

Advanced Android Interface and Activities

*Lecturer: Srinivas Shakkottai**Scribe: Self*

1 Introduction

In this machine problem will learn how to start a new activity, read files, and perform multimedia operations.

2 Problem statement

Part I: Music Player App

In this part, you will learn how to use several GUI Widgets and the media player. This app will have a single activity. The user sees a vertical sequence of 3 radio buttons, each with a music file name next to it. When any radio button is clicked, the appropriate music should start playing, and a picture corresponding to the music (like a cover illustration) should appear below the radio buttons. Finally, a set of music control buttons (start/stop etc.) appears below the image.

Resources: Place three mp3 song files in “raw”, and place three images corresponding to these songs in drawable.

XML: The layout of this app will consist of a title (TextView), a vertical list of radio buttons (Radio Group), an image (ImageView) and several buttons to control the player (start/stop Button).

JAVA: When the radio button is pressed, the corresponding music file must be played [2, 3], and the image in the ImageView is set to the one corresponding to the song that was clicked.

Part II: Quiz App

The objective is to start new activities corresponding to buttons pressed in MainActivity, and to pass data back and forth between the calling and called activities. This app will have two activities. The first activity shows a list view with quiz topics. It also shows a score board of your performance thus far (initialized to 0), and a reset button. You must have at least 3 quiz topics.

When a quiz topic is clicked, a new activity opens that shows a question about that quiz topic, and presents an editText to enter the answer. The answer is verified, and control is passed back to the main activity which updates the scoreboard based on whether the answer was correct or wrong.

Resources: Add arrays of strings corresponding to questions and answers in “raw”. More in file handling [4].

XML: The layout of the MainActivity will consist of a title (TextView), a vertical list of quiz topics (ListView), and a scoreboard (TextView) and a reset button (Button). The secondary activity layout consists of an app title (TextView), a TextView to display the question, an EditText to enter the answer, and a Button to press when you get done.

JAVA: When a quiz topic in main activity is pressed, the topic is sent via an intent to the secondary activity. The secondary activity identifies the topic from the intent, and displays a quiz question corresponding to the chosen topic. If the answer is entered correctly, a value of 1 sent back to the Main Activity. Else

the value of 0 is sent back. The scoreboard is updated, and the user may choose again.

First read through “Starting another Activity” in [1] and Chapters 7 and 8 in HFA [6]. You will also need to create `LinearLayouts` and `GridLayouts` for the appropriate display activities.

Don’t forget to declare your different display activities in the manifest (typically will happen automatically). You can read more on how to modify the Manifest on page 111 in HFA [6].

3 Evaluation Guidelines

1. Check operation using the interface.
2. Creativity carries bonus points.
3. Your application must run on either a physical or a virtual device.

References

- [1] Android developer: Building Your First App
<http://developer.android.com/training/basics/firstapp/index.html>
- [2] Android developer: Building Apps with Multimedia
<http://developer.android.com/training/building-multimedia.html>
- [3] Starting a media player after stopping.
<https://stackoverflow.com/questions/4486828/android-media-player-restart-audio-after-calling-stop>
- [4] Android developer: Saving Data
<http://developer.android.com/training/basics/data-storage/index.html>
- [5] Android developer: List View
<http://developer.android.com/guide/topics/ui/layout/listview.html>
- [6] Head First Android