

# 오픈소스 소프트웨어 프로젝트 최종보고서 발표

2020-2-OSSP-HotSource

2016112548 김정률 2016112587 김택원 2017112447 전영인

# 목차

기존 프로젝트 소개

업무 분담 및 일정

개선 사항



# 기존 프로젝트 소개

2020-2-OSSP-HotSource

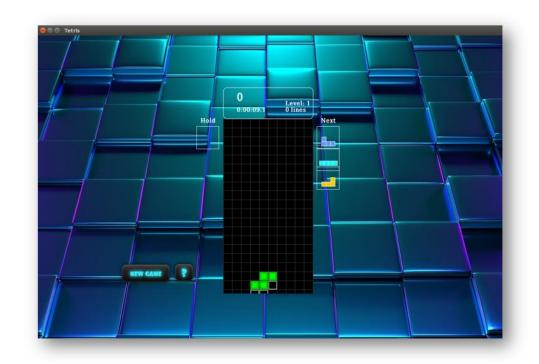


• 원작자 PSNB92

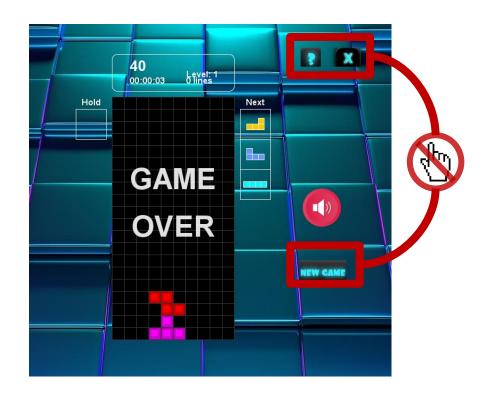
• 2차 저작자 2019년 1학기 PAIS Team팀

• 3차 저작자 2019년 2학기 OS팀

## 기존 프로젝트 문제점



단조로운 플레이 환경



각종 오류 등 미흡한 부분 多

⇒ 프로젝트 주제 선정

## 프로젝트 주요 목표



└ 기존 기능의 개선 □ 게임 종료 후 오류 개선 □ 게임 진행 오류 개선 ┛필요 기능의 개발 □ 시간 단축에 대한 추가 점수 부여 ┙ 게임 시작 전 준비 시간 부여 ┙ 쌓인 블록 구현 - HARD MODE □ 음소거 기능 UX/UI 디자인 개선 ☐FeedBack 반영 ┙ 랭킹시스템 개선 (AWS RDS 활용) Frame Resizing



# 개선 사항(완료)

2020-2-OSSP-HotSource

## 게임 종료 후 오류 개선



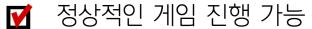
기존 실행화면

게임이 종료 된 후 마우스를 사용할 수 없음

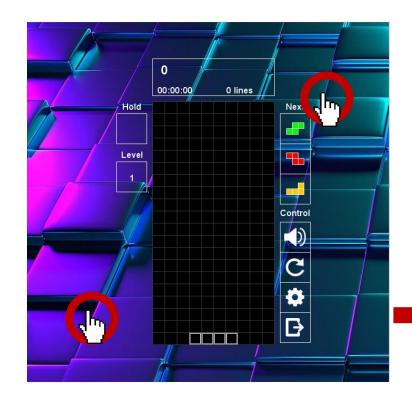


개선된 실행화면

ActionEvent가 제대로 작동

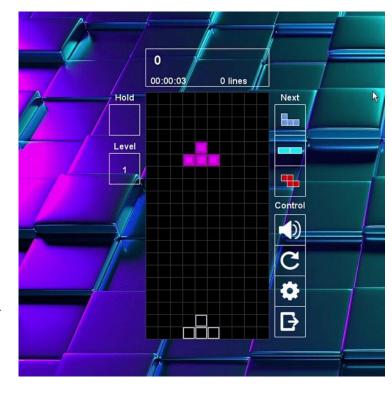


## 게임 진행 오류 개선



background.setBounds(0, 0, 700, 720);
background.setBorderPainted(false);
background.setContentAreaFilled(false);
background.setFocusPainted(false);
background.setVisible(true);
background.setFocusable(false);

코드추가



개선된 실행화면

포커싱 오류 개선

☑ 정상적인 게임 진행 가능

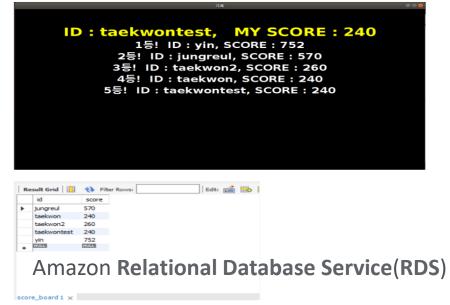
기존 실행화면

배경 클릭시 게임 진행 불가 키보드 먹통

## 최고 점수 갱신

```
DBInsert.java
                 ScoreFrame.java
                                                      Main.java
                                    DBSelect.java
   package com.ok.gamedb;
 3⊕ import java.sql.Connection;
   public class MyDB {
        static Connection conn = null;
        static PreparedStatement pstmt = null;
        static Statement stmt = null;
        static ResultSet rs = null;
 13
 14
        // RDS SETTING
 15
        static String url = "DB URL";
        static String DB ID = "ID";
        static String DB PW = "PASSWORD";
        static String TABLE = "TABLE";
19
```

3. Ranking System with DataBase



• Scoreboards that were only updated locally were synchronized anywhere by using the DataBase.

### AWS RDS 인스턴스 생성

외부에서도 접근할 수 있는 DB 구축

기 존 : Text 파일 사용하여 랭킹시스템 구현 ⇒ 각자 랭킹 현황이 다름

개선 후 : AWS DB 사용하여 랭킹시스템 구현 ⇒ <mark>동일한 랭킹 현황 갱신</mark>

## 최고 점수 갱신

- 데이터베이스를 다루는 새로운 패키지를 구성
- MyDB에는 insert와 select에 사용되는 리소스들을 관리

```
// RDS SETTIME
static String url = "idbc:mysql://horsourcegamedb.rds.amazonaws.com/test?serverTimezo
static String DB_ID = "yin9931";
static String DB_PW = "yin9931";
static String TABLE = "`horsource_gamedb`.`gamedb`";
```

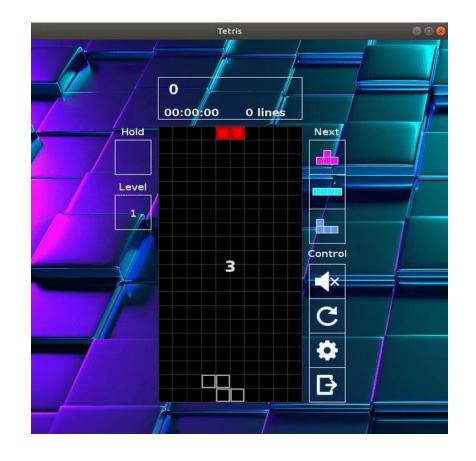
- DBSelect를 통해 유저의 데이터 중 상위 10개를 가져옴
- DBInsert를 통해 데이터를 RDS 서버로 전송

mysql 드라이버 로딩 RDS에 연결하기 SQL 쿼리 수행

☑ 다른 플레이어들과 경쟁하면서 성취감을 느낄 수 있음

## 시간 단축에 대한 추가 점수 부여

```
public void firmDrop()
31
32
           if(1 \le ty \&\& ty < 5)  {
33
                score += (FirmDropPlusScore - (ty * DropDelayMinusScore));
34
            else if(0 >= ty) {
35
                score += FirmDropPlusScore;
37
38
            int oldy = ty;
            while (pieceLegal() == LEGAL)
39
                ty++;
            if (oldy != ty)
                lastMoveRotate = false;
                delays = 0;
47
48
```



## 블록 <u>빠르게 Drop 시 보너스 점수</u> 부여

지연 시간에 따라 최대 20점, 최소 0점

☑ 이용자에게 더 빠른 판단과 스피드를 요구하며 게임의 승부욕을 북돋아줄 수 있음.

## 게임 시작 전 준비시간 부여

#### 카운트 다운 띄우기

```
// Tetris.java 232 line
private int countdown_number = 3;

// Tetris.java 564 line
protected void countdown() {
    countdown_number = 3;

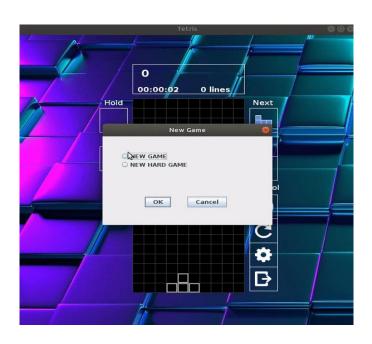
// 게일 말추기
    paused = true; // stop game
    Timer paused_timer = new Timer();
    TimerTask paused_task = new TimerTask() {

        @Override
        public void run() {
            paused = false; // game start!
        }};
        paused_timer.schedule(paused_task, delay*(getcount()+1)); //run after 1200*4ms
```

```
// Tetris.java 1108 line
// 카운트 다운 도중 게임을 종료하는 경우를 고려하 | dead==false인지 확인
// countdown number>0일 때는 countdown number
를 띄우기

else if (countdown_number>0 && dead==false) {
    g.setColor(new Color(0, 0, 0, 80));
    g.fillRect(x, y, FIELD_M, FIELD_H);
    g.setColor(C_NOTICE);
    drawCentered(g, countdown_number+"", x + FIELD_W / 2, y + 5 + FIELD_H / 2);
}

// 카운트 다운 도중 게임을 종료하는 경우를 고려하여 dead==false인지 확인
// countdown_number==0일 때는 GO!값을 띄우기
else if (countdown_number==0 && dead==false) {
    g.setColor(new Color(0, 0, 0, 80));
    g.fillRect(x, y, FIELD_M, FIELD_H);
    g.setColor(C_NOTICE);
    drawCentered(g, "GO!", x + FIELD_M / 2, y + 5 + FIELD_H / 2);
}
```



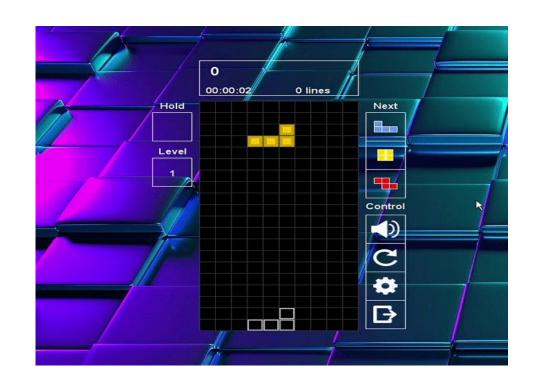
자바의 Timer, TimerTask객체를 이용 게임 구역의 중심에 3, 2, 1 또는 go! 띄우기

☑ 플레이어가 집중력 있게 게임에 임할 수 있는 환경을 제공

## 음소거 기능 추가

```
public void play(){
    clip.setFramePosition(0);
    clip.start();
}
public void stop() {
    clip.stop();
}

private void mute() {
    if(soundplay) {
       bgm.stop();
       soundplay = false;
    }else {
       bgm.play();
       soundplay = true;
    }
}
```

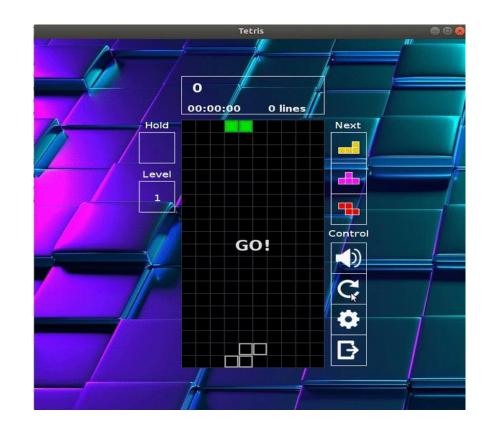


게임 화면에 음소거 버튼 추가 버튼을 누를 시 음소거 가능 버튼을 다시 누를시 BGM 재생

☑ 다양한 플레이 환경을 제공

## 쌓여 있는 블록 - HARD MODE

```
//Already Stacked Block
351
352
            Random rand = new Random();
353
            int mapsize = 4;
354
            int n = rand.nextInt(mapsize);
355
356
            switch(n) {
                case 0: // 낭떠러지
357
358
                    //게임 시작하자마자 쌓여있는 블록 생성
359
                    for(int i=0; i<8; i++) {
360
                        for(int j=12; j<20; j++) {
361
                            board[i][j] = Block EXIST;
362
                    }
363
364
                    break;
365
                case 1: // 협곡
366
                    //게임 시작하자마자 쌓여있는 블록 생성
367
                    for(int i=0; i<5; i++) {
368
                        for(int j=12; j<20; j++) {
369
                            board[i][j] = Block EXIST;
370
371
372
                    for(int i=7; i<10; i++) {
373
                        for(int j=12; j<20; j++) {
374
                            board[i][j] = Block EXIST;
375
376
377
                    break;
```



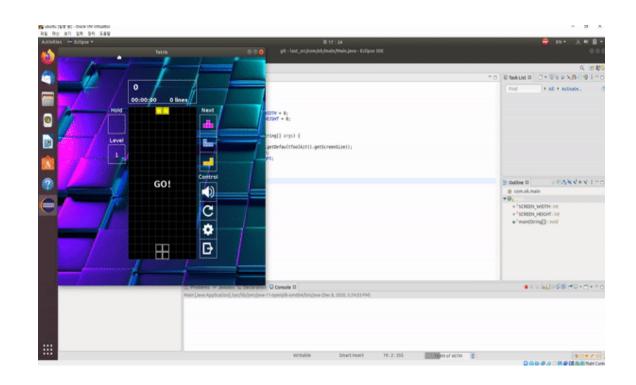
자바의 Random 객체를 이용

HARD MODE 선택 시 플레이어에게 랜덤으로 맵 부여

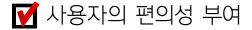
☑ 유저의 플레이 환경의 다양성 부여

## Frame Resizing

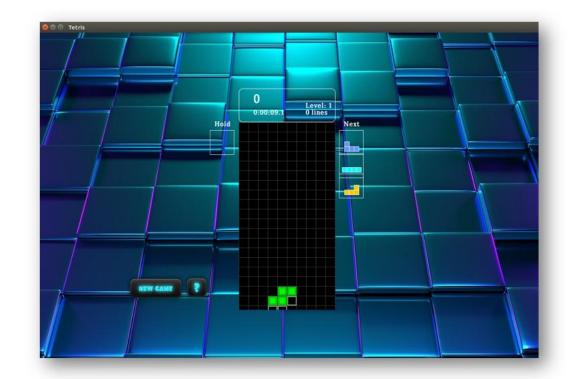
```
game.drawTo((Graphics2D)(g), (int)(frame.getSize().width - game.xoffset*2 -
       game.boxsize*2 - game.FIELD W)/2, (int)(frame.getSize().height/8));
TetrisRenderer.muteButton.setSize(boxsize,boxsize);
TetrisRenderer.muteButton.setLocation(right x, right y);
g.drawRect(right x, right y, boxsize, boxsize);
TetrisRenderer.soundButton.setSize(boxsize,boxsize);
TetrisRenderer.soundButton.setLocation(right x, right y);
g.drawRect(right x, right y, boxsize, boxsize);
TetrisRenderer.newButton.setSize(boxsize,boxsize);
TetrisRenderer.newButton.setLocation(right x, right y +boxsize);
g.drawRect(right x, right y +boxsize, boxsize, boxsize);
TetrisRenderer.keyButton.setSize(boxsize,boxsize);
TetrisRenderer.keyButton.setLocation(right x, right y +boxsize*2);
g.drawRect(right x, right y +boxsize*2, boxsize, boxsize);
TetrisRenderer.homeButton.setSize(boxsize,boxsize);
TetrisRenderer.homeButton.setLocation(right x, right y +boxsize*3);
g.drawRect(right x, right y +boxsize*3, boxsize, boxsize);
```

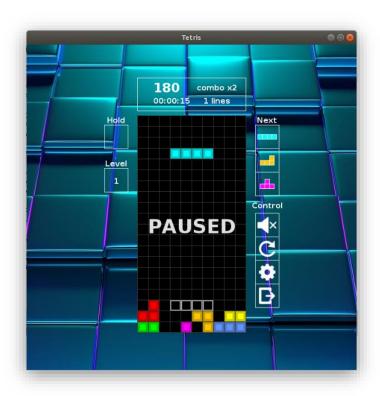


창 크기에 따른 이미지,버튼 위치 수정 사용자의 모니터 크기에 맞게 창 크기 조절 가능



## 배경화면 UX/UI 개선



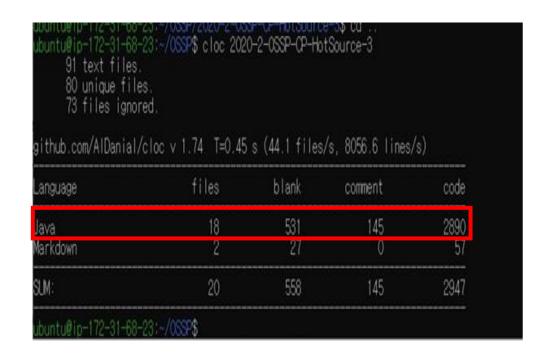


버튼 위치 수정, 버튼 아이콘 수정



☑ 편안한 플레이 환경 구성

## 프로젝트 전후 코드라인 수 비교



| 106 text file:<br>103 unique fi<br>79 files igno | les.<br>ored.        |       |                   | /r\  |
|--|----------------------|-------|-------------------|------|
|  | l/cloc v 1.74 T=0.75 |       | /s, 8019.2 ttiles |      |
| Language   | files                | blank | comment           | code |
| Java   | 20                   | 552   | 238               | 3267 |
| VLIT   | i                    | Ū     | Ū                 | 2123 |
| Markdown   | 2                    | 79    | 0                 | 110  |
| XML  | 6                    | 5     | 0                 | 61   |
| INI  | 1                    | 0     | 0                 | 3    |
| SUM:   | 30                   | 636   | 238               | 5566 |
| ubuntu@ubuntu-Virt                               | ualBox:~/ossp\$      |       |                   |      |

기존 코드의 오류 수정, 여러 기능 추가, 불분명한 숫자 변수정리 등으로 전체 코드 라인 수 증가



# 업무 분담 및 일정

2020-2-OSSP-HotSource

## HotSource 팀 진행 상황

현재

| ACTIVITY               | PLAN START | PLAN<br>DURATION | ACTUAL<br>START | ACTUAL<br>DURATION | PERCENT<br>COMPLETE | PERIODS  9 10 11 12 13 14 15 |  |  |  |  |  | PARTICIPANTS |             |
|------------------------|------------|------------------|-----------------|--------------------|---------------------|------------------------------|--|--|--|--|--|--------------|-------------|
| 프로젝트 주제 선정 및<br>제안서 작성 | 9          | 1                | 9               | 1                  | 100%                |                              |  |  |  |  |  |              | 김택원,김정률,전영인 |
| 게임 오류 사항 수정            | 10         | 2                | 10              | 2                  | 100%                |                              |  |  |  |  |  |              | 김택원,김정률,전영인 |
| 블록 모드 구현               | 12         | 1                | 13              | 2                  | 100%                |                              |  |  |  |  |  | 1            | 김택원         |
| 시작 전 카운트 구현            | 12         | 1                | 12              | 1                  | 100%                |                              |  |  |  |  |  |              | 전영인         |
| 기록 갱신                  | 12         | 1                | 12              | 1                  | 100%                |                              |  |  |  |  |  |              | 김정률         |
| 시간 단축에 따른 추가 점수 부여     | 13         | 1                | 13              | 1                  | 100%                |                              |  |  |  |  |  |              | 김택원,전영인     |
| 배경 음악                  | 13         | 1                | 13              | 1                  | 100%                |                              |  |  |  |  |  |              | 김정률         |
| 오류 사항 수정 및 해결 못한 기능 추가 | 14         | 1                | 14              | 1                  | 100%                |                              |  |  |  |  |  |              | 김택원,김정률,전영인 |
| 검토 및 피드백 / 디버깅         | 15         | 1                | 15              | 1                  | 50%                 |                              |  |  |  |  |  |              | 김택원,김정률,전영인 |

☑ 모든 기능 개선과 오류사항 해결 완료!

☑ 교수님 피드백 반영 완료!

#### 1

## 역할 분담



## 김택원

AWS DB구축 게임 중 오류 사항 수정 보너스 점수 구현 쌓인 블록 모드



## 김정률

음소거 기능 구현 게임 중 오류 사항 수정 Frame Resizing 주석 및 불필요 코드 정리



전영인

시작 시 카운트 다운 구현 게임 중 오류 사항 수정 JDBC 활용한 랭킹 시스템 구축 불필요한 코드 정리 배경화면 UX/UI 개선

# Y Thank You



EQ&A