package cn.com.cyj.mouse.services;

import java.util.HashMap;

import java.util.Random;

import android.content.Intent;

import android.os.Bundle;

import android.os.Handler;

import android.view.View;

import android.view.View.OnClickListener;

import android.widget.Button;

import android.widget.ImageButton;

import android.widget.TextView;

import cn.com.cyj.mouse.R;

import cn.com.cyj.mouse.ui.BaseActivity;

import cn.com.cyj.mouse.ui.MouseStart;

/\*\*

\* 游戏开始的界面：游戏中有12个ImageButton,每个ImageButton背景设置成地鼠洞,游戏中开启一个线程控制游戏时间

\*

\* @author cyj

\*

\*/

public class GameRun extends BaseActivity {

/\*\*

\* 线程睡眠时间

\*/

public static final int THREAD\_SLEEP\_TIME = 500;

/\*\*

\* 游戏时间

\*/

public static final int TIME = 30;

private ImageButton one;

private ImageButton two;

private ImageButton three;

private ImageButton four;

private ImageButton five;

private ImageButton six;

private ImageButton seven;

private ImageButton eight;

private ImageButton nine;

private ImageButton ten;

private ImageButton eleven;

private ImageButton twleve;

// 显示时间

private TextView showTime;

// 显示分数

private TextView score;

MyClick click;

private Random random;

// 游戏当前时间

private int time;

// 游戏总时间

private int totalTime;

// 老鼠下一次出现位置

private int next;

// 游戏当前分数

private int nowScore;

// 游戏线程

private Thread t;

// 存放按钮和next的映射

HashMap<ImageButton, Integer> battle;

HashMap<Integer, ImageButton> nextMap;

public Handler handler = new Handler() {

public void handleMessage(android.os.Message msg) {

changeUI();

};

};

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_gamerun);

battle = new HashMap<ImageButton, Integer>();

nextMap = new HashMap<Integer, ImageButton>();

initImageButton();

initOnClick();

initbattleMap();

initNextMap();

next = -1;

random = new Random();

totalTime = TIME;

time = 0;

nowScore = 0;

showTime.setText(TIME + "");

}

@Override

protected void onResume() {

super.onResume();

if(t == null){

// 控制游戏时间

t = new Thread(new Runnable() {

@Override

public void run() {

try {

while (totalTime != 0) {

Thread.sleep(THREAD\_SLEEP\_TIME);

next = random.nextInt(12) + 1;

time++;

handler.sendEmptyMessage(1);

}

} catch (Exception e) {

e.printStackTrace();

}

if (totalTime == 0) {

Intent intent = new Intent(GameRun.this, GameOver.class);

// 该参数跳转页面不会触发onUserLeaveHint()方法

intent.addFlags(Intent.FLAG\_ACTIVITY\_NO\_USER\_ACTION);

intent.putExtra("score", "" + nowScore);

GameRun.this.startActivity(intent);

finish();

}

}

});

}

t.start();

}

// 初始化按钮

private void initImageButton() {

one = (ImageButton) findViewById(R.id.first);

two = (ImageButton) findViewById(R.id.second);

three = (ImageButton) findViewById(R.id.three);

four = (ImageButton) findViewById(R.id.four);

five = (ImageButton) findViewById(R.id.five);

six = (ImageButton) findViewById(R.id.six);

seven = (ImageButton) findViewById(R.id.seven);

eight = (ImageButton) findViewById(R.id.eight);

nine = (ImageButton) findViewById(R.id.nine);

ten = (ImageButton) findViewById(R.id.ten);

eleven = (ImageButton) findViewById(R.id.eleven);

twleve = (ImageButton) findViewById(R.id.twelve);

showTime = (TextView) findViewById(R.id.showtime);

score = (TextView) findViewById(R.id.score);

}

// 给按钮添加点击事件

private void initOnClick() {

click = new MyClick();

one.setOnClickListener(click);

two.setOnClickListener(click);

three.setOnClickListener(click);

four.setOnClickListener(click);

five.setOnClickListener(click);

six.setOnClickListener(click);

seven.setOnClickListener(click);

eight.setOnClickListener(click);

nine.setOnClickListener(click);

ten.setOnClickListener(click);

eleven.setOnClickListener(click);

twleve.setOnClickListener(click);

}

// 按钮id和next映射关系

private void initbattleMap() {

battle.put(one, 1);

battle.put(two, 2);

battle.put(three, 3);

battle.put(four, 4);

battle.put(five, 5);

battle.put(six, 6);

battle.put(seven, 7);

battle.put(eight, 8);

battle.put(nine, 9);

battle.put(ten, 10);

battle.put(eleven, 11);

battle.put(twleve, 12);

}

// next和按钮id的映射关系

private void initNextMap() {

nextMap.put(1, one);

nextMap.put(2, two);

nextMap.put(3, three);

nextMap.put(4, four);

nextMap.put(5, five);

nextMap.put(6, six);

nextMap.put(7, seven);

nextMap.put(8, eight);

nextMap.put(9, nine);

nextMap.put(10, ten);

nextMap.put(11, eleven);

nextMap.put(12, twleve);

}

/\*\*

\* 更新小老鼠出现位置和显示游戏剩余时间

\*/

private void changeUI() {

// 更新显示剩余时间，游戏0.5s更新一次，因此更新次数是游戏时间的二倍

if (time % 2 == 0) {

showTime.setText(--totalTime + "");

}

if (next == -1)

return;

// 每次出地鼠时将按钮背景初始化

reImageButton();

// 获得next对应的按钮

ImageButton bt = nextMap.get(next);

// 给按钮设置地鼠图片

bt.setBackgroundResource(R.drawable.end);

}

// 按钮背景初始化

private void reImageButton() {

one.setBackgroundResource(R.drawable.start);

two.setBackgroundResource(R.drawable.start);

three.setBackgroundResource(R.drawable.start);

four.setBackgroundResource(R.drawable.start);

five.setBackgroundResource(R.drawable.start);

six.setBackgroundResource(R.drawable.start);

seven.setBackgroundResource(R.drawable.start);

eight.setBackgroundResource(R.drawable.start);

nine.setBackgroundResource(R.drawable.start);

ten.setBackgroundResource(R.drawable.start);

eleven.setBackgroundResource(R.drawable.start);

twleve.setBackgroundResource(R.drawable.start);

}

/\*\*

\* 点击事件，判断是否打中

\*

\* @author cyj

\*

\*/

class MyClick implements OnClickListener {

@Override

public void onClick(View v) {

// 是否的分的标记

Boolean isScore = false;

// 获取点击按钮对应next

int battleId = battle.get(v);

// 如果点击按钮为next得分

if (battleId == next) {

// 得分为true

isScore = true;

}

if (isScore) {

// 设置打中的图片

v.setBackgroundResource(R.drawable.zhong);

if (MouseStart.controller.isPlay()) {

// 打中的音效

MouseStart.controller.playSound(R.raw.hathit);

}

// 加分

score.setText((nowScore += 10) + "");

} else {

// 设置没打中的图片

v.setBackgroundResource(R.drawable.meizhong);

if (MouseStart.controller.isPlay()) {

// 没打中的音效

MouseStart.controller.playSound(R.raw.dismistake);

}

}

}

}

@Override

protected void onStop() {

// 停止线程

super.onStop();

if (t != null) {

t.interrupt();

t = null;

}

}