Assignment 2 – KimsGame - PG4600-14 Anders Mikkelsen

Screenshots:

/docs/screenshots/

UML:

/uml/

Javadocs:

/docs/javadocs/index.html

References:

Hiding of Soft Keyboard:
     - [Hide Soft Keyboard](#)

Kims icon:
     - [Kims Ridiculous Icon](#)

Code Style:

[Google Code Style with 100 char breadth.](#)

Design:

After last assignment Ive been thinking about how to properly use activities and fragments. After working a lot with MVC in web lately I figured I wanted to try that approach for android as well. So my utils are my models, my activites are my controllers and my fragments are my view. That way an activity is not locked to a certain layout and I can easily modify or add any fragments at my own leisure if I want to change / modify the program without breaking earlier code. My activities are more like handlers since they are directly coupled with logic in my fragments but I think that is in line with what I perceive to be the design philosophy for Android. So now in my activities I can just swap between the fragments I need! No messy cleanups needed. I think I made a pretty strong design decision in this assignment, because it will be very easy to extend!

Implementation:

This was a bit different than my last assignment so I had to think abit about how I wanted my fragments and activites coupled but it was pretty straight forward from then on. I have interperated your mention of "getting information from a database" as being applicable to a Content Provider and I wanted to show I can use them so I get the word information from the contact names as I supsect they will never be empty and will have a wide variety of names and I store and retrieve results in an application database.

I made this app "rotation proof" by locking to portrait as I believe I successfully showcased that I am capable of handling rotation in my last assignment.

I also finish the activity when it is done because there is no reason to go back in this app.

Known Weakness:

Rounds and points are stored in sharedpreferences. This is to show I know how to use it, I could store them in variables and transfer with an intent. This would be safer as a user could edit the SharedPreferences file during play and modify the round / points values. On the other hand this is a single player app and keeping it in sharedPreferences is more robust.

I interpreted the "exit" button as only available on the high score screen. You might have intended for this to be available on all screens. The user can exit with the back button in any other scenario anyway.

The game will reset every round if you close the app without resetting to round 1. However it will not help you if you exit when you are supposed to enter a highscore and it also introduces a weakness where a savvy user could go to the guess screen and exit if he doesnt know the word to get a free new try by opening the app again.

Tests:

Implemented some simple tests for experimentation. Testing wasnt part of the assignments as far as I understood, although they are important!