Project 1 05-4/633 B
Due: Wednesday, September 23rd by 11:59 pm SSUI Mobile Lab

PROJECT 1

Project Overview

For this project you will create a simple puzzle game. The puzzle consists of tiles of images and one empty tile. The player plays the game by selecting/pressing on tiles that are adjacent to the empty tile, which moves the tile into the empty space leaving the tiles original space empty. The game should implement the following features:

1. Moving of the tiles as described above.
2. Score tracking, which increments each time the user presses on a tile and displays this information to the player.
3. Starting a new game after the player presses on the “New Game” button. This should also randomize the puzzle tiles.
4. A message that tells the player she won after solving the puzzle.

PART 1: Designing the UI of Your Puzzle Game

Goal

Sketch the user interface for your puzzle game and specify the springs and struts layout for your interface. This will help you implement the layout in part 2 of the assignment.

Project Overview

This part of the project will help you plan your UI for the puzzle game.

Grading

In this part of the assignment you start with 100 pts, but some examples of where points will be deducted (down to minimum of 0 pts) include:

Nothing submitted	-100 pts
Everything needs to be attached to a spring or a strut (can’t have two things side by side in a row, for example, or above another in a column, without a spring or strut between)	each -10 pts
Every container with multiple elements that has no springs or struts between them	each -10 pts
The window has to start with a single container that contains everything else. So the outermost container must be a row or a column	-10 pts
Every row/column needs to have specification of the alignment in the opposite direction	each -5 pts
Any required UI elements are missing	each -10 pts

Turning Your Program In
The part of the assignment is due Wednesday, September 23, 2015 at 11:59 pm together with part 2. Create a document containing the images of your sketches in either a PDF or Microsoft Word (.doc, .docx, or .rtf) file. Name the document p1_LASTNAME_FIRSTNAME (for example, p0_banovic_nikola.pdf). Then include the document in your project file at the same directory level as your README.txt (see next part).

PART 2: Puzzle Game Implementation

Goal
Make a simple puzzle game using the Android framework.

Files Provided

P1.zip	Android Studio project starter code (will be available online on Blackboard).

Part Overview
As mentioned above, for this project you will create a simple puzzle game. The puzzle consists of tiles of images and one empty tile. The player plays the game by selecting/pressing on tiles that are adjacent to the empty tile, which moves the tile into the empty space leaving the tiles original space empty. The game should implement the following features:

5. Moving of the tiles as described above.
6. Score tracking, which increments each time the user presses on a tile and displays this information to the player.
7. Starting a new game after the player presses on the "New Game" button. This should also randomize the puzzle tiles.
8. A message that tells the player she won after solving the puzzle.

Note: Please follow this naming convention for the application and project names in Android Studio: p1_lastname_firstname, for example "p1_banovic_nikola"

Bells and Whistles
Completing the above requirements and having properly documented code with no errors will get you 90 out of 100 points, which is an A-. You must complete one of the below 'bells and whistles' to get full points. You can complete as many of these as you want, but you will get no more than 15 points total. Different bonuses are worth different numbers of points, based on difficulty. You may receive up to 15 points for bells and whistles giving you an option to earn 5 bonus points. Be sure to let me know in the README which of these you chose to implement.

1. (5 points) Show last move button, which highlights the last tile that the player moved to the empty space.
2. (5 points) Show puzzle button, which shows the solved puzzle (the final image) to the player and allows the player to return to the puzzle afterwards.
3. (10 points) Implement functionality that allows the player to replace the default puzzle image with another image. Note that you will have to plan early in your "software design phase" to include this feature.

If you can think of something else cool that you'd like to do for this section, let me know and I'll let you (and the rest of the class) know if it's an acceptable option for this portion and how much it's worth.

Show and Tell

The course instructors will be picking the nicest game implementations and show them to the class. They may also be put on a public website. Please let me know if you do not want your solution shown in the README file in your assignment.

Turning Your Program In

The program is due Wednesday, September 23, 2015 at 11:59 pm. You should turn in your assignment on Blackboard by going to the assignment link, attaching a zip file with the contents below, and pressing SUBMIT. Please write a README.txt file which mentions any online sources you used to help with the project, as well as any notes about your project (i.e. if you couldn't get a particular part of the project to work). Create a zip file that contains the entire program directory from your workspace, and the README.txt file. Then, name the file p1_lastname_firstname.zip. For example:

p1_banovic_nikola.zip

Once you have made this zip file, go to the assignment page on blackboard, attach this file, and submit.

Grading

This part of your assignment will be graded as follows:

Turn in is correct (has required features), compiler code has no errors or warnings	30 pts
Puzzle game layout is correct and displays and updates in response to user input properly on variety of (virtual) devices	30 pts
Puzzle game functions correctly for any user input	10 pts
Errors are correctly detected and handled	5 pts
The user is properly notified when an error is encountered	5 pts
Bells and whistles	5 – 15 pts
Comments and Coding style	10 pts

Format your code to be readable (e.g., use indentation) and to make it easy to find methods and other declarations (e.g., via white space and/or explicit separators). Include clear and descriptive comments in your code that make the function of your code apparent.

Note 1: You will be graded on the quality of your code and commenting. In particular, submissions that do not contain clear and useful comments throughout their code will be penalized. For coding style, this also includes factors such as modularity, sensible method and variable names, and overall clarity. You may find this reference useful: http://particletree.com/features/successful-strategies-for-commenting-code