**QA Concept**

In our quality assurance concept, we want to target 3 main things: the efficacy of our code, the documentation of the methods and processes and the total lines of code per class. These 3 main things will help us understand our code better, thus making it easier to improve the project moving further. The total lines of code will also help us visualize how we’ve progressed through the analyzed points in time.

The *efficacy* of our code will help us in realizing what parts of our code is redundant. As we are a group of 4 people, with different coding styles and backgrounds, it is inevitable that some parts of code will be redundant and not put to efficient use. The provided code coverage will help us in identifying these parts, so that we can look at them critically and make sure we can code more efficiently in the future. It will also help us work better in a team, as we can analyze each other’s styles and tendencies.

The *documentation* analysis will show us how well we’re commenting our code and describing each process, method and class. Documentation is extremely important, especially working in a team, as the other team members need to be able to understand what each individual was trying to achieve. We all write code quite differently, and as such, analyzing how many comment/javadoc descriptions have been written is very useful to see what methods and classes will need to be improved.

The *total lines of code* metric will show us which classes we have invested the most work in and will also show the average length of classes. This is very useful to have an oversight of all classes and to see if there are classes where things can be shortened/branched off. It will also help keep track of our project through the the timeline and to compare with our GANTT Project Timeline. It is a good way to see how well we’ve stuck to our project plan and if the parts of code were expanded around the time they were supposed to be expanded at.

***Code Coverage***