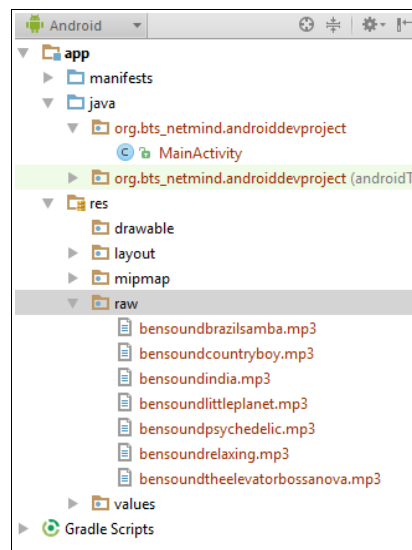# ANDROID DEVELOPMENT ASSIGNMENT

Exercise proposal

Create an Android application which allows to play a certain set of audio files provided in the resources' *raw* folder.

The developer will have to use the `MediaPlayer` API for this purpose. Follows a snippet about a straightforward usage of the `MediaPlayer` class.

```
[...]
    private MediaPlayer mPlayer;

    @Override
    protected void onCreate(Bundle savedInstanceState)
    {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        this.mPlayer = MediaPlayer.create(this, R.raw.bensoundbrazilsamba);
            this.mPlayer.start();
    }
[...]
```

For more information about the `MediaPlayer` API, check the official documentation at http://developer.android.com/guide/topics/media/mediaplayer.html.

The developer has to provide a set of `Buttons` to manipulate the player. For instance, two different `Buttons` to start/resume and stop the reproduction. Furthermore, all songs have to be selectable, for example, adding another `Button` to swap to another song.

In the host `Activity` it is convenient to override the `onDestroy()` method in

order to stop the `MediaPlayer` in case the application is going to finish. Bear in mind that a screen orientation change also calls `onDestroy()`, but the track should not be stopped then (use an if clause to check the return value of the `isFinishing()` method).

The app has to include a notification on the status bar when any song is playing, including the default app launcher icon (included in the resources' mipmap folder), the title of the song in course and a little description about it.

Follows a table with some information about the music tracks involved:

Song title	Duration	Country	Comments
bensoundbrazilsamba.mp3	*04:00*	Brazil	*Samba* is a Brazilian musical genre and dance style, with its roots in Africa via the West African slave trade and African religious traditions, particularly of Angola
bensoundcountryboy.mp3	*03:27*	USA	*Country music* is a genre of American popular music that originated in the Southern United States in the 1920s
bensoundindia.mp3	*04:13*	India	The music of India includes multiple varieties of folk music, pop, and Indian classical music. India's classical music tradition, including Hindustani music and Carnatic, has a history spanning millennia and developed over several eras
bensoundlittleplanet.mp3	*06:36*	Iceland	The music of Iceland includes vibrant folk and pop traditions. Well-known artists from Iceland include medieval music group Voces Thules, alternative rock band The Sugarcubes, singers Björk and Emiliana Torrini, post-rock band Sigur Rós and indie folk/indie pop band Of Monsters and Men
bensoundpsychedelic.mp3	*03:56*	South Korea	The Music of South Korea has evolved over the course of the decades since the end of the Korean War, and has its roots in the music of the Korean people, who have inhabited the Korean peninsula for over a millennium. Contemporary South Korean music can be divided into three different main categories: Traditional Korean folk music, popular music, or K-pop, and Western-influenced non-popular music

bensoundrelaxing.mp3	04:48	Indonesia	The music of Indonesia demonstrates its cultural diversity, the local musical creativity, as well as subsequent foreign musical influences that shaped contemporary music scenes of Indonesia. Nearly thousands of Indonesian islands having its own cultural and artistic history and character
bensoundtheelevatorbossanova.mp3	04:14	Brazil	*Samba* is a Brazilian musical genre and dance style, with its roots in Africa via the West African slave trade and African religious traditions, particularly of Angola

According to the official documentation it is a best practice to run the `MediaPlayer` in a background thread, such as a `Service` executed within a `Thread` object, for instance.

The minimum SDK has to be set up to API 11 (Android 3.0 Honeycomb).

Program performance

Minimum-requirements

The app has to allow the user to play any of the songs provided, including a notification which has to be visible during the reproduction. If the notification is clicked, a new `Activity` or `Dialog` has to be started showing the information provided in the Comments column. The application must include the launcher icon provided in the assignment data package.

It is strongly recommended to **include at least one** of the "Extra tasks" mentioned below.

In order to get extra marks, the developer can choose to implement one or various of the following extra tasks.

Extra task 1

Utilise a `ListView` or similar to show a list of the available songs. When an item is clicked, the corresponding song has to be played (for more information about `ListViews`, check http://developer.android.com/guide/topics/ui/layout/listview.html).

Extra task 2

Make use of any type of menu available in the Android platform (standard options

menu, ActionBar, Context Menu, Toolbar, etc.) implementing any functionality which may be appropriate for the app. For isntance, a context menu when long clicking an item in the list (if task 1 has been addressed brefore) showing some information about the audio track.

Extra task 3

Include a background thread (such as an `AsyncTask` or a `BoundService` within a `Thread` object) where the `MediaPlayer` performs.

Extra task 4

Include the table data into a database to be accessed from the app. It is also convenient to implement a Content Provider to make the data shareable with any other app with appropriate permissions.

Extra task 5

Add a Google Map Activity which includes `Markers` on the locations indicated on the above list relating to each song. When a `Marker` is clicked, an info window pops up showing the information provided in the Comments column. If this latter window is also clicked, the associated song is played. The Google Map Activity has to be accessed either by pressing a Button or implementing any navigation feature (such as a navigation drawer, configuring the Toolbar/ActionBar, etc.).

The submission date has been set up in class for:

29/04/2016