

FINAL PROJECT

Team 1

Randomization of shortcut play

□ Issue:

- ▣ In the original design of this app only allows default settings for randomization.
 - Artist, album or genre playlist will be shuffled.
 - Self-defined, favorite, or last-added playlist will not be shuffled.

□ Refactor:

- ▣ Add an option for customer to choose from.

What next?

- Find out the mechanism for Android settings.
 - Need to change the XML files.
 - SharedPreferences
- Check out the architecture for Apollo Music Player.
 - PreferenceUtils
 - ShortcutActivity
- Do refactoring.
- Write the code!

Mechanism for Android settings

- ❑ Settings for each app must have their default value when they were written in XML files.
- ❑ When you launch an app, settings for this app will be loaded into the SharedPreferences automatically.
- ❑ You may do some changes to the settings while you are using it.
- ❑ Record the settings into an XML files after you finish using it.
 - ❑ `/data/data/YOUR_PACKAGE_NAME/shared_prefs/YOUR_PREFS_NAME.xml`
 - ❑ `/data/data/YOUR_PACKAGE_NAME/shared_prefs/YOUR_PACKAGE_NAME_preferences.xml`

Architecture for Appolo Music player

□ PreferenceUtils

- ▣ PreferenceUtils is used to deal with all the settings of Appolo Music Player.

□ ShortcutActivity

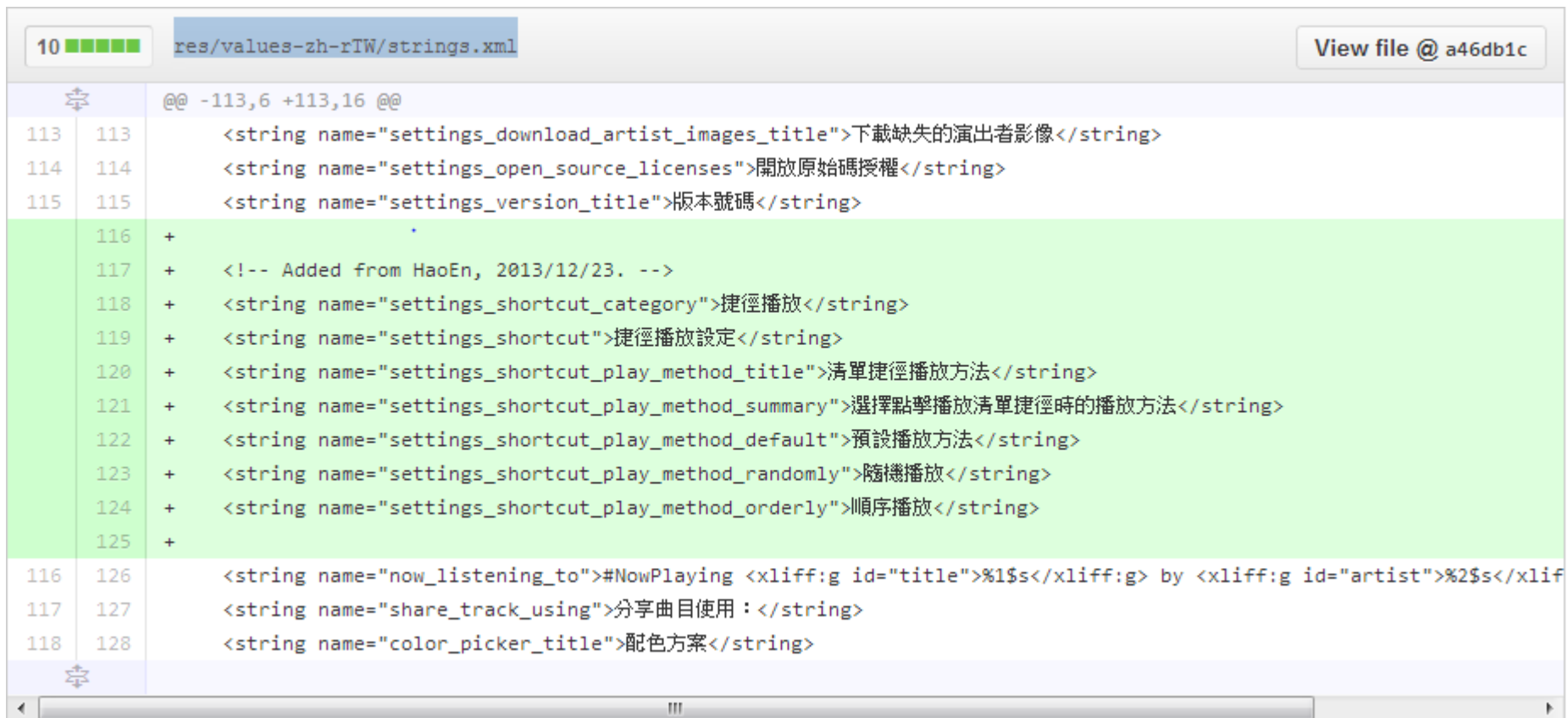
- ▣ ShortcutActivity is a shortcut to launch the music player from homescreen.
- ▣ It will enter the view for playing music directly, and determine whether shuffle the playlist or not.

Refactoring

- Add a list for customer to choose from.
 - ▣ And a block and alternative window for it.
 - ▣ Create one array for choice description and one for the value of choice.
- Add some predicates in `ShortcutActivity.java`, to make decision for randomization of playlist.
 - ▣ Default
 - ▣ Shuffle
 - ▣ Not shuffle

Write the code!

- res/values-zh-rTW/strings.xml
(Only change the 繁體中文 version now)



```
10 ■■■■■ res/values-zh-rTW/strings.xml View file @ a46db1c  
@@ -113,6 +113,16 @@  
113 113     <string name="settings_download_artist_images_title">下載缺失的演出者影像</string>  
114 114     <string name="settings_open_source_licenses">開放原始碼授權</string>  
115 115     <string name="settings_version_title">版本號碼</string>  
116 +  
117 +     <!-- Added from HaoEn, 2013/12/23. -->  
118 +     <string name="settings_shortcut_category">捷徑播放</string>  
119 +     <string name="settings_shortcut">捷徑播放設定</string>  
120 +     <string name="settings_shortcut_play_method_title">清單捷徑播放方法</string>  
121 +     <string name="settings_shortcut_play_method_summary">選擇點擊播放清單捷徑時的播放方法</string>  
122 +     <string name="settings_shortcut_play_method_default">預設播放方法</string>  
123 +     <string name="settings_shortcut_play_method_randomly">隨機播放</string>  
124 +     <string name="settings_shortcut_play_method_orderly">順序播放</string>  
125 +  
116 126     <string name="now_listening_to">#NowPlaying <xliff:g id="title">%1$s</xliff:g> by <xliff:g id="artist">%2$s</xliff:g>  
117 127     <string name="share_track_using">分享曲目使用：</string>  
118 128     <string name="color_picker_title">配色方案</string>
```

Write the code!

□ res/values/arrays.xml

13  res/values/arrays.xml [View file @ a46db1c](#)

		@@ -24,5 +24,18 @@
24	24	<item>@string/page_songs</item>
25	25	<item>@string/page_genres</item>
26	26	</string-array>
	27	+
	28	+ <!-- Added from HaoEn, 2013/12/23. -->
	29	+ <!-- Two choice for shortcut randomization -->
	30	+ <string-array name="settings_shortcut_play_method_choice">
	31	+ <item>@string/settings_shortcut_play_method_default</item>
	32	+ <item>@string/settings_shortcut_play_method_randomly</item>
	33	+ <item>@string/settings_shortcut_play_method_orderly</item>
	34	+ </string-array>
	35	+ <string-array name="settings_shortcut_play_method_value">
	36	+ <item>"default"</item>
	37	+ <item>"randomly"</item>
	38	+ <item>"orderly"</item>
	39	+ </string-array>
27	40	
28	41	</resources>

Write the code!

□ res/xml/settings.xml

14

res/xml/settings.xml

[View file @ a46db1c](#)

```
@@ -58,6 +58,20 @@
58      android:summary="@string/settings_delete_cache_summary"
59      android:title="@string/settings_delete_cache_title" />
60  </PreferenceCategory>
61  +
62  +  <!-- Added from HaoEn, 2013/12/23. -->
63  +  <!-- Shortcut motivation -->
64  +  <PreferenceCategory android:title="@string/settings_shortcut_category" >
65  +      <!-- Shortcut randomization -->
66  +      <ListPreference
67  +          android:key="shortcut_random"
68  +          android:summary="@string/settings_shortcut_play_method_summary"
69  +          android:title="@string/settings_shortcut_play_method_title"
70  +          android:entries="@array/settings_shortcut_play_method_choice"
71  +          android:entryValues="@array/settings_shortcut_play_method_value"
72  +          android:defaultValue="default" />
73  +  </PreferenceCategory>
74  +
61  75  <!-- About -->
62  76  <PreferenceCategory android:title="@string/settings_about_category" >
63  77      <PreferenceScreen android:title="@string/settings_about_apollo" >
```

Write the code!

- src/com/andrew/apollo/ui/activities/ShortcutActivity.java
(continue with next page)

19 ■■■■■ src/com/andrew/apollo/ui/activities/ShortcutActivity.java View file @ a46db1c

```
@@ -37,6 +37,7 @@
37 37 import com.andrew.apollo.utils.Lists;
38 38 import com.andrew.apollo.utils.MusicUtils;
39 39 import com.andrew.apollo.utils.MusicUtils.ServiceToken;
40 +import com.andrew.apollo.utils.PreferenceUtils;
40 41
41 42 import java.util.ArrayList;
42 43 import java.util.List;
@@ -88,6 +89,11 @@
88 89     * Used with the loader and voice queries
89 90     */
90 91     private final ArrayList<Song> mSong = Lists.newArrayList();
92 +
93 + /**
94 +  * Used to determine play at random or in order.
95 +  */
96 + private PreferenceUtils mPreferences;
91 97
92 98     /**
93 99     * {@inheritDoc}
@@ -106,6 +112,8 @@ protected void onCreate(final Bundle savedInstanceState) {
```

```

106 112 // Get the voice search query
107 113 mVoiceQuery = Capitalize.capitalize(mIntent.getStringExtra(SearchManager.QUERY));
108 114
115 + // Get the preferences
116 + mPreferences = PreferenceUtils.getInstance(this);
109 117 }
110 118
111 119 /**
@@ -183,6 +191,17 @@ public void run() {
183 191     }
184 192     }
185 193     // Finish up
194 +
195 + // Added from HaoEn, 2013/12/23.
196 + // And also cancel all 'mShouldShuffle'
197 + // Set mShouldShuffle according to settings.
198 + if (mPreferences.shortcut_rand().equals("randomly"))
199 +     mShouldShuffle = true;
200 + else if (mPreferences.shortcut_rand().equals("orderly"))
201 +     mShouldShuffle = false;
202 + else if (mPreferences.shortcut_rand().equals("default"))
203 +     ; // Do nothing.
204 +
186 205     allDone();
187 206 }
188 207 });

```

Write the code!

□ src/com/andrew/apollo/utils/PreferenceUtils.java

16

src/com/andrew/apollo/utils/PreferenceUtils.java

View file @ a46db1c

```
@@ -77,6 +77,12 @@
77 77
78 78 // Key used to set the overall theme color
79 79 public static final String DEFAULT_THEME_COLOR = "default_theme_color";
80 +
81 + /**
82 +  * Added from HaoEn, 2013/12/23.
83 +  */
84 + // Key used to determine randomization for shortcut
85 + public static final String SHORTCUT_RAND = "shortcut_random";
86
87 private static PreferenceUtils sInstance;
88
@@ -382,5 +388,13 @@ public boolean isGridLayout(final String which, final Context context) {
382 388 final String defaultValue = "simple";
383 389 return mPreferences.getString(which, defaultValue).equals(grid);
384 390 }
385 -
391 +
392 + /**
393 +  * Added from HaoEn, 2013/12/23.
394 +  * @return True if random play from shortcut
395 +  *         false otherwise
396 +  */
397 + public final String shortcut_rand() {
398 +     return mPreferences.getString(SHORTCUT_RAND, "default");
399 + }
400 }
```