Lappeenrannan teknillinen yliopisto
School of Business and Management

Software Development Skills

Marta Xiaoyang Moraga Hernández, 001101267

LEARNING DIARY, MOBILE MODULE

LEARNING DIARY

08.02.2023

Android Studio For Beginners

In this tutorial I learnt about the basics of Android Studio, like adding components to the view and basic scripting, and debugging. Debugging is similar to other programs. I did the simple app example.

I set up the repository as I did for other courses with *git* init and *git remote add* and downloaded Android Studio. The program has changed a lot from the video examples, for instance, to change to the preferences recommended in the tutorial, I had to access *settings*, which were represented as a gear symbol. While, on the tutorial, said options are accessed by pressing the program title.

Furthermore, in the current version it is necessary to confirm the refactor when changing the id of an app element and the options for debugging are in *Run->Debugging actions…*

Instead of concatenating *result+""* to show the result in the *TextView* as the tutorial did, I used *Integer.toString(result)*, because the program showed a warning.

09.02.2023

Android Studio For Beginners Part 2

I followed the tutorial and learnt about Activities and Intentions, and reinforced the knowledge from the first video. Android Studio has changed a bit from the tutorial version, it asks for more confirmations when adding an Activity than in the video, but it works the same.

The biggest issue I encountered was that the button supposed to open *google.com* was not working, probably because the difference in versions. After looking online I added the following filter to *AndroidManifest.xml*:

```
<activity ...>
    <intent-filter>
        <action android:name="android.intent.action.VIEW" />
        <!-- Include the host attribute if you want your app to respond
             only to URLs with your app's domain. -->
        <data android:scheme="http" android:host="www.example.com" />
        <category android:name="android.intent.category.DEFAULT" />
        <!-- The BROWSABLE category is required to get links from web
pages. -->
        <category android:name="android.intent.category.BROWSABLE" />
    </intent-filter>
</activity>
```

This appears in the documentation for Android Studio: https://developer.android.com/guide/components/intents-common#ViewUrl
About how to load a web URL.

I also learnt that you can send data between Activities as key-value pairs and how to use said data.

10.02.2023

Android Studio For Beginners Part 3

I followed the tutorial and learnt about *ListView* and *ImageView*, and reinforced the knowledge from the other videos. As always there were some differences because of the Android Studio version, but everything worked the same. It seems nowadays *ListView* is not used as much, as it appeared in the *Legacy* section of the palette.

When creating the first *my_listview_detail.xml* layout I changed the text size by adding *android:textSize="32dp"* manually instead of editing an attribute, as I did not find the option.
https://stackoverflow.com/questions/13264794/font-size-of-textview-in-android-application-changes-on-changing-font-size-from
Thanks to that I know learnt a bit about changing attributes by changing the script instead of looking for it in the attributes section.

The video also teaches how to add files directly to the folder structure. I accessed the drawable folder on my computer by right clicking the folder on Android Studio and copying the path/reference. I do not own the images used; they are the first results of looking up the words on Google.

I understand better the usual workflow, organisation, and development structure.

Personally, the most interesting parts were working on *DetailActivity.java*, since it reminds me of Object-Oriented Programming and Web Development. In this section I also learnt about useful features such as *BitmapFactory.decodeResource()*, which allows for a resource to be checked without loading it.

13.02.2023

Initial plan

My initial plan is to make a simple recipe app. In the home page there will be three buttons and a switch. The first button will lead to a recipe list similar to the one from the last tutorial, which will show the recipes. The second one will launch an outside activity; it will work as a link to the Lidl page. The third and last button will lead to a random generator, that will recommend a recipe to try out. The switch will turn on and off night mode.

Main page

I start by setting up the first page that shows when opening the app, which is going to be very simple. I make sure to add to *strings.xml* any text that will be needed for this, so that the text on the buttons will use *@strings*.

Buy Button

I start with the button that leads outside the app, as it was the easier one. I remembered to add the filters necessary to *AndroidManifest.xml* as I did when doing the tutorial.

Night/Day mode Switch

I found a quick tutorial to enable the Night/Day mode Switch. I really like to change to dark mode, so I thought it would be a good little addition.
Tutorial: https://www.youtube.com/watch?v=uIdL-0Ekrj8

RecActivity and RecipesActivity

I create *RecActivity* and *RecipesActivity*, with their related layouts. I had a bit of trouble because, instead of selecting the *Activity* option, I created Java Classes. Thankfully, I checked the options a second time.

Now, the Recommendation and Recipes buttons start their corresponding activities. However, they are still empty.

15.02.2023

RecipesActivity

RecipesActivity works like the *ListApp* made in the tutorial. A *ListView* occupies the whole screen and an *ItemAdapter* that extends *BaseAdapter* handles the recipes to show. When a recipe is touched, *RecipeDetailActivity* launches and shows more details about the recipe.

When making the layout for the *ListView* items, an accessibility warning that led to the following link appeared in Android Studio:
https://support.google.com/accessibility/android/answer/12159181
This helped me make small changes so that the text would fit better, most notably I changed measurements to sp and set:

android:layout_width="match_parent"
android:layout_height="wrap_content"

Each *ListView* item, when pressed, leads to *RecipeDetailActivity*. It sends the id, the name and the description as extras, while the instructions string will be accesses from the new activity.

I downloaded some photos to add a *ImageView* to the *RecipeDetailActivity*. The easiest way to adjust them was to set *android:scaleType="centerCrop"*.
https://developer.android.com/jetpack/compose/graphics/images/customize
The photos are not mine; they are some of the first results after a google search. Their rights belong to their owners.

RecActivity

It is composed of a simple *TextView* and a *Button*. Each time the button is pressed; a different recipe recommendation is shown. It works with basic random int generation.

Languages

I right clicked the *strings.xml* file and opened the *Translation Editor*, where I created another file for a Spanish translation. I added translations for all strings used, except the Lorem Ipsum extracts. Changing the system language will change the app language accordingly. I also added Finnish; but translations are from Google translate, as I do not speak it.

Conclusion:

From this course I learnt how to make basic apps, work with containers and handle views and activities. Most importantly, it has taught me about how to work with Android Studio and how to look for solutions when trying to do something new.