Project Proposal

Marc Fervil

Context

When developing software, the developers must take several metrics into account in order to ensure that the project code is usable, functional, readable, and portable. Without software metrics, medium to large sized projects have the potential to become unmaintainable behemoths that frighten even the most competent developers. These metrics include but are not limited to weighted methods per class, depth of inheritance tree, number of children, coupling between objects, and lack of cohesion methods. Due to the fact that these metrics are correlated with the quality and maintainability of a project, several tools have been created to help us determine these metrics.

Motivation / Problem

At this point in the class we have used both Understand and Designite to generate software metrics for various projects. However, my largest gripe with both tools is that neither provided an intuitive interface to generate and analyse these metrics. Designite was fairly easy to use, but it lacked any form of interface and you were limited to analysing the metrics in a spreadsheet. Designite did however, provide easily understandable metrics that the average user could understand. While Understand did have *some* interface, I was not able to immediately grasp the steps that one needs to take to generate the metrics, analyse the metrics, and interpret the metrics.

Expected Contributions (Solution) and objective

My proposed solution to this problem is to create a basic python metrics generation tool that has an extremely intuitive UI/UX . The full scope of this solution is as follows:

1. **[Objective 1]** Generate useful code metrics for a javascript project
    a. Lines of Code
    b. Depth of inheritance tree
    c. Number of children
    d. Weighted methods per class
    e. Cyclomatic complexity
2. **[Objective 2]** Intuitive UI
    a. Analytics view (metrics for a class)
    b. Package view (view metrics on a package level)
    c. Average view (compare class metrics vs average metrics)
3. **[Objective 3]** Graph view
    a. Provide a graph for any given set of metrics
    b. View to compare graphs side by side

Task Management

	Week 1	Week 2	Week 3	Week 4
Objective 1				
Objective 2				
Objective 3				