

Course Name: CMPE 277 – Smartphone Application

Homework 2 – Activity Life Cycle

Student's Name and ID: **Sih-Han Chen / 011498552**

Instructor: **Chandrasekar Vuppalapati**

Semester: **Spring, 2017**

Learning Objective:

The purpose of the app is to understand activity life cycle and the role of Android OS plays in handling life cycle events.

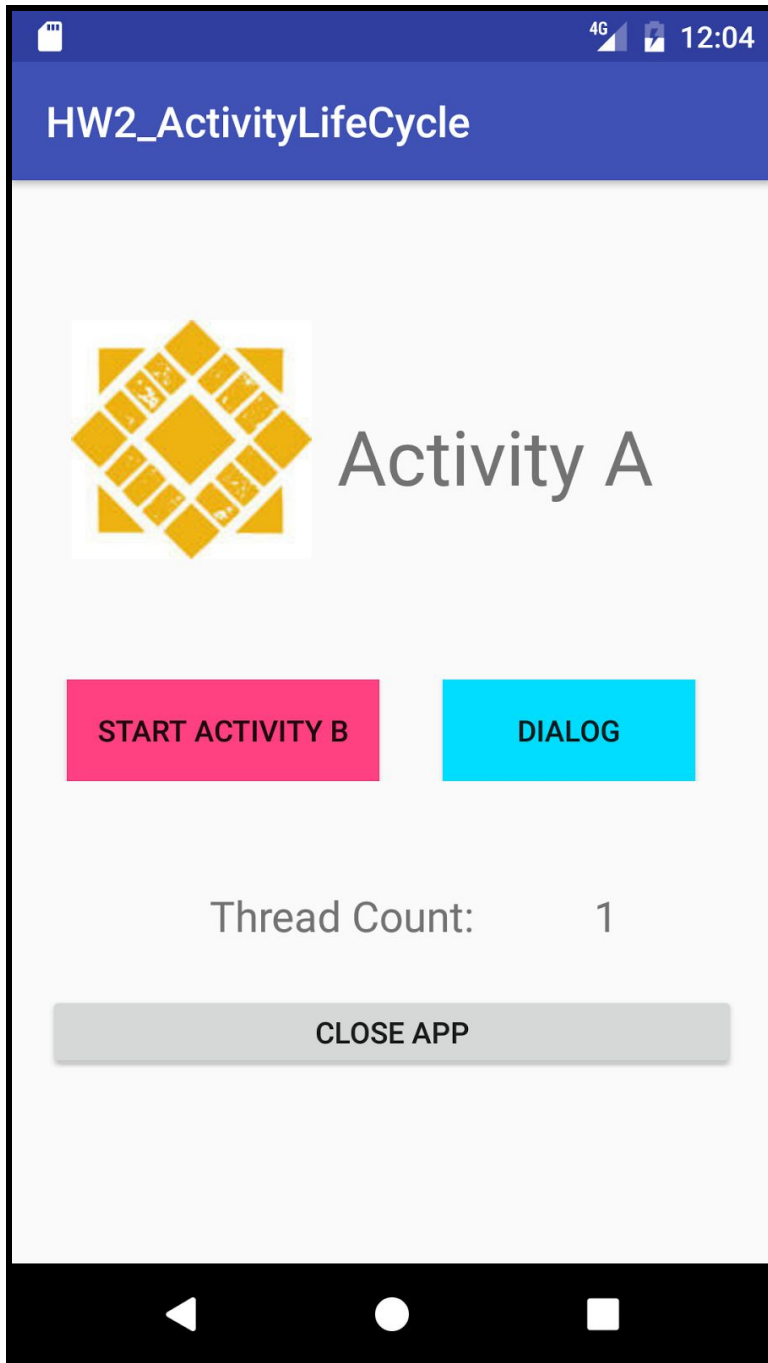
Please submit screen captures of the App and complete source code.

GitHub

https://github.com/stephen-sh-chen/CMPE277_Smartphone_App/tree/master/HW2_ActivityLifeCycle

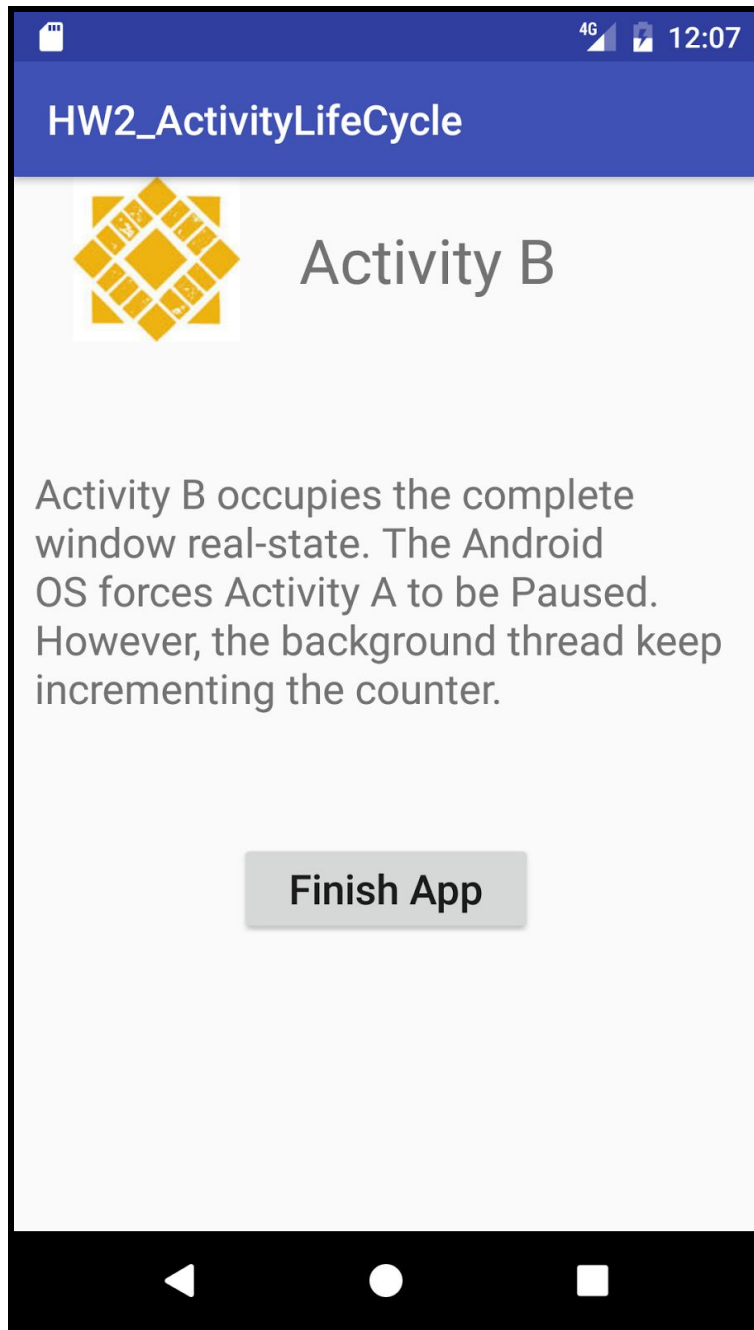
1. Main Activity Page

This initial Thread Count is 1 when the main page first time loaded.



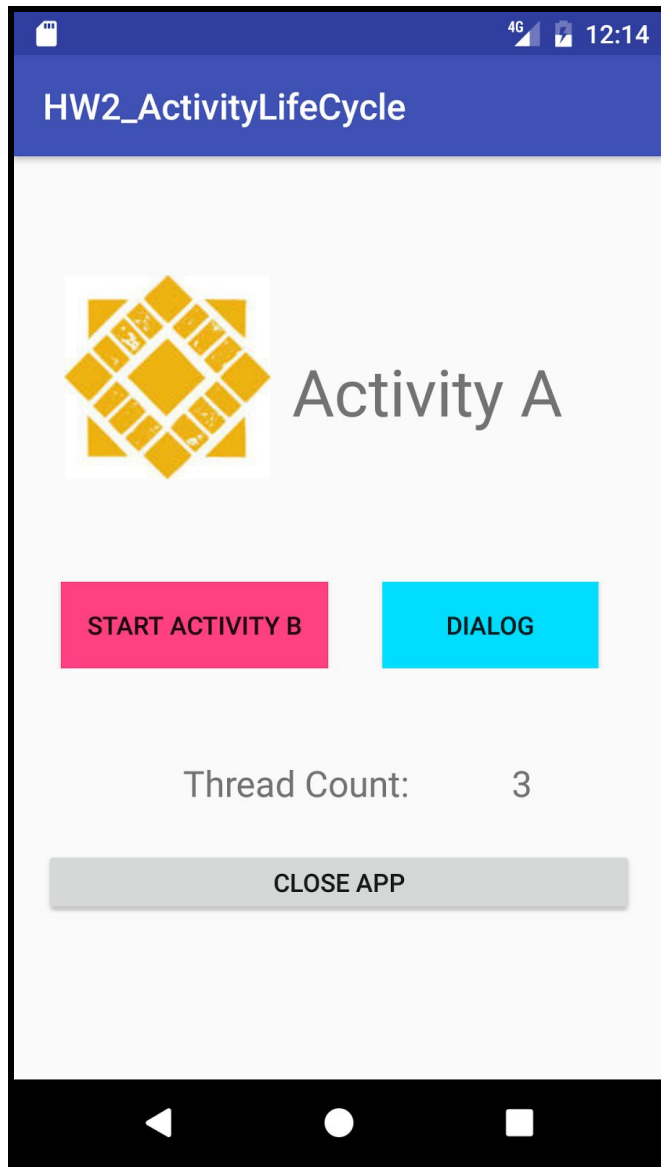
2. Start Activity B from Activity A

On the previous Activity A, we click the button “Start Activity B”, then it will switch to Activity B page. In the background the Thread Count ($==1$) will be treated as a parameter of intent to deliver to Activity B. In the `onResume()` of Activity B, the received thread count which got from Activity A will be plus 1, so the Thread Count now is 2.



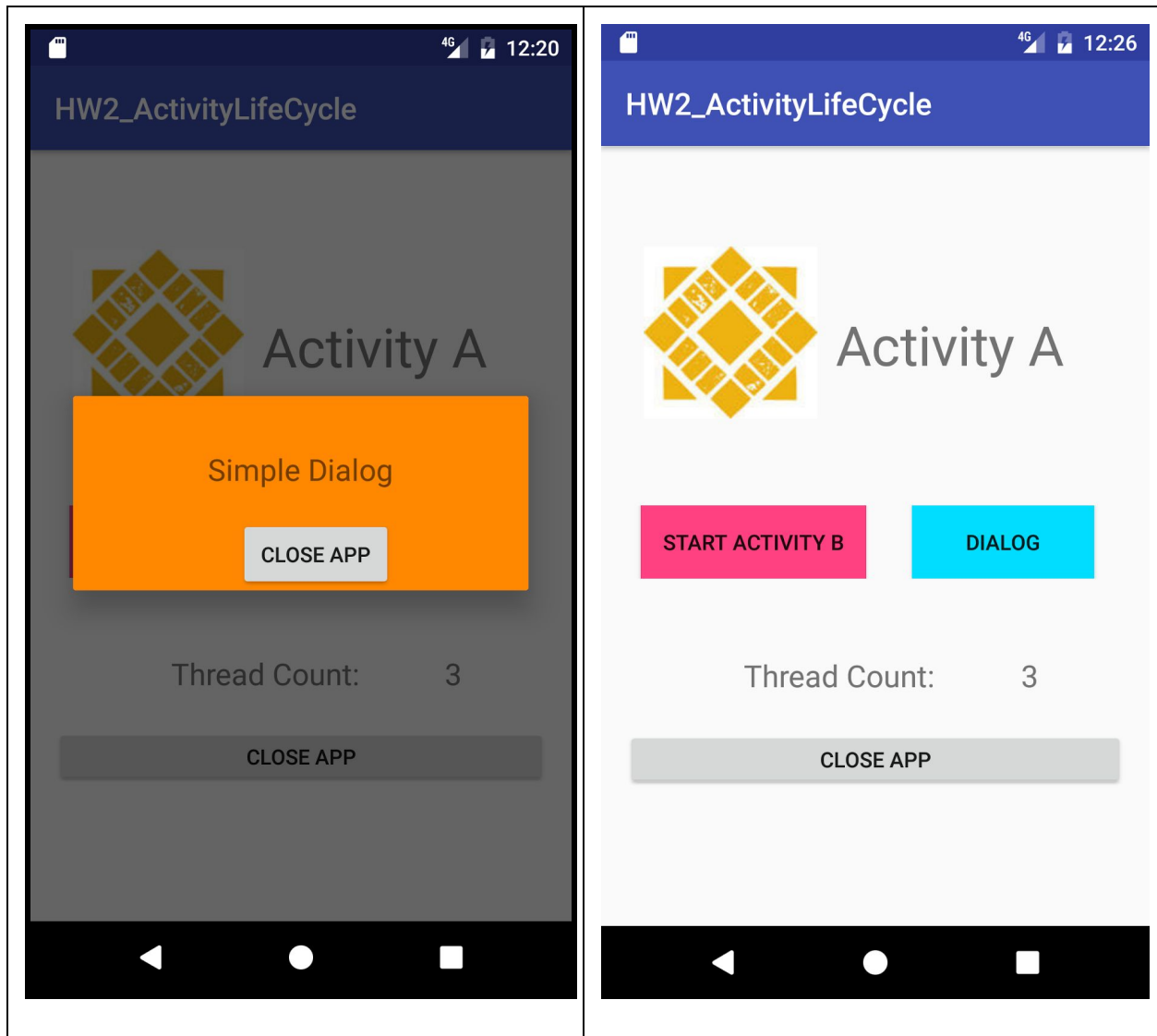
3. Switch Back to Activity A from Activity B

When click the “Finish App” button of the above Activity B. The page will switch back to Activity A. In the background code operation, when the “Finish App” button is clicked, the increased Thread Count ($==2$) will be treated as a parameter to send back to Activity A. And in the `onResume()` of Activity A, the Thread Count($==2$) will be plus 1, so that is why now the Thread Count is 3.



4. Click Dialog Button on the Activity A

Once the click the “Dialog” button on the Activity A, it will pop up a floating dialog activity. As we can see the Activity A is still running in the screen. That means the `onResume()` of Activity A won't be invoked when the Simple Dialog closed. That is why we still see the Thread Count keep as 3 at this moment.



5. Close the App then Launch again

Click the “Close App” of Activity A then launch it again, the thread will be reset as 1.

