

Project preparation/proposal

The following enumeration are the sections that you should include in your proposal.

1. Team Formation

List the team members and their basic responsibilities/roles (a table is preferred).

2. Type of the study

Type 1: Correlation of metrics with other software artifacts/activities. List your hypotheses (Note that you may mention more than one hypothesis in the draft. You are not committed to implement all of them in the final project). Then you can build a linear regression model to explain the relationship between a dependent variable and one or more explanatory variables (independent variables).

Type 2: Examine the evolution of metrics by mining repositories. For example, you can use the **design flaw detection strategies** proposed by Marinescu¹ to detect problematic classes or methods.

3. Related studies

Describe very briefly the relevant studies that you found in the literature. Provide references in the **References** section.

4. Projects

List the projects that you will analyze in the final project (Again you may list more projects than those that you will actually analyze in the final project). Justify their selection based on your goals (i.e., examined research questions). Give details about:

4.A the versions/revisions that you selected

4.B the presence of complementary software artifacts that you will use in your study (e.g., bug reports, feature requests)

4.C additional tools that you may use for extracting information (e.g., applied refactorings)

How many systems to study?

At least 3 systems, from different sw. ecosystems (Apache, Eclipse, etc.) for standard applications. Mobile/embedded applications should contain larger datasets, since these systems are typically small in size. (20+)

In case of system evolution analysis, you are required to study at least 5 release versions, preferably 10.

Characteristics/constraints of the studied systems?

- Open-source systems are preferred over proprietary systems, unless a justification is provided for studying proprietary systems.

¹Radu Marinescu, "Measurement and Quality in Object-Oriented Design," PhD thesis, Politechnica University of Timisoara, Romania, 2002.

- Active projects (At least one commit in the last 30 days)
- More than one developer/contributor
- Size over 100K SLOC

Quality of the system studied.

It is expected that you study projects that are used by several users, and not prototypes or school assignments.

For example, if you are studying projects coming from Github, you could select projects that have at least 20 stars ,or number of forks that a project has.

You could also propose to study existing research datasets. For example: Evosuite SF110 (or a representative subset of it) if you were interested on study automatic test case generation. Or any relevant research dataset. (e.g., PROMISE).

<http://promise.site.uottawa.ca/SERepository/datasets-page.html>

5. Metrics

List the set of metrics you are planning to implement and use in your study. Include references to papers that contain a formal definition for the selected metrics.

In general, you expect to define a set of dependent (control variables)

Product metrics (e.g. size and complexity)

Process (code churn, change entropy)

Human factors (total authors, author ownership)

And one or more metrics for each research object of interest (RQ or Hypothesis)

These could be external quality metrics (maintainability, testability) non-functional attributes (performance)

The minimum number of metrics to be analyzed in total is 6

See the following reference for inspiration:

McIntosh, S., Kamei, Y., Adams, B., & Hassan, A. E. (2014, May). The impact of code review coverage and code review participation on software quality: A case study of the qt, vtk, and itk projects. In *Proceedings of the 11th Working Conference on Mining Software Repositories* (pp. 192-201). ACM.

6. Resource planning

Create a Gantt chart with tasks and the team members that will work on them. The tasks may (should in my opinion) overlap. For example, while the source code is developed, some team members could work on studying about statistics and exploring statistical tools, or writing the related work section for the final report. The parallelization of tasks and the efficient team management are key factors for a successful project.

7. References

Use IEEE citation style²

² [IEEE Citation Reference](#)