File Name (i.e. LoginController.java): DashboardView.java

Author(s) (i.e. Risa (frontend), Kris (backend)): Risa (frontend)

Reviewer(s) (i.e. Ashley (design)): Heming (frontend)

yes no Buttons are in line with the design spec in terms of functionality  
 Comments:

Conference info button does not work. None of the action bar buttons work.

yes no EditTexts are in line with the design spec in terms of functionality (input validation feedback)  
 Comments:

Click here to enter text.

yes no Images are in line with the design spec in terms of pixels (for different screen sizes)

yes no 480 x 800

yes no 480 x 854

yes no 720 x 1280

Comments:

All images are off center in the WXGA screen.

yes no Text is in line with the design spec in terms of pixels (for different screen sizes)

yes no 480 x 800

yes no 480 x 854

yes no 720 x 1280  
Comments:

“Opencomm” needs to be moved to the left. Again, the text for the four icons are off center in the WXGA screen.

yes no EditTexts are in line with the design spec in terms of pixels (for different screen sizes)

yes no 480 x 800

yes no 480 x 854

yes no 720 x 1280

Comments:

Click here to enter text.

yes no Classes are commented with a javadoc comment at the top  
 Comments:

Click here to enter text.

yes no Methods are commented with a javadoc comment above them  
 Comments:

Click here to enter text.

yes no Comments exist within methods to show the flow of code  
 Comments:

Click here to enter text.

yes no All exceptions are handled appropriately: don’t ignore exceptions, and don’t catch generic exceptions

void setServerPort(String value) {

try {

serverPort = Integer.parseInt(value);

} catch (NumberFormatException e) { }

}

// Don’t do this!!

try {

someComplicatedIOFunction(); // may throw IOException

someComplicatedParsingFunction(); // may throw ParsingException

someComplicatedSecurityFunction(); // may throw SecurityException

// phew, made it all the way

} catch (Exception e) { // I'll just catch all exceptions

handleError(); // with one generic handler!

}

Comments:

Click here to enter text.

yes no All imports are fully qualified (i.e. import foo.Bar; not import foo.\*;)  
 Comments:

Click here to enter text.

yes no All methods are 40 lines or shorter  
 Comments:

Click here to enter text.

yes no All lines are 100 characters or shorter  
 Comments:

Click here to enter text.

yes no TODO comments are properly added and assigned to the correct team/team member

Comments:

Click here to enter text.

yes no Standard Brace Style and Indentation is used (i.e. Ctrl+Shift+F)

Comments:

Click here to enter text.

yes no Used enhanced For Loop syntax (i.e. For-each syntax)

Comments:

Click here to enter text.

yes no All constants are stored in an XML file or labeled as “static final”

Comments:

Click here to enter text.

yes no Log TAG and Debugger Boolean is generated and stored at the beginning of the class

Comments:

Click here to enter text.

yes no Proper Log classes are used (i.e. if (D) Log.d(…) for logs implemented for debugging purposes)

Comments:

Click here to enter text.

yes no All onClicks handled in XML file (android:onClick) and method implemented in class

Comments:

Click here to enter text.