MOBILE PROGRAMMING

Term Project

1816373 박해민 # 1813756 임연정 # 1814392 홍소현

Index

#0. 기획의도

#1. 클래스 관계도

#2. 기능별 JAVA 코드 설명

#3. 실행결과

#0

기획의도

힐링 게임, 2030세대를 위로하다

지소윤 기자 승인 2018.05.25 10:00 댓글 0

2030세대의 힐링 니즈가 부각되면서 무자극 힐링 게임이 인기를 끌고 있다. 힐링 게임들은 게임 조작이 간단하고, 경쟁이나 어려운 미션 등을 주지 않는다. 방치하거나 그냥 접속하는 것만으로도 진행되는 쉬운 게임들이 이에 속한다.

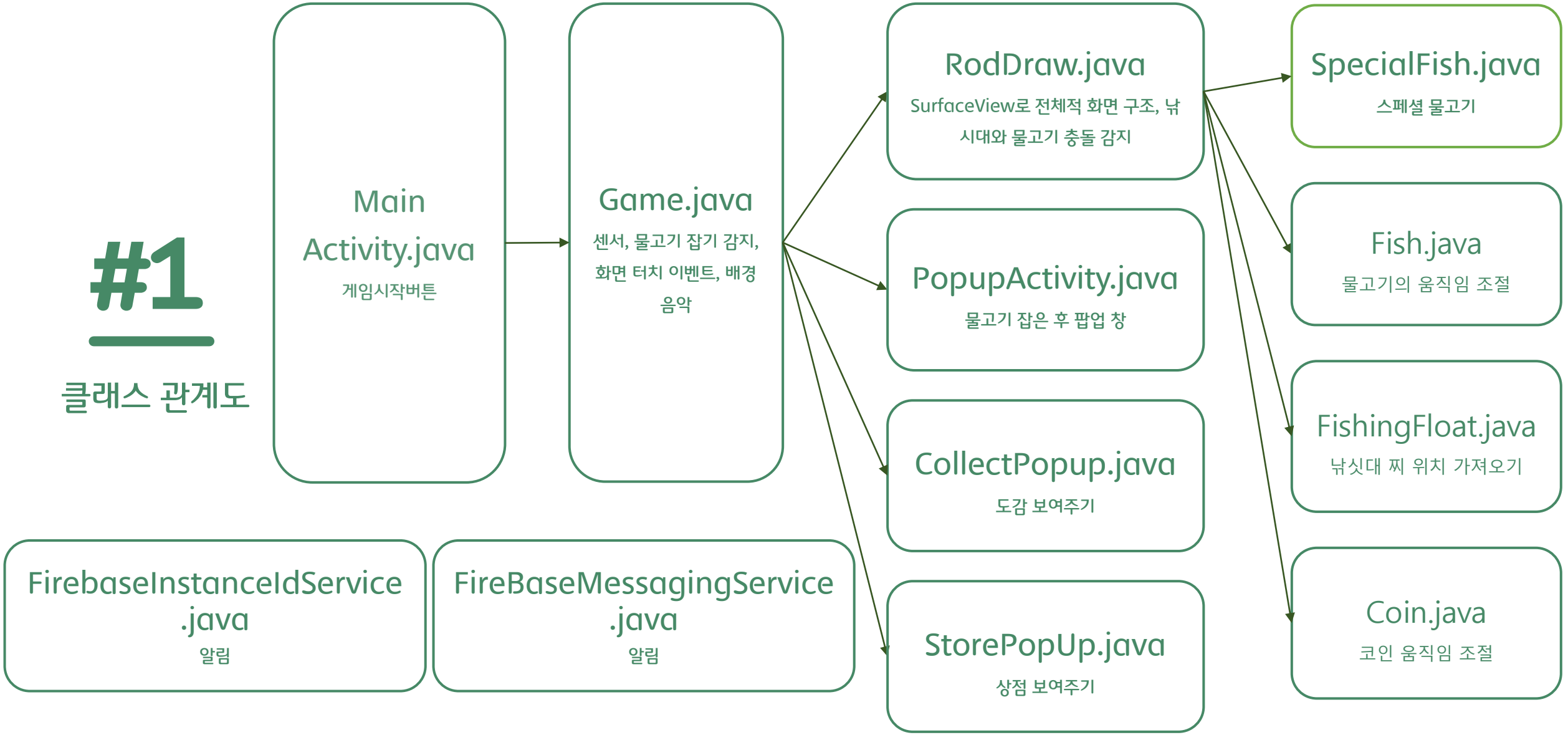
힐링 게임 '마이 오아시스'는 지난 2017년 8월 누적 다운로드 100만 이상을 기록했으며, 2018년 5월 누적 다운로드 수 600만을 돌파했다. 탭을 통해 자신만의 오아시스를 꾸며 나가는 동시에, 용기를 북돋는 갖가지 문구들을 보게 되는 방식이다. 게임 제작자인 김도형 대표는 힐링이라는 핵심 가치를 구현하기 위해 아름다운 그래픽, 감성적인 음악, 그리고 위로의 글로 게임을 구성했다고 밝혔다.

마이 오아시스 [사진 출처=Google play]

“일상 속 힐링이 될 수 있는 간단한 게임! ”

#1

클래스 관계도


Main Activity.java
게임시작버튼
Game.java
센서, 물고기 잡기 감지, 화면 터치 이벤트, 배경 음악
RodDraw.java
SurfaceView로 전체적 화면 구조, 낚시대와 물고기 충돌 감지
PopupActivity.java
물고기 잡은 후 팝업 창
CollectPopup.java
도감 보여주기
StorePopUp.java
상점 보여주기
SpecialFish.java
스페셜 물고기
Fish.java
물고기의 움직임 조절
FishingFloat.java
낚싯대 찌 위치 가져오기
Coin.java
코인 움직임 조절
FirebaseInstanceIdService .java
알림
FireBaseMessagingService .java
알림

#2

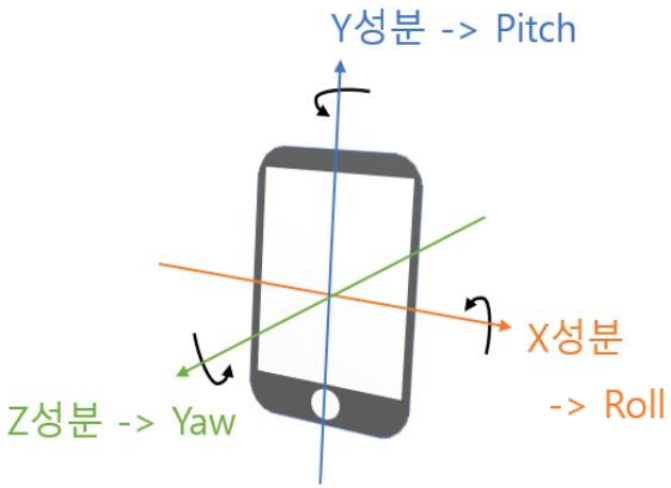
JAVA 코드 설명

Game.java
RodDraw.java

#2

드 설명

Gyroscope Sensor



각속도 측정

Accelerometer Sensor

가속도 측정

Sensor

```
private class Listener implements SensorEventListener {

    @Override
    public void onAccuracyChanged(Sensor sensor, int accuracy) { ; }

    @Override
    public void onSensorChanged(SensorEvent event) {
        if(count==0) {if(event.sensor.getType() != Sensor.TYPE_GYROSCOPE)  return;}
        if(count==1) {if(event.sensor.getType() != Sensor.TYPE_ACCELEROMETER)  return;}

        switch (mDisplay.getRotation()) {
            case Surface.ROTATION_0:
                if(count==0)ss.move1(event.values[DATA_X],
                        event.values[AXIS_Y]);
                else ss.move2(-event.values[DATA_X],
                        event.values[AXIS_Y]);
                break;
            case Surface.ROTATION_90:
                if(count==0)ss.move1(-event.values[DATA_X],
                        event.values[AXIS_Y]);
                else ss.move2(-event.values[DATA_X],
                        event.values[AXIS_Y]);
                break;
            case Surface.ROTATION_270:
                if(count==0)ss.move1(event.values[DATA_X],
                        -event.values[AXIS_Y]);
                else ss.move2(-event.values[DATA_X],
                        -event.values[AXIS_Y]);
                break;
        }
    }
```

Game.java
RodDraw.java

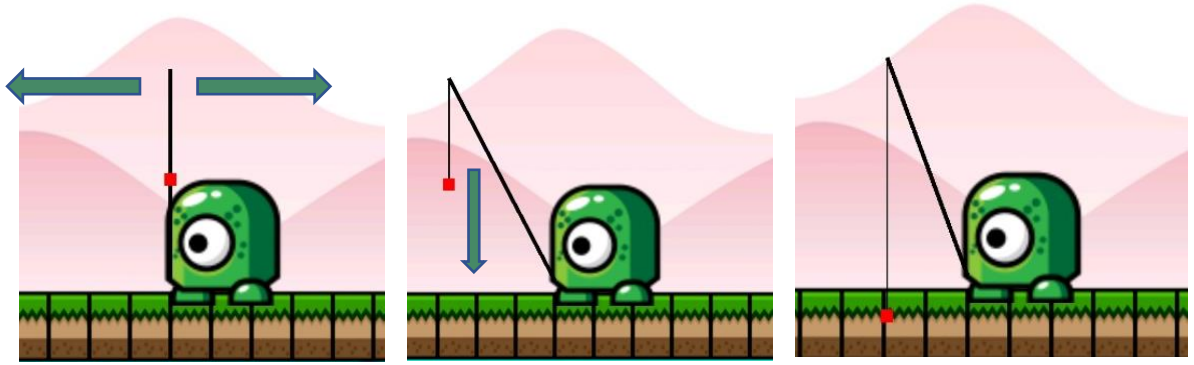
#2

코드 설명

```
paint.setStrokeWidth(3);
canvas.drawLine( startX: Width/2-y,  startY: Height/4,  stopX: Width/2-y,  stopY: Height/3, paint);
canvas.drawLine( startX: Width/2-y,  startY: Height/4,  stopX: Width/2-y,  stopY: Height/3+accY/6*5, paint);
```

처음 화면 터치 다운

Gyro 센서로 낚시대의 위치 조절



```
@Override
public boolean onTouchEvent(MotionEvent event) {
    int action = event.getAction();

    switch(action) {
        case MotionEvent.ACTION_DOWN ://화면을 터치했을때
            if(count==0){
                sm.registerListener(gyroL, gyroSensor, SensorManager.SENSOR_DELAY_FASTEST);// 가속도
                break;
            }
            else if(count==1){
                sm.registerListener(accL, accSensor, SensorManager.SENSOR_DELAY_FASTEST);
                count=2;
                break;
            }
            else
                break;
```

두번째 화면 터치 다운

Accel sensor로 낚시줄 조절.

```
paint.setStrokeWidth(8);
canvas.drawLine( startX: Width/2,  startY: (Height/5)*2, stopX: Width/2-y,  stopY: Height/4, paint);
```

#Fish.java(물고기 동작 원리)

#2

코드 설명

```
public Fish(int screen_width, int screen_height) {
    this.screen_width = screen_width;
    this.screen_height = screen_height;
}
```

스크린 사이즈 가져오기

-> 물고기 움직이는 범위 지정

```
public void MakeFish(int x, int y, int obstacle_num) {

    this.x = x;
    this.y = y;
    this.obstacle_num = obstacle_num;
```

물고기 생성 위치 지정

```
public void ResetSprite() {

    status = ENTER;
    isDead = false; //----------------------------------------
    isWrongObstacle = false;
}
```

셋팅 초기화. 죽은 물고기 감지, 장애물 감지

```
public void EnterSprite() {

    final int v_x, v_y;
    Random r_time = new Random(); //객체생성
    v_x = (r_time.nextInt( bound: 10)+5);
    v_y = (r_time.nextInt( bound: 50)+10);

    if (--delay >= 0) return;// delay time이 끝나지 않

    if (back_x == 0){//왼쪽에서 오른쪽으로 이동
        x += v_x;//속도 랜덤으로 하자
        if (x+w >= screen_width) {
            back_x = 1;
        }
    }else if(back_x == 1){
        x -= v_x;
        if (x <= 0) {
            back_x = 0;
        }
    }
```

물고기의 움직임 조절(위아래/좌우)

SpecialFish.java(스페셜 피시 동작 원리)

#2

코드 설명

```
x += v_x * 5;//속도 랜덤으로 하자
```

```
y -= v_y * 5;//속도 랜덤으로 하자
```

SpecialFish 속도 높이기

RodDraw.java

```
public void MoveAll() {
    if(timer >= 5 && timer <= 8){
        specialFish.Move();
        if (timer == 8){
            timer = 0;
            specialFish.y = Height/9 * 10;
            specialFish.x = r_width;
        }
    }
```

moveAll 함수

– – – · 5~8초 동안 스페셜 피시가 활동

스페셜 피시 출몰 시간 계산 – – –

```
public class CheckTime extends Thread{
    CheckTime(){
    }

    public void run(){
        while(true){
            try {
                // 스레드에게 수행시킬 동작들 구현
                Thread.sleep( millis: 1000); //Thread를 잠
            } catch (InterruptedException e) {
                e.printStackTrace();
            }
            timer ++;//1초마다 증가
        }
    }
}
```

물고기 움직이게 하기!
RodDraw.java

#2
코드 설명

Class GameThread

void DrawAll(Canvas canvas)

```
public void DrawAll(Canvas canvas) {

    for (int i = 0; i < sprite_Thread.length; i++) {
        if (sprite_Thread[i].back_x == 0 && change_inspec[i] == 1) {
            final int r_img_n_1;
            //객체생성
            r_img_n_1 = r_time.nextInt( bound: 8);//이부분을 바꾸면 물고기 이외의 장애물 나옴
            imgfish[i] = BitmapFactory.decodeResource(RodDraw.mContext.getResources(), obstaclesID[r_img_n_1]);

            obstacle_inspec[i] = r_img_n_1;//----------------------------------------------------------------
            change_inspec[i] = 0;
        }
        if (sprite_Thread[i].back_x == 1 && change_inspec[i] == 0)
            change_inspec[i] = 1;
    }
```

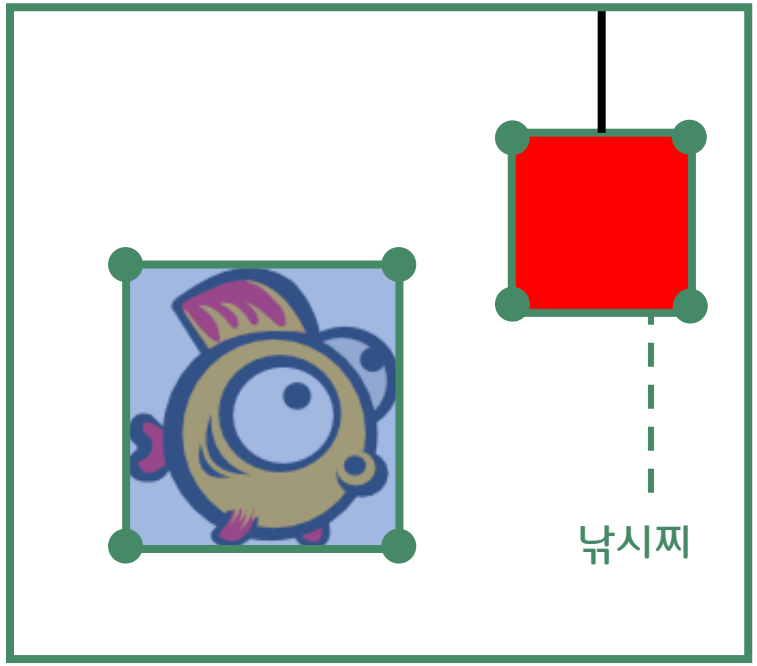
imgfish[i]를 랜덤으로-> 물고기 왕복 시 바꾸기

obstacle_inpsec[i]-> 물고기 종류 저장

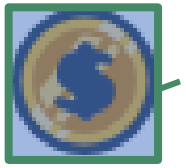
(chang_inspec[i], back_x 확인)-> 순간 확인

RodDraw.java

CheckCollision




```
if (g.x <= fx && g.x + g.w >= fx || fx <= g.x && fx + fw >= g.x) {
    if (g.y <= fy && g.y + g.h >= fy || fy <= g.y && fy + fh >= g.y) {
        if(obstacle_inspec[i] <5){//물고기의 개수에 따라 변경하기----------
            sprite_Thread[i].isDead = true;
            StopThread();
            num=i;
            isResult=true;
        }else{
            sprite_Thread[i].isWrongObstacle = true;//-----------------
            StopThread();
            num=i;
            isResult=true;
        }
    }
}
```



```
if (g.x <= cx && g.x + g.w >= cx || cx <= g.x && cx + cw >= g.x) {
    if (g.y <= cy && g.y + g.h >= cy || cy <= g.y && cy + ch >= g.y) {

            coin_Thread[i].isDead = true;
            coin++;
            StopThread();

            isCoin=true;

    }
}
```

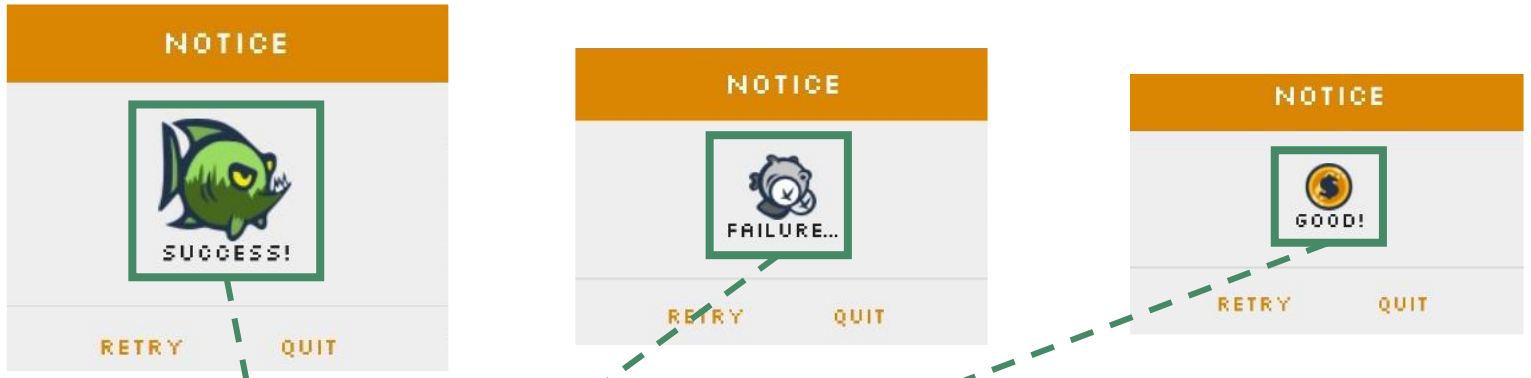
#2

코드 설명

PopupActivity.java

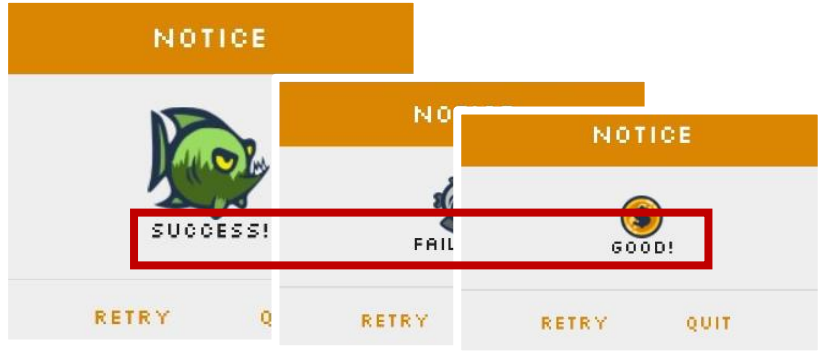
#2

코드 설명




```
Integer[] fishID = {R.drawable.fish1, R.drawable.fish2, R.drawable.fish3, R.drawable.fish4, R.draw
```

```
//데이터 가져오기
Intent intent = getIntent();
String data = intent.getStringExtra( name: "result");
int img_num= intent.getIntExtra( name: "img_num", defaultValue: 0);
txtText.setText(data);
imgView.setImageResource(fishID[img_num]);
```



PopupActivity.java

#2

코드 설명

```
public void retry(View v){

    finish();
    Intent refresh = new Intent( packageContext: this, Game.class);
    startActivity(refresh);

}
```

```
public void mOnClose(View v){

    finish();
}
```

Game.java(음악, 백버튼 스레드 종료)

#2

코드 설명

TimeTask

```
TimerTask tt=new TimerTask() {
    Intent intent = new Intent( packageContext: Game.this, PopupActivity.class);

    @Override
    public void run() {

        if(ss.isResult)
        {

            if(ss.obstacle_inspec[ss.num]>=5&&ss.obstacle_inspec[ss.num]<8) {
                mediaPlayer_bg.stop();
                intent.putExtra( name: "result", value: "FAILURE...");
                mediaPlayer_ob_caught = MediaPlayer.create( context: Game.this, R.raw.fail);
                mediaPlayer_ob_caught.start();
```

Obstacle_inspec 사용!

성공, 실패 음악, 배경음악 멈추기

Class BackPressCloseHandler

```
public void onBackPressed(){
    if(System.currentTimeMillis() > backKeyClickTime + 2000){
        backKeyClickTime = System.currentTimeMillis();
        showToast();
        return;
    }
if (System.currentTimeMillis() <= backKeyClickTime + 2000){
```

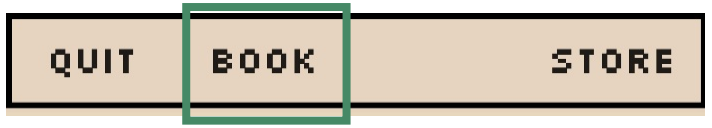
한 번 클릭 시 토스트

```
    mediaPlayer_bg.stop();
    ss.StopGame();
```

두 번 클릭 시 게임스레드, 배경음악 종료

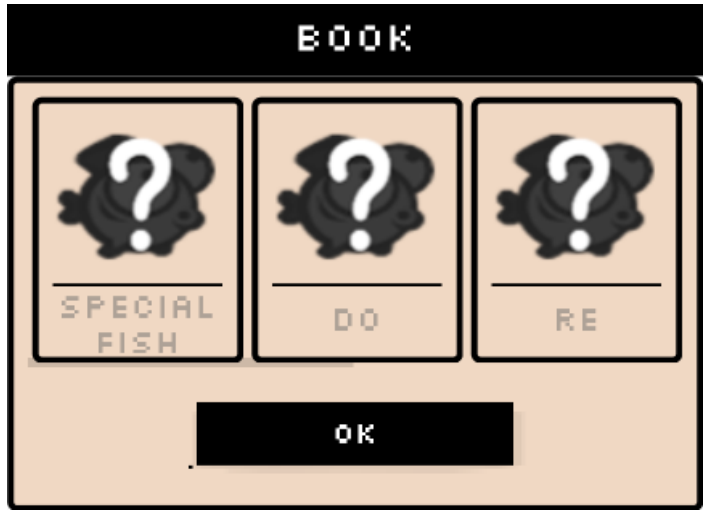
CollectPopup.java

Book



#2

코드 설명



SharedPreference로 데이터 저장

물고기를 잡으면 도감에 물고기 표시

```
intent.putExtra( name: "img_num", ss.obstacle_inspec[ss.num]);
startActivity(intent);
ss.isResult=false;
SharedPreferences sharedPreferences = getSharedPreferences( name: "sFile2", MODE_PRIVATE);
//저장을 하기위해 editor를 이용하여 값을 저장시켜준다.
SharedPreferences.Editor editor = sharedPreferences.edit();
editor.putInt( s: "img"+ss.obstacle_inspec[ss.num],ss.obstacle_inspec[ss.num]); // key, value를 이용하여 저장하는 형태
//최종 커밋
editor.commit();
finish();
```

```
for(int i=0;i<ss.obstacle_inspec.length;i++){
    num[i]= sf.getInt( s: "img"+i,  i: 6);
    intent.putExtra( name: "img"+i,num[i]);
```

```
for(int i=0; i<fishID.length;i++) {
    num[i] = intent.getIntExtra( name: "img" + i,  defaultValue: 0);
}
for(int i=0; i<fishID.length;i++) {
    if(num[i]<5){
            ImageView imgv = (ImageView) findViewById(layoutID[num[i]]);
            imgv.setImageResource(fishID[num[i]]);
            TextView txtText = (TextView) findViewById(TextID[num[i]]);
            txtText.setTextColor(BLACK)
        }
```

```
else if(num[i]==8){
TextView txtText1=(TextView) findViewById(R.id.sfish_name);
ImageView img=(ImageView)findViewById(R.id.sfish_image);
img.setImageResource(fishID[8]);
txtText1.setTextColor(BLACK);
txtText1.setText("#");
```

StorePopup.java

Store

#2

코드 설명

BUY 버튼을 누르면 코인을 차감해 캐릭터 구매

```
final Button b2=(Button)findViewById(R.id.character2);

final LinearLayout char1=(LinearLayout) findViewById(R.id.char1);
final LinearLayout char2=(LinearLayout)findViewById(R.id.char2);


b2.setOnClickListener((view) → {
        if(buttonClick[1]==0){
            if(coin<=0) Toast.makeText(getApplicationContext(), text: "코인이 부족합니다.", Toast.LENGTH_LONG).show();
            else{
                buttonClick[1]=1;
                coin--;
                Toast.makeText(getApplicationContext(), text: "캐릭터를 구매하였습니다.", Toast.LENGTH_LONG).show();
                b2.setText("SOLD");}}
        else  Toast.makeText(getApplicationContext(), text: "이미 구매하셨습니다.", Toast.LENGTH_LONG).show();
});
```

StorePopup.java

Store

#2

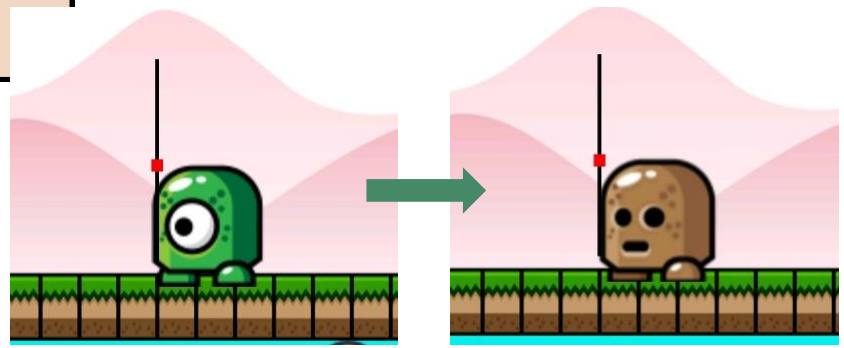
코드 설명

구매한 캐릭터를 선택하고 OK버튼을 누르면 캐릭터가 바뀐다.

```
final Button b2=(Button)findViewById(R.id.character2);

final LinearLayout char1=(LinearLayout) findViewById(R.id.char1);
final LinearLayout char2=(LinearLayout)findViewById(R.id.char2);

char2.setOnClickListener((view) → {
        if(buttonClick[1]==0)
            Toast.makeText(getApplicationContext(),  text: "캐릭터를 구매해야합니다.", Toast.LENGTH_LONG).show();
        else{
            character=1;
            char2.setBackgroundTintList(ColorStateList.valueOf(Color.parseColor( colorString: "#FOD8C3")));
            char1.setBackgroundTintList(ColorStateList.valueOf(Color.parseColor( colorString: "#FFFFFF")));
    }});
```



FirebaseInstanceIDService.java
FireBaseMessagingService.java

Firebase Cloud Messaging

#2

코드 설명

```
<service
    android:name=".FireBaseMessagingService">
    <intent-filter>
        <action android:name="com.google.firebase.MESSAGING_EVENT"/>
    </intent-filter>
</service>


<service
    android:name=".FirebaseInstanceIDService">
    <intent-filter>
        <action android:name="com.google.firebase.INSTANCE_ID_EVENT"/>
    </intent-filter>
</service>
```

Manifest.xml 에 <service> 추가

onMessageReceived

```
@Override
public void onMessageReceived(RemoteMessage remoteMessage) {
```

onMessagedReceived 메서드를 재정의하면
수신된 RemoteMessage 객체를 기준으로 작업
을 수행 및 메시지 데이터를 가져옴

sendNotification

```
private void sendNotification(String messageBody) {
```

알림 채널 설정
(안드로이드 8.0 이상에서는 알림 채널 설정 필요)

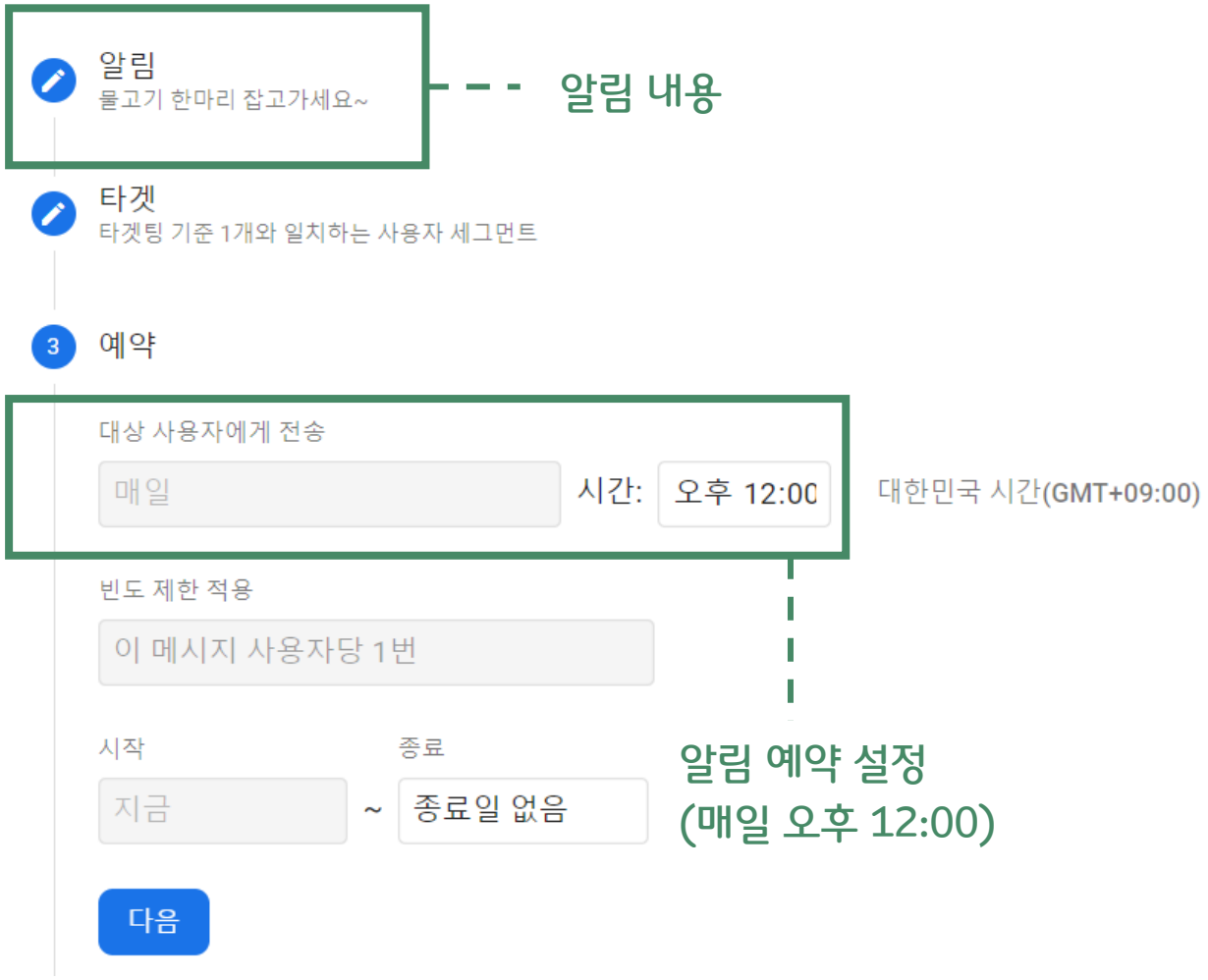
FirebaseInstanceIDService.java
FireBaseMessagingService.java

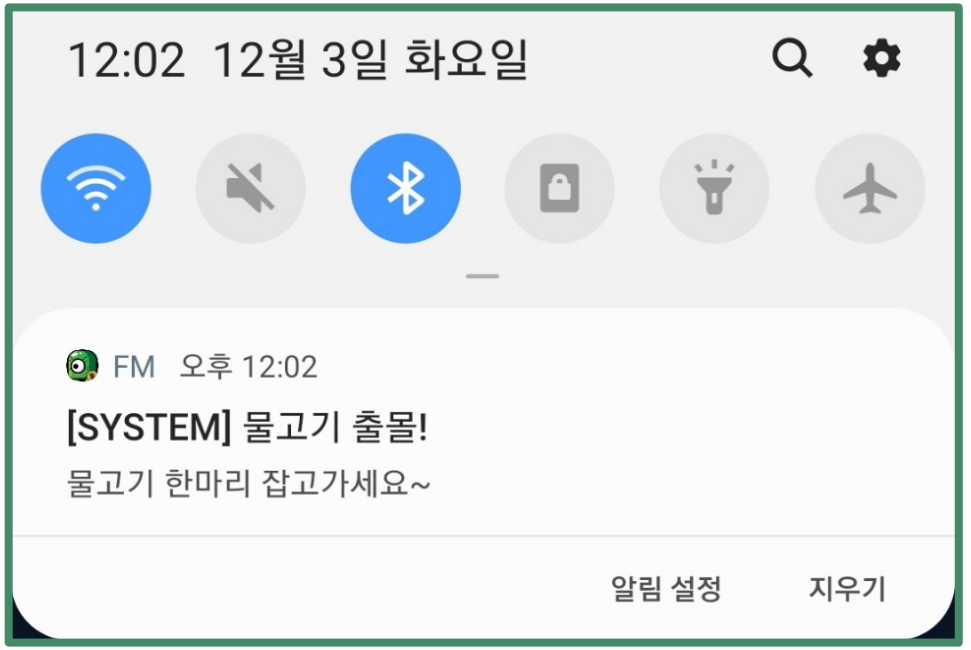
Firebase Cloud Messaging

#2

코드 설명

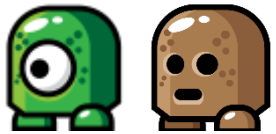
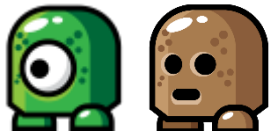
서버에서 푸시 알림 보내기





#3

실행 결과



THANKS