Nick Mariasi

Final Project: Mobile Computing and Programming

Statement of Idea/Target Users

The application I developed is a simple coding concepts guide for three different languages: C++, Java, and Python. In each language, there are 3 different concepts that are explained: different types of loops, input methods, and output methods. At the end of each tutorial, a small test can be taken which is basically 3 questions with a hidden answer that must be revealed by pressing a button once you guess what the answer might be. Intended users for such an application would be beginner programmers so they can learn the very basic concepts or intermediate programmers looking for a short description to refresh their memory on these simple concepts should it slip their mind.

Basic Features:

1. The user interface is clean, with simple button navigation and a menu for when the user is in the fragments should they want to navigate to a different section or back to the homescreen/about page.
2. There are a multitude of java classes that implement the model-view-controller design
3. There are a lot of drawable resources used. Initially I had just 3 for the main screen, but all the fragments now contain drawable resources to display the information
4. All the buttons in the program respond to touch in addition to the fragment’s menu bar responding to swipes left and right to scroll through the option (depending on screen size).
5. In all the fragments, if you click on ‘continue to test’ and wish to view the information again, clicking the back button will restore you back to whatever tab you were on in the tutorial
6. I used a Handler to implement the test feature as with the handler I was able to only show the answer if the user clicked the ‘reveal answer’ button

Elective Features:

1. Fragments: I used a total of 9 fragments in this application, 3 for each of the tutorials which made navigation much easier and look much better than just using buttons to navigate between sections.
2. ActionBar and Menus: I used a menu system that’s active in each of the fragments for better navigation. Should the user’s back button not work, or they just don’t want to use it then, the menu in the top right allows for navigation to the homescreen, about screen, and the different tutorials. In fact, the **about** screen is only accessible through the menu bar.
3. Audio: Audio was implemented through the sound that the buttons make when clicked on the homescreen. If you click the C++ button, the sound it makes is going to be different from that of the Java or Python button.