Thesis Proposal (MA)

A Way-Back Code Analysis Machine

Outline

Did you ever contribute to a bigger software project? Did you write tests that do not only produce binary results (pass or fail)? Performance evaluations are such kind of tests. Results of a performance test have a meaning when having an isolated look at them: an overall latency of 70ms for a task means that the task is perceived as interactive by a human. However, when looking at a series of results in correlation to code changes at the development of a software, a relative meaning is added: it becomes visible which impact certain code changes had on the overall latency of a system. In an agile software development process such a knowledge can be used for continuously optimizing the latency of a software.

The testing of a software is highly dependent on the test scenarios. A problem when adapting test scenarios at a certain point in time is that the historic dimension is missing for the data evaluation. Only test results from the time after the test was designed are available. With version control systems such as GIT or SVN, a software project brings its code history. The goal of this thesis is to develop and implement an automated test workflow that runs a test scenario over past versions of the software project to gather historic data from the software versions before the test was designed.

As a second focus, suitable representations for the obtained data will be analyzed and developed. Graphs plotting the measured values over the revision history could already be a suitable visualization.

Possible Structure

- Analysis
  - Analysis of testing tools and methodologies, version control.
- Related work
  - What do other projects do that answer related questions?
- Design
  - Which components do you need?
  - Which are options for the design? Why are your choices good?
- Implementation
  - Relevant details such as frameworks used.
- Evaluation
  - How well does it work?
    - Metrics!

Requirements

Curiosity, Joy to work in a team, Knowledge in Java.
Ability to write good code (including unit tests and documentation).

Contact

If you are interested, please send an email briefly explaining why you think to be the right person for this thesis to:

Marc-Oliver Pahl (pahl@net.in.tum.de)
http://s2o.net.in.tum.de/

Distributed Smart 2pace Orchestration System

ds2os.org/