

中山大学数据科学与计算机学院本科生实验报告

(2017 年秋季学期)

课程名称：手机应用开发

任课教师：刘宁

年级	2015	专业 (方向)	计应
学号	14970011	姓名	宋思亭
电话	18826073511	Email	1056284551@qq.com
开始日期	2017.11.13	完成日期	2017.11.15

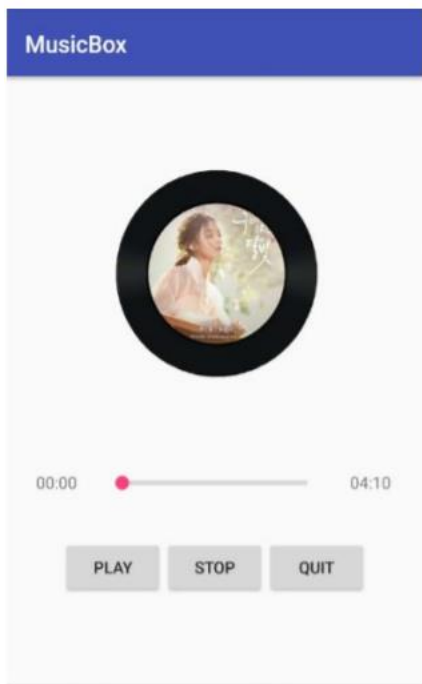
一、 实验题目

1. 学会使用 MediaPlayer ；
2. 学会简单的多线程编程，使用 Handle 更新 UI ；
3. 学会使用 Service 进行后台工作 ；
4. 学会使用 Service 与 Activity 进行通信。

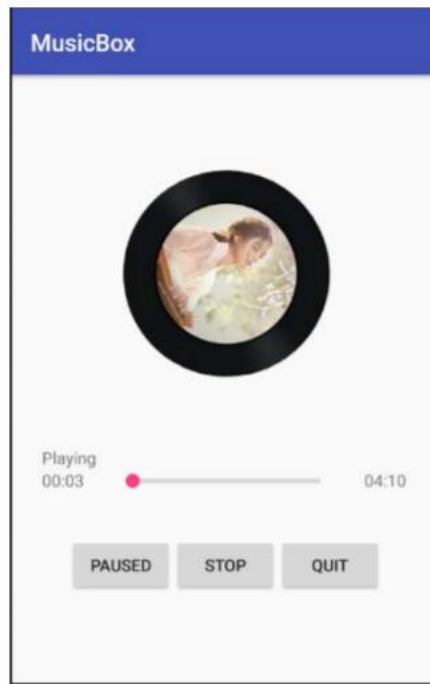
二、 实现内容

实现一个简单的播放器，要求功能有：

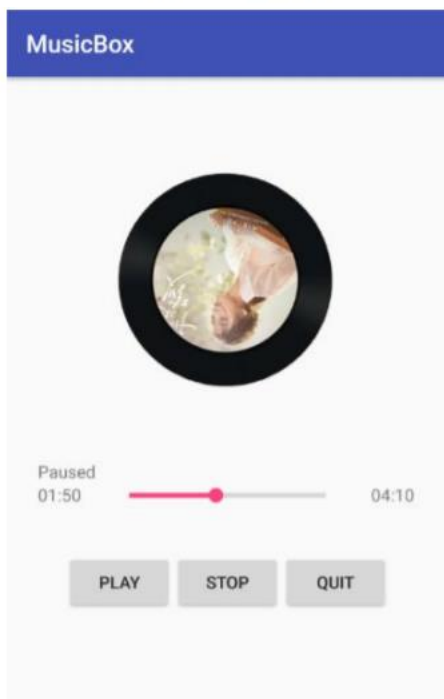
1. 播放、暂停，停止，退出功能；
2. 后台播放功能；
3. 进度条显示播放进度、拖动进度条改变进度功能；
4. 播放时图片旋转，显示当前播放时间功能；



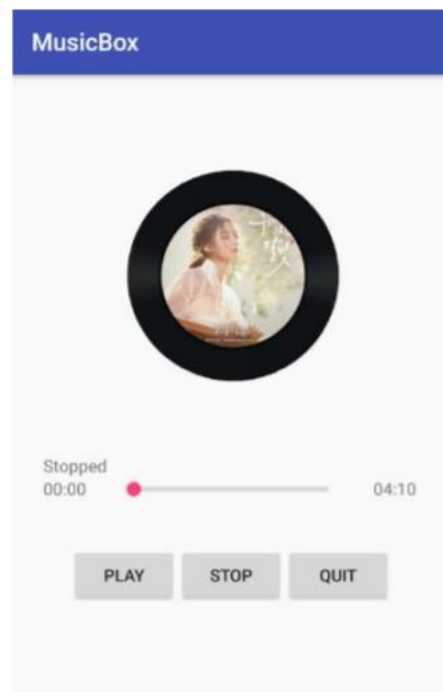
打开程序主页面



开始播放



暂停

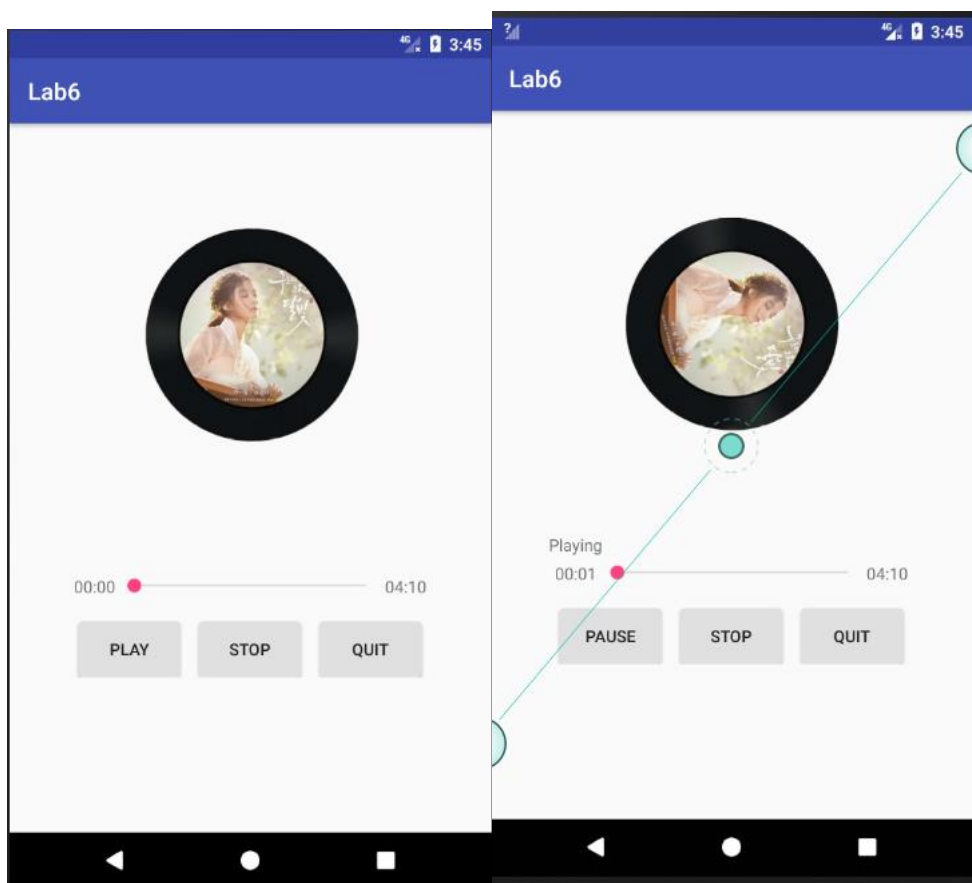


停止

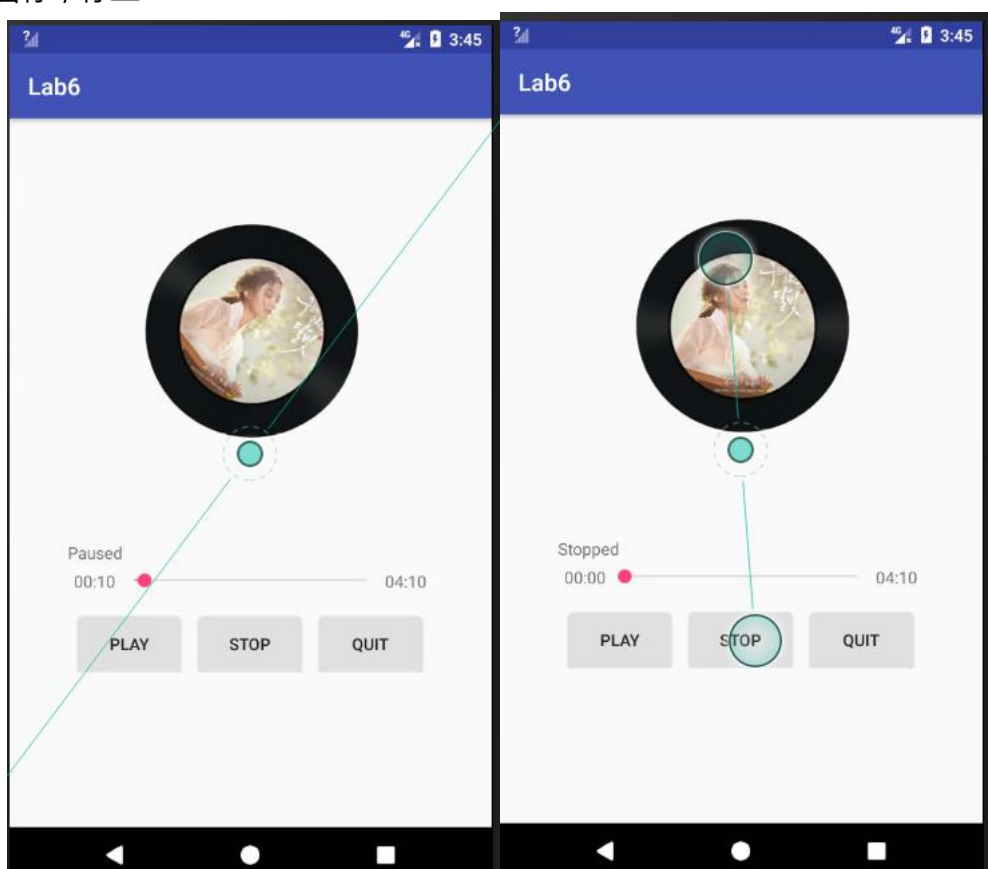
三、 课堂实验结果

(1) 实验截图

打开程序，开始播放



暂停，停止



(2) 实验步骤以及关键代码

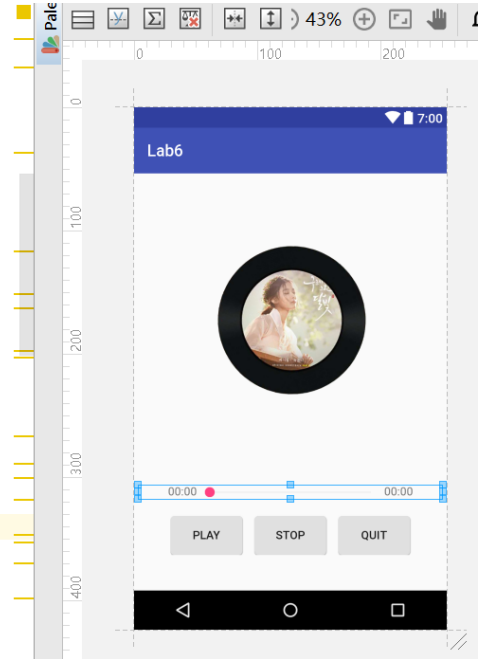
1. 写 activity_main.xml , 设置外观样式

```
<RelativeLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_gravity="center"
    android:layout_margin="5dp"
    android:gravity="center_horizontal">

    <TextView
        android:id="@+id/MusicTime"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="00:00" />

    <SeekBar
        android:layout_width="230sp"
        android:layout_height="wrap_content"
        android:id="@+id/MusicSeekBar"
        android:layout_weight="1"
        android:max="100"
        android:layout_toRightOf="@+id/MusicTime"/>

    <TextView
        android:id="@+id/MusicTotal"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="00:00" />
```



写 shape.xml , 设置按钮样式

```
<solid android:color="#E0E0E0" />
<corners android:radius="5dip" />
<padding
    android:bottom="2dp"
    android:left="10dp"
    android:right="10dp"
    android:top="2dp" />
```

2. 写 MusicService 类

```
public MediaPlayer mediaPlayer;
public boolean tag = false; //表示播放状态
```

// 初始化MusicService类

```
public MusicService() {
    mediaPlayer = new MediaPlayer();
    try {
        mediaPlayer.setDataSource("/storage/emulated/0/Music/melt.mp3");
        mediaPlayer.prepare();
        mediaPlayer.setLooping(true);
    } catch (Exception e) {
        e.printStackTrace();
    }
}
```

// 保持Activity和服务通信

```
public MyBinder binder = new MyBinder();
public class MyBinder extends Binder {
    MusicService getService() { return MusicService.this; }
}
```

```
public void playOrPause() {
    if (mediaPlayer.isPlaying()) {
        mediaPlayer.pause();
    } else {
        mediaPlayer.start();
    }
}
```

```
public void stop() {
    if (mediaPlayer != null) {
        mediaPlayer.stop();
        try {
            mediaPlayer.prepare();
            mediaPlayer.seekTo(0);
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}
```

并在 AndroidManifest.xml 文件里注册 Service

```
<service
    android:name=".MusicService"
    android:enabled="true"
    android:exported="true" />
```

在 MainActivity 中绑定 Activity 与 Service

```
// 保持与服务通信
private void bindServiceConnection() {
    Intent intent = new Intent(MainActivity.this, MusicService.class);
    startService(intent);
    bindService(intent, serviceConnection, this.BIND_AUTO_CREATE);
}

// 绑定Activity与服务
private ServiceConnection serviceConnection = new ServiceConnection() {
    @Override
    public void onServiceConnected(ComponentName name, IBinder service) {
        musicService = ((MusicService.MyBinder) (service)).getService();
        musicTotal.setText(time.format(musicService.mediaPlayer.getDuration()));
    }

    @Override
    public void onServiceDisconnected(ComponentName name) { musicService = null; }
};
```

停止服务时，解除绑定

```
// 停止服务时解除绑定
btnQuit.setOnClickListener((v) -> {
    handler.removeCallbacks(runnable);
    unbindService(serviceConnection);
    Intent intent = new Intent(MainActivity.this, MusicService.class);
    stopService(intent);
    try {
        MainActivity.this.finish();
    } catch (Exception e) {
        e.printStackTrace();
    }
});
```

3. 使设置 Seekbar 改变与时间同步

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

```

seekBar.setOnSeekBarChangeListener(new SeekBar.OnSeekBarChangeListener() {
    @Override
    public void onProgressChanged(SeekBar seekBar, int progress, boolean fromUser) {
        if (fromUser == true) musicService.mediaPlayer.seekTo(progress);
    }

    @Override
    public void onStartTrackingTouch(SeekBar seekBar) {
    }

    @Override
    public void onStopTrackingTouch(SeekBar seekBar) {
    }
});

```

使用 Seekbar 显示播放进度，设置当前值与最大值，通过 Handler 更新 UI 上的组件状态

// 通过Handler更新UI上的组件状态

```

public Handler handler = new Handler();
public Runnable runnable = () -> {
    musicTime.setText(time.format(musicService.mediaPlayer.getCurrentPosition()));
    seekBar.setProgress(musicService.mediaPlayer.getCurrentPosition());
    seekBar.setMax(musicService.mediaPlayer.getDuration());
    musicTotal.setText(time.format(musicService.mediaPlayer.getDuration()));
    handler.postDelayed(runnable, 200);
};

```

4. 设置动画实现图片旋转

// 设置图片旋转

```

ImageView 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);

```

设置开始/暂停点击事件

```

// 由tag的变换来控制事件的调用
if (musicService.tag != true) {
    // 开始播放音乐并设置按钮和状态显示
    btnPlayOrPause.setText("PAUSE");
    musicStatus.setText("Playing");
    musicService.playOrPause();
    musicService.tag = true;

    // 设置图片旋转
    if (isRotating == false) {
        animator.start();
        isRotating = true;
    } else {
        animator.resume();
    }
} else {
    // 暂停播放音乐并设置按钮和状态显示
    btnPlayOrPause.setText("PLAY");
    musicStatus.setText("Paused");
    musicService.playOrPause();
    animator.pause();
    musicService.tag = false;
}

```

设置停止按钮点击事件

```

// 停止播放音乐并设置按钮和状态显示
musicStatus.setText("Stopped");
btnPlayOrPause.setText("PLAY");
musicService.stop();
//animator.pause();
musicService.tag = false;

// 设置图片停止旋转
animator.end();
isRotating = false;

// 设置进度条位置到起点
seekBar.setProgress(0);

```

5. 向虚拟机添加文件，刚开始添加失败，在 adb 中获取权限后成功添加

```

C:\Users\StellaSong>adb root

C:\Users\StellaSong>adb remount
remount succeeded

C:\Users\StellaSong>

```

▼ storage	100 2017-11-14
> 13F7-120B	512 1970-01-01
▼ emulated	4096 2017-09-11
▼ 0	4096 2017-09-11
> Alarms	4096 2017-09-11
> Android	4096 2017-09-11
> DCIM	4096 2017-09-15
> Download	4096 2017-09-11
> Movies	4096 2017-09-11
▼ Music	4096 2017-11-14
melt.mp3	10097... 2016-10-23
> Notifications	4096 2017-09-11
> Pictures	4096 2017-09-11
> Podcasts	4096 2017-09-11
> Ringtones	4096 2017-09-11
> obb	4096 2017-09-11
> self	60 2017-11-14

6. 动态文件读取权限申请，在 AndroidManifest.xml 文件里注册权限

```

<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
<uses-permission android:name="android.permission.READ_EXTERNAL_STORAGE" />

```

在代码中添加权限确认函数

```

private static final int REQUEST_EXTERNAL_STORAGE = 1;
private static String[] PERMISSIONS_STORAGE = {
    Manifest.permission.READ_EXTERNAL_STORAGE,
    Manifest.permission.WRITE_EXTERNAL_STORAGE
};

// 获取读取系统文件权限
public static void verifyStoragePermissions(Activity activity) {
    try {
        int permission = ActivityCompat.checkSelfPermission(activity, "android.permission.READ_EXTERNAL_STORAGE");
        if (permission != PackageManager.PERMISSION_GRANTED) {
            ActivityCompat.requestPermissions(activity, PERMISSIONS_STORAGE, REQUEST_EXTERNAL_STORAGE);
        } else {
            hasPermission = true;
        }
    } catch (Exception e) {
        e.printStackTrace();
    }
}

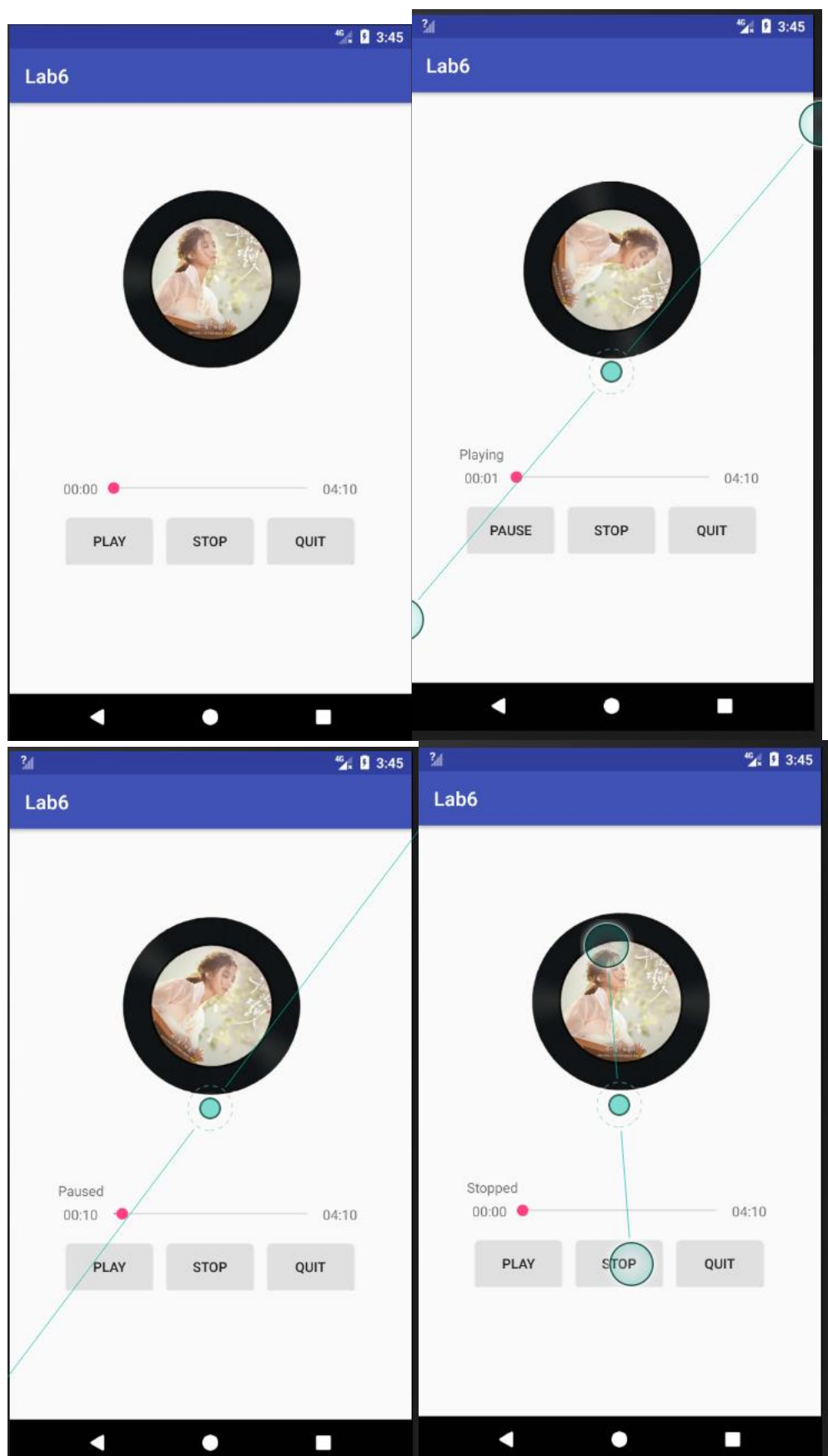
```

(3) 实验遇到困难以及解决思路

1. 向虚拟机添加文件时由于没有权限添加失败，查找相应解决方案后，在系统路径中添加 android 的 tool 路径，执行 adb 操作，root 获取权限，而后成功导入文件
2. 导入文件无法被正确的路径识别，更换文件存放路径解决

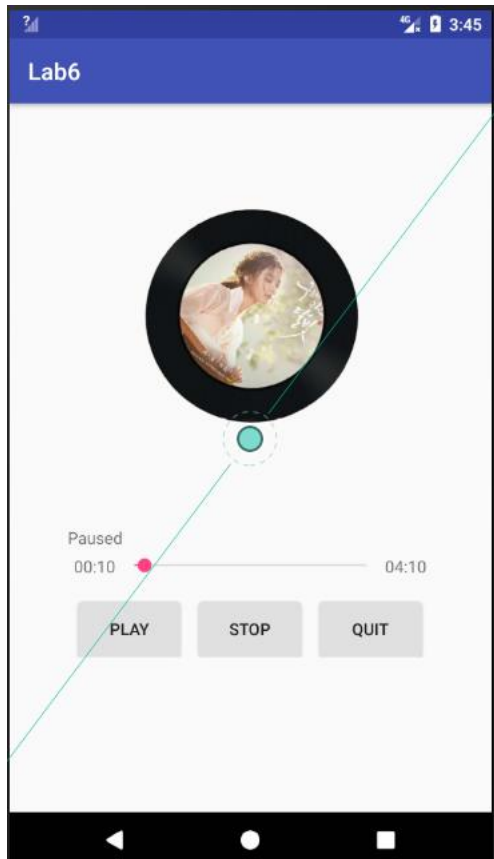
四、课后实验结果

1. 播放、暂停，停止，退出功能正常

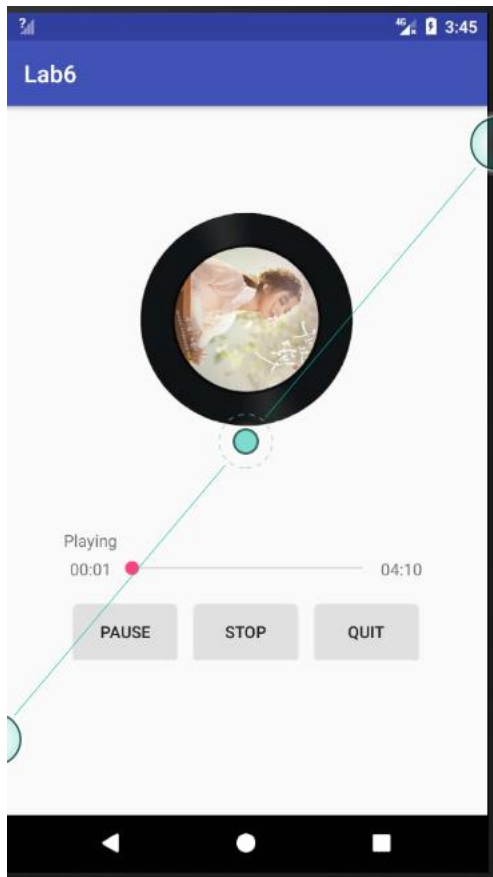


2. 后台播放功能正常

3. 进度条显示播放进度、拖动进度条改变进度功能正常



4. 播放时图片旋转，显示当前播放时间功能正常



五、 实验思考及感想

1. 安卓开发过程中有很多坑等我们去跳，需要多用搜索引擎多看文档
2. 遇到奇怪的问题不知道怎么解决时，可以重启试试