A Moodle Plugin to detect code similarity – Educating computer programming students about plagiarism

Project Team

Angela Carbone, Judy Sheard, Margot Schuhmacher, Tri Le (Monash University)
Michael de Raadt (Moodle HQ)
Chris Johnson (ANU)

Source code plagiarism detectors

❖ Freely available to use, e.g. MOSS, JPlag, YAP3

❖ Good algorithm to overcome popular disguising techniques.

❖ However:

- Unfriendly user interface
- High overhead of adoption: impose additional work for lecturers
- No integration to popular learning environment: difficult to adopt at the institutional level
- Lack of educational features to provide students with feedback

⇨ very few study on the educational benefits of these tools

- Current developments: Prototype plugin to integrate two third party tools: JPlag and MOSS into popular VLE -- Moodle

Development sponsored by Netspot Innovation fund (Le at al., 2012)

Plugin/Tool design

submit

students

scanning services

feedback

report

lecturers

tutors
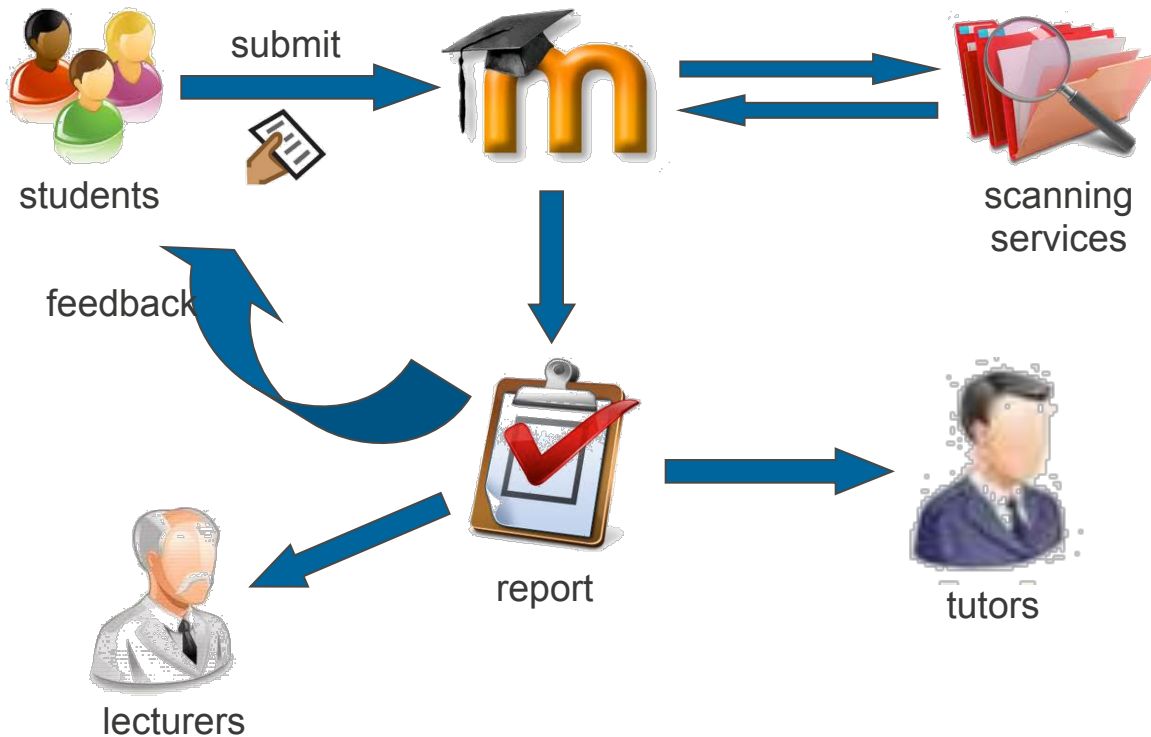
Special consideration:

- Institutional policy regarding students' privacy

- Usability

- Robustness

- Seamless integration

- Educational features: draft submission and formative feedback

Screen shots and short videos

Creating a code matching assignment
http://www.screenr.com/Pzl8

Purpose of our project

❖ To investigate the *useability* and *educational benefit* of the plugin

  ❖ **Usability:** Intuitiveness of interface | Performance of the tool | Relevance of the report
  ❖ **Educational features**
    • provide students with an experimental learning environment (like TurnItIn)
    • Lecturer and students comments about draft submissions
    • Comments around publishing the similarity report to students

❖ Research Approach

  – **For academics**:
    • a simulation of real assignment submission with a sandbox server
    • 30 academic invited; 7 responded
  – **For students**:
    • trialled the tool using a Java assignment
    • 13 submitted assignment; 9 responded

Limitation and future work

❖ Limitation

- Limited number of academic and student participants
- Homogenous participant background
- Trialling of the tool not in real setting

❖ Future work (next phase)

- Enhance the tool using participants' feedback: code seeding, excluding common base code.
- Carry out trialling in real settings
- Extend the trial to multiple institutions
- Develop deeper insights into the educational benefits of formative feedback and draft submissions on academic integrity awareness

Getting the Plugin

- Git
  - https://github.com/thanhtri/plagiarism_programming

- Moodle docs
  - http://docs.moodle.org/22/en/Programming_Code_Plagiarism_Plugin

- Forum
  - http://moodle.org/mod/forum/discuss.php?d=204648

- Interested in trialling the plugin and participating in a research study?

  - Contact Angela Carbone angela.carbone@monash.edu