Android Application Development, COMP 10073

Lab #2 - Overview:

Create an application that switches between 3 activities and tracks how often each activity is visited, and which activity was visited last. The app should handle rotation.

General Requirements

R1: Create a new Android Application using the Empty Views Activity Template. Select API 34 (Android 14.0) as the Minimum SDK. Use a domain naming pattern of ca.mohawk.lastname when you create your project (lastname is sufficient for project name). Include the following statement of authorship in a comment at the top of your MainActivity java code:

```
I, <your_name>, 000123456 certify that this material is my original work.
No other person's work has been used without due acknowledgement.
```

R2: The displayed name ("app_name") on the Action Bar should be changed to the following, with your name and student number. The ActionBar must be visible.

Lab 2 - LastName 000123456

R3: Create 3 activities, each activity layouts is similar and has the following elements:

a) Two buttons using the labels PREV and NEXT for switching between activities.

b) Three TextView widgets for displaying the current activity name, the previous state, and the total number of visits to all of the activities.

c) Each activity should use a different background color. Use partially transparent colours (the A value should be 30%-50%), otherwise the text can be too hard to read.

Don’t use the default textSizes, choose larger sizes at least 24sp. Use the search tool in the Android Studio widget attribute editor to search for **background** and **textSize**.

R4: Avoid using Lambda fns. When the PREV or NEXT Button is clicked:

a) The NEXT button should switch to the next activity, i.e. 1->2, 2->3, 3->1, the PREV button should start activities in the reverse order, i.e. 1->3, 3->2, 2->1.

b) Report the previous state, either just started, or the last activity visited. You MUST pass the previous state to the next activity through an Intent.

c) Track the total number of visits to all activities. You MUST use a getter / setter.

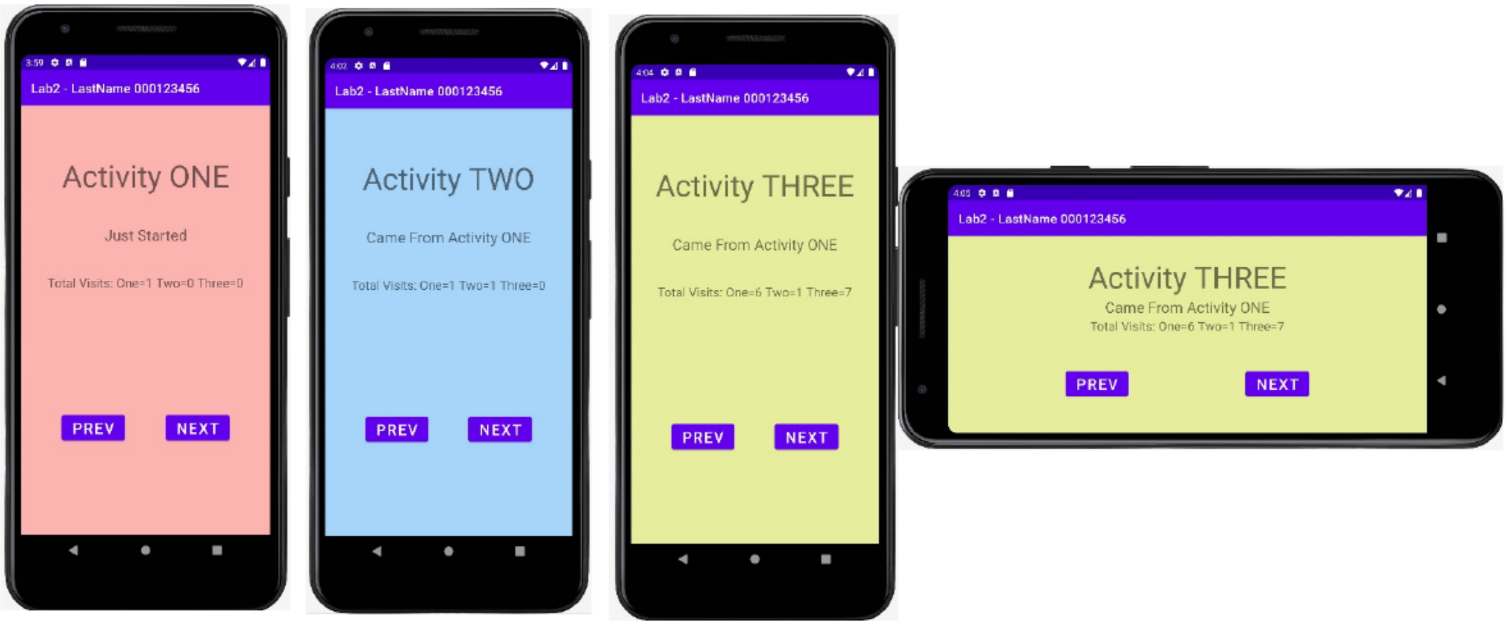
R5: If the Android back button (◀) is pressed visit counts should only go up, never down.

a) Update the visit counts via onResume() using the getter() / setter().

b) Last activity visited should NOT be updated when the back button is pressed, meaning the stale value should still be shown.

Sample Output

Your output should be similar to this example, although you should choose different colors (avoid these shades of red, blue and green), and you may choose any theme.


Lab2 - LastName 000123456
Activity ONE
Just Started
Total Visits: One=1 Two=0 Three=0
PREV
NEXT
Lab2 - LastName 000123456
Activity TWO
Came From Activity ONE
Total Visits: One=1 Two=1 Three=0
PREV
NEXT
Lab2 - LastName 000123456
Activity THREE
Came From Activity ONE
Total Visits: One=6 Two=1 Three=7
PREV
NEXT
Lab2 - LastName 000123456
Activity THREE
Came From Activity ONE
Total Visits: One=6 Two=1 Three=7
PREV
NEXT

Submission

Use Android Studio to Export your project, including the .java code, related .xml files, and all other project elements to a ZIP archive. Do not use Windows to create the .zip file.

Upload your .zip file to myCanvas by the deadline. If Canvas records your upload as late, i.e. you miss the deadline you may demonstrate your application in the next lecture with a 25% penalty.

Be prepared to demonstrate your application to your professor in class. You will have about 1 minute to demonstrate its behaviour. You are expected to demo your work on the lab computers, not your personal laptop, or your hardware Android device.

Grading

Labs are graded out of 4 points. You must meet all requirements to get a full score.

- Canvas upload must be correct and received by the deadline. If the package appears to be too large or too small, missing files, or late, you will lose points.
- The layout must be correct and follow all the details specified in the assignment, and match the sample output as closely as possible.
- The behaviour must be correct. Your application should not crash, it should prevent the user from entering faulty values, and exhibit the behaviour described in the requirements.
- If the app crashes at any point your maximum score will be 2/4.