Vrije Universiteit Brussel

Proposal bachelor thesis

Title: Detecting Design and Implementation Defects in Chamilo

Promotor: Coen De Roover

Includes preparation course: Yes

Context

Chamilo (http://www.chamilo.org/) is open source e-learning and collaboration software used by universities around the world. It is known as Pointcarré (http://pointcarre.vub.ac.be) at our own.

Reliability (i.e., absence of bugs) and maintainability (i.e., effort required to act upon change requests) are important quality attributes of the open source code. Several tools have been developed to evaluate the design and implementation of software systems on these attributes, as well as to prevent both from deteriorating over time:

> *Smell detector*s, for instance, measure aspects of a system's design and warn about abnormal values called bad smells. Examples include excessively long methods, classes accessing more fields from others than of their own, or a class that is referenced by every other class in the system. *Bug detector*s, on the other hand, scan the code for mistakes that developers often make such as using an assignment = rather than a comparison == operator in a conditional. Finally, *clone detectors* look for duplicate (e.g., copy-pasted) or very similar (e.g., insufficiently abstracted) code throughout the software.

In general, the above tools flag generic design and implementation defects. They are incapable of detecting defects that are specific to a particular software system such as violations of coding conventions (e.g., classes for GUI elements should be named *Widget) or of design invariants (e.g., every subclass of X should override a particular method).

Proposal bachelor thesis

The goal of this bachelor thesis is two-fold:

1) To report on how state of the art smell, bug and clone detection tools fare in practice on the source code of Chamilo. For instance, we foresee that most tools will have difficulties handling multiple programming languages in the same file (e.g., PHP, JavaScript, SQL). Your report will list attention points for researchers in software quality assurance tools.

2) To interview the Chamilo and Pointcarré developers about the coding conventions and design invariants that they often violate unknowingly, and about which they would like to be warned by a tool.
You will implement such a tool by extending an existing state of the art detector for generic defects with custom, application-specific traversals of the source code. To this end, you will compare the internals of each tool and report on which design lends itself best to such customizations. Apart from the tool, the result will be a recommendation for the kinds of extension interfaces software quality assurance tools should provide.

Preparatory course bachelor thesis

To prepare for this thesis, you will perform a literature study on the state of the art in smell, bug and clone detectors. The focus will be on the techniques and source code representations used by these tools to detect design and implementation defects.