

# SOCKET PROCESSING OMOK PROJECT

**SPOP**

---

## PROJECT OVERVIEW

- 2019

**SPOP** is Omok network game that uses Processing Library.

- SPOP uses TCP / IP sockets to implement communication between the server and the client.
- Structural design is more important than implementation in programming.
- This program aims to become proficient at documenting the project.

Developer  
Information

- JUNSEOP LIM , HYUNJUN KIM

-

PC & Mobile App  
Java JDK 8  
Processing 3.5.3

© 2019. Thethelab Inc. All Rights Reserved.

SOCKET  
PROCESSING  
OMOK  
PROJECT

# ABOUT SPOP

---

## WHAT - WE DESIGN

SPOP 은 TCP/IP 소켓을 이용한 네트워크 오목 게임으로써, 프로세싱 라이브러리를 통해 유저 인터페이스를 구현합니다.  
**네트워크 통신 구조에 대한 이해와 프로젝트 내용의 문서화에 익숙해지기 위한 프로젝트입니다.**

---

PC & Mobile App  
Java JDK 8  
Processing 3.5.3

---

---

## Requirements

/ 요구사항

- SPOP은 TCP/IP 소켓을 이용해 만든 네트워크 오목게임이다.
  - 클라이언트와 서버는 지정된 프로토콜에 따른 메시지를 주고 받는다.
  - SPOP은 ① 대기, ② 게임, ③ 종료, 세 가지의 상태를 갖는다.
  - ① 대기
    - 클라이언트의 정보를 확인할 수 있다.
    - ‘게임’ 상태로 넘어가기 위한 ‘READY’ 버튼이 있다.
    - 서버로 부터 게임 시작 메시지를 받으면 카운트 다운, 게임 시작 문구가 출력된다.
  - ② 게임
    - 클라이언트의 게임 순서를 확인하기 위한 애니메이션이 출력된다.
    - 각각의 순서에 따라 누구의 차례인지 확인할 수 있다.
    - 마우스 클릭을 이용해 게임을 진행한다.
  - ③ 종료
    - 게임 결과를 확인하며, 일정 시간 후 '대기' 상태가 된다.
-

Protocol

/ 프로토콜

클라이언트의 이벤트 발생에 따라, 서버와 문자열 메시지를 주고받으며 통신. 서버와 클라이언트는 '#' 구분자로 메시지를 구분하여 수행 기능과, 전송 데이터를 구분한다.



클라이언트



서버

| 헤더     | 데이터              | 기능   | 예시                        |
|--------|------------------|--|---------------------------|
| Client | 클라이언트 번호         | 서버 → 클라이언트<br>연결 순서에 따라 클라이언트 번호를 지정한다.              | Client#1                  |
| Ready  | 클라이언트 번호         | 클라이언트 → 서버<br>클라이언트 준비 완료 시, 서버에 알려준다.               | Ready#1                   |
| Dice   | 주사위 값 1, 주사위 값 2 | 서버 → 클라이언트<br>두 클라이언트 준비 완료 시, 주사위 값들을 클라이언트에게 알려준다. | GameStart#gamestart#4#3#1 |
| Check  | X 좌표, Y 좌표, 흑백   | 클라이언트 → 서버<br>마우스 클릭 시, 좌표와 흑백 값을 서버에 알려준다.          | Check#8#15#BLACK          |
| Do     | X 좌표, Y 좌표, 흑백   | 서버 → 클라이언트<br>돌을 놓을 수 있는 위치인지 체크 후, 클라이언트에게 알려준다.    | Do#8#15#BLACK             |
| Out    | 클라이언트 번호         | 서버 → 클라이언트<br>상대 클라이언트 접속이 끊기면 클라이언트에게 알려준다.         | Out#0                     |



SOCKET  
PROCESSING  
OMOK  
PROJECT  
**DATA FLOW**

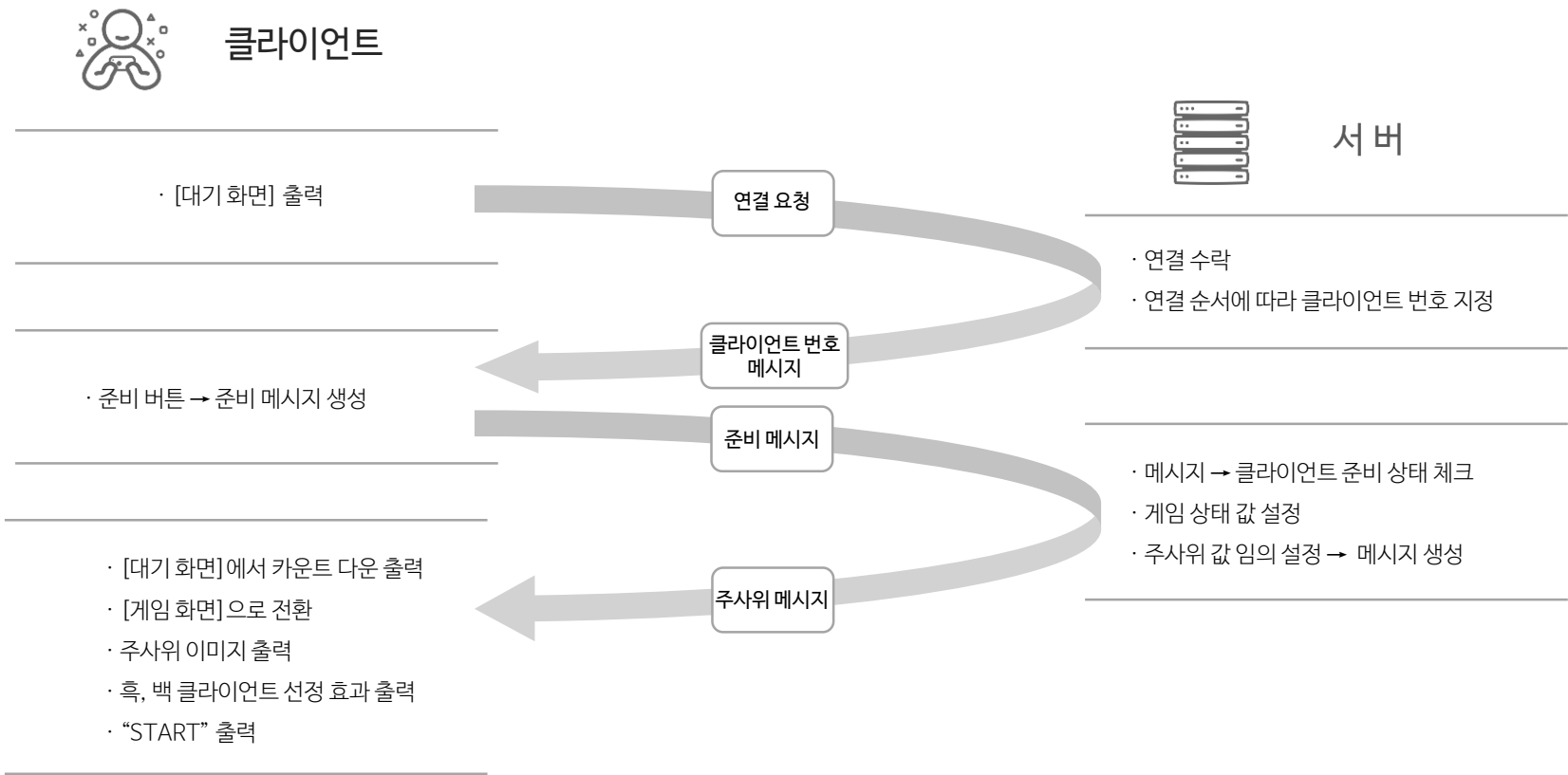
---

DATA FLOW  
of SPOP

DATA FLOW 장에서는 서버와 클라이언트가 어떤 식으로 데이터를 주고 받는지 그 흐름을 보여줍니다.

PC & Mobile App  
Java JDK 8  
Processing 3.5.3

---





클라이언트

- 돌을 놓을 수 있는 위치 → 마우스 커서 변경
- 마우스 클릭 → 좌표 값, 흑백 값 담긴 메시지 생성

- 돌을 오목판에 표시
- 오목 확인 시, 오목 하이라이트 효과 출력
- [결과 화면] 출력
- 일정 시간 후, [대기 화면] 으로 전환

- 클라이언트 종료 텍스트 출력
- [대기 화면] 으로 전환
- 클라이언트 필드 초기화



서버

- 메시지 → ① 좌표 범위, ② 중복, ③ 금수, ④ 오목
- 조건 만족 시,  
좌표 값, 흑백 값, 게임 상태 담긴 메시지 생성

- 게임 상태 값 변경
- 클라이언트 번호 재설정을 위한 메시지 생성

확인 메시지

수행 메시지

클라이언트  
연결 끊김

초기화 메시지

# SOCKET PROCESSING OMOK PROJECT UI / UX

---

UI / UX  
Overview

---

PC & Mobile App  
Java JDK 8  
Processing 3.5.3

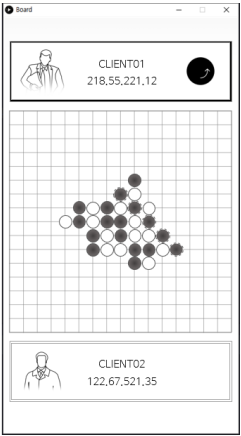
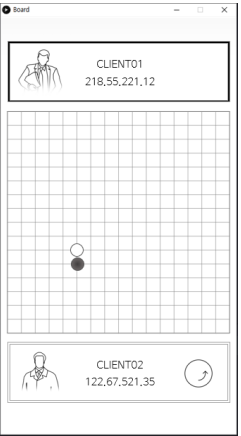
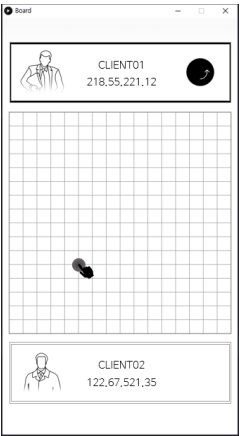
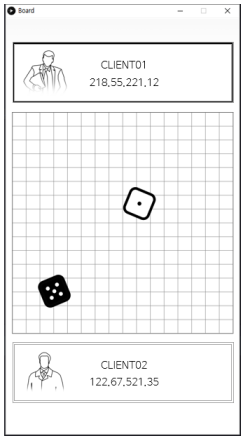
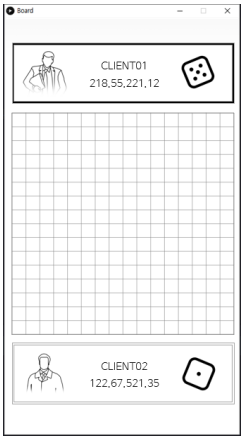
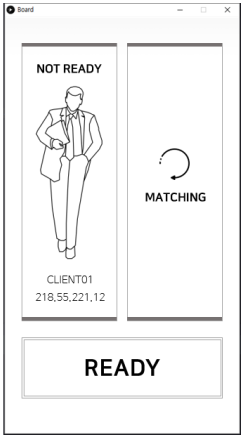
---



Overview

/ 화면 명세서 요약

\* 보는 순서



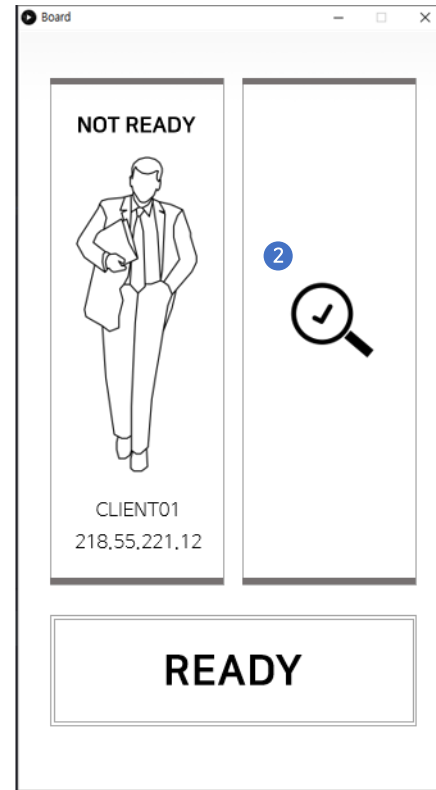
### Matchmaking

/ 대기화면

① 클라이언트를 실행하면 상대 클라이언트를 찾을 때까지 표시를 계속해서 화면에 출력한다.

② 상대 클라이언트를 찾으면, 매칭이 완료되었다는 표시를 띄운다.

③ 표시를 2초간 띄운 후, 상대 클라이언트의 프로필과 정보, 준비 상태를 화면에 출력한다.



\* ① ② ③ 순서가 진행되어 상대 클라이언트 컴포넌트 내부의 이미지가 변경될 때마다 페이드 인 / 아웃 효과를 적용한다.

### Preparation Sequence

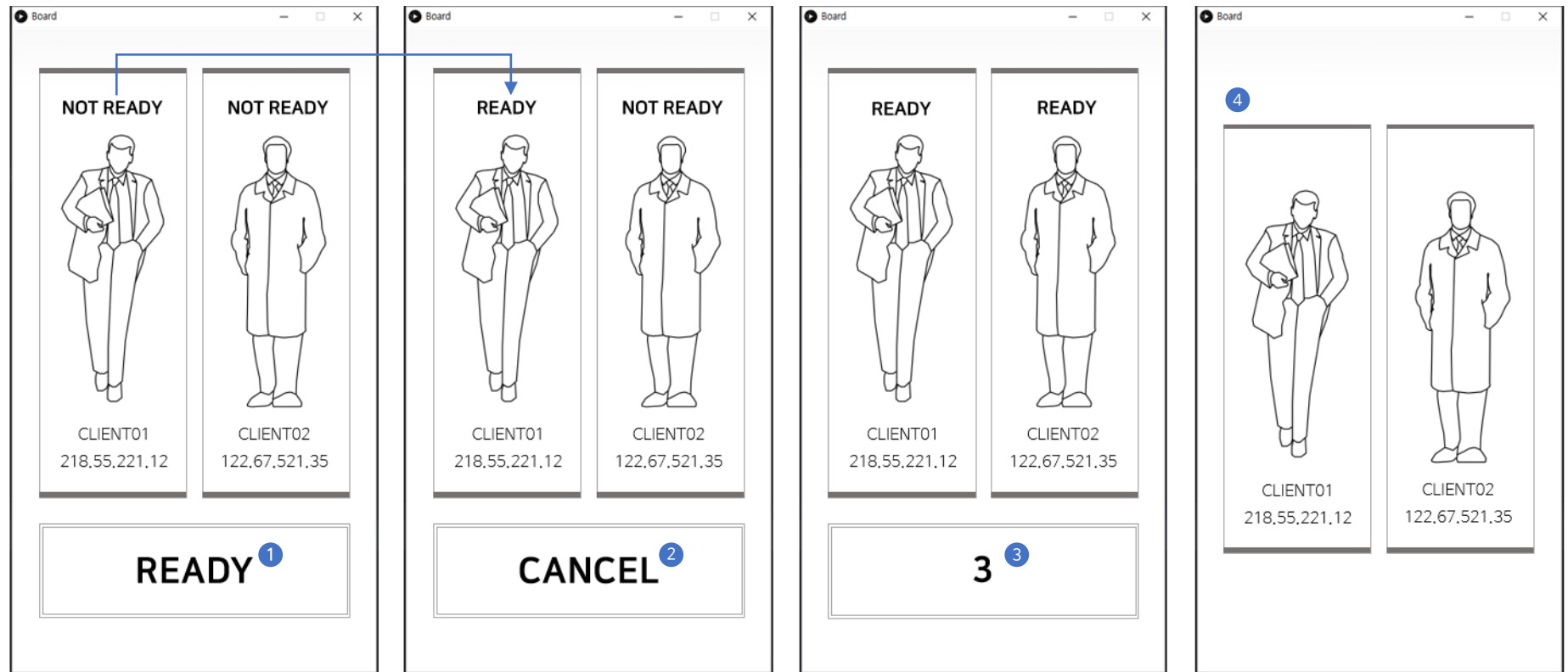
/ 준비 화면

① 준비 버튼을 누르면 프로필 위의 준비 상태 텍스트가 'NOT READY'에서 'READY'로 변경되고

② 준비 버튼의 텍스트는 'READY'에서 'CANCEL'로 변경된다.

③ 두 클라이언트가 모두 준비 상태일 경우, READY 버튼 위의 텍스트가 3초의 카운트 다운으로 변경된다.

④ 카운트 다운이 완료되면 준비 버튼 창이 사라지고, 클라이언트의 프로필이 중앙으로 내려온다.



### Game Screen

/ 게임 화면

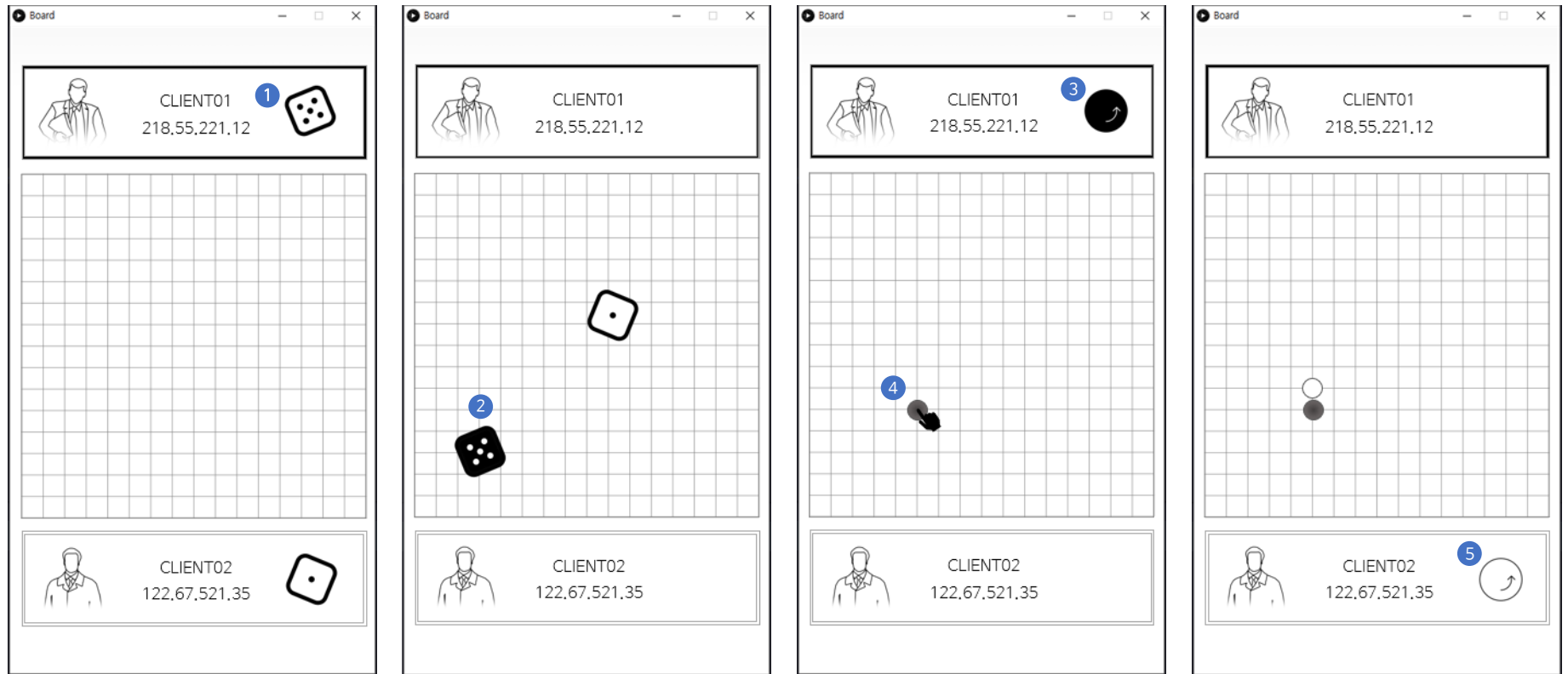
① 준비 화면에서 게임 화면으로 변경 후, 두 클라이언트는 동시에 주사위를 던진다.

② 두 개의 주사위 중 큰 값이 나온 주사위의 색이 검정색으로 변하며, 해당 주사위를 던진 클라이언트가 흑이 된다.

③ 누구의 턴인지 클라이언트 정보 우측에 버튼이 표시되며, 해당 버튼을 누르면 돌을 한 수 물릴 수 있다.

④ 돌을 놓을 수 있는 위치에 마우스 커서를 놓으면 커서가 손 모양으로 바뀌며, 클릭 시 돌이 놓이게 된다.

⑤ 돌을 놓으면 상대 클라이언트에게 턴이 넘어가게 되고, 동일하게 버튼이 표시된다.



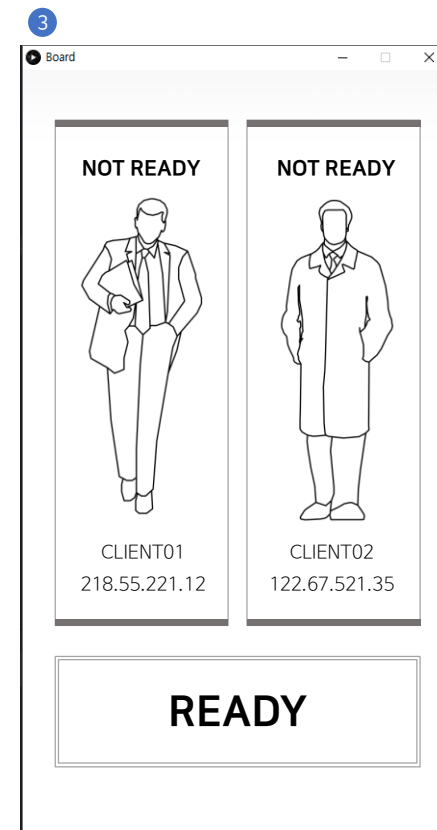
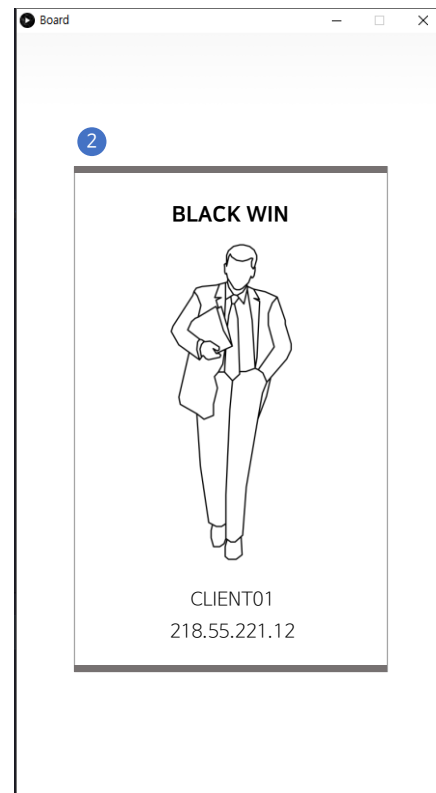
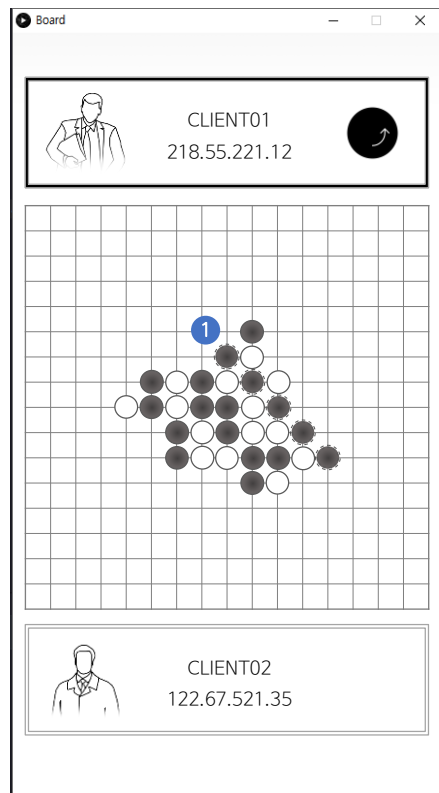
### Result Screen

/ 결과 화면

① 게임 진행 중 오목이 완성되면, 3초 동안 점선으로 외곽을 강조해준다.

② 3초가 지나면 승리한 클라이언트의 프로필과 정보를 5초 동안 보여준다.

③ 5초가 지나면 자동으로 준비 화면으로 돌아간다.





SOCKET  
PROCESSING  
OMOK  
PROJECT

**UI / UX** with Animation

---

UI / UX  
Overview

---

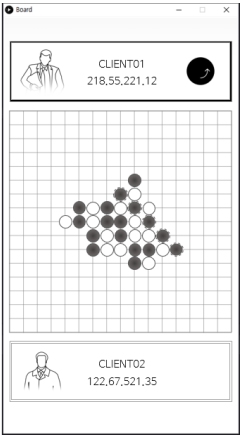
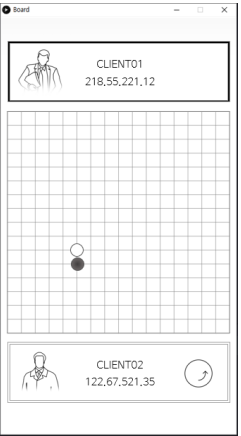
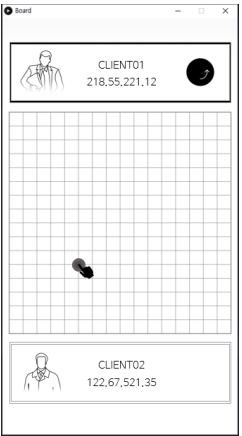
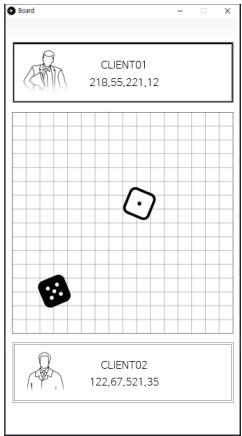
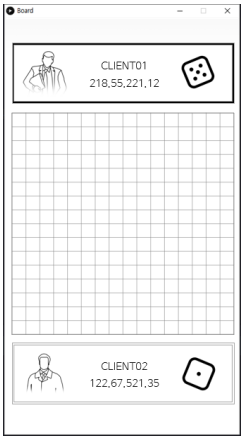
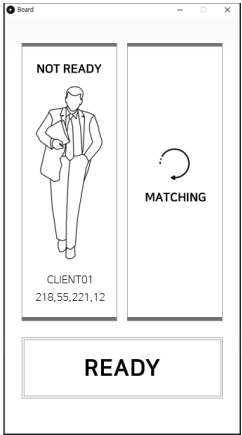
PC & Mobile App  
Java JDK 8  
Processing 3.5.3

---

Overview

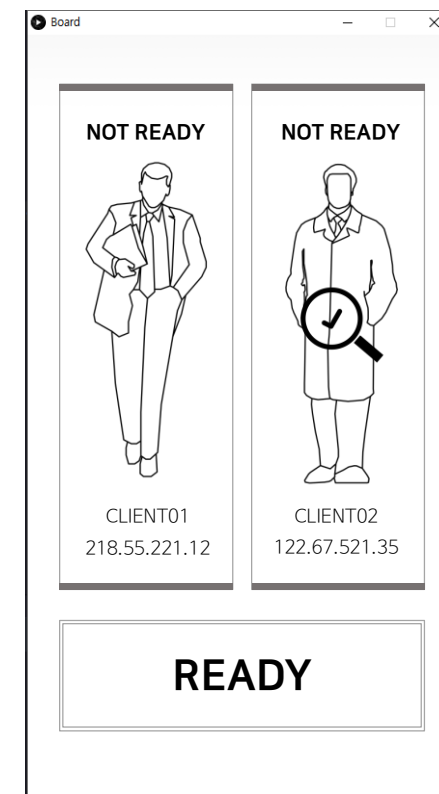
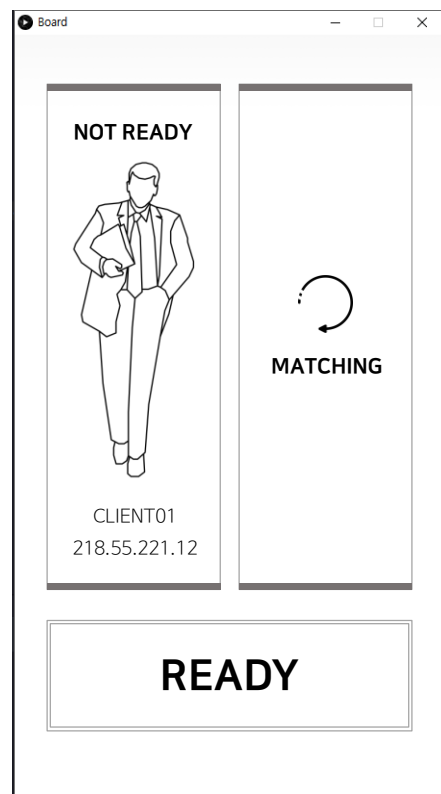
/ 화면 명세서 요약

\* 보는 순서



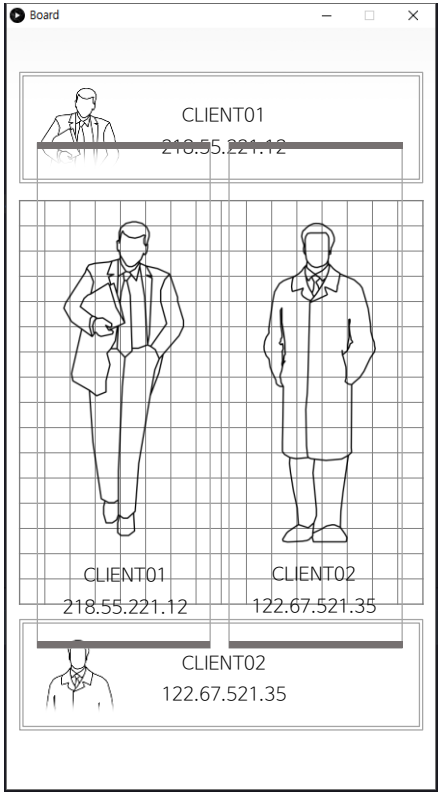
## Matchmaking

/ 대기화면



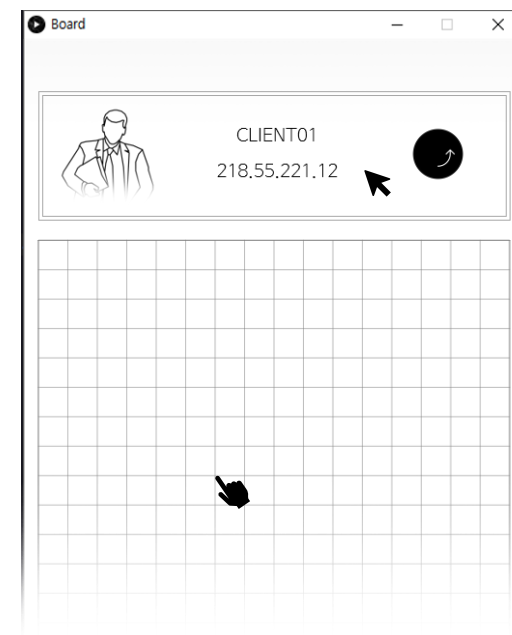
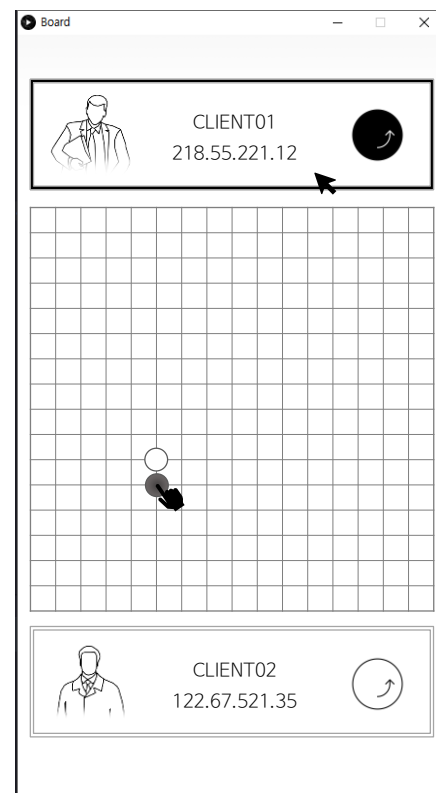
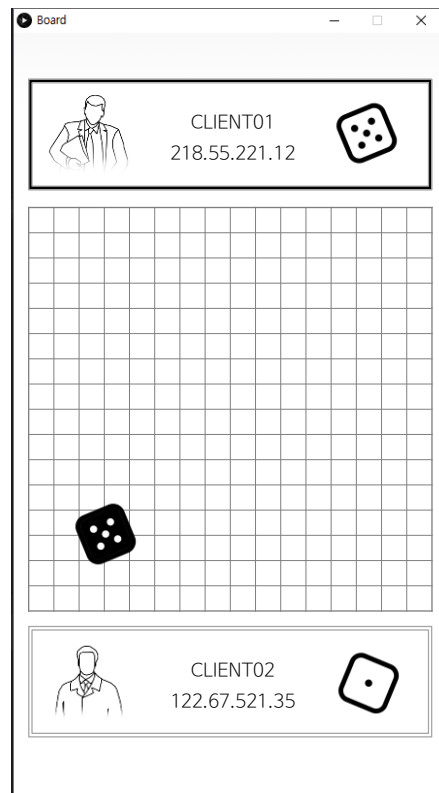
Preparation Sequence

/ 준비 화면



## Game Screen

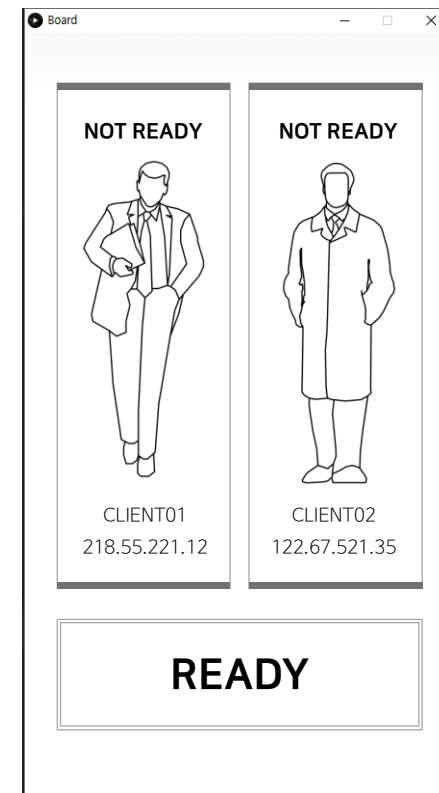
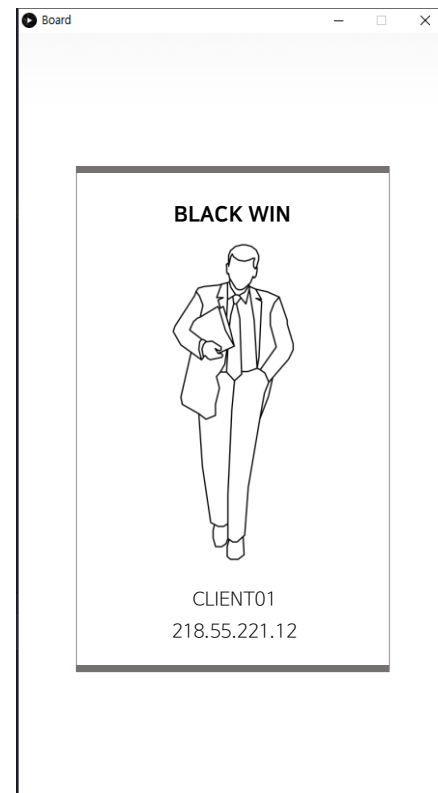
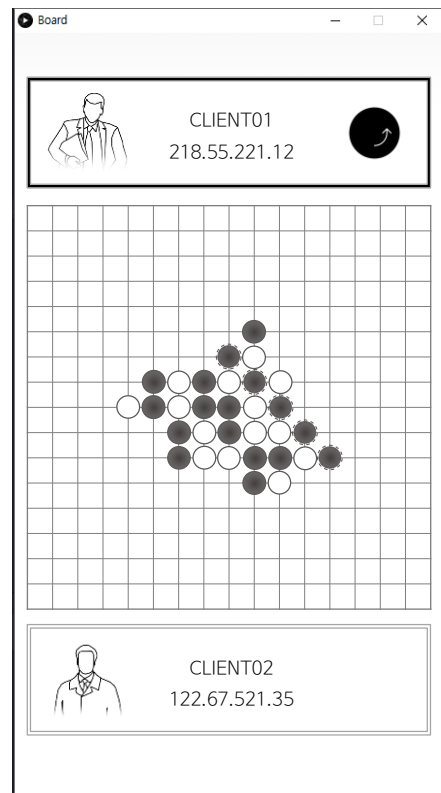
/ 게임 화면





## Result Screen

/ 결과 화면



# SOCKET PROCESSING OMOK PROJECT STRUCTURE

---

Structure  
Overview

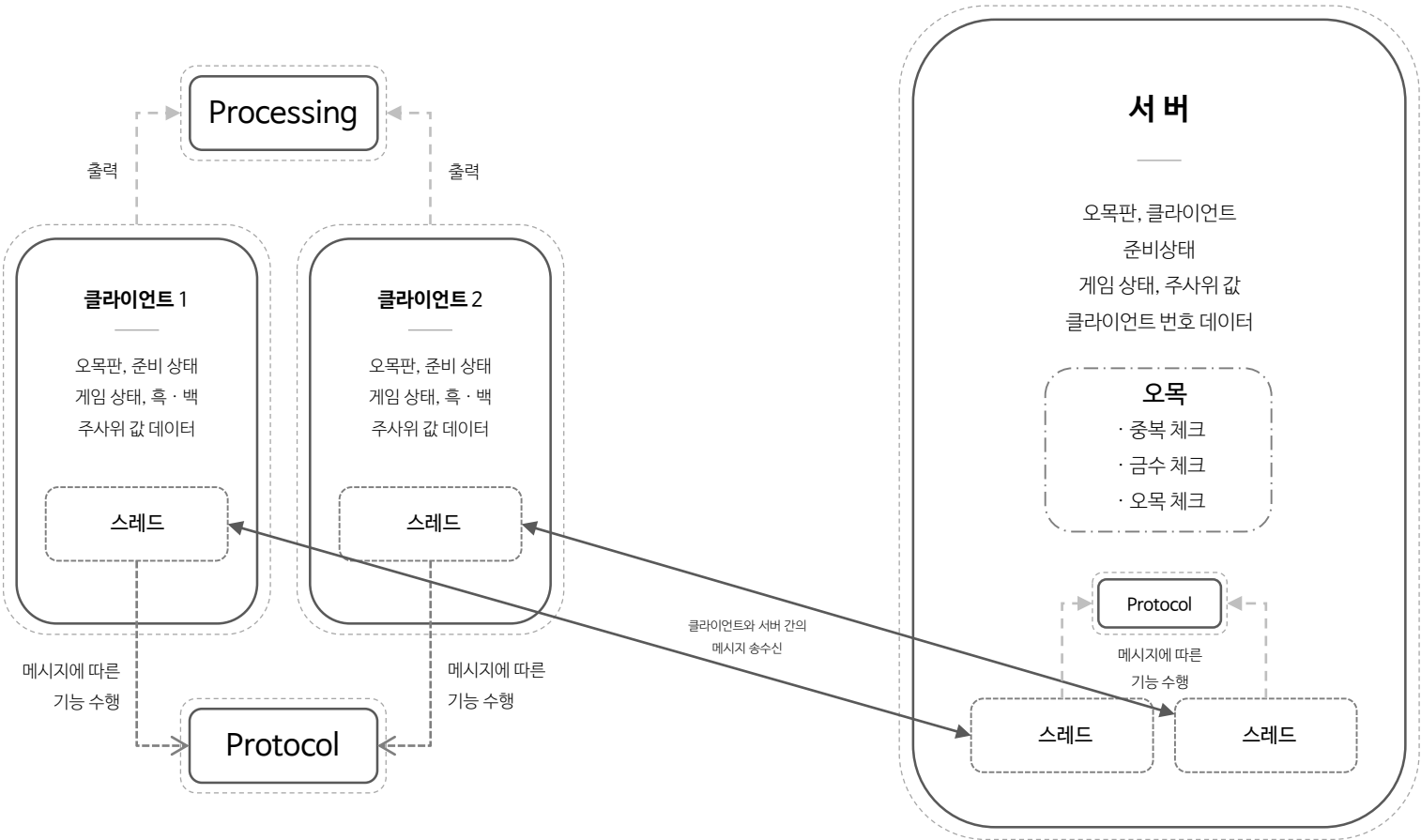
---

PC & Mobile App  
Java JDK 8  
Processing 3.5.3

---

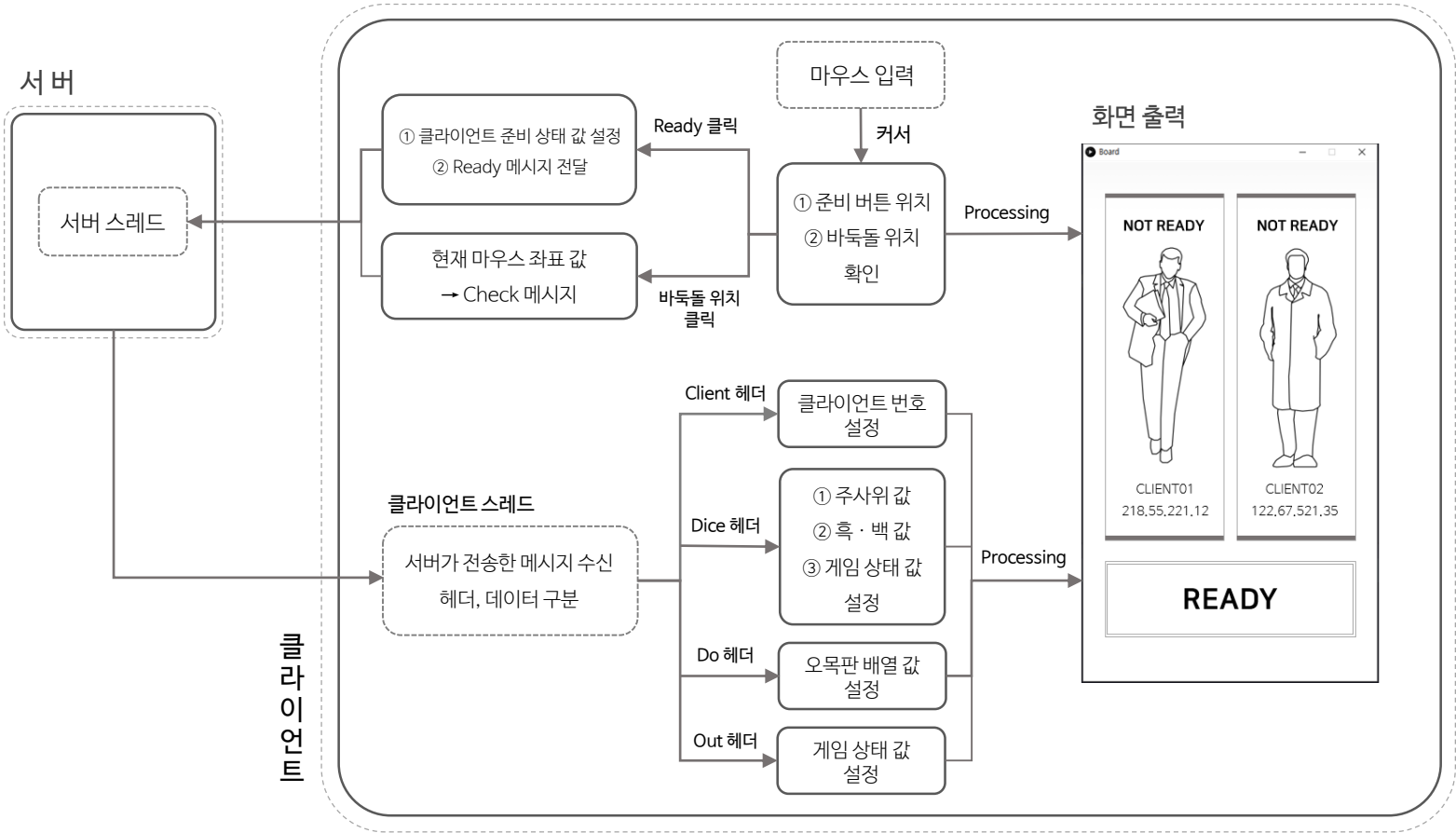
Overall Architecture

/ 전체 프로젝트 구조



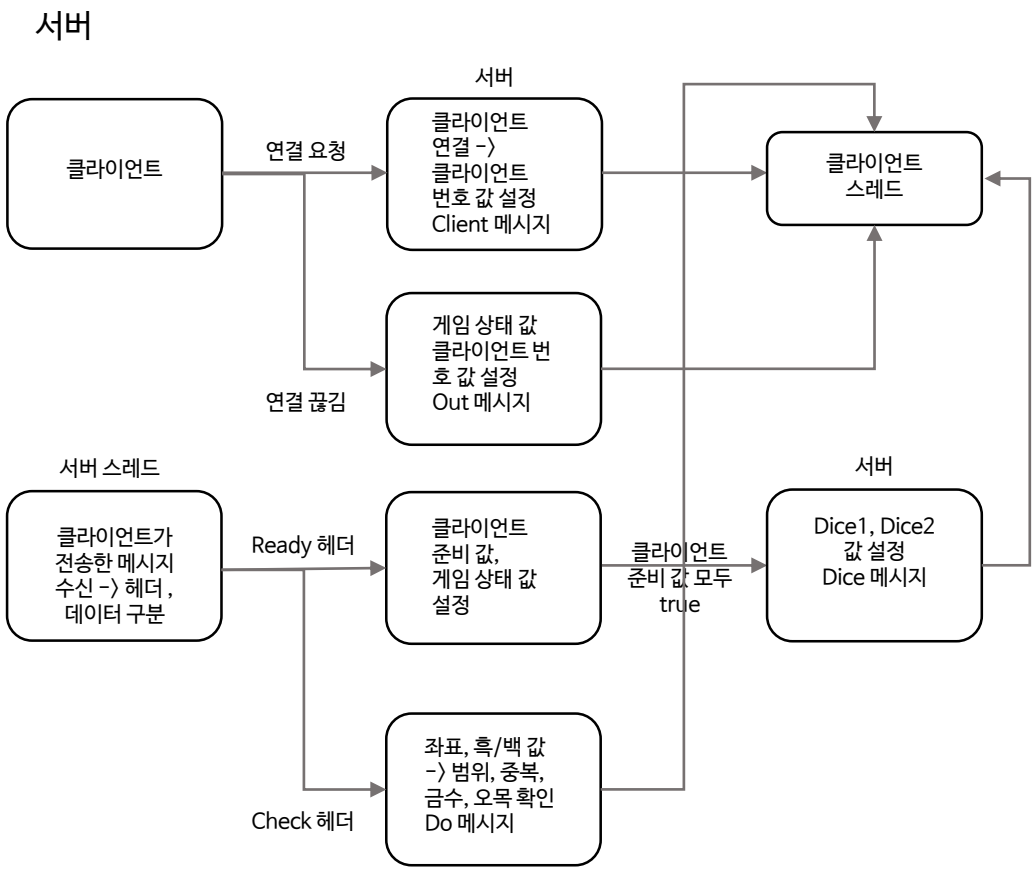
Client Architecture

/ 클라이언트 프로젝트 구조



Server Architecture

/ 서버 프로젝트 구조





END  
OF  
DOCUMENT  
**THANK YOU**

---