

In the context of the expertise group Software Engineering & Technology (SET) and the Laboratory for Quality Software (LaQuSo), both of the TU/e Department of Mathematics and Computer Science, the following M.Sc. thesis proposal is available for students with experience in programming, solid knowledge of statistics and a keen interest in software metrics, maintainability, statistical methods and tool development. The project will be supported by

Supervisor dr. Alexander Serebrenik (a.serebrenik@tue.nl)

Tutor drs. Reinier Post (r.d.j.post@tue.nl)

### **Software Metrics Validation using Version Control**

Maintainability of a software system can be understood by means of efficiency of the development effort: how easy is it to modify a system in order to implement new functionality or fix a bug? Numerous maintainability metrics can be found in software engineering literature. In this graduation project we focus on validating the metrics proposed. To this end we look for relations between the values of maintainability metrics, on the one hand, and the development effort (expressed as e.g., number of changes or number of lines of code affected), on the other hand. The underlying idea is that should such relations be found, we will be able to estimate the development effort based on the maintainability metrics.

The graduation project proposed will consist of the following steps:

1. Based on the literature study list relevant software maintainability metrics (e.g., McCabe's complexity or the maintainability index).
2. Based on the literature study list development effort measurements (e.g., number of changes or number of lines of code affected by the changes).
3. Chose available tools for computing metrics listed in (1), and assessing development effort measurements listed in (2). The tools should be applicable to industrial-size systems.
4. Validate correctness of the results obtained using the tools chosen in (3).
5. List requirements that should be satisfied by open source software systems to serve as a benchmark. For each project considered a significant number of software versions should be analyzed. Moreover, since we are interested in validating the existing metrics rather than developing the new ones, projects should be chosen in function of availability of tools in (3).
6. Select a number of open source software systems.
7. Apply the tools developed in (3) to the open source systems selected in (6).

- ★
8. Study correlations between the maintainability metrics and the development effort measures on across different systems. To this end such tools as R can be used and both strength and significance of the correlation should be studied.
  9. Study correlations between the maintainability metrics and the development effort measures on different versions of the same system. Can one observe change points the in metrics-measurements correlation?
  10. Study possible threats to validity of the results.

Special attention should be paid to applicability of the tools and techniques developed or used to industrial-size software systems, as well as to the reusability of the tools and techniques. In particular, tools developed in (3) should be easily replaceable by alternative implementations without compromising (8) and (9).

