

CS478: Software Development for Mobile Platforms

Project #1

Due time: 11:59 pm on 2/12/2023

Submit using Blackboard web site

Total points: 100

Instructor: Ugo Buy

TAs: Sasanka Veleti, Anupreet Paulkar and Harshil Bhatt

Project functional specification. Design and implement an app with the following functionality. The app starts off with a main activity containing a welcome text and two buttons. The two buttons are arranged vertically (i.e., with button 1 above button 2) when the device is in portrait mode and horizontally (i.e., with button 1 to the left of button 2) when the device is in landscape mode. Either way, the welcome text is centered in the display above the buttons.

Regardless of orientation, upon pressing the first button your app displays a second activity containing a read-only text field and an edit text field. The read-only text field prompts the device user to enter a phone number in the edit text field. A phone number consists of nine digits. The digits may appear consecutively (e.g., as in 3125551212) or using parentheses, an optional space and a mandatory dash to separate the various groups of digits (e.g., as in (312) 555-1212). No other formats are accepted.

When the user is done entering the number, she will press the “done” or return key in the soft keyboard. The activity must now check whether a phone number is in one of the 2 formats was entered by the user. In this case, the activity will set a result code of “RESULT_OK”; otherwise, the activity sets a result code of “RESULT_CANCELED”. Either way, the second activity terminates itself, thereby causing the first activity to become visible again.

Upon returning from the second activity, the first activity checks whether the result code was “RESULT_OK”. In the case, the user may press the second button in the first activity causing the device to display the dialer activity in the phone app. This activity will show the phone number that was entered by the user in the second activity of your app, but without dialing that number.

You must use the existing *Phone* app pre-installed in your device when displaying the dialer activity. (You are not allowed to specify what app and app component should be used for dialing the number.) However, if the result code was “RESULT_CANCELED”, when the user presses the second button, the first activity displays a toast message informing that device user that she entered an incorrect number and includes the number in the toast message.

Your app must be able to handle device configuration changes. For instance, if the device is switched from portrait to landscape mode or vice versa after the user entered a number, your app should “remember” the number that was entered.

Implementation notes. Your app has no knowledge of the specific app to be invoked for displaying the dialer. You are not responsible for coding or downloading additional *Phone* apps; you may assume that a suitable app is already installed on your device even though you don’t know what that app is.

Use intent extras to pass the number from the second activity to the first activity.

Create an instance of the *Toast* class with the static message *makeText()*, which takes as input 3 arguments. You can then display the toast by calling method *show()* on the instance. See the online documentation for additional details.

The Android manifest file should contain the following line before the “application” tag:

```
<uses-permission android:name="android.permission.QUERY_ALL_PACKAGES"/>
```

Use an *EditText* widget to enter the number in the second activity. When defining a listener for the edit text, implement Java interface *TextView.OnEditorActionListener* . Read the Android documentation on the *TextView* class in order to define the listener.

Android platform. For this project use a Pixel 5 AVD running Android 13–API 33 that you downloaded for Homework 1. You are not required to provide backward compatibility with previous Android versions.

Submission instructions. *You must work alone on this project.* Submit the entire Studio project as a zip archive using the submission link in the assignment’s page on Blackboard. No late submissions will be accepted.