CMSC 436 Lab 3

Building a Dynamic UI with Fragments

Overview

- In this lab you will create a simple UI using Fragments
- The Fragments will be added or removed from an Activity by clicking buttons
- You will also add Fragment transactions to the Back Stack so that the user can undo them by pressing back
- Finally, you will create an alternate UI for landscape orientation by rearranging the Fragments

Overview

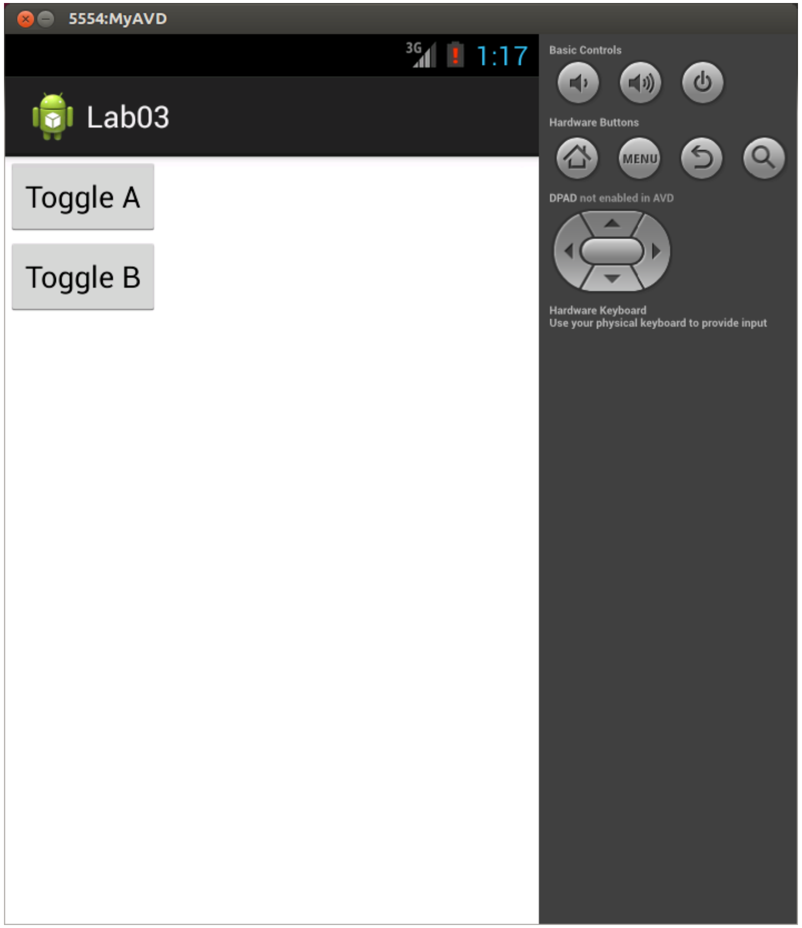
- The information you will need for this lab can be found on the Android developer site at

  http://developer.android.com/guide/components/fragments.html

  http://developer.android.com/training/basics/fragments/
- The developer site discusses how to support Fragments on older API versions as low as 4 by using the Support Library, but for this lab you can ignore this and just set the minimum required API version to 11

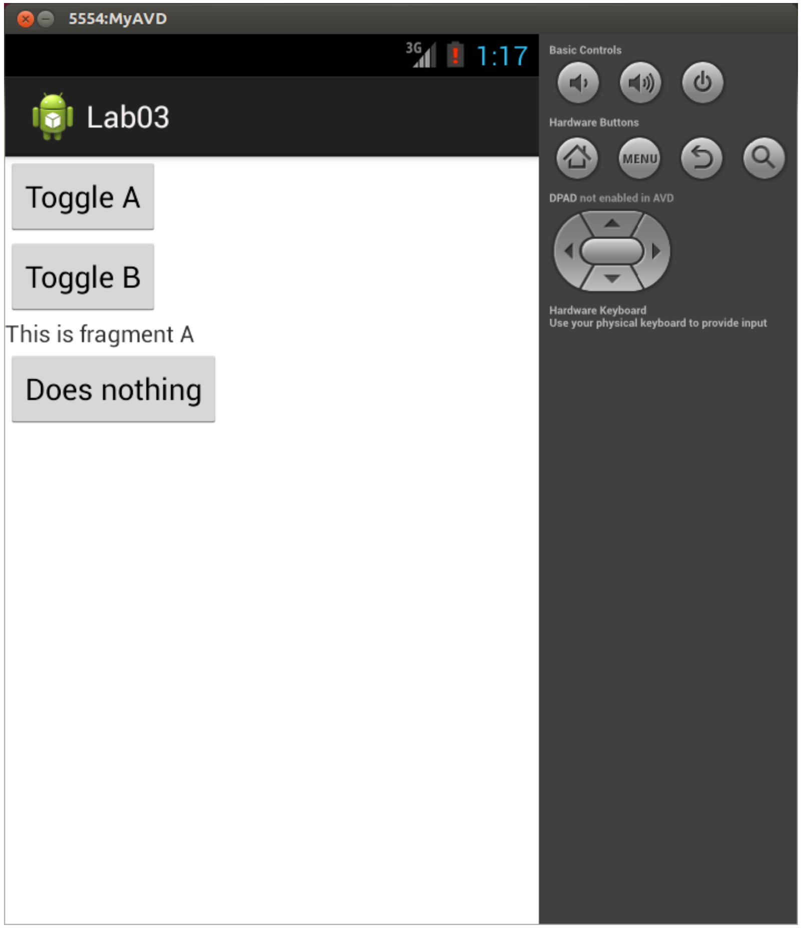
Fragment Toggle

- Initially the UI should have only two buttons with no Fragments visible
- The xml should contain two FrameLayout tags to reserve room for two Fragments, A and B



Fragment Toggle

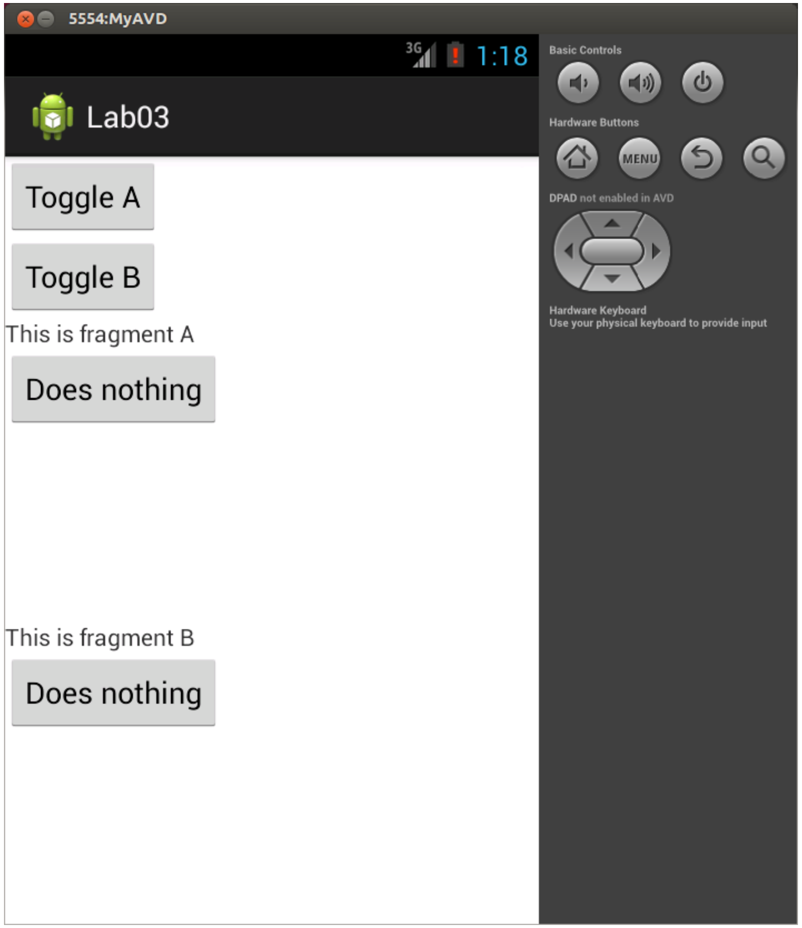
- Clicking Toggle A will cause Fragment A to appear
- The Fragment contains some text and a button that does nothing



This is fragment A

Fragment Toggle

- Clicking Toggle B will cause Fragment B to appear

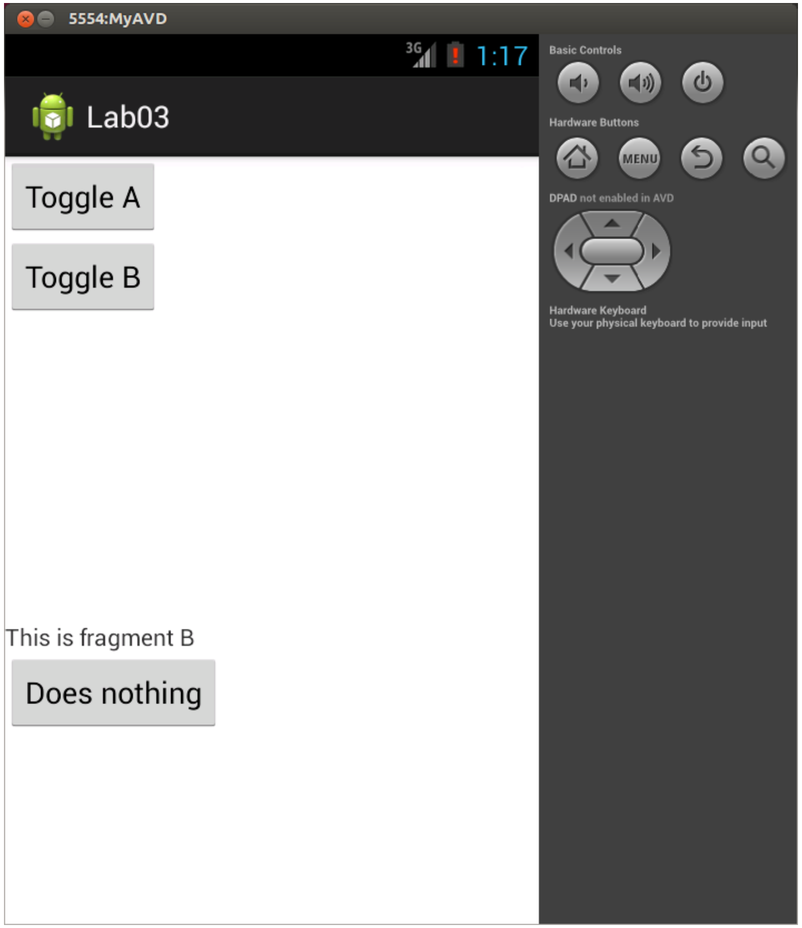


This is fragment A

This is fragment B

Fragment Toggle

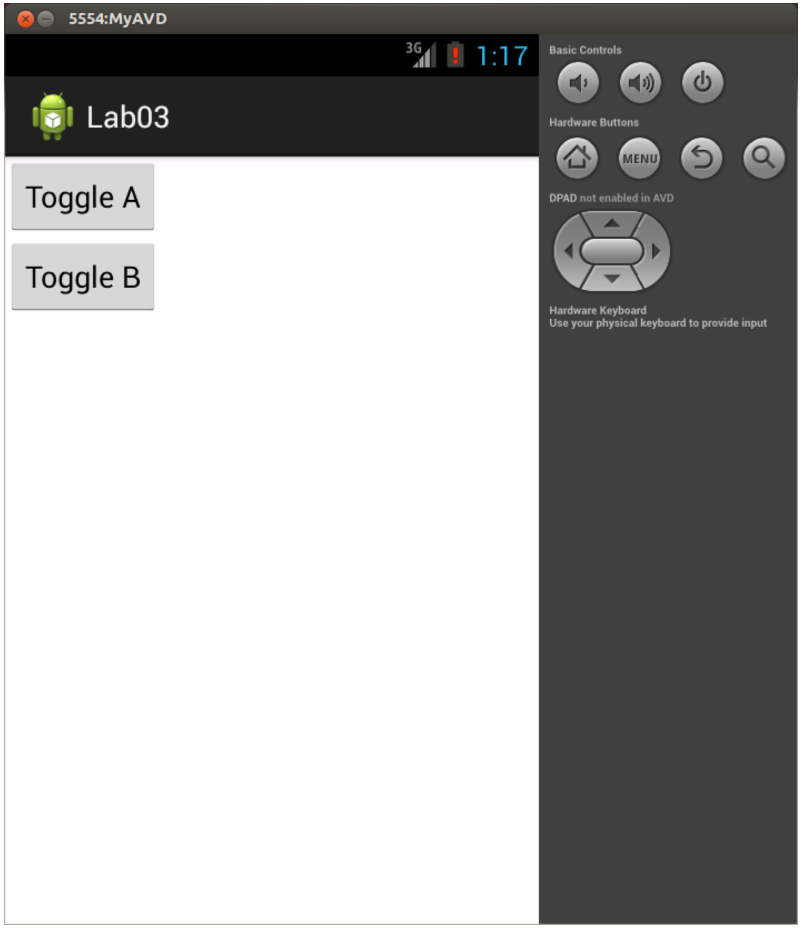
- Clicking Toggle A again will remove Fragment A



This is fragment B

Fragment Toggle

- Clicking Toggle B again will remove Fragment B



Fragment Toggle

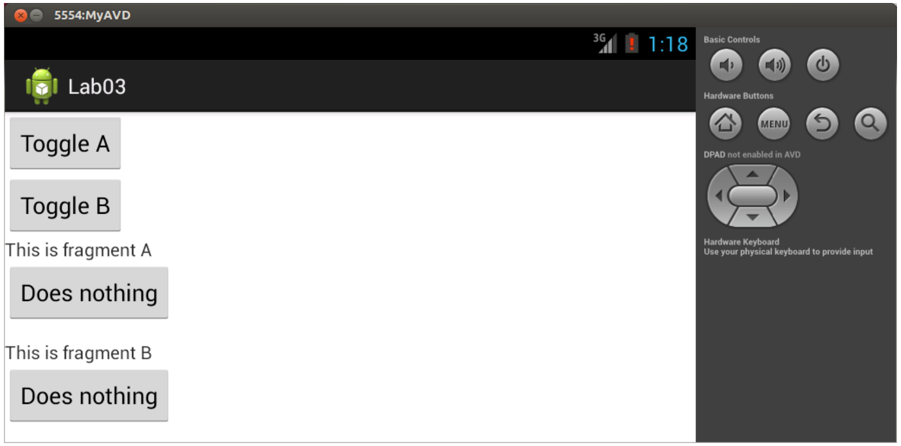
- Fragments are added and removed with transactions generated by the FragmentManager
- You can also use the FragmentManager to check if a Fragment currently exists by using findFragmentById

Back Stack

- By default, when the user clicks Back, the entire Activity will be exited
- In some situations, however, it would make sense for clicking Back to only undo the most recent Fragment transaction
- Add all the Fragment add and remove transactions to the back stack so that clicking Back will undo these transactions

Landscape Layout

- By default, when you rotate the device to landscape orientation, the Fragments and other UI will be arranged the same way as in portrait orientation



This is fragment A

Does nothing

This is fragment B

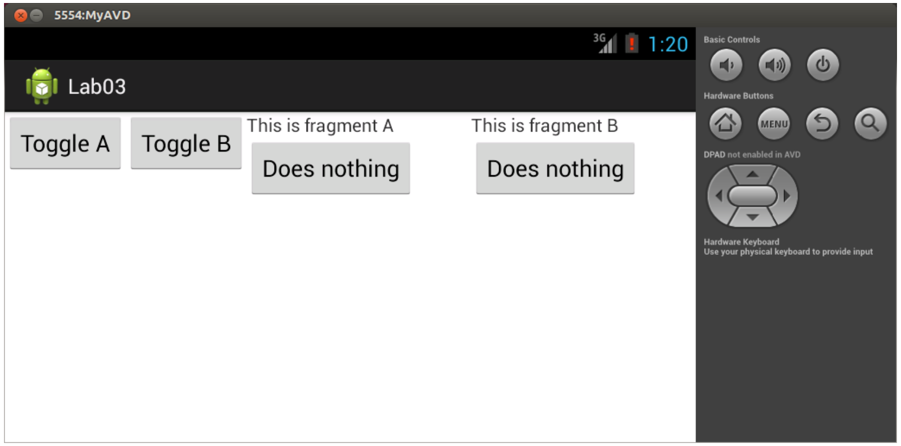
Does nothing

Landscape Layout

- One of the benefits of Fragments is that they can act as modular units that can be rearranged to create a different interface
- The xml files for the default layout are stored in the res/layout folder, but you can create a res/layout-land folder that contains xml files that will automatically be used when the device is in landscape orientation

Landscape Layout

- Create a landscape layout so that the Fragments and UI are arranged horizontally when in landscape orientation, as shown below



This is fragment A

This is fragment B