

APP DEVELOPMENT

IN ANDROID STUDIO



HÁSKÓLINN Í REYKJAVÍK
REYKJAVÍK UNIVERSITY

LAB 3: COMPONENTS

OCTOBER 18, 2017

JÓN STEINN ELÍASSON

JONSTEINN@GMAIL.COM

Contents

1	Activities	2
1.1	Starting a new Activity	2
1.2	Lifecycle	2
1.3	Passing data between activities	2
2	Fragments	2
3	Services	2
4	Assignment	2

List of Tables

List of Figures

List of Listings

1 Activities

We have already used an activity without going into too much detail what it is. An activity is a single screen unit (usually full screen) with an user interface. So far we have only worked with a single activity but an Android app can have multiple activities. It breaks the app up into sections with different purposes and UI. For example, a menu in an email app could be an activity while composing an email might be another, opened from the menu.

One activity serves as a launcher activity. It is our starting point when running the app (opposed to a C++ main function) and from there on we can start navigating to other activities if any. Every app must have a launcher activity. All activities must be declared in our app's manifest¹ and the launcher activity is also determined there.

Each activity uses a layout file that defines their UI at least partially (some of it may be done dynamically in Java). In the `onCreate` function we have been setting the activity's layout with the `setContentView` method. Activities can share layouts although it serves a limited purpose unless most of the UI is dynamic. We will look at better ways to share UI in Fragments.

1.1 Starting a new Activity

Let's start by creating a new activity with `File > New > Activity > Empty Activity` and name it `SecondActivity` and leave the other options as they are. Before proceeding you should inspect your manifest. Add a button to the launcher activity which calls the method `clicked()` upon being clicked and add some text to the second activity so it differentiates from the other one. Finally add the following method to the launcher activity and run the program.

```
1 public void clicked(View view) {
2     startActivity(new Intent(MainActivity.this, SecondActivity.class));
3 }
```

The `startActivity` function takes `Intent` as a parameter. We will look at that class and its parameters in more detail later but for now, you can think of it as a bridge between two activities.

1.2 Lifecycle

1.3 Passing data between activities

2 Fragments

3 Services

4 Assignment

¹Android Studio does this automatically when an activity is created