

네프 프로토콜 구상 – 클라이언트쪽 [2017-11-13-월~... 2017-12-10]

모두 클라이언트쪽을 의미함.

클라이언트 -> 서버로 메시지 전달.

C_ClientThread [로그인전 제어] : 로그인, 회원가입, (강제)접속종료

- 로그인

GUI_LoginJFrame에서 LoginOK_jButtonActionPerformed() 수행시,

- c_ClientThread.c_Login.Login(name, pwd)
: [LOGIN]/id/pwd 전송

- 회원가입

GUI_CreateAccJFrame에서 CreateAccOK_jButtonActionPerformed() 수행시,

- c_ClientThread.c_Login.Join(newName, newPwd)
: [JOIN]/id/pwd 전송

- (강제)접속종료

GUI_LoginJFrame에서 Back_jButtonActionPerformed() 수행시,

- c_ClientThread.quit()
: [QUIT] 전송

C_WaitingRoom [대기실 제어] : 이름, 방[대기실 등] 입장, 방 생성, 채팅[메시지], (강제)접속종료

- 클라이언트 이름

C_WaitingRoom.controlStart()시작하자마자, [NAME]/userName 전송

- 방[대기실 등] 입장

GUI_JoinRoomJFrame에서 JoinRoomOK_jButtonActionPerformed() 수행시,

- c_WaitingRoom.JoinRoom(roomName, roomPwd)
[JOIN_ROOM]/roomNumer/roomPwd 전송

- 게임방 생성

GUI_CreateRoomJFrame에서 CreateRoomOK_jButtonActionPerformed() 수행시,

- c_WaitingRoom.createRoom(roomName, roomPwd)
: [CREATE_ROOM]/roomName/roomPwd/userName 전송

- 채팅[메시지]

GUI_WaitingJFrame에서 inputChat_jTextFieldActionPerformed() 수행시,

- c_WaitingRoom.sendTalk()

: [MSG]/message 전송

- (강제)접속종료

GUI_WaitingJFrame에서 Quit_jButtonActionPerformed() 수행시,

- c_ClientThread.quit()

: [QUIT] 전송

C_GameRoom[게임방 제어] : 채팅[메시지], 방 나가기, 레디, 자리(열고 닫기), 강퇴, 게임 시작, (강제)접속종료

- 채팅[메시지]

GUI_GameRoomJFrame에서 inputChat_jTextFieldActionPerformed() 수행시,

- c_GameRoom.sendTalk()

: [MSG]/message 전송

- 방 나가기

GUI_GameRoomJFrame에서 Back_jButtonActionPerformed() 수행시,

- c_GameRoom.exitRoom()

: [EXIT_ROOM]/userName 전송

- 레디

GUI_GameRoomJFrame에서 payer*Ready_jComboBoxActionPerformed() 수행시,

- c_GameRoom.ready()

: [READY]/index/Ready or Not ready 전송

- 자리(열고 닫기) [방장만이 보낼 수 있음]

GUI_GameRoomJFrame에서 payer*_jComboBoxActionPerformed() 수행시,

- c_GameRoom.place()

: [PLACE]/index/Open or Close 전송

- 강퇴 [방장만이 보낼 수 있음]

GUI_GameRoomJFrame에서 payer*_jComboBoxActionPerformed() 수행시,

Close를 했는데 해당 위치에 사람이 있을 경우,

- c_GameRoom.fire()

: [FIRED]/userName 전송도 같이 전송

- 게임 시작 [방장만이 보낼 수 있음]

GUI_GameRoomJFrame에서 GameStart_jButtonActionPerformed() 수행시,

- c_GameRoom.gameStart()
: [GAME_START] 전송

- (강제)접속종료

그냥 에러시 try/catch문에서 [QUIT] 전송

[게임은 서바이벌 모드로 진행. 점수 상관없이 오래 사는 놈이 이김]

C_TetrisRoom [테트리스 시작한 방 제어] : 테트리스 명령어, 채팅, 이김, 게임오버[대기] & 게임 종료, 방 나가기, (강제)접속종료

- 테트리스 명령어

GUI_MultiTetrisJFrame에서, 어디든간에 해당 테트리스 명령어를 눌렀을 때,

- c_TetrisRoom.tetrisCMD(keyCode)
[TETRIS_CMD]/userName/keyCode 전송

~~- 채팅[메시지] 패널 제어문제로 기능 삭제함~~

~~MultiTetrisJFrame에서, inputChat_jTextFieldActionPerformed() 수행시,
-- c_ClientThread.c_WaitingRoom.c_GameRoom.c_TetrisRoom.sendTalk()
-- : [MSG]/message 전송~~

~~- 이김~~

- 게임오버[대기] & 게임 종료

GUI_MultiTetrisJFrame에서, 어디든간에 해당 클라이언트가 게임오버를 당했을 때,

- c_TetrisRoom.gameOver()
[GAME_OVER]/userName 전송

- 방 나가기

GUI_MultiTetrisJFrame에서 Back_jButtonActionPerformed() 수행시,
- c_TetrisRoom.exitRoom()
: [EXIT_ROOM]/userName 전송

서버 -> 클라이언트로 메시지 받음.

C_ClientThread [로그인전 제어] : 로그인, 회원가입, 서버(강제) 종료

- 로그인

C_ClientThread에서 [socket1로 받음]

[LOGIN]/SUCCESS/id : 로그인 성공 및 성공한 계정의 id값을 의미함

GUI_LoginJFrame에서 LoginOK_jButtonActionPerformed() 과정 도중 [socket2로 받음]

[LOGIN]/SUCCESS or FAILURE : 로그인 성공여부를 받음.

[LOGIN]/FAILURE/INVALID_VALUE or ALREADY_CONNECTED : 잘못된 값 혹은 이미 접속 중

- 회원가입

GUI_CreateAccJFrame에서 CreateAccOK_jButtonActionPerformed() 과정 도중 [socket2로 받음]

[JOIN]/SUCCESS or FAILURE : 회원가입 성공여부를 받는다.

[JOIN]/FAILURE/ALREADY_JOIN : 이미 중복된 계정의 아이디

- 서버(강제) 종료

C_ClientThread에서 [socket1로 받음]

[QUIT] : 이상한 이유로 서버가 꺼졌음을 의미함. [서버의 강제 종료 등]

C_WaitingRoom [대기실 제어] : 채팅[메시지], 방[대기실 등] 입장함, 방 생성됨, 접속자 목록, 입장, 퇴장, 연결 끊김, 서버(강제) 종료, 방 가득참 & 방 비밀번호 틀림

C_WaitingRoom에서 [socket1로 받음]

- 채팅[메시지]

[MSG]/message : 메시지를 받음을 의미함.

=> recvTalk(rmsg[1]); 메시지를 화면에 보여줘야 함.

- 방[대기실 등] 입장함

[JOIN_ROOM]/roomNum/roomName/roomOwnerName : 자신이 roomNum의 새로운 방에 접속에 성공했음을 의미함. 방주인 이름도 받음.

=> 클라이언트의 새로운 방 번호를 지정하고,

클라이언트의 방 이름을 지정한 후,

- 이후, 게임방 제어 C_GameRoom.controlStart() 에서 통제함. //방주인 이름도 넘김

- 방 생성됨

[CREATE_ROOM]/roomNum/roomName : 방번호, 방이름의 방이 생성되었음을 의미함.

// 방 목록에 방을 추가 rList.add(new C_Room())

// 방 목록을 갱신한다. roomsInfo();

- 접속자 목록

[PLAYERS]/..... : 대기실의 접속자 userName 목록을 의미함. userName1/tuserName

// pList를 갱신한다.

// 화면상의 userList를 갱신한다.

- 입장

[ENTER]/userName : 다른 클라이언트인 userName가 대기실에 들어왔음을 의미함.

```
// pList를 갱신한다.  
// 화면상의 userList를 갱신한다.  
// 메시지를 화면에 보여준다.
```

- 퇴장

[EXIT]/userName : 다른 클라이언트인 userName가 대기실에서 나갔음을 의미함.

```
// pList를 갱신한다.  
// 화면상의 userList를 갱신한다.  
// 메시지를 화면에 보여준다.
```

- 연결 끊김

[DISCONNECT]/userName : userName의 다른 클라이언트가 연결을 끊었음을 의미함.

```
// pList를 갱신한다.  
// 화면상의 userList를 갱신한다.  
// 메시지를 화면에 보여준다.
```

- 서버(강제) 종료

[QUIT] : 이상한 이유로 서버가 꺼졌음을 의미함. [서버의 강제 종료 등]

```
// 메시지를 화면에 보여준다.  
// 장비를 정지합니다.
```

- 방 가득참 & 방 비밀번호 틀림

GUI_JoinRoomJFrame에서 JoinRoomOK_jButtonActionPerformed() 과정 도중 [socket2로 받음]

[LOGIN]/SUCCESS or FAILURE : 방 입장의 성공여부를 받음.

[LOGIN]/FAILURE/FULL or WRONGPWD : 방이 이미 찼음 혹은 방의 비밀번호가 틀림

C_GameRoom[게임방 제어] : 채팅[메시지], 방 나가기, 레디, 자리(열고 닫기), 강퇴, 게임 시작, 접속자 목록, 입장, 퇴장, 연결 끊김, 방 터짐, 서버(강제) 종료

C_GameRoom에서 [socket1로 받음]

- 채팅[메시지]

[MSG]/message : 메시지를 받음을 의미함.

```
=> recvTalk(rmsg[1]); 메시지를 화면에 보여줘야 함.
```

- 방 나가기

[EXIT_ROOM] : 자신이 방에서 나와서 대기실에서 감을 의미함.

```
=> // 클라이언트의 대기실 방 번호0을 지정하고,  
// 클라이언트의 방 이름[대기실]을 지정한 후,  
// 루프를 빠져나온다. 이후, 게임방 제어 C_WaitingRoom.controlStart() 에서 통제함.
```

- 레디

[READY]/index/Ready or Not ready

=> 해당 index 위치의 값을 Ready or Not ready로 적절하게 바꿔준다.

- 자리(열고 닫기)

[PLACE]/index/Open or Close

=> 해당 index 위치의 값을 Open or Close로 적절하게 바꿔준다.

- 강퇴

[FIRED]/userName

=> 본인이 강퇴되었는지 확인한다. [본인 이름 == userName]

if 본인이 강퇴되었다면,

// 본인이 방장에 의해 강퇴되었다고 알린다.

// 클라이언트의 대기실 방 번호0을 지정하고,

// 클라이언트의 방 이름[대기실]을 지정한 후,

// 본인과 연결된 서버쪽으로 무의미한 메시지를 하나 날린다.

// 루프를 빠져나온다. 이후, 게임방 제어 C_WaitingRoom.controlStart() 에서 통제함.

else 타인이 강퇴되었다면,

// 타인이 방장에 의해 강퇴되었다고 알린다.

- 게임 시작

C_GameRoom에서 [socket1로 받음]

[GAME_START] : 게임이 시작했음을 의미함.

=> 카운트 다운 5초를 세며 보여준 후 넘어간다.

이후는 C_TetrisRoom.controlStart()에서 다시 통제함.

GUI_GameRoomJFrame에서 GameStart_jButtonActionPerformed() 과정 도중 [socket2로 받음]

[GAME_START]/SUCCESS or FAILURE : 로그인 성공여부를 받음.

[GAME_START]/FAILURE/NOT_START_ALONE : 혼자 시작할 수 없음을 의미.

or NOT_READY_ALL : 모든 인원이 레디를 하지 않음

- 접속자 목록

[PLAYERS]/..... : 게임방의 접속자 userName 목록을 의미함. userName1/tuserName

// pList를 갱신한다.

// 화면상의 userList를 갱신한다.

- 입장

[ENTER]/userName : 다른 클라이언트인 userName가 게임방에 들어왔음을 의미함.

// pList를 갱신한다.

// 화면상의 userList를 갱신한다.

// 메시지를 화면에 보여준다.

- 퇴장

[EXIT]/userName : 다른 클라이언트인 userName가 게임방에서 나갔음을 의미함.

// pList를 갱신한다.

// 화면상의 userList를 갱신한다.

// 메시지를 화면에 보여준다.

- 방 터짐

[ROOM_DESTROYED]

// 방장이 나가서 방이 터졌음을 알린다.

// 클라이언트의 대기실 방 번호0을 지정하고,

// 클라이언트의 방 이름[대기실]을 지정한 후,

// 본인과 연결된 서버쪽으로 무의미한 메시지를 하나 날린다.

- 연결 끊김

[DISCONNECT]/userName : userName의 다른 클라이언트가 연결을 끊었음을 의미함.

// pList를 갱신한다.

// 화면상의 userList를 갱신한다.

// 메시지를 화면에 보여준다.

- 서버(강제) 종료

[QUIT] : 이상한 이유로 서버가 꺼졌음을 의미함. [서버의 강제 종료 등]

// 메시지를 화면에 보여준다.

// 장비를 정지합니다.

C_TetrisRoom 테트리스 시작한 방 제어] : 테트리스 명령어, 채팅, 이김, 게임 오버[대기], 게임 종료, 방 나가기, 게임 포기, 퇴장, (강제)접속종료

- 테트리스 명령어

[TETRIS_CMD]/userName/keyCode : 해당 클라이언트의 테트리스 명령어를 의미

=> 해당 userName에 따라 keyCode 내용을 반영한다.

- 채팅[메시지]

[MSG]/message : 메시지를 받음을 의미함.

=> recvTalk(rmsg[1]); 메시지를 화면에 보여줘야 함.

- 이김

- 게임오버[대기]

[GAME_OVER]/userName : 해당 클라이언트가 게임오버 되었음을 의미함.

=> 해당 클라이언트가 게임오버되었다고 표시함. [패널을 바꾸던, 텍스트로 표시하던 둘다던]

게임 종료

[GAME_END]/winnerName : 게임이 종료되었음을 의미함/ 승자의 이름도 받음

=> 해당 클라이언트가 이겼다고 표시함. 누가 이겼다, ~~당신은 N등이다~~ 등을 ~~표시~~ 이걸 보류 이후 대기실로 돌아가야함.

=> 클라이언트의 대기실 방 번호0을 지정하고,

클라이언트의 방 이름[대기실]을 지정한 후,

- 이후, 게임방 제어 C_WaitingRoom.controlStart() 에서 통제함.

[사실상 반복문을 빠져나오면 됨]

- 방 나가기

[EXIT_ROOM] : 자신이 방에서 나와서 대기실에서 감을 의미함.

=> 클라이언트의 대기실 방 번호0을 지정하고,

클라이언트의 방 이름[대기실]을 지정한 후,

- 이후, 게임방 제어 C_WaitingRoom.controlStart() 에서 통제함.

[사실상 반복문을 빠져나오면 됨]

- 게임 포기

[GIVE_UP]/userName : 다른 클라이언트인 userName가 게임을 포기하고 나갔음 의미함.

// 메시지를 화면에 보여준다.

// 해당 클라이언트를 게임오버 처리해주고 멈춰줘야함..

- 퇴장

[EXIT]/userName : 다른 클라이언트인 userName가 게임방에서 나갔음을 의미함.

// 메시지를 화면에 보여준다.

- (강제)접속종료

[QUIT] : 이상한 이유로 서버가 꺼졌음을 의미함. [서버의 강제 종료 등]

// 메시지를 화면에 보여준다.

// 장비를 정지합니다.