

ITIS/ITCS 4180/5180 Mobile Application Development

Homework 3

Date Posted: 09/01/2015 at 12:00

Due Date: 09/05/2015 at 23:55

Basic Instructions:

1. In every file submitted you **MUST** place the following comments:
 - a. Assignment #.
 - b. File Name.
 - c. Full name of all students in your group.
2. Each group should submit only one assignment. Only the group leader is supposed to submit the assignment on behalf of all the other group members.
3. Your assignment will be graded for functional requirements and efficiency of your submitted solution. You will lose points if your code is not efficient, does unnecessary processing or blocks the UI thread.
4. Please download the support files provided with this assignment and use them when implementing your project.
5. Export your Android project and create a zip file which includes all the project folder and any required libraries.
6. Submission details:
 - a. All the group members should submit the same zip file.
 - b. The file name is very important and should follow the following format:
Group#_HW03.zip
 - c. You should submit the assignment through Moodle: Submit the zip file.
7. **Failure to follow the above instructions will result in point deductions.**

Homework 1 (100 Points)

In this assignment you will build a simple Memory game for Android. The game is played on a 4-by-4 grid with 16 square blocks labeled with pictures that are hidden. The goal is to find the pairs of matching images. Images should be downloaded using either a Thread or AsyncTask and loaded into the game board. This is a single player game.

Figure 1, Application User Interface

Part 1 (30 Points): Building the Interface

The interface should be created to match the user interface (UI) presented in Figure 1. You will be using layout files, strings.xml, and drawable files to create the user interface. The layout XML file can be modified through the raw xml, or through the GUI tools provided within Android Studio. To build the UI, please follow the following tasks:

1. Your application should have an application launcher icon, please select your launcher icon to represent your app.
2. The string values used for the text labels, button labels and hints should be read from the strings.xml file and should not be hardwired in the layout file.
3. At the start of a new game, the 16 selected images should be randomly positioned on the 4x4 game board, and a placeholder image should be used to cover each one.
4. When the player first clicks on an image, the placeholder should be replaced to reveal the corresponding image.
5. When the player clicks on a second image, the placeholder should again be replaced with the corresponding image. If this image matches the first uncovered image, the two images should remain uncovered, and a Toast message saying "Match found" should be displayed. See Figure 2(b).
6. If the second image that is uncovered does not match the first image, a Toast message saying "Not a match" should be displayed, and there should be a 3-second delay before covering both images with the placeholder image. See Figure 2(c).
7. When the player has matched all pairs of images, show an alert indicating that the user has found all the matches. See Figure 2(c).
8. Clicking the "New Game" button should create a new game board by reinitializing the game.
9. The application should be destroyed if the "Exit" button is clicked.