

# \* Mafia Game

범인은 이 안에 있다!!!

김웅태, 정상구 ,정원석

- \* 기존 선생님께서 제공해 주신 채팅 프로그램 기반
- \* **Database**는 서버에서만 접속 가능하게 구현
- \* **Client**만 재배포 가능한 버전 구현
- \* 게임 시작 자동
- \* 투표시 이벤트창 자동으로 생성
- \* 게임 끝나면 시간지연 후 자동으로 다시 게임 시작
- \* 서버에서 게임 내용 실시간 감시 가능

\* 구현 목표

## Server 측

### Server Main

#### DB Connection

-DB접속

#### GameRoom

-User클래스를 제네릭으로 갖는 벡터

#### User

-쓰레드

#### GameSystem

-마피아 결정  
게임 시작  
이벤트 발생

## Client 측

### Clinet main

#### Login Client

-최초 접속 시 서버로 로그인 정보 전송

#### Mafia Client

-게임 실행 시 쓰는 클라이언트 기본 채팅 정보 및 사용자 목록을 가짐

#### Event

-투표시 이벤트창  
사용자 목록과 디자인을 가지고 있음

#### Memberdesign

로그인 /001/USER\_ID<UserPASS

로그아웃 /002/USER\_ID/

채팅 /003/USER\_ID/data

투표내용 /004/USER\_ID/data

이벤트 /005/Event 실행유무 (true/false)

사용자 정보 /007/User\_List

마피아 유무 /008/실행유무 (true/false)

\*패킷

```

@Override
public void actionPerformed(ActionEvent e) {
    String packet = null;
    String pass = new String(pass_jt.getPassword());
    pw.println("/001/"+id_jt.getText()+"<"+pass);
    String data = null;
    System.out.println("Client >> Server : "+"/001/"+id_jt.getText()+"<"+pass);
    try {
        packet = br.readLine();
        String opcode = (String) packet.subSequence(0, 5);
        data = (String) packet.subSequence(5, packet.length());
        System.out.println("Client 받음 : "+ packet);
    } catch (IOException e1) {
        e1.printStackTrace();
    }
    if(data.equals("1")){
        try{
            new MafiaClient(id_jt.getText(),this.so,this.br, this.pw);
            this.dispose();
        }catch(Exception ea){
            ea.printStackTrace();
        }
    }else if(packet.equals("2")){
        JOptionPane.showMessageDialog(this, "패스워드가 틀렸습니다");
    }else if(packet.equals("3")){
        JOptionPane.showMessageDialog(this, "회원가입을 해주세요");
    }
}

```

로그인 정보가 맞으면  
MafiaClient 객체 생성

```

public void run() {
    try{
        while(flag) {
            String packet = br.readLine();
            String opcode = (String) packet.subSequence(0, 5);
            String data = (String) packet.subSequence(5, packet.length());

            if(opcode.equals("/003/")) {
                System.out.println("Client 발음 : " + packet);
                if(data != null && !data.equals("")) {
                    ta1.append("[ "+(String) data.subSequence(0, data.indexOf("/"))+" ] "+(String) data.subSequence(1, data.length()));
                }
            }
            if(opcode.equals("/002/")) {
                ta1.append(data);
            }

            if(opcode.equals("/005/")) {
                if(data.equals("true")) {
                    ev = new Event(this.List, this.so, this.pw, this.br);
                    ta1.append("게임이 시작 되었습니다.\n");
                }
            }

            if(opcode.equals("/007/")) {
                if(!(List.contains(data))) {
                    List.add(data);
                }
                System.out.println(data);
                jl.setListData(List);
            }

            if(opcode.equals("/008/")) {
                if(data.equals(this.id)) {
                    ta1.append("당신은 마피아로 선정 되었습니다.\n");
                }
            }
        }
    }
}

```

\* 클라이언트에서 패  
킷 정보 분류해서 받  
는 부분

```

public void check(String packet) {
    try{
        String opcode = (String) packet.subSequence(0, 5);
        String data = (String) packet.subSequence(5, packet.length());

        if(opcode.equals("/001/")){ //LOGIN 처리부
            DBConnection db = new DBConnection(); //DBConnection 클래스 하나 생성
            try {
                this.Id = ((String) data.subSequence(0, data.indexOf("<"))); //Id 변수에 접속된 클라이언트 id저장
                db.logIn(this.Id);
                send("/001/"+db.confirm(this.Id, //데이터 부분에서 <전까지 (ID)
                    (String) data.subSequence(data.indexOf("<")+1, data.length()))); //데이터 부분에서 <다

                if(db.confirm((String) data.subSequence(0, data.indexOf("<")), //데이터 부분에서 <전까지 (ID)
                    (String) data.subSequence(data.indexOf("<")+1, data.length()))).equals("1")){
                    MainServer.room.UserList.add((String) data.subSequence(0, data.indexOf("<")));

                    db.logIn((String) data.subSequence(0, data.indexOf("<")));
                    for(int i=0; i<MainServer.room.UserList.size(); i++){
                        MainServer.room.sendAll("/007/"+MainServer.room.UserList.get(i));
                        System.out.println("Server >> Client : "+"/007/"+MainServer.room.UserList.get(i))
                    }
                }
            } catch (IOException e) {
                e.printStackTrace();
            }
            System.out.println("Server >> Client "+ db.confirm((String) data.subSequence(0, data.indexOf("<"))
                (String) data.subSequence(data.indexOf("<")+1, data.length()))); // 클라이언트로 보내는 정보
        }

        if(opcode.equals("/004/")){
            if(data.equals("0"))
                MainServer.room.vote[0] +=1;
            if(data.equals("1"))
                MainServer.room.vote[1] +=1;
            if(data.equals("2"))
                MainServer.room.vote[2] +=1;
            if(data.equals("3"))
                MainServer.room.vote[3] +=1;
            if(data.equals("4"))
                MainServer.room.vote[4] +=1;
            System.out.print(MainServer.room.vote[0]); // 투표가 화이

```

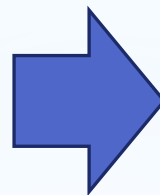
우리가 만든 OP코드를 붙여  
패킷형태로 데이터 전송.

로그인

ID: jws

PASS: •

확인 회원가입



main

jws		



```
/008/null
Server >> Client : /007/jws
Server >> Client 1
Server >> Client : /007/jws
Server >> Client : /007/jws
Server >> Client 1
투표 진행중!!!
1 0 0 0 0
```



\*마피아 제거 실패

\*게임이 끝나지 않음

\*로그인시 서버가 다른 일을 하고 있으면 로그인  
을 무한정 대기

\*마피아 선정에 실패할 때가 생김

\***아쉬운 점**