

CIS 3515 Lab

Worksheet 5

Instructions: We are going to extend your previous application by adding two features:

1. localize your application by adding a Spanish translation
2. Refactoring our app to use a single activity along with a Fragment instead of 2 activities

You will then generate a signed APK with a new version number.

Section 1

You will update the application you worked on in the last lab to ensure that all text labels are taken from a strings resource file. You will then provide a translation for a second language.

1. Add string resources to your application to provide the labels necessary to display the word for each color in your grid. One approach is to use individual String resource elements, but a better approach is to use a String-array resource:

<https://developer.android.com/guide/topics/resources/string-resource.html>

2. Modify your application code to use the string resources to display the cell labels in your grid view. You can retrieve a string-array resource as follows:

```
Resources res = [context.]getResources();  
String[] gridLabels = res.getStringArray(R.array.grid_array);
```

3. Create a new **strings.xml** file for your Spanish translations

1. In Android Studio's **Project** view, right click on the **values** folder under **res** and select **New** → **Android resource file**
 2. Enter **strings** in the File name field and select **main** Source set and enter **values** under Directory name
 3. Select **Locale** under Available qualifiers and click the >> button
 4. Select **es:Spanish** under Language. Leave *Any region* selected on *Specific Region Only*
 5. Once complete click on **OK**
4. Copy your string-array resource (and all other string resources) from your default strings resource file, into your new Spanish translation file, and replace all the text values with the proper translation.

1. Ways to get translations
 1. Good → Google translate
 2. Better → Ask your Spanish speaking friend
 3. Best (but not practical for us) → Pay a translation service (Google provides one)
5. Test your application by changing the Language of your virtual device from English to Spanish. You will find the Language configuration under Device Settings. Good luck changing it back from Spanish to English.

Section 2

Adapt your application to utilize a fragment for the color canvas, instead of a second activity (you can leave the activity in your project if you'd like, but we'll no longer be using it). You will allow a user to select a color from an adapter view in the activity, and once selected, the information should be used to create and attach a fragment to the activity whose background color will be set to the color selected in the adapter view.

1. Create a fragment called CanvasFragment.
2. When a user selects a color from the Adapter View, an instance of CanvasFragment should be created (using a factory method with the selected color passed as an argument) and attached to the activity.
3. The fragments should be added to the activity's backstack so that all transactions are tracked and pressing the back button will undo the last transaction.
4. The fragment will use the color that was passed as an argument to set its background color.

Once your application has been tested, update the version number and generate a new APK

1. Modify the file **build.gradle (Module:app)** found under **Gradle Scripts** in the project view
 1. Under default config, change **versionCode** to **2**, and **versionName** to **1.0.1-spanish**
2. Click on **Build** and select **Generate Signed APK**
3. On the presented dialog, click on **Choose existing** and locate the certificate you generated during the previous lab. Select the alias you had created and enter the requested passwords, then click on **Next**.

Please note: If you are unable to find your previous key store, or are unable to recall one or both passwords, then you can create a new key store and key. Keep in mind however that had this application been previously distributed on Google Play or another distribution platform, you would have lost the ability to update the application currently deployed and would instead have to change the package name (your apps unique identifier), lose your current user base and start all over.

4. Select the destination folder of the APK and set Build Type to **release**
5. Click on **Finish**
6. Locate and rename your generated APK to **app-release-1.0.1.apk**

You have one week to complete this assignment. Upload your project (or a GitHub link) as well as the signed APK to Canvas, or make arrangements to show your work to the TA BEFORE the beginning of next week's lab.

NOTE

You're being asked to make major changes to a project (it's basically a new project at this point). This is a great opportunity to practice using a *version control system* (VCS) such as Git. A VCS (at a minimum) allows you to keep track of old versions of code. That way, even even though you'll be changing the overall structure of your project and losing code, you can always go back and see the original source code.