Programmation d'applications mobiles - Android

Practical - Part 1

8h

Goal

Learn some of the basic of the Android developpement

Technical base

The main docs : https://developer.android.com/guide/index.html

Kotlin docs : https://kotlinlang.org/docs/reference/

Compose docs : https://developer.android.com/jetpack/compose/documentation

Kotlin Primer

Follow the Kotlin tutorial on the Kotlin official site : https://play.kotlinlang.org/koans

- Introduction
- Classes
- Conventions
- Collections
- Properties

Initialization

1. **Workspace**
    a. At home:
        - Start Android Studio
        - Use Standard initialization, SDK
        - To run the Android emulator : https://developer.android.com/studio/run/emulator
        - on an Intel CPU use Intel HAXM
        - On AMD, you'll need to enable SVM in your BIOS
        - You can also use your personal Android phone https://developer.android.com/studio/run/device
    b. (in ISIMA):
        - Start Android Studio
        - If it asks you to Import Settings, choose No
        - **If it asks** for an SDK, enter the current location : /Applications/sdk, if not, just press next

2. **Create a project**
    - Use **Empty Views Activity** to start fresh.
3. **Android emulator**
    - In the right sidebar, there is a **Device Manager** button
    - It allows to create differents size of devices to test your application on multiple resolutions
    - Choose an existing emulator or create one

First Project Activity (xml)

You can find some information here : https://developer.android.com/guide/topics/resources/layout-resource#idvalue (see the example also)

- **You had to use Empty Views Activity to create the project**
- Add a button, which when pressed, will display the text of your choice.
- Add an EditText, and when the button is pressed, replace the TextView's text by the EditText content.
- Use Log class to display logs in the onCreate, onStart, onResume, onPause, onStop, onDestroy of your Activity
- Try to figure out what triggers each methods (like enable rotation on your device and rotate your device, press the Home Button…)
- What happens to your TextView when you rotate the device ?

Resources

- Add a new color and use it in your code to change the button background
- Display a Toast when the button is pressed with a string from the resources

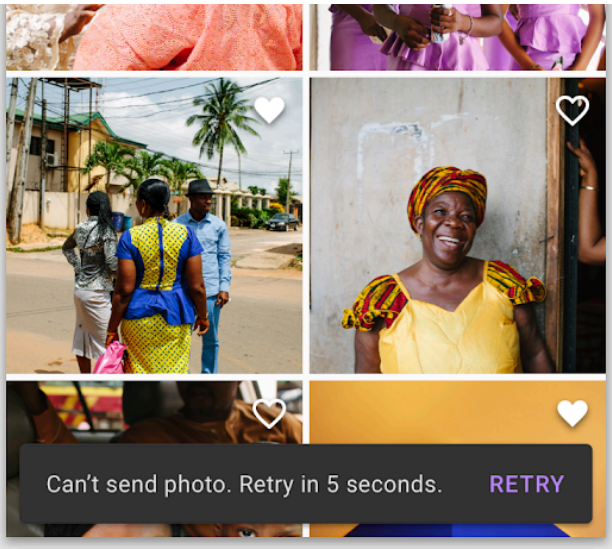
Intent

- Add a second activity and try to display it with your button (Doc)
- Ask the system to display the camera app (Doc)

Compose

Now we'll start the Compose part. Please read the documentation at
https://developer.android.com/jetpack/compose/documentation

- Create a new project and this time use **Empty Activity to create the project**
- Remake your previous FirstActivity in Compose (TextView => Text, EditText = TextField)
- Rotate the device and see what happens
- Remove the Text (leaving only the TextField and Button), and use a Scaffold to display a Snackbar (see below) containing the text in the TextField



List

- Use the List tutorial to display a list of element
- Add a Button at the bottom that will add a new cell with a random text

JSON

- Copy the following JSON extract from https://cataas.com/ and put it in a variable.

```
[{"_id":"VOXNL4u88baKyyjI","mimetype":"image/png","size":47320,"tags":["sil
ly","gray","cute"]},{"_id":"fogXmHYpuMV51EaN","mimetype":"image/jpeg","size
":24964,"tags":["Cute","fluffy","belly","black and
white","black","white","furry","ragamuffin","floof","curled","paw","floor",
"lying"]},{"_id":"Gcip5bmpURqUjzR8","mimetype":"image/jpeg","size":49961,"t
ags":["cute","smol","two
cats","bitting","brown","little","small","young","proud"]}]
```

Follow the documentation to add Kotlin Serialization to parse this JSON string in your project.

If you're stuck you can read https://developermemos.com/posts/kotlin-serialization-android

- Create a class CatPicture to store the JSON data.
- Display a list with the _id of the picture for now
- Bonus : display the tags using a Chip

Picture

- Integrate Coil into your project. It will allow you to download and display pictures from our JSON
- PS : the url to display the image is https://cataas.com/cat/{_id}

Putting every together

- Find a good API from this list https://github.com/public-apis/public-apis or create a hand-made JSON to make an app about your favorite subject!

Finished early ?

Use a ViewModel (and here) to handle the data part of your application.

Add Navigation to allow users to click on a picture and make it fullscreen, on another screen