

# 作业 5

——我的日记本 App（背景音乐）

软件名称: \_\_\_\_\_ MyDiary2.0 \_\_\_\_\_

完成人: \_\_\_\_\_ 唐福梅 \_\_\_\_\_

学号: \_\_\_\_\_ 20152100025 \_\_\_\_\_

班级: \_\_\_\_\_ 15 级软件 5 班 \_\_\_\_\_

完成时间: \_\_\_\_\_ 2017-12-12 \_\_\_\_\_

## 一、软件内容简介

在上次“我的日记本”APP的基础上，使用 service 的方法播放背景音乐。使用 bindService 方式启动 service。

将 service 绑定在 MainActivity 上，点击 menu 上上的“音乐”会将 MainActivity 的布局换成“activity\_music”，从而进入到音乐播放的界面。

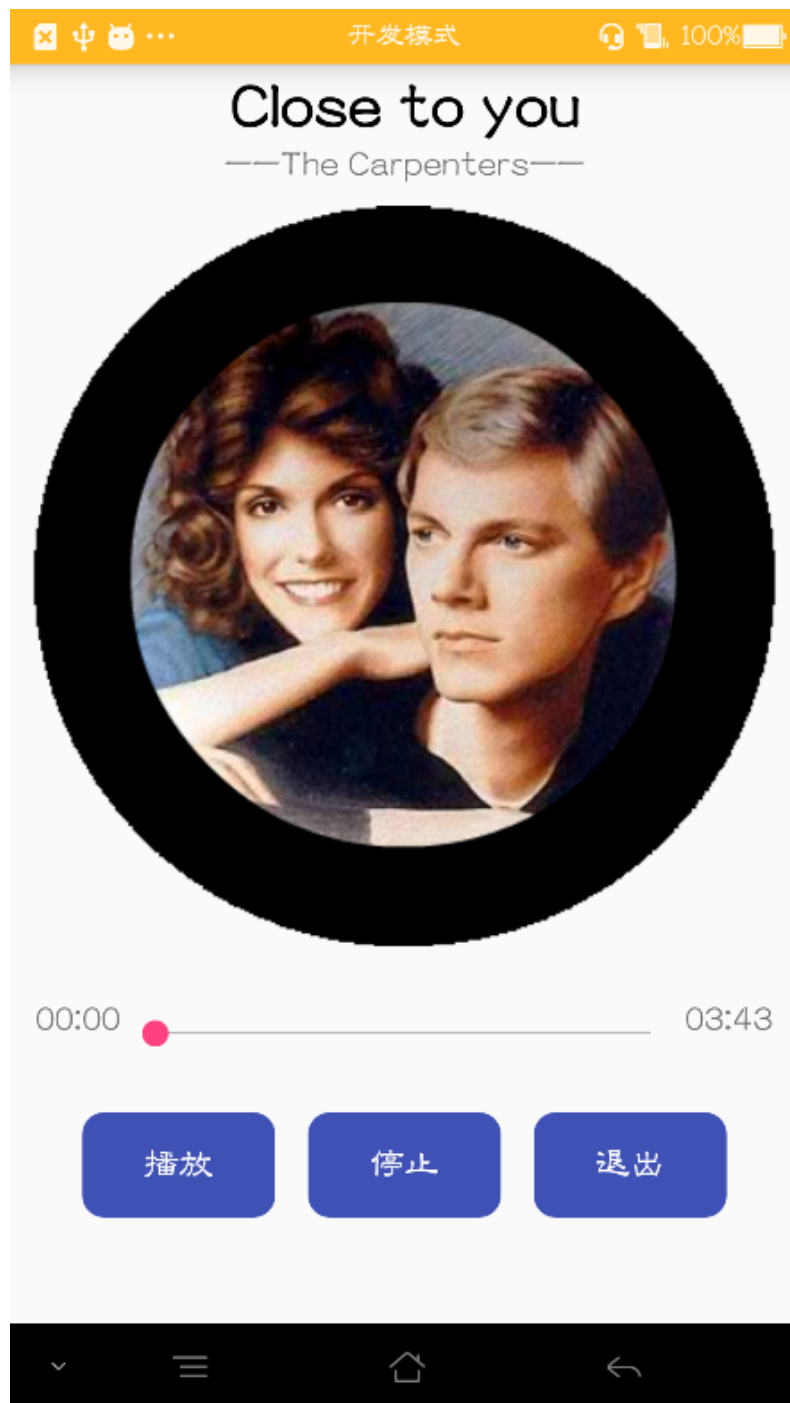
播放的音乐是程序中已经写好的（Close to you），并设置为单曲循环。在播放界面可以看到音乐播放的进度，可以播放/暂停、停止音乐，点击退出返回到日志列表（将 MainActivity 的布局换成“activity\_main”），此时音乐就会在后台播放。

## 二、界面设计

1. 在 menu 菜单中加入音乐 item



## 2. 播放音乐的界面



## 三、代码设计

### (一) 开发步骤

(1) 按课本的代码实现音乐播放器（这个时候没有加到 **MyDiary** 项目中）。

- 1) 将播放、暂停放到 menu 菜单中
- 2) 通过多级菜单来选择可以播放的歌曲，而不是让用户自己输入

#### 【缺点】

- 1) 根据多级菜单来选择播放的歌曲很不方便。
- 2) 不能在播着一首歌的时候，播放另外一首歌（逻辑没有设计好）
- 3) 没有办法循环播放

#### (2) 在 MyDiary 项目中新建了一个 **MusicActivity** 用来绑定 **Service** 播放音乐。

- 1) 播放音乐的界面放在 **activity\_music** 布局文件中（支持播放/暂停、停止操作）
- 2) 像网易云音乐的播放界面一样，将（歌手的）图片设置成随着音乐播放一直旋转。
- 3) 有进度条显示音乐的播放进度

#### 【缺点】

按了退出键就解除绑定服务，所以一返回到日志列表，音乐就停止了。不能做到运行软件的整个过程中一直播放背景音乐。

#### (3) 将在 **MusicActivity** 中实现的代码放到 **MainActivity** 中。

### (二) 具体代码设计

(1) 创建一个 **Service** 类，指定名称为 **MusicService**，该类用于实现音乐的播放和暂停的功能

```
//获取默认播放歌曲
public String getPath(){
    String pathway="Music/a.mp3";
    File SDpath= Environment.getExternalStorageDirectory();
    File file=new File(SDpath,pathway);
    return file.getAbsolutePath();
}

//构造函数
public MusicService() {
    mediaPlayer = new MediaPlayer();
    try {
        mediaPlayer.setDataSource(getPath()); //设置默认播放的歌曲
        mediaPlayer.prepare();
        mediaPlayer.setLooping(true); //设置为循环播放
    } catch (Exception e) {
        e.printStackTrace();
    }
}
```

```

//创建服务的代理，其实例化的对象作为onBind的返回值
public class MyBinder extends Binder {
    MusicService getService() {
        return MusicService.this;
    }
}

```

```

//播放、暂停键
public void playOrPause() {
    if (mediaPlayer.isPlaying()) {
        mediaPlayer.pause(); //当前处于播放状态则按下暂停
    } else {
        mediaPlayer.start(); //当前处于暂停状态则按下播放
    }
}

```

```

//停止键
public void stop() {
    if (mediaPlayer != null) {
        mediaPlayer.stop();
        try {
            mediaPlayer.reset(); //调用reset使mediaplayer类重新实例化一次
            mediaPlayer.setDataSource(getPath());
            mediaPlayer.prepare();
            mediaPlayer.seekTo(0);
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}

```

## (2) 编写界面交互代码

### 1) 进度条的实现

定义 Handler: run 函数中进行更新 seekbar 的进度在类中定义简单日期格式，用来显示播放的时间，用 time.format 来格式所需要的数据，用来监听进度条的滑动变化：

```

private SimpleDateFormat time = new SimpleDateFormat("mm:ss");

```

```
// 通过 Handler 更新 UI 上的组件状态
public Handler handler = new Handler();
public Runnable runnable = new Runnable() {
    @Override
    public void run() {
        musicTime.setText(time.format(musicService.mediaPlayer.getCurrentPosition()));

        //使用 Seekbar 显示播放进度，设置当前值与最大值
        seekBar.setProgress(musicService.mediaPlayer.getCurrentPosition());
        seekBar.setMax(musicService.mediaPlayer.getDuration());
        musicTotal.setText(time.format(musicService.mediaPlayer.getDuration()));
        handler.postDelayed(runnable, 200);
    }
};
```

## 2) 音乐播放中，实现图片旋转以及其实时更新

为了完成图片旋转以及其实时更新，我使用了 animator 属性。首先实例化了一个 ObjectAnimator，然后设置所需的参数：imageView：需要更改的 View，动画类型，动画范围，并在对应的点击事件中引用 animator 中的一些事件，实现动画的点击动作。

```
imageView = (ImageView) findViewById(R.id.Image);
final ObjectAnimator animator = ObjectAnimator.ofFloat(imageView, "rotation", 0f, 360.0f);
animator.setDuration(10000);
animator.setInterpolator(new LinearInterpolator());
animator.setRepeatCount(-1);
```

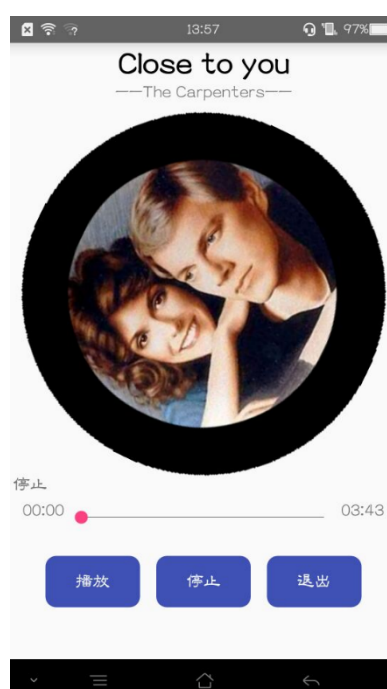
## 四、软件操作流程

### （一）流程说明

- (1) 点击 menu 上的音乐进入播放界面
- (2) 在播放界面上选择相应操作（播放/暂停、停止，退出）

### （二）测试截图

播放与停止



## 五、难点（或遇到的问题）和解决方案

### 【问题 1】

因为书上说 `Environment.getExternalStorageDirectory()` 返回的手机 SD 卡的目录，所以将背景音乐放到 SD 卡对应目录中了，结果点击播放总提示文件不存在。

### 【解决 1】

步骤（1）在用来测试的真机（android4.0 版本）上 `Environment.getExternalStorageDirectory()` 返回的是手机内部存储目录，当把音乐放到手机存储对应位置就好。

步骤（2）在清单文件中添加 SD 卡的读写权限

### 【问题 2】

退出 activity（没有按停止键）后再次打开软件音乐还在播放但是音乐状态已经回到初始状态。

【解决 2】在 MainActivity 中获取并设置返回键的点击事件

// 获取并设置返回键的点击事件

**@Override**

```
public boolean onKeyDown(int keyCode, KeyEvent event) {  
    if (keyCode == KeyEvent.KEYCODE_BACK) {  
        moveToBack(false);  
        return true;  
    }  
    return super.onKeyDown(keyCode, event);  
}
```

## 六、不足之处和今后的设想

不足之处：

- 1) 只实现了单曲循环固定的音乐
- 2) 只能在 MainActivity 中停止音乐

今后的设想：

- 1) 多选几首背景音乐让用户选择
- 2) 给软件加上一些动效，以实现更好的交互，让用户体验更好。

## 七、参考文献

[1] zejia\_. 关于 Android Service 真正的完全详解，你需要知道的一切[DB/OL].

<http://blog.csdn.net/javazejian/article/details/52709857>, 2016,2016-10-07/2017-12-12

[2] guolin. Android Service 完全解析，关于服务你所需知道的一切(上) [DB/OL].

[http://blog.csdn.net/guolin\\_blog/article/details/11952435/](http://blog.csdn.net/guolin_blog/article/details/11952435/), 2013,2013-10-31/2017-12-12

[3] 雨点点. Android 获取内外置存储卡方法[DB/OL].

[http://blog.sina.com.cn/s/blog\\_5da93c8f0102vcam.html](http://blog.sina.com.cn/s/blog_5da93c8f0102vcam.html), 2014,2014-11-27/2017-12-12

[4] yanglh6. Android 开发 6: Service 的使用（简单音乐播放器的实现） [DB/OL].

[https://www.cnblogs.com/yanglh6-jyx/p/Android\\_Service\\_MediaPlayer.html#\\_labelTop](https://www.cnblogs.com/yanglh6-jyx/p/Android_Service_MediaPlayer.html#_labelTop), 2016, 2016-11-13/2017-12-12