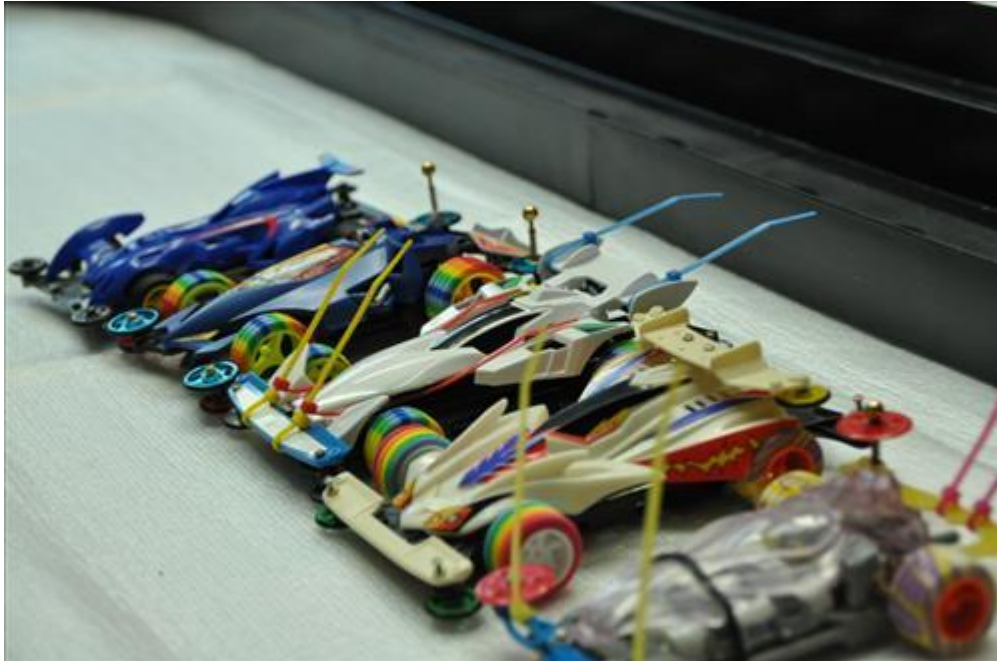


JAVA MINI PROJECT



김한울

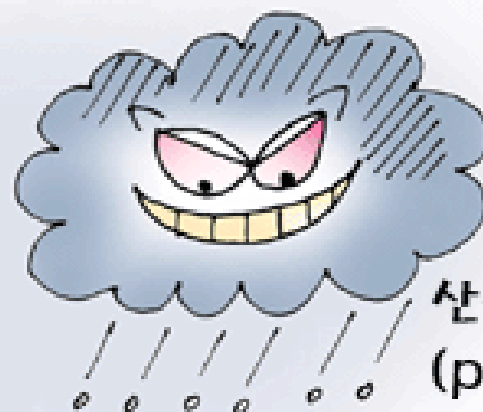
TABLE OF CONTENTS

- ✻ 프로젝트 소개 및 목표
- ✻ 개발일정
- ✻ 주요기능
- ✻ 핵심소스구현
- ✻ 개발환경
- ✻ 시연행사
- ✻ Q&A

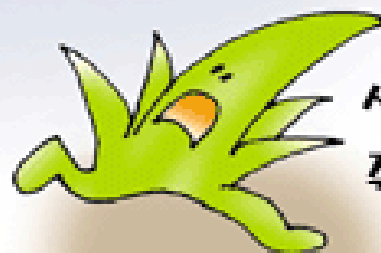
프로젝트 소개 : 산성비

A.K.A ACID RAIN

아황산가스, 또는
질산가스



산성비
(pH 5.0이하)



살려
주세요 !?

프로젝트 소개 : 산성비 A.K.A ACID RAIN

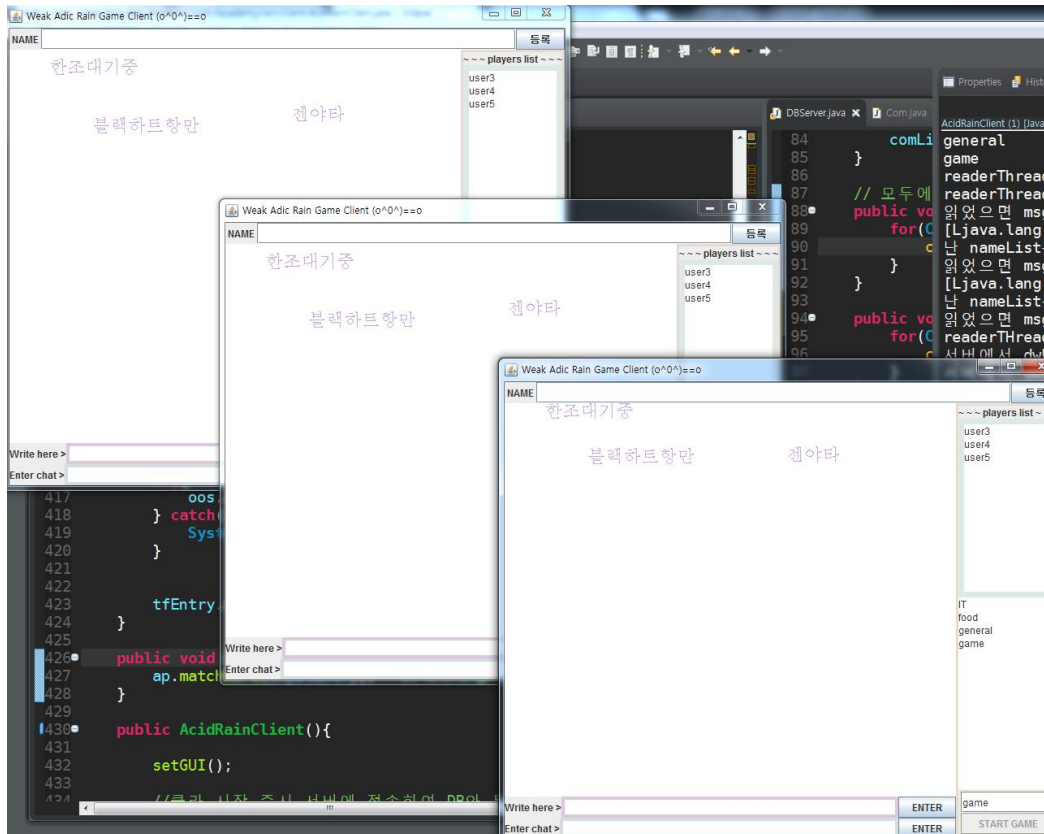


추억의 고전게임, 한컴 타자
산정비 기반

-
-
-

프로젝트 소개 : 산성비

A.K.A ACID RAIN



- 자동 레벨업 없음
- 멀티플레이 추가
- Stage 선택 개념
- Stage별 단어<-> DB

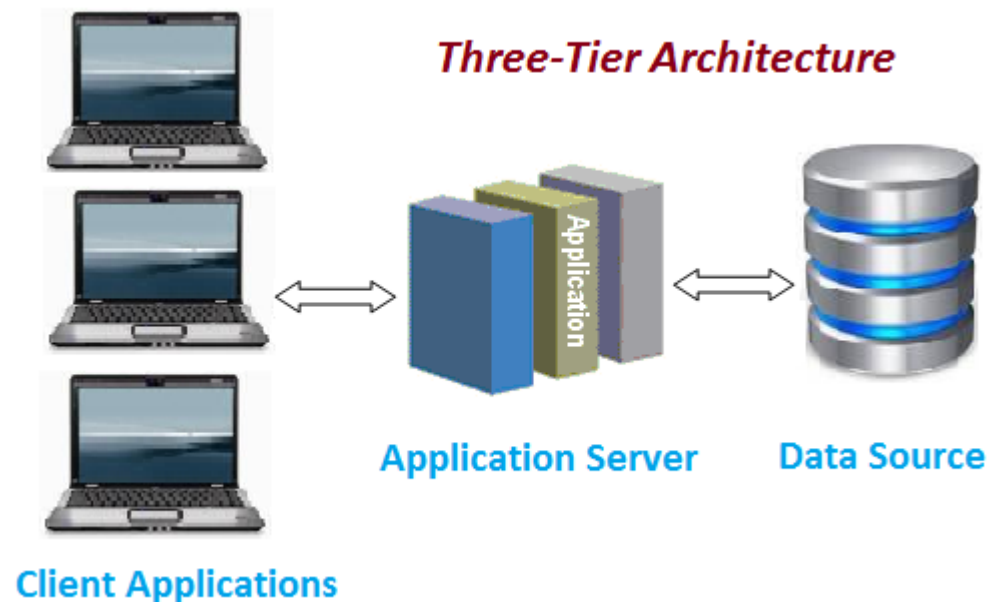
프로젝트 구현 목표

3 Tier system 네트워크 통신

DB 쿼리문 실행을 통한 **CRUD**의 구현

모든 클라에 동일한 화면을 보이게 함으로서

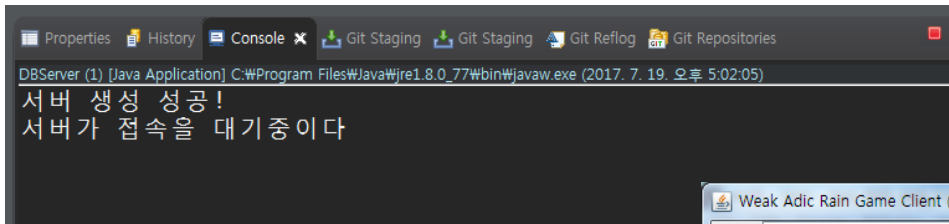
동시 실행 개념에 대한 이해



개발일정

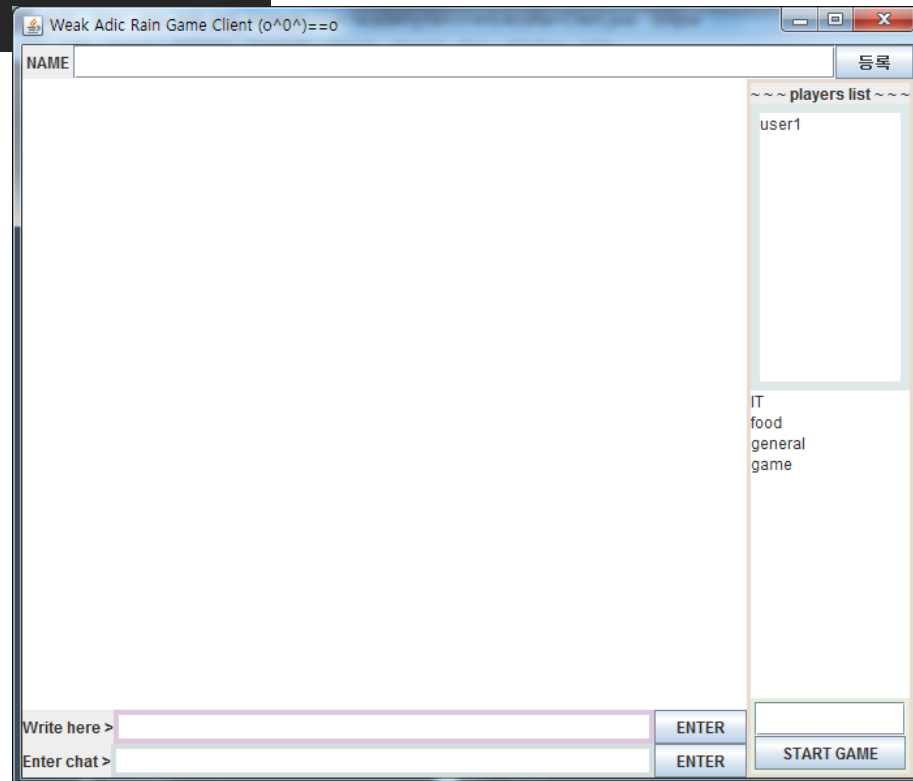
일정							
날짜	07월 14일	07월 15일	07월 16일	07월 17일	07월 18일	07월 19일	07월 20일
목록	업무분석	DB설계 및 쿼리문 작성	프로토콜 설계	기능구현			디버그
	DB설계		UI				발표

주요기능 - 실행화면



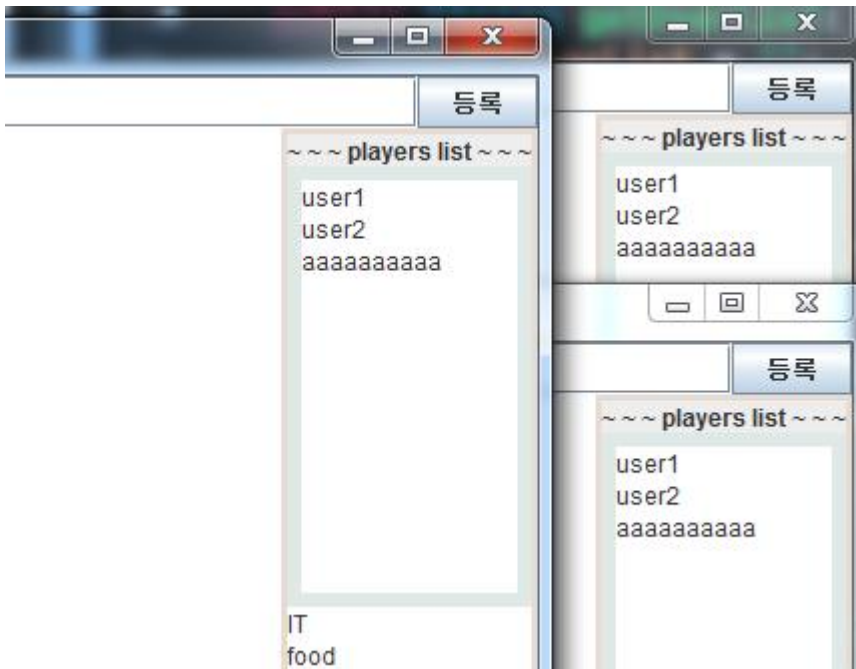
서버는 별개의 **GUI** 없음

클라이언트는
시작과 동시에 서버에
접속하여 네 종류의
테마를 받아온다



주요기능 - 실행화면

유저명 입력하고 등록을 누르면
DB에 insert가 되도록 구현



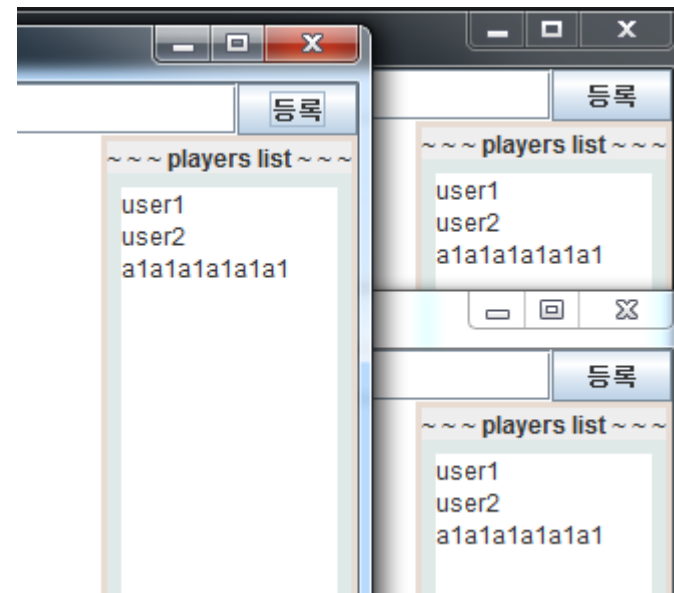
```
String insertSQL = "insert into users(username, ip) "  
    + "values(?, ?);";
```

34	aaaaaaaaaa	0	2017-07-19 17:05:27	192.168.205.128
----	------------	---	---------------------	-----------------

주요기능 - 실행화면

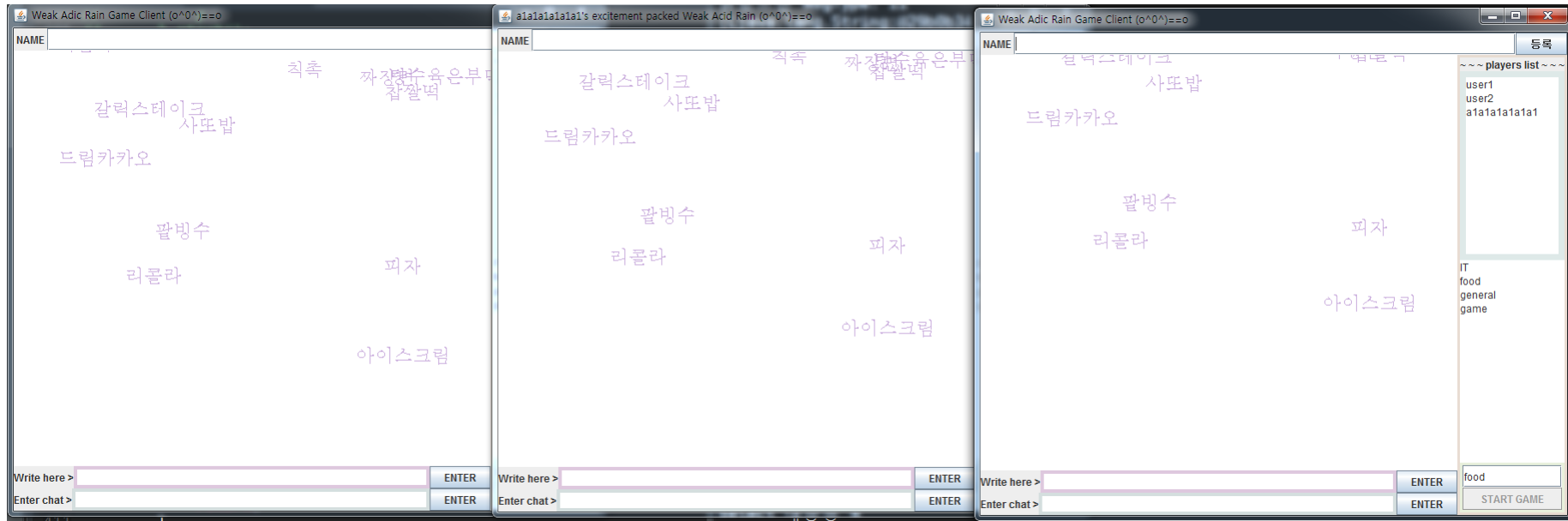
동일한 버튼에 **Flag**를 나누어 처음 눌렀을시 **DB에 insert**,
그 이후로는 지속적으로 **update**가
되도록 구현

```
private int nameNeverChanged = 1; // true, 0: false  
  
case "I":  
    configNameInsertUpdate();  
  
public void configNameInsertUpdate(){  
    switch(nameNeverChanged){  
        case 0:  
            rename();  
            updateUser();  
            break;  
        case 1:  
            insertUser();  
            nameNeverChanged = 0;  
            break;  
    }  
}
```



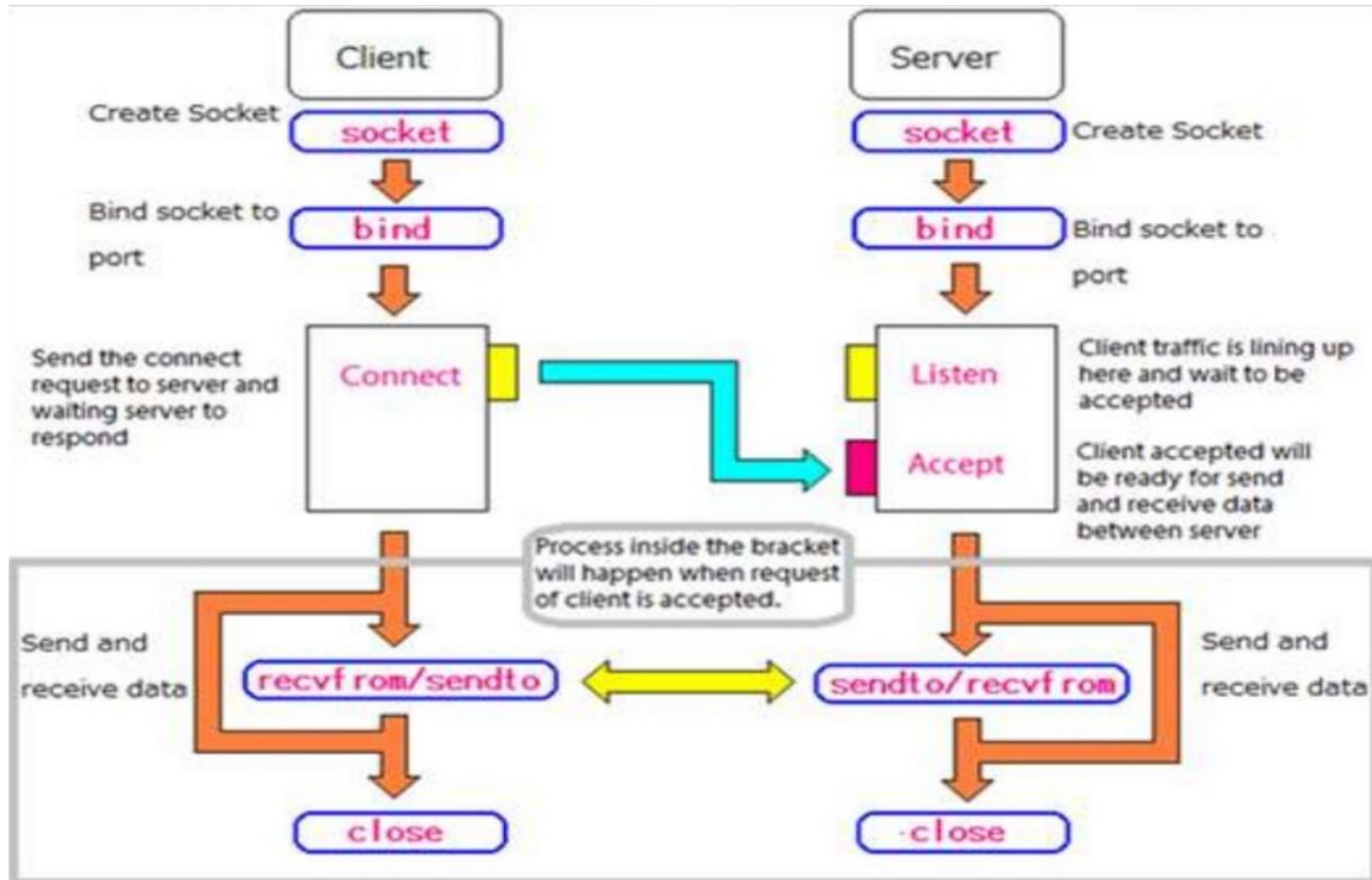
34	a1a1a1a1a1a1	0	2017-07-19 17:05:27	192.168.	0
----	--------------	---	---------------------	----------	---

주요기능 - 실행화면

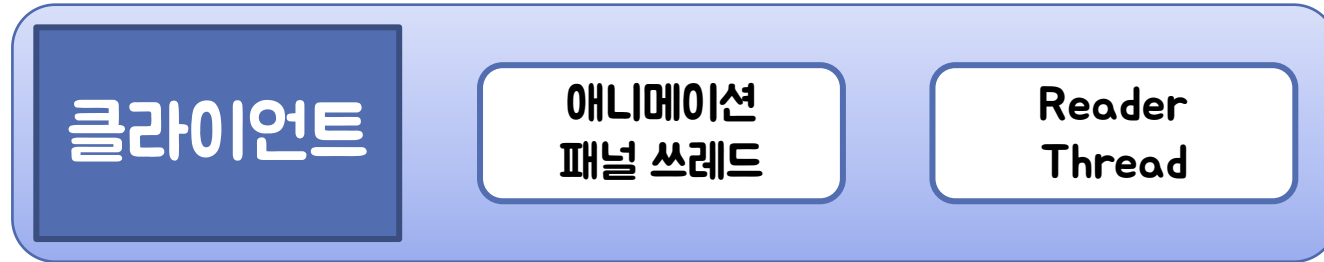


타입 선택하고 **start**버튼을 클릭하면 서버에서 모든 클라이언트에 DB에서 뽑아온 단어, x,y좌표와 속도 조절을 위한 **deltaY**를 담은 **ArrayList**를 동일하게 뿌려준다

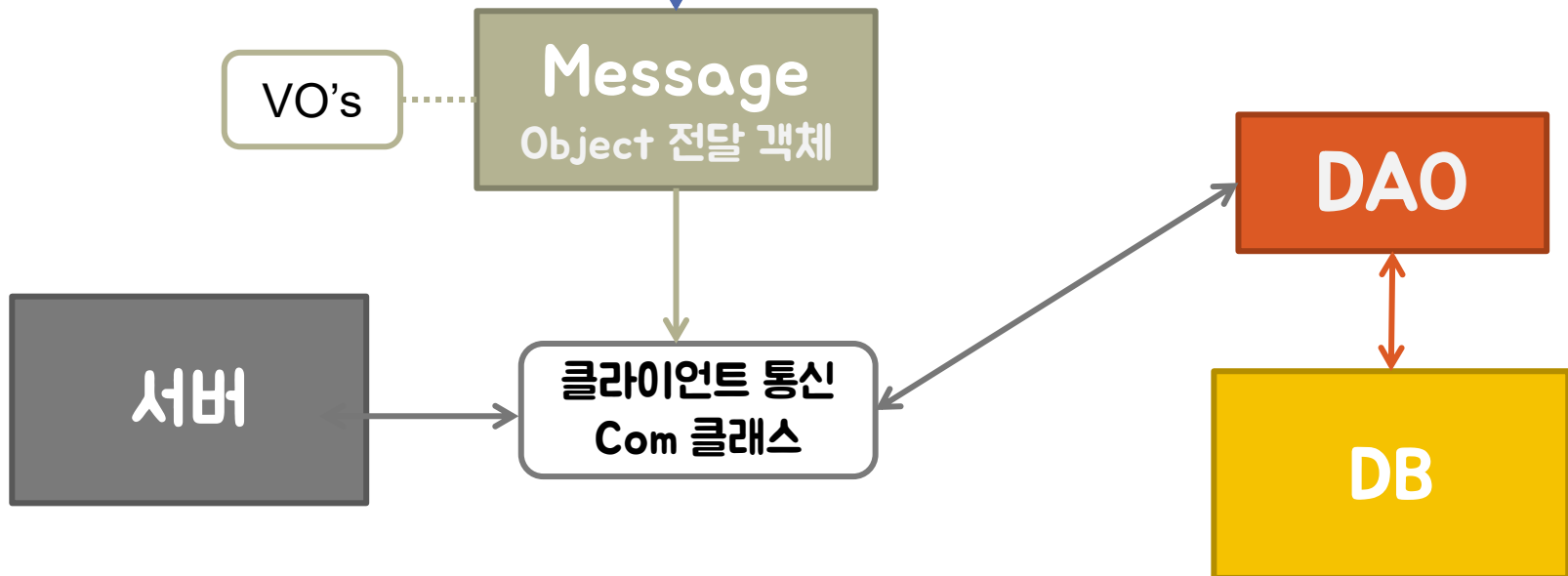
PROGRAM FLOWCHART



주요기능 - 구조도 및 관계도



- ▼ > com.acid.rain.client
 - > AcidRainClient.java
 - > AcidRainClientPanel.java
 - > Player.java
 - > ReaderThreadClient.java
 - > ThreadTest.java
- ▼ > com.acid.rain.dao
 - > AcidRainDAO.java
 - > JDBCUtil.java
- ▼ > com.acid.rain.server
 - > Com.java
 - > DBServer.java
- ▼ > com.acid.rain.vo
 - > AcidRain.java
 - > DrawWord.java
 - > Message.java



핵심구현내용

```
//서버서 단 한개의 DWList를 만들어 각각 com에 전달해준다
public void createDrawWordList(ArrayList<AcidRain> list){
    //리스트 초기화
    dwList = new ArrayList<DrawWord>();
    ArrayList<AcidRain> rList = list;

    int xCoord = 0;
    int yCoord = 0;
    int deltaY = 0;

    for(int i = 0; i < list.size(); i++){
        xCoord = random.nextInt(450);
        yCoord = random.nextInt(600) - 600;
        deltaY = random.nextInt(10) + 15 * currentLevel;

        dwList.add(new DrawWord(xCoord, yCoord,
                                rList.get(i).getWord(), deltaY));
    }
}
```

플레이어 선택에 따라

단어 리스트를 가지며

X좌표, y좌표, delta값을

단어별로 지정하여 리스트에
저장

```
public void sendDrawWordList2All(){
    for(Com cm : comList){
        cm.sendDWList(101, dwList);
    }
}
```

모든 클라이언트에 리스트를
뿌려준다

핵심구현내용

```
public void run(){ //여기서는 내려주는거랑 계속 그려주는거
    while(onAir){
        if(panelState == PANEL_STATE_START_SIGN_FIRED){
            drawWords();

            repaint();
        }
    }
}
```

drawWords()로
deltaY만큼 y 값
변경

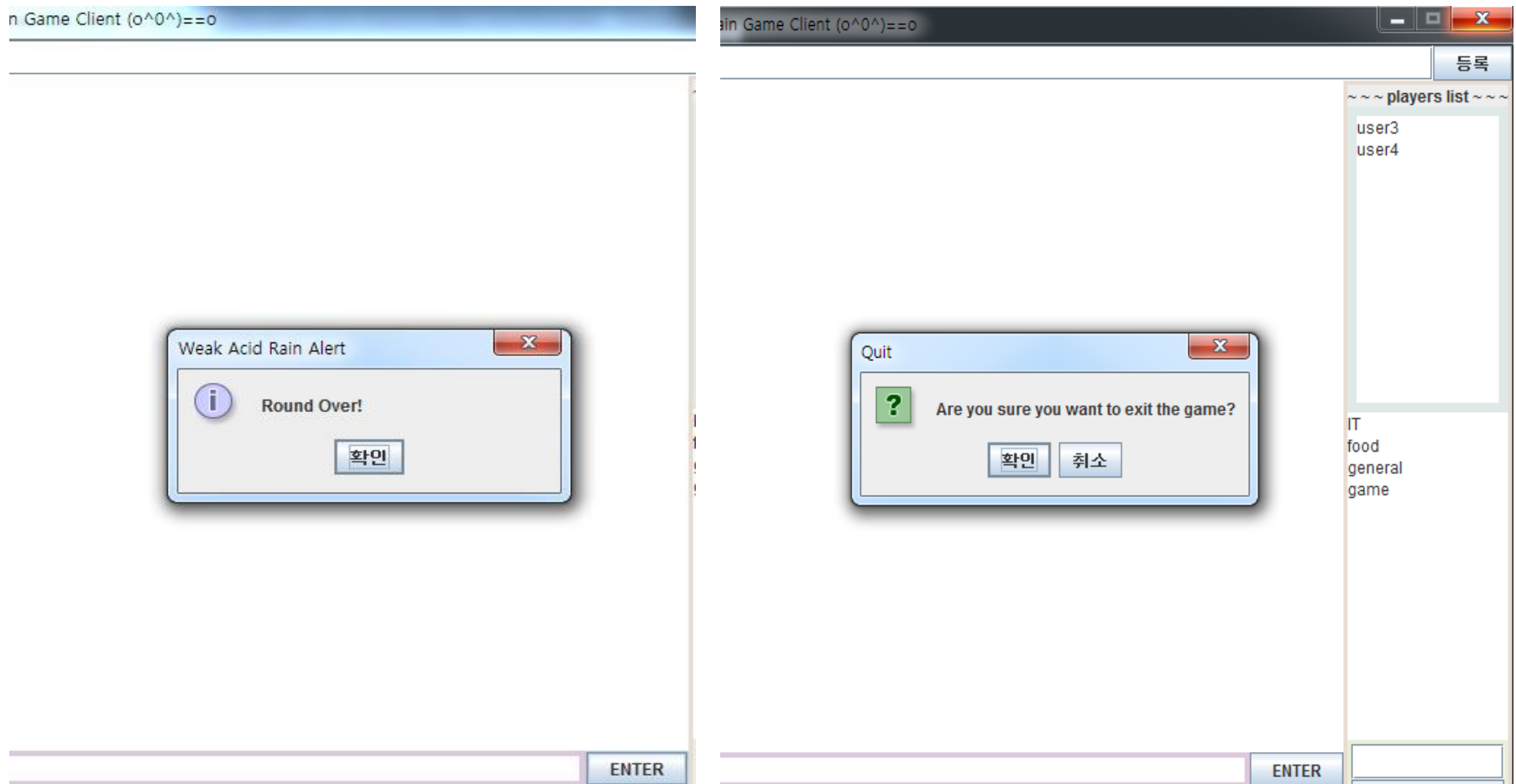
```
// ===== 페 인 트 =====
@Override
public void paint(Graphics g) {
    setGraphicsSetting(g);

    g.setFont(font);
    g.setColor(new Color(200, 170, 220));
    if(dwList != null){
        for(int i = 0; i < dwList.size(); i++){
            g.drawString(dwList.get(i).getText(),
                dwList.get(i).getX(), dwList.get(i).getY());
        }
    }
}
```

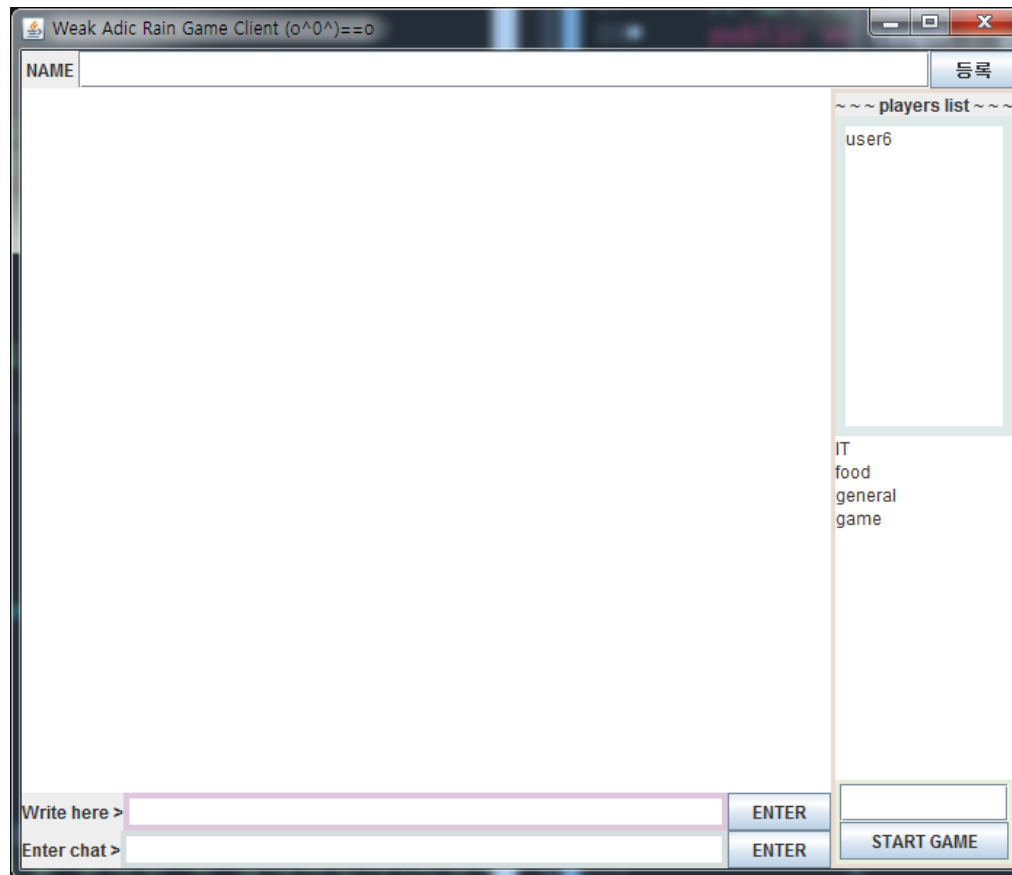
Repaint()로 paint()를 호출하여
단어의 x,y 좌표를 받아 해당 위치에 문자 출력

부가내용

JOptionPane을 이용한 상황별 팝업 출력



시연행사



개발환경



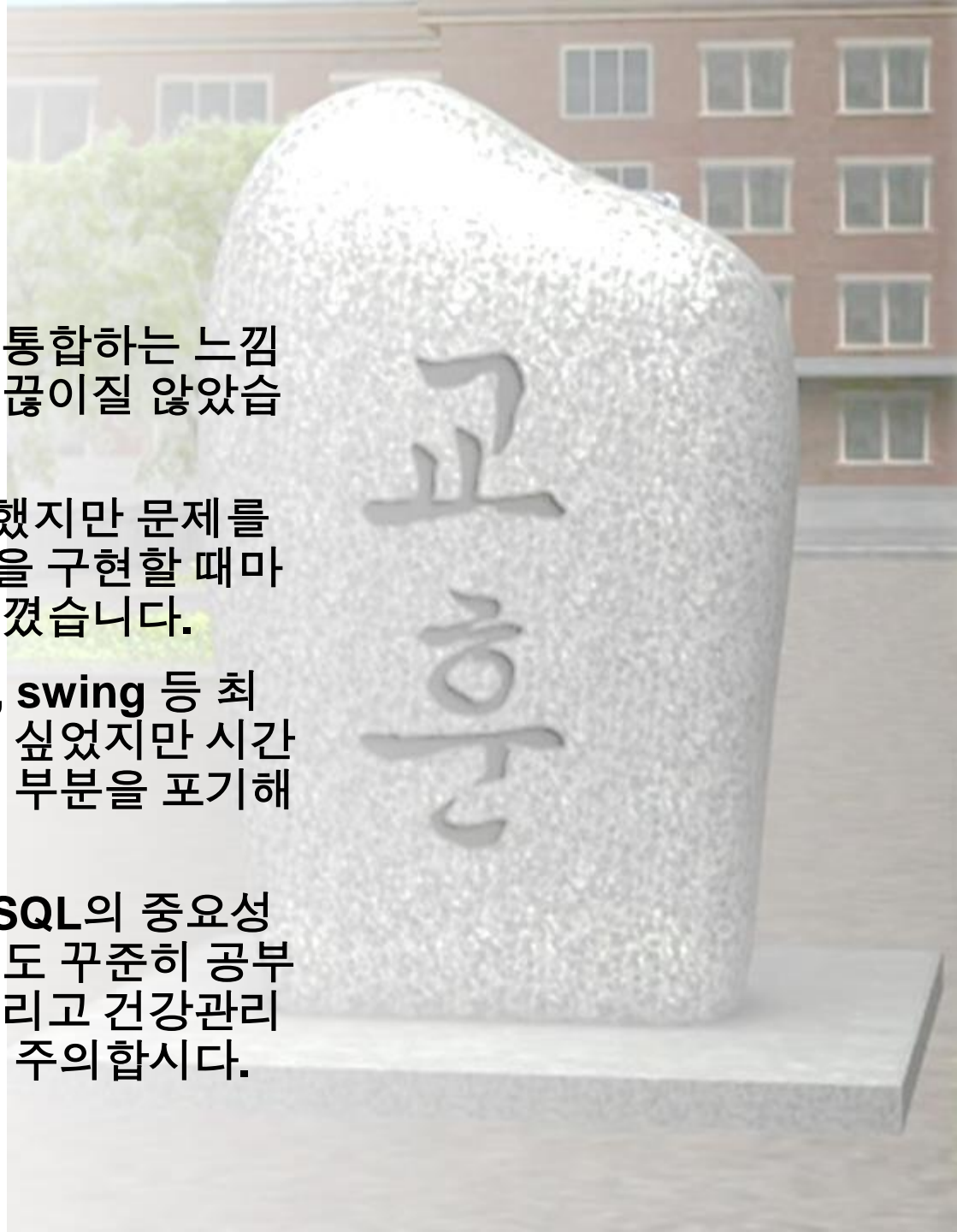
소감

여태까지 배웠던 많은 것들을 통합하는 느낌으로 진행을 하다보니 에러가 끊이질 않았습니다.

건강문제까지 겹쳐 지치기도 했지만 문제를 하나 하나 해결해가며 기능들을 구현할 때마다 그 이상의 카타르시스를 느꼈습니다.

네트워크, **JDBC**연동, 쓰레드, **swing** 등 최대한 많은 것들을 구현해 보고 싶었지만 시간과 실력의 제약으로 인해 많은 부분을 포기해야 했던 것이 아쉽습니다.

프로젝트를 진행하며 자바와 **SQL**의 중요성을 지속적으로 느꼈고, 이후에도 꾸준히 공부를 해야겠다고 느꼈습니다. 그리고 건강관리도 철저히 하여 아프지 않도록 주의합시다.



SPECIAL THANKS TO:

```
case 1:
    System.out.println("select 대장정 5(com 1번에 막 들어옴) typeid:" + msg.g
    System.out.println("select 대장정 5.05 com의 panelstate: " + panelState);
    System.out.println("select 대장정 6(최종 typeid받아옴: )" + typeIdX);
    System.out.println("select 대장정 7(받아온 typeid를 acidrain에 넣어줌)");
    System.out.println("select 대장정 8: selectwords하고 난 tempMsg");
    System.out.println("모두에게 보낼 리스트 만들기 전 tempMsg의 길이: " + temp
    System.out.println("서버가 보낼 리스트의 길이: " + tempMsg.getList().size()
    System.out.println("서버서 모두에게 dwList를 보냈서!");
    System.out.println("새 이름: " + msg.getAcidrain().getUsername());
    System.out.println("현 이름: " + msg.getNameString());
    System.out.println("EntryCheck 대장정 3: 서버 프로토콜 34번국도 타고왔으~");
    System.out.println("dao가기전 새 이름: " + acidrain.getUsername()
        + ", 구 이름: " + oldName);
    System.out.println(msg.getList().size());
    System.out.println("이름바꿨어알~ 새 이름: " + acidrain.getUsername()
        + ", 현 이름: " + oldName);
    System.out.println("nameupdate " + (resultCnt == 0 ? "FAILED!" : "SUCCESS!"));
    System.out.println("readerThread의 selectWords프로토콜타고까지 왔다 1");
    System.out.println("읽었으면 msgType: " + msgType);
    System.out.println("서버에서 dwList를 받았다 1");
    System.out.println("서버에서 dwList를 받았다 2");
    System.out.println(nameList);
    System.out.println("난 nameList를 client로 전달하였다");
    System.out.println("panelState받아온거: " + panelState);
    System.out.println("난 a패널이고 나의 dwList길이는: " + this.dwList
    System.out.println("스타트 사인 주어졌다!");
    if(s.equals(dwList.get(i).getText())){
        dwList.remove(i);
        System.out.println(s + " 입력!");
```

Q&A



감사합니다

