Music Application - Android

Shravan Varma Epuri

08

**Fall**

This zip file consists of two Android apps. The first app, named *ServiceMusic(Server)* stores a number of audio clips, such as songs or other recordings. The clips are numbered 1 through *n*, where *n* is the total number of clips. The app contains a service intended to be bound (as opposed to started), which exposes an API for clients to use. The API supports such functionality as playing one of the audio clips, pausing the clip, resuming the clip and stopping the playing of the clip altogether. In addition, this app maintains an SQLite database that keeps track of all the requests that were received by the *ServiceClient* app. For each request, the database records the date and time (including seconds) when the request was issued, the kind of request (e.g., whether to stop or resume a clip), the number of the clip (if applicable) and the current state of the service when the request was received (e.g., playing clip number 3, paused while playing clip number 4, etc.) Additional functionality exposed by the *ServiceMusic* API allows a client application to query the database for all transactions that were recorded thus far in the database. The service broadcasts an implicit intent when a clip finishes playing. The application (for testing) includes 3 audio clips of variable duration.

The second app, *ServiceClient* consists of an activity that exposes functionality for using the *ServiceMusic* and binds to the service for playing desired audio clips. The interface includes appropriate *View* elements for the following functionality:

(1) Playing a given clip (by number),

(2) Pausing the playback,

(3) Resuming the playback,

(4) Stopping the player, and

(5)Getting a record of all *ServiceMusic* transactions (i.e., requests) recorded so far.

The transactions are shown in a second activity that contains a *ListView*. When the client activity is stopped, the service continues playing; however, the service is unbound and stopped if the activity is destroyed. Finally, when a broadcast is sent indicating that an audio clip has finished playing, the app displays an appropriate toast message on the device’s screen.

***Implementation notes***. Use a Nexus 5 device running the latest Android platform available (API 21—Lollypop).

Table layout is in such a way that it will display in portrait mode.

Backward compatibility with previous Android versions is not provided but can be done with minimal work if needed.