## Androidの基礎プロジェクト自己評価

**インストラクション:** 最終課題が終わったら、以下の質問に答えてください。これは開発中のあなたの思考過程を評価者に簡潔に説明するチャンスです。答えを書き終えたら、ソースコードと添付ファイルと一緒に送信してください。

# 必須コンポーネントに関する質問

## パーミッション

**自分のアプリで選んだパーミッションについて、選んだ理由を詳しく説明してください。**

|  |
| --- |
| <uses-permission android:name="android.permission.READ\_EXTERNAL\_STORAGE"/>  <uses-permission android:name="android.permission.MOUNT\_UNMOUNT\_FILESYSTEMS" />  As a local Music Player App it can read and delete audio file from external storage.  <uses-permission android:name="android.permission.INTERNET" />  In the NetSearchFragment the App get data online, so the Internet permission is also necessary.  <uses-permission android:name="android.permission.WAKE\_LOCK"/>  Use PowerManager to let play music service run even power is locked. |

## コンテンツプロバイダ

**コンテンツプロバイダの名前とそれが裏で使っているもの**

**（例えば、Sunshineのコンテントプロバイダは WeatherProvider という名前で、weatherとlocationという2つのテーブルを持つSQLiteデータベースを裏で使っています）**

|  |
| --- |
| MediaProvider which Android system provides. Use ContentResolver to query and get data.  In the background ,it use MediaScanner and MediaStore which have a series data tables. |

**どんなバックエンドと通信していますか？**

**（例えば、SunshineはOpenWeatherMap APIと通信しています）**

|  |
| --- |
| Get Recommendation data from <http://music.baidu.com/top/new/?pst=shouyeTop> |

**アプリがSyncAdapterを使用している場合、その名前は？　ネットワーク通信には実際にどんな仕組みが使われていますか？　（例えば、SunshineはHttpURLConnection を使ってネットワーク通信を行いますが、あなたのアプリはサードパーティのライブラリを使って通信しているかもしれません）**

|  |
| --- |
| Use third-party library “jsoup-1.7.3.jar” connect to network and get data from website. |

**どんなローダー／アダプタを使用していますか？**

|  |
| --- |
| MusicListAdapter and SearchResultAdapter |

## ユーザ／アプリの状態

**アプリがユーザもしくはアプリの状態を正しく保存および復元を行っている方法／場所を詳しく説明してください。（この質問の解答例については評価基準表を参照してください）**

|  |
| --- |
| In the MusicPlayerApp, screen rotation is locked . To add the rotation function next version. |

# オプションのコンポーネントに関する質問

あなたの最終課題に当てはまる質問に答えてください

## 

## 通知

**アプリで通知を実装した方法／場所について詳しく説明してください。**

|  |
| --- |
| Notification is created when PlayService’s onCreate() method be called.  User can use buttons in notification to control music player.  package org.loader.liteplayer.service.PlayService.onCreate() |

## 

## ShareActionProvider

**ShareActionProviderを実装した方法／場所について詳しく説明してください。**

|  |
| --- |
| Didn’t use shareActionProvider |

## 

## ブロードキャストイベント

**ブロードキャストイベントを実装した方法／場所について詳しく説明してください。**

|  |
| --- |
| private BroadcastReceiver mScanSDCardReceiver = new BroadcastReceiver() {  public void onReceive(Context context, Intent intent) {  L.l(TAG, "mScanSDCardReceiver---->onReceive()");  if(intent.getAction().equals(Intent.ACTION\_MEDIA\_SCANNER\_FINISHED)) {  MusicUtils.initMusicList();  ((LocalFragment)mFragments.get(0)).onMusicListChanged();  }  }  };  package org.loader.liteplayer.activity.MainActivity |

## 

## カスタムビュー

**カスタムビューを実装した方法／場所について詳しく説明してください。**

|  |
| --- |
| CDView simulates a revolving record when the music is playing.  LrcView shows lrcs dynamically.  package org.loader.liteplayer.ui.CDView  package org.loader.liteplayer.ui.LrcView |