Peer review of group 10

Do the design and implementations follow any design principles?

Looking through the code you will see it follows the Single Responsibility Principle with how the classes are built, very class taking care of one thing. The code also follows the Liskov Substitution Principle, given how the classes Notebook and Book extends Item.

Consistent coding style?

The code style is consistent since it uses the MVC pattern. Since the code is divided like it should be the model-classes contains the same type of code and the controller-classes contains the same type of code.

Is the code documented?

Yes, for the most part, especially for the methods. There are comments in all the classes, except MainActivity and MassagePage. Would like to see more comments about the classes, like what they do.

Are proper name used?

Looking through the package there are three classes, whose names may not give away exactly what they do. The classes are ListingAdapter, ListingPageActivity and Listing. But after looking into the different classes and looking at the methods you get a grip of what they do.

The names of the methods and variables are good and easy to understand.

Is the design modular?

The code is to some extend modular. The code has no circular dependencies , since the code is divided into smaller classes that does one thing. There are also no unnecessary dependency in the code, thou there are some strong. Both Book and Notebook extends an abstract class Item. Having looked at the code, the dependency makes sense.

Is the code well tested?

The model package is well tested, but since the group write themselves in their SDD that there are some functionality in the ControllerView package, it should also be tested.

Are there any security problem?

One security problem that was found were that if the user were to write two different passwords in the register page, the user would still get logged in.

We also tried sending in an ad without a picture, which made the program exits and the ad were not saved.

Is the code easy to understand?

The code is easy to understand since its names of methods, classes, and variables. The classes are relatively small which makes it easy to understand what the class are supposed to do and therefore also how the methods work. Since the code follows the MVC pattern it is also easy to understand what the different classes are responsible for.

Does the code have a MVC structure?

Yes, somewhat. The Model are separated from the View and Controller. The controllers and views are in the same package but seeing how the code is structured it makes sense to have them in the same package.

Can the design or code be improved?

The design is very easy on the eye and the code looks good, so no.