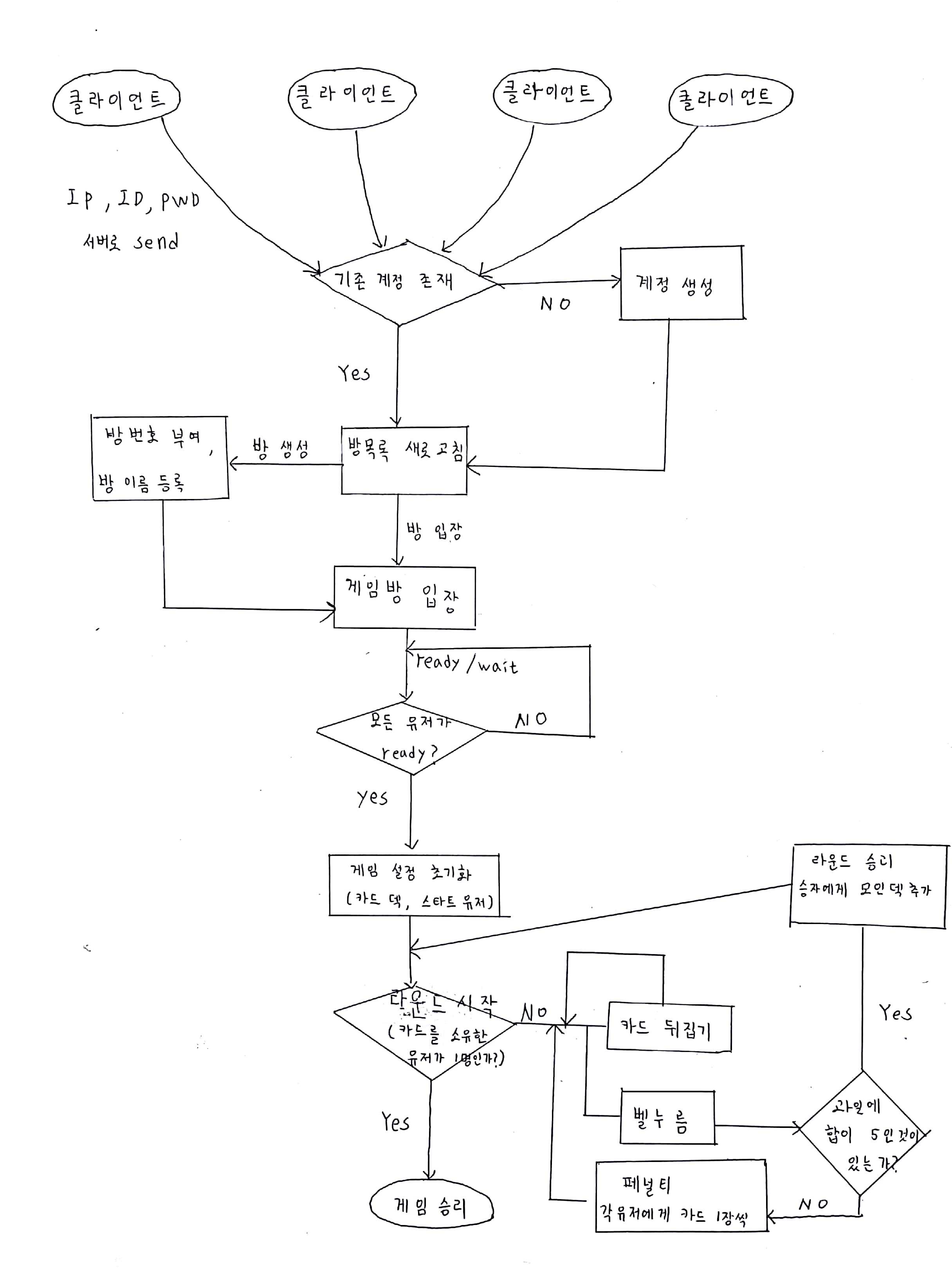
**네트워크 프로그래밍 (B)**

**<팀프로젝트>**

**1496091 이재하**

**1491012 박성영**

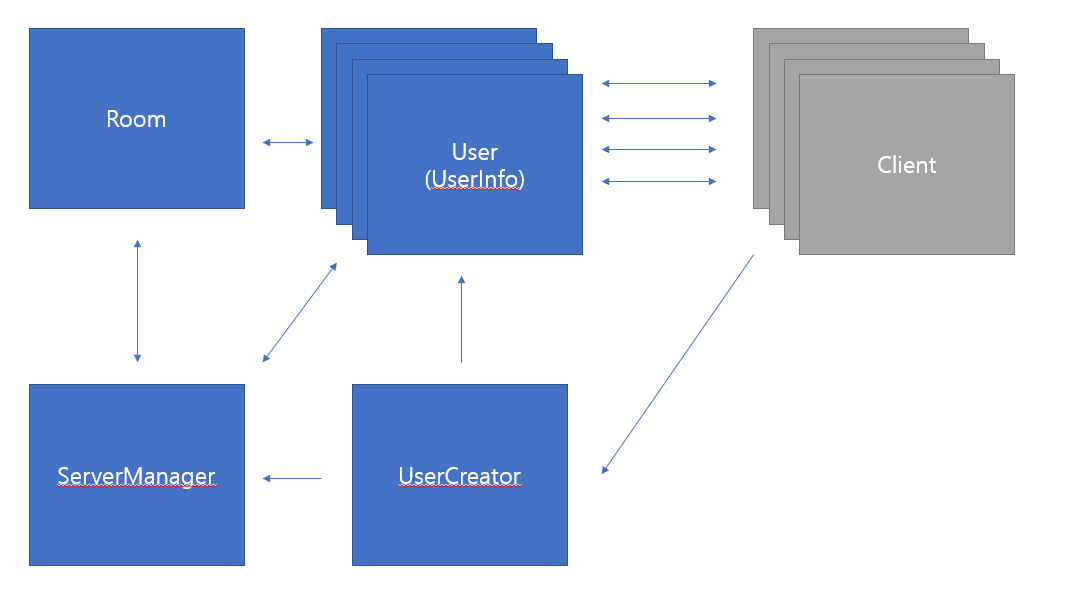
**플로우 차트**



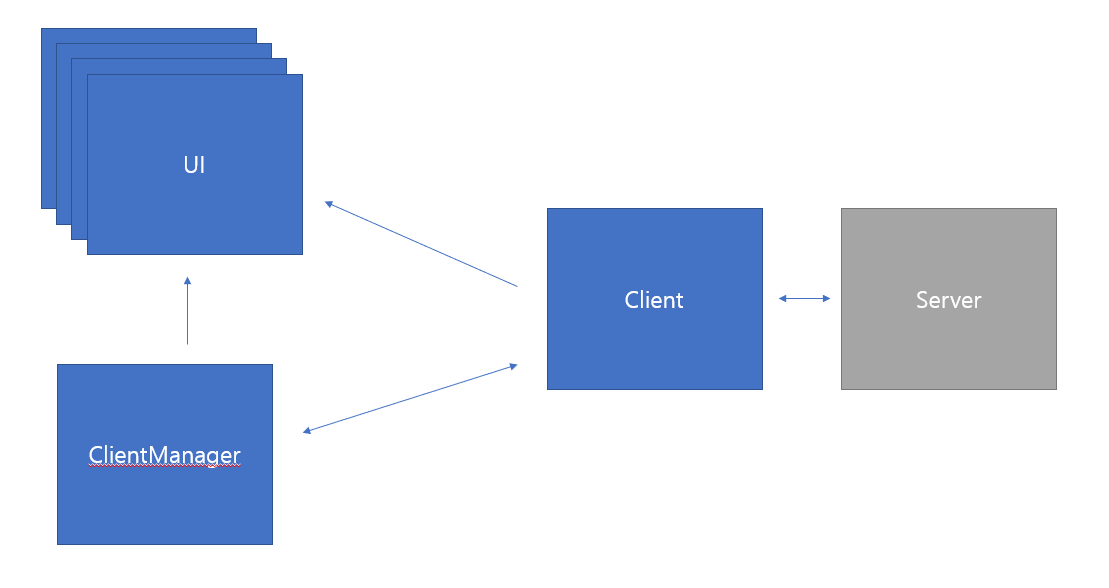
**프로토콜 표**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **번호** | **기능** | **인자 값** | | | |
| **1** | 메시지 | 아이디 | 메시지 | | |
| **6** | 로그인 전송 | 아이디 | 비밀번호 | | |
| **7** | 로그인 응답 | 결과 | 아이디 | | |
| **8** | 로그아웃 | 아이디 | | | |
| **10** | 방 생성 전송 | 제목 | | | |
| **11** | 방 생성 응답 | 결과 | 방 번호 | 아이디 | |
| **12** | 방 참가 전송 | 방 번호 | | | |
| **13** | 방 참가 응답 | 결과 | 방 번호 | | 아이디 |
| **14** | 방 새로 고침 | 방 개수 | 방 정보, … , … | | |
| **15** | 방 나가기 | 방 번호 | | | |
| **16** | 준비 요청 | 결과 | | | |
| **17** | 준비 응답 | 결과 코드 | | | |
| **18** | 아이디 업데이트 | 아이디 그룹 | | | |
| **19** | 게임 시작 | 아이디 인덱스 | | | |
| **20** | 게임 방 정보 요청 |  | | | |
| **31** | 라운드 시작 | 라운드 | 현재 차례 | | |
| **32** | 뒤집기 요청 | 요청 유저 | | | |
| **33** | 뒤집기 응답 | 카드 번호 | 다음 차례 | | 카드 개수 |
| **34** | 벨 누름 요청 | 요청 유저 | | | |
| **35** | 벨 누름 응답 | 결과 | 요청 유저 | | 카드 개수 |
| **36** | 라운드 종료 | 유저 상태 | | | |
| **37** | 게임 종료 | 이긴 유저 | | | |
| **38** | 차례 변경 | 해당 유저 | | | |

**서버 구성도**



**클라이언트 구성도**

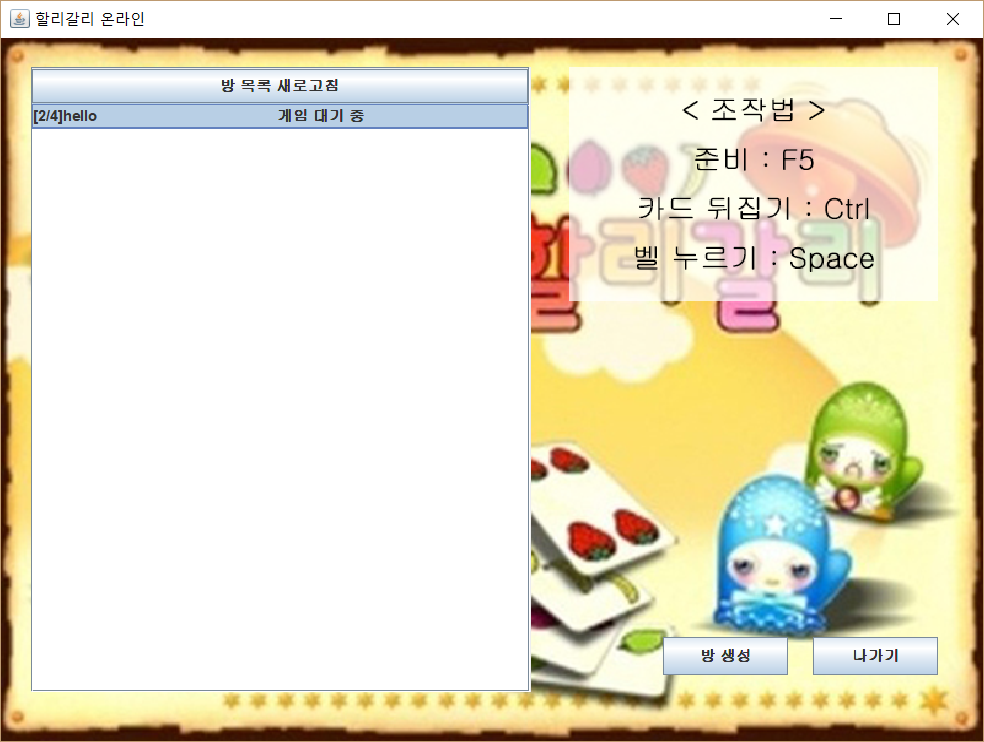


**실행 화면 <윈도우(자바)>**

**로그인**



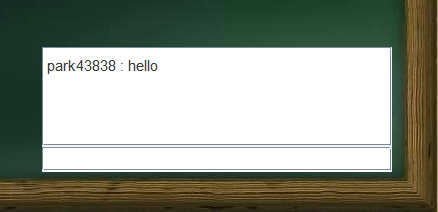
**게임 방 목록**



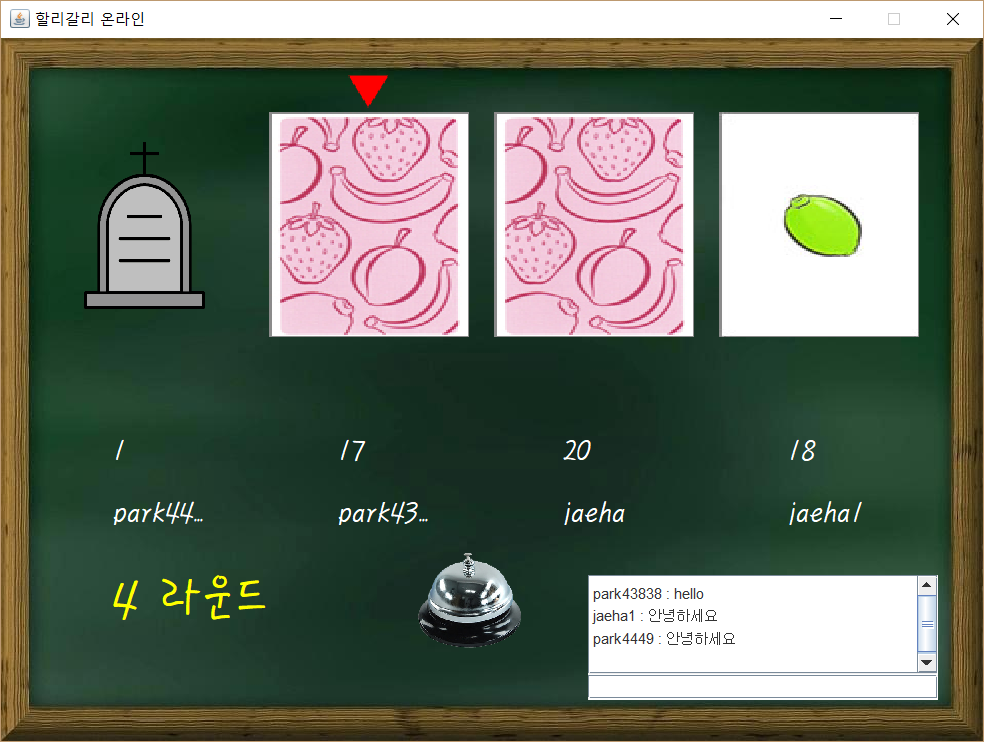
**게임 화면**



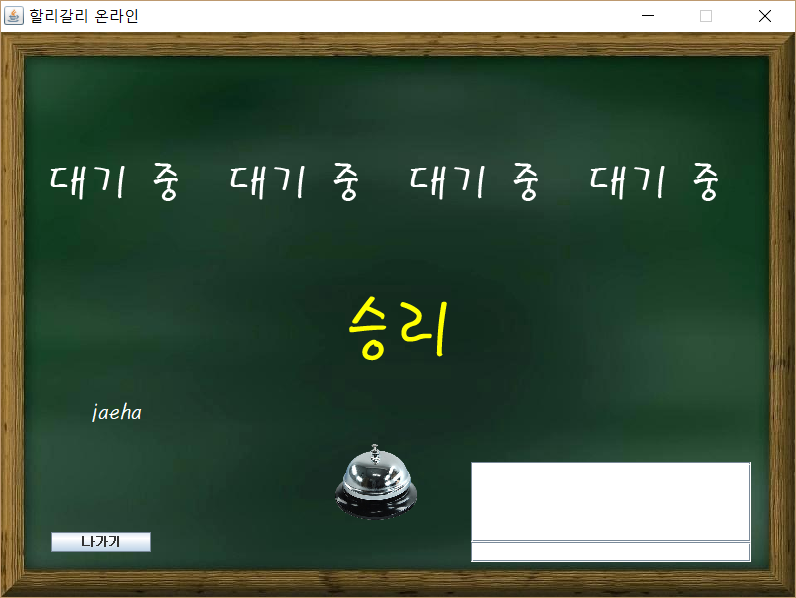
**채팅(확대)**



**1명이 죽은 상태**



**게임 종료**



**실행화면 <안드로이>**

**로그인**

****

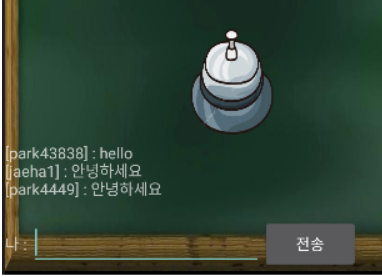
**게임 방 목록**



**게임 화면**

****

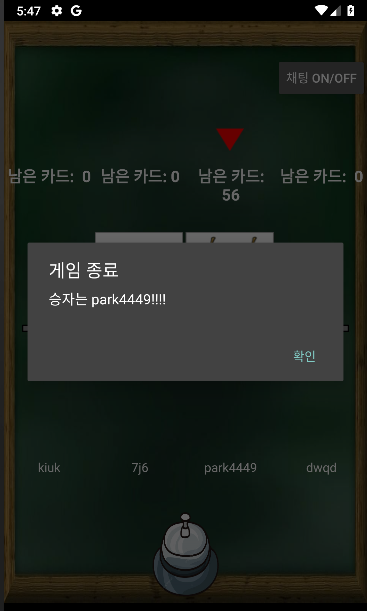
**채팅(확대)**



**1명이 죽은 상태**

****

**게임 종료**



**프로젝트 소감**

**1496091 이재하**

프로젝트를 하면서 느낀 점은 네트워크 프로그래밍에 있어서 프로토콜(약속)을 만드는 것은 코딩하는 것만큼이나 중요하다는 것을 알았다. 한번 정해진 약속은 다시 고치는데 나 뿐만 아니라 다른 사람에게도 영향을 미치기 때문에 효율적이지 않은 프로토콜을 계속해서 사용해야했다.

또한 이때까지는 클라이언트 영역에 가까운 곳을 설계했다면, 이번에는 클라이언트와 서버 동시에 설계해야 했기 때문에 더 많은 신경을 요했다.

같은 이유로, 코딩에 있어서 설계가 얼마나 중요한지 깨닫게 되었다. 내가 만약 설계를 잘못하여 코딩하는 도중에 그것을 발견했다면, 그것을 고치는데 이러한 네트워크 프로그래밍의 경우 정말 많은 노력이 필요하다는 것을 알게 되었다.

마지막으로 이러한 네트워크 프로그래밍을 하면서 성능 문제에 대하여 고민하게 되었다. 요즘 컴퓨터들은 워낙에 빨라서 네트워크 환경을 이용하지 않는다면 성능이 조금 떨어져도 체감이 안될 정도로 빠르다. 하지만 네트워크 환경은 어쩔 수 없는 외부 요인 때문인지 조금 잘못 구현하게 되면 성능이 눈에 보이게 떨어지고 심한 경우에는 동기화가 제대로 되지 않거나 메시지의 순서가 일정치 않는 등, 내가 원하지 않는 결과를 낳게 된다. 그래서 그런 문제를 해결해보고 싶었지만 그런 쪽으로는 지식이 부족하여 더 깊게 생각해볼 수 없었다는 것이 매우 아쉽다. 만약 기회가 된다면 이번 프로젝트를 계기로 네트워크 관련 성능과 오류가 생기는 것들에 대하여 공부를 더 해보고 싶다.

**1491012 박성영**

이번 네트워크 기말 프로젝트를 하면서 초반 설계에 중요성을 크게 느꼈다. 처음 설계를 잘 못하게 되면 코드도 길어지고 오류 수정에 어려움도 있다는 것을 몸소 느꼈다.

또한 서버에 설계에 편리함을 느꼈다. 작년 프로그래밍 랩 프로젝트로 게임을 만들었을 때는 서버 없이 1인용 게임을 만들었지만 이번 프로젝트보다 훨씬 더 까다로웠었는데, 이번 프로젝트에서는 서버를 만들어서 신경써야 할 연산을 서버에서 처리하고 클라이언트에서는 그 값을 받기만 하여 ui로 보여주는 형식으로 짰는데, 서버만 잘 만들어 놓으면 클라이언트 구성은 쉽게 할 수 있었던 것 같다.

그리고 자바와 안드로이드로 프로그램을 짰기 때문에 공부도 2배가 되었던 것 같다. 처음에는 안드로이드가 자바 기반이라 쉽게 생각했지만, 안드로이드와 자바 사이에 쓰는 함수들이 다른 것들이 많아 조금 까다로웠던 것 같다. 하지만 안드로이드에서 지원하는 AP를 이용하면 자바에서 복잡하게 짜야 되는 기능들을 안드로이드에서는 API를 이용해서 쉽게 구현할 수 있었고 화면 구성에 제약이 있었지만 xml을 이용한 안드로이드는 자바 스윙을 이용한 화면 구성보다 훨씬 간편하였다.

마지막으로 이번 네트워크 프로젝트를 통해 소켓 통신에 대해 몸소 느낄 수 있는 기회가 되었고,. 다음 졸업프로젝트 등 다른 프로젝트가 있다면 UDP등 다른 방식에 통신 방법으로 프로그램을 짜봤으면 좋겠다.