CS478: Software Development for Mobile Platforms

Project #4

Due time: 7:00 pm on 4/24/2015
Submit using Blackboard web site
Total points: 100
Instructor: Ugo Buy

TAs: Nianzu Ma and Xiangyu Liu

For this project you will design and code two Android apps. The first app, named *AudioServer* 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 *AudioServer* 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 *AudioServer* 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 should include at least 3 audio clips of variable duration. However, the duration of Clip #1 should be between 30 seconds and 3 minutes.

The second app, *PlayerClient* consists of an activity that exposes functionality for using the *AudioServer* and binds to the service for playing desired audio clips. Your interface should mininally include 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 *AudioServer* transactions (i.e., requests) recorded so far. The transactions are shown in a second activity that contains a *ListView*. You are not required to keep the database in a content provider, you can just return an appropriate *List* from the audio server to the client. When the client activity is stopped, the service should continue playing; however, the service should be unbound and stopped if the activity is destroyed. Finally, when a broadcast is sent indicating that an audio clip has finished playing, your app should display an appropriate toast message on the device's screen.

Hints. You are at liberty to choose the audio clips from segments pictures publicly-available (and not copyrighted or otherwise protected) on the Internet. When testing your application, make sure to upload *AudioServer* app first, or else the client app may fail to initialize properly. Use static method *Calendar.getInstance()* to get the current day and time.

Implementation notes. As with the previous project, use a Nexus 5 device running the latest Android platform available (API 21—Lollypop). Design your table layout in such a way that it will display in portrait mode. You are not required to provide backward compatibility with previous Android versions.

You must work alone on this project. Submit the entire Eclipse project as a zip archive using the submission link in the assignment's page on Blackboard. Alternatively, you may submit an Android Studio project. No late submissions will be accepted.