Metrics Tracking for JXI

Product Metrics

- Test Cases / Method
  - This metric will help track the “coverage” of our code. It is a very crude measurement of code coverage but we will control quality through well-designed and well-written code and tests. Our goal is to have at least a 1:1 test to method ratio.
- Total Number of Tests
  - This metric will track the sheer volume of tested code in our system. Ideally, this number will be much greater than the number of methods we write.
- Methods / Class
  - This metric will help us ensure code quality. We will attempt to reduce the number of methods per class in order to keep code simple and readable. The goal is to have less than 20 methods/class.
- Reek (code smells)
  - This metric will keep track of how many code smells remain in our code during each release. This will help us track how well our code is written and designed. The goal of our application is to maintain less than 27.6 code smells for the application.
- Saikuro (Cyclomatic Complexity)
  - This metric will focus on paths through our code. Based on the paths, it will assign a cyclomatic complexity score to each method, and will illuminate where inelegant code lies. The goal for each release is under a complexity of 2.2.
- Flog (Cyclomatic Complexity)
  - This metric also measures cycloamtic complexity, but uses an “ABC” (Assignment, Branching, Calls) calculation method. This metric will assign a score to each method, and an average score for the application. This slightly more in-depth look at the code will ensure we are not producing unnecessarily intricate code. The goal for each release is under a complexity of 10.1.

Communication Metrics

- Client Communications per Week
  - This will involve tracking the number of times that the client is contacted (via

meetings, discussions, e-mail, etc.) about the project.

Unless otherwise stated, each metric will be tracked weekly via running “rake metrics:all” on the project.

The goals for our metrics are not random, they are community averages of rails applications, of which we want to be better than most.