

peach³

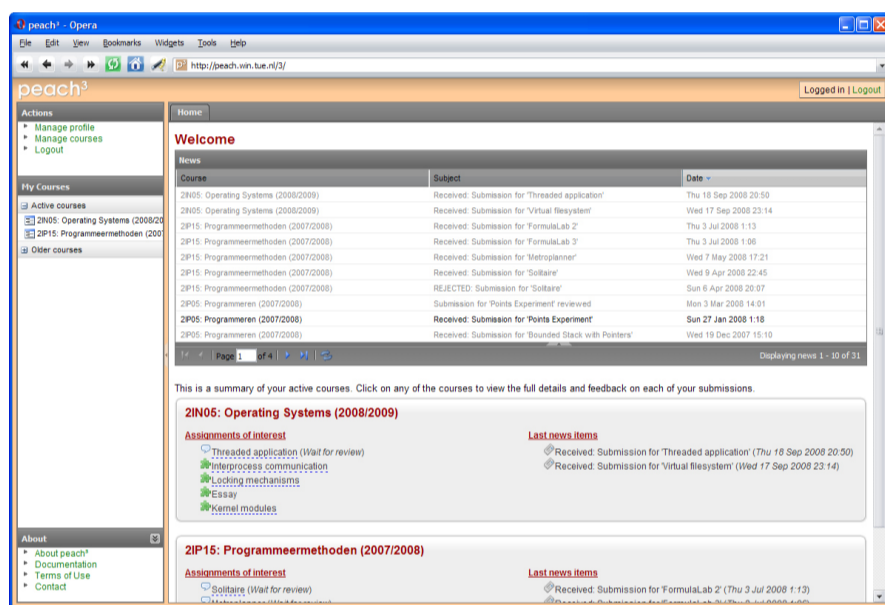
Web-based course management system

ir. E.T.J. Scheffers,
dr. ir. T. Verhoeff

Software Engineering and Technology Group,
Department of Mathematics and Computer
Science,
Eindhoven University of Technology,
The Netherlands.

peach³ is a web-based application for the presentation of assignments, the collection, storage, and automated and/or manual evaluation of work submitted for assignments, and the administration of results.

Originally, the system was developed with a focus on programming education assignments and programming contest problems, but nowadays it can be used for any kind of assignments that require submission of work in the form of one or more files.



Participants need to register with the system and join the course(s) of interest. Assignments can be grouped, availability of assignments can be timed, and there are group and individual submission deadlines. For each assignment it is possible to script automatic evaluation ranging from a simple check on file type/size to compiling submitted programs, running the executable with various input cases, capturing the produced output, checking the correctness of the output, and turning that into a numeric score.

peach³ is the third generation of Peach. The first two generations have been in use since 2001 at various institutions, including Eindhoven University of Technology in the Netherlands, where it was developed, and University of Tampere in Finland. The third generation has been in use since August 2007, and is in active development as an open-source project.

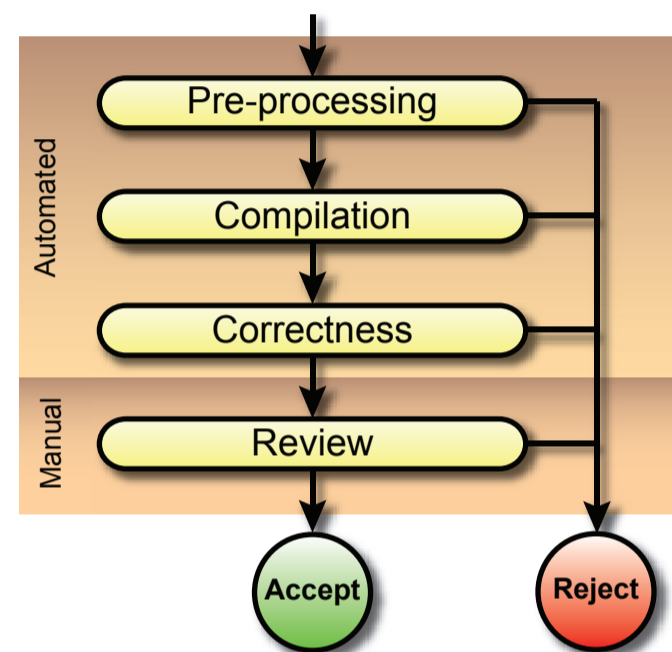
More information, documentation, source code and a live demo is available at the peach³ website:

<http://peach3.nl/>

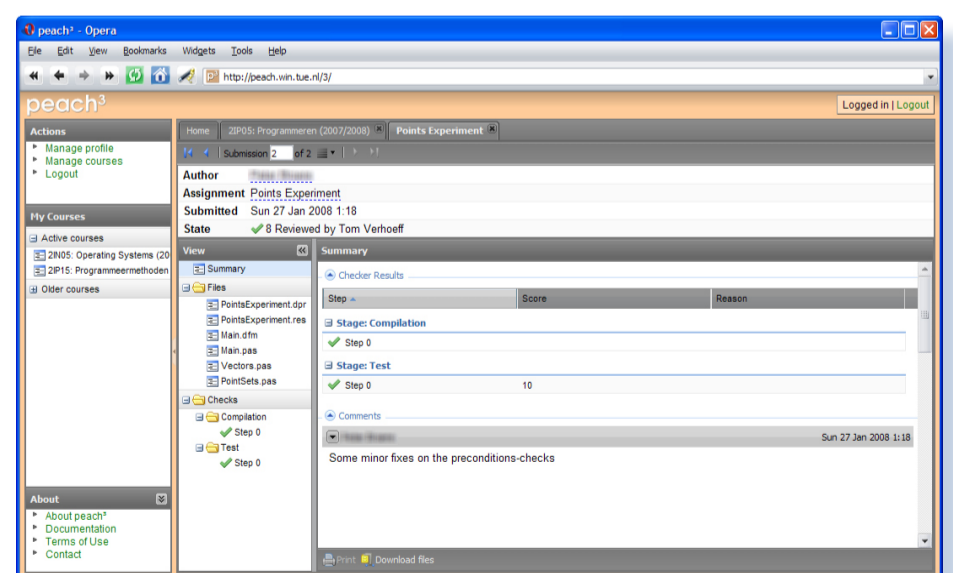
CAN PEACH³ ASSIST YOU?

Automated evaluation

Peach³ uses scripts (written in Python) to automatically evaluate submissions. For computer programs, this evaluation usually consists of 3 stages: pre-processing, compilation and correctness tests.



The pre-processing stage can test the submissions for properties like a minimum percentage of comments. The compilation stage will compile the submitted program, and if that passes, the correctness test stage can supply input to the program and can verify the correctness of the output. The correctness test can also simulate a (text-based) interactive session with the submitted program.



The result of the automated validation can be directly used to grade a submission, or can be used as input for manual grading. Because the automated validation can reject submissions that do not meet a minimum required level of quality, manual grading is only required for submissions that pass the automated validation.