 코드 정리 |  | 최종 점검 |  |  |

1주차 (11/4 ~ 11/9) 100%:   
LockQueue  
서버에서 공용으로 사용하는 큐를 동기화하기 위한 LockQueue class 작성 완료  
PacketManager  
클라이언트 패킷 매니저의 초기화와 send, recv 구현 완료  
키 입력시 PacketManager  
서버에서 클라이언트와 패킷을 주고 받을 쓰레드 함수 초기화 및 send, recv 부분 완료  
쓰레드 메인 루프에서 너무 긴 시간 점유하지 않도록 타임아웃 옵션 설정 완료

변경사항:  
기존 계획에서는 난이도 수신을 wait server에서 받고 Client-server 스레드에 전달할 계획 이였는데, Client-server 스레드에서 초기화 작업을 하는 과정에서 난이도를 전달 받는것으로 수정

2주차 (11/11 ~ 11/16):

3주차 (11/18 ~ 11/23):

다른 플레이어 랜더링 구현.

플레이어가 상하좌우 모든 방향으로 움직일 수 있도록 수정

PacketManager를 통해 서버와 정보를 주고받는 부분 구체화

다른 플레이어의 정보 갱신

4주차 (11/25 ~ 11/30)

성현석

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 일 | 월 | 화 | 수 | 목 | 금 | 토 |
| 10/29 | 30 | 30 | 11/1 | 2 | 3 | 4 |
|  |  |  |  |  |  | InGame  Initialize() |
| 5 | 6 | 7 | 8 | 9 | 10 | 11 |
| 졸작 회의 | 시험 | 시험 | InGame  Loop 구조  Lobby  GameStart() | InGame  PushPaket() |  | InGame  Physics Class 테스트 환경 제작 |
| 12 | 13 | 14 | 15 | 16 | 17 | 18 |
| 졸작 회의 | InGame  Physics Class 멤버 함수 | InGame  Physics Class 멤버 함수 | InGame  Collision Manager | InGame  Collision Manager |  | InGame  Update() |
| 19 | 20 | 21 | 22 | 23 | 24 | 25 |
| 졸작 회의 | InGame  Collision Manager | 시험 | 시험 | 플레이 가능 브렌치 생성 | 알파 테스트 |  |
| 26 | 27 | 28 | 29 | 30 | 12/1 | 2 |
| 졸작 회의 | 버그 픽스 |  |  |  | 베타 테스트 |  |
| 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| 졸작 회의 | 문서 정리 | 코드 정리 |  | 최종 점검 |  |  |

1주차 (11/4 ~ 11/9) 100%:   
InGameThread 초기화와 각 클라이언트의 EventQueue의 주소를 담을 새로운 구조체 EventQueues 구현  
InGameThread Loop 구조 구현  
EventQueue의 내용 읽기(pop)를 위해 ToSeverQueueCheck() 구현  
EventQueue에 내용을 보내기(push)를 위한 PushPacket() 구현

변경사항  
InGameThread애서 ClientInfoQueue의 데이터를 읽는게(pop) 아닌 LobbyThread에서 읽는 것(pop)으로 변경

2주차 (11/11 ~ 11/16):

InGamePhysics 클래스 구조 구현

CollisionManager 클래스 구조 구현

추가로 필요한 PlayerInfo 구조체 생성

Physics 플레이어끼리의 충돌은 운동량 보존 법칙이용

벽이나 플레이어간의 충돌 여부를 체크하기위한 InGamePhysics 클래스의 멤버함수 구현

충돌후 가속도를 구현하기 위한 물리법칙을 적용한 CollisionManager 클래스의 멤버함수 구현

<변경 사항>

시험 일정 변경으로 모든 일정이 2일 당겨짐

<변경할 점>

오류를 방지하기 위해 충돌 시 오차 구현

운동량 보존 법칙에서 작용 반작용 법칙을 적용

3주차 (11/18 ~ 11/23)

InGamePhysics 클래스 맴버 함수를 운동량 보존 법칙에서 작용 반작용 법칙으로 변경

오류를 방지하기 위해 CollisionManager 클래스 보완

Client move() 추가 구현

클라이언트의 강제종료 여부 판단, 패킷을 상황에 맞게 전송등 InGameThread 구조 개선

4주차 (11/25 ~ 11/30)

권순원

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 일 | 월 | 화 | 수 | 목 | 금 | 토 |
| 10/29 | 30 | 30 | 11/1 | 2 | 3 | 4 |
|  |  |  |  | 시험 |  | wait server  Initialize() |
| 5 | 6 | 7 | 8 | 9 | 10 | 11 |
| wait server  AcceptClient() | Lobby  Initialize() | Lobby  Loop 구조 | Lobby  GameStart() | Client  추가 플레이어 모델 로드 |  | 졸작 회의 |
| 12 | 13 | 14 | 15 | 16 | 17 | 18 |
| ~~Client~~  ~~움직임 방식 변경~~ | 시험 | 시험 | 시험 | ~~Client~~  ~~타 플레이어 랜더링~~ |  | 졸작 회의 |
| 19 | 20 | 21 | 22 | 23 | 24 | 25 |
| Client  움직임 방식 변경 | Client  타 플레이어 랜더링 | Client  장애물 패턴 추가 | Lobby  테스트 및 버그픽스 | 플레이 가능 브렌치 생성 | 알파 테스트 | 졸작 회의 |
| 26 | 27 | 28 | 29 | 30 | 12/1 | 2 |
|  |  | 버그 수정 |  |  | 베타 테스트 | 졸작 회의 |
| 3 | 4 | 5 | 6 | 7 | 8 | 9 |
|  | 문서 정리 | 코드 정리 |  | 최종 점검 |  |  |

1주차 (11/4 ~ 11/9):   
MainThread  
서버가 시작되고 LobbyThread 호출  
대기소켓 생성 후 Client 요청이 있을 시 accept하고 client-server thraed 호출

LoobyThread  
MainThread에서 생성 요청을 받아 생성  
ClinetInfoQueue에서 인원을 체크하고 InGame thread를 호출

변경 사항  
원래 계획은 InGame thread를 초기화 하면서 ClientInfoQueue에서 WaitPop을 통해 플레이어와 통신할 수 있는 큐를 가져오는 것이었는데, thread로 동작하기 때문에 초기화를 다 하기 전에 접근해 InGame thread가 두 번 호출되는 문제가 발생해 Lobby thread에서 WaitPop을 해서 전달해주는 것으로 변경  
1주차 계획 내용 전부 구현.

2주차 (11/11 ~ 11/16):

3주차 (11/18 ~ 11/23):

다른 플레이어 랜더링 구현.

플레이어가 상하좌우 모든 방향으로 움직일 수 있도록 수정

장애물 생성 방식 변경 및 장애물 패턴 추가

4주차 (11/25 ~ 11/30)