

A Web-Based Peer Evaluation Tool

**David W. Manry, DE and
Hollis G. Bray, Jr, DE, PE, CPC**
University of Louisiana at Monroe
Monroe, Louisiana

Vir V. Phoha, Ph.D.
Louisiana Tech University
Ruston, Louisiana

Designing and implementing peer evaluation in a course requires the instructor consider many factors. A literature review notes these factors, along with a review of several web-based peer evaluation programs. Based on the authors' experiences, desirable features for a computer-based system are listed. A feature-rich web-based peer evaluation tool (PET) that was developed to simplify data collection and analysis in an undergraduate class is described. The PET has unique and customizable features, including the algorithm and rubric. For analysis, the results can be displayed in both graphical and tabular form.

Key Words: Peer Evaluation Tool, Peer Grading, Group Projects, Rubrics, Web-Based

Introduction

Designing and implementing peer evaluation in courses requires significant time for the instructor. The instructor must consider the size and composition of the group, the peer evaluation rubric, the grade distribution algorithm and the weight of the peer evaluation. Additional factors include the time required to compile and report the results of the peer evaluation. An instrument that allows the instructor to be productive in these activities while providing flexibility and selectable options would be a valuable tool. A computer-based tool saves time for the instructor, allowing the instructor to focus on the important task of evaluating and grading the work, and promoting objective peer evaluations by students in the group.

Strategies for Peer Assessment and Grading

Strategies for peer evaluation and grading require sound rubrics and an algorithm for compiling the results (Michaelsen, 1988). The tool used to perform and analyze the peer evaluation must be able to fit the strategy. Therefore, the following factors relate to not only the strategy, but the evaluation tool:

- Training in team dynamics.
- Allowing students to set expectations and norms.
- Addressing the "free-rider" or "loafer."
- Creating an atmosphere of "positive interdependence."
- Method of establishing small groups.
- Progress report requirements.
- Direct assignment of grades by students.
- Using students' evaluations of performance to assign grades.
- Adjustment of grades by the instructor.
- Anonymous or identified review?
- Does the student evaluate himself?
- Partial or total distribution of team points? (Foundation, 2002; Walker, 2008; Kelley & Sadowski, 2005)

A survey of the literature reveals a number of resources available to assist in developing the strategy for peer assessment and grading. Educational institutions provide general resources for instructors to assist in peer evaluation. These items generally involve strategies or forms for students to use in structuring the evaluation. Forms typically include static rubrics and suggestions for how instructors may facilitate group work (Centers, 2010).

Peer evaluation rubrics may be as simple as three to five items where students rate team members on a Likert scale or as many as twenty items with subjective observations regarding the strongest and weakest team members (Luse, 2011). Some instructors prefer that students rate themselves while others insist that students do not rate their own work. Peer evaluation algorithms allow a student to rate peers in proportion to perceived contributions (Vaidyanathan, 2008). Peer grades may or may not be anonymous, depending on the instructor's preference. Instructors could expect students to develop their own rubric for the peer evaluation. Additionally, students may not be eager to rate their peers; therefore, instructors must obtain buy-in to peer grading schemes. Instructors should anticipate and address students' reluctance to participate in peer grading (Harreid, 2010).

Computer-Based Systems

Programmers developed a web-based peer evaluation system to facilitate peer evaluation of programs written by computer science students. It employed structured query language (SQL) and PHP: hypertext preprocessor (PHP) scripts. The system included an administrator interface for adding instructors and students, assigning courses and students, and other administrative details. An instructor interface allowed posting, review and grading of assignments and design of rubrics for peer grading. A student interface allowed submittal of assignments, review of grades, and peer review by students. Students were limited to the instructor supplied rubric. No mathematical algorithm was presented to calculate grades based on peer review. The assignments were individual rather than group work (Triveldi, 2003).

A commercial web-based program developed for managing courses with writing assignments is extensive in that it allows students to develop their own rubrics. The student rubrics are calibrated to referenced writing samples and standardized by comparing the results to the instructor's rubric for the evaluation. However, there are no mathematical algorithms provided for peer grading and the system does not support group assignments (Calibrated, 2001).

iRubric is a rubric development and management tool that allows instructors to share, edit and deploy rubrics via the web. However, the system does not support peer grading (iRubric, 2011).

Development of a Web-Based Peer Evaluation Tool

Previous works by the authors recount their experiences using peer evaluations with construction management students. A hands-on, active learning approach to teach reinforced concrete and formwork design is described in this earlier literature. Each fall during a concrete class, a scale model of a reinforced concrete structure is built. The class is divided into groups, each representing a different functional area (e.g., project administration or ironworker). In the first couple of years, the peer evaluations from these groups were turned in on paper before being entered into Excel for analysis (Bray & Manry, 2007).

In later years, to reduce the time required for compiling and calculating the individual student scores, the authors developed a computer-based instrument for peer evaluation. This instrument uses a menu-driven program written in Visual Basic for Applications (VBA). The application has been deployed in both the Blackboard and Moodle learning management systems (LMS). The peer evaluation tool (PET) is loaded with information, including the group roster and rubric. The files generated when the student responds to the PET are combined to produce an aggregate table of data. A grade distribution algorithm is applied and the peer-adjusted grades are calculated. The final grades must be manually entered into the LMS (Bray & Manry, 2010).

Desirable Features for a Computer-Based System

The authors' experiences led to the identification of several shortcomings of the VBA application. A set of desired features is identified below based on the authors' experiences in applying peer assessment in class:

- The PET should be independent from the LMS.
- Several grading algorithms should be available.
- Rubric items should be easily customized.

- A simple interface is needed to customize the PET for each use.
- Graphical and tabular display of peer evaluation data are needed for instructors' analysis.
- Multiple mathematical algorithms are needed to apply peer grades.

A PET that is independent of the LMS gives the instructor several advantages. One of the greatest burdens in conducting peer evaluation is the collection and compilation of the results. Whether done by hand or by spreadsheet, the amount of time required to collect, compile, and report student results can be substantial relative to other graded class material. If a LMS is used to facilitate the distribution and collection of data from students, a procedure must be developed especially for that LMS system, and much of it may require manual operation by the instructor in order to collect and compile data. A web-based system operating independently of the LMS avoids this problem and allows more time for the instructor to actually review results.

A simple user interface was developed to customize the PET. Customizable features include:

- Option to allow the instructor to require student's self-evaluation.
- Option to change the number and description of rubric items.
- Option to require forced-ranking.
- Ability to set weights for peer evaluation items.
- Ability to upload rosters from text files.
- Graphical and tabular display of data.

In earlier work it was shown that students can be reluctant to critically evaluate their peers, especially when the group grade is not particularly low (Bray & Manry, 2007). One simple way to increase the distribution of peer evaluations is to require a forced-ranking of peers. This will prevent students from giving the same rating to all peers. With no ties, the distribution of scores will necessarily be increased. Uploading rosters from a text file assists the instructor in deploying and using the PET.

The graphical and tabular display of data allows the instructor to make a quick check on whether students are performing critical evaluation of peers. By decreasing the amount of time required to collect, compile and analyze the peer evaluation data, the instructor has the flexibility of assessing the student effort that goes into peer evaluation and to give feedback.

Web-Based PET

Incorporating the desired features, a web-based PET has been developed using the Java programming language coupled with the corresponding OpenGL graphical library implementation. Both the student and the instructor will access the PET through the designated website. One caveat of a Java/OpenGL implementation is that the PET can be constructed so that it can run either as a Java applet in a web browser or as a stand-alone program. This setup will allow the instructor to work offline, if necessary.

The advantage of the web-based Java program is that data for peer evaluations are collected and compiled independently of the LMS used by the institution. In most cases, only the final peer evaluation grade needs to be input manually by the instructor into the LMS. The Java applet and corresponding input files would be stored in a web directory and the applet accessed via a web browser.

The PET consists of two components – the peer evaluator and the instructor evaluation compiler. Both the peer evaluator and the evaluation compiler use two comma-delimited ASCII text files as input. The first of these files is the student roster. Each student's peer group is denoted in this file. The second file contains each group's rubric evaluation items and their respective descriptions. The number and name/description of evaluation items can vary between groups. Options, such as allowing self-evaluation and/or required forced-ranking, would also be specified in this second file.

Besides the two comma-delimited text files, the evaluation compiler also needs to know the directory that contains all the individual peer evaluations. These peer evaluations are also comma-delimited ASCII text files that contain each evaluator's ratings of his peers.

To initiate an evaluation, the student simply navigates to the website and accesses the Java applet. Currently the student logs in using a username/password combination. To provide better security, a future implementation will use the username/password along with a user authentication based on biometric measures (Sheng, Phoha & Rovnyak, 2005). Once logged in, the student will begin the rating process.

The screenshot (see Figure 1) shows a customized rubric with five evaluation items (tabs). The student rates each of his peers according to the rubric description on each tab. If desired, the instructor may configure the evaluation to allow self-evaluation and/or to require forced-ranking. Once the student has completed the evaluation, the file is saved in a directory along with all other peer evaluations. The peer evaluations contained in this directory provide the input for the instructor's evaluation compiler.

Individual	Strongly Disagree	Disagree	Neutral	Agree	Strongly Agree
Anderson, Steve					●
Freeman, Robert				●	
Holmes, Jarred				●	
Neighbors, Jeremy				●	
Vice, Matthew				●	
Yates, Garrett				●	

Figure 1: Peer evaluation item input form screenshot.

Once the peer evaluations have been completed, the instructor can start the analysis using the evaluation compiler. The instructor chooses file and directory locations. Any missing files from students not completing the evaluation are noted on the class info tab, but are ignored in the results. Once the locations are defined, hitting the "Prepare Charts" button will provide each group with a tab after the "Class Info" tab (see Figure 2).

Clicking on any of the group tabs will bring up the screen shown (see Figure 2) below. Notice that each group tab has three tabs on a sub-level. These three tabs contain different ways of looking at the data. The first tab "Line Chart" contains the data by category on a line graph. Above each category's line, individual evaluator ratings are shown. The top row of circles is for the ratings from the first evaluator (alphabetically). Looking at the legend, the top is for Steve Anderson's evaluation of the group. Using the legend to find the color identifying the person evaluated, we see that Anderson, Freeman, Holmes, Neighbors, Vice, and Yates were rated 10, 9, 8, 7, 6, and 5, respectively, for Leadership.

Because it is hard to look at individual evaluator rows, individual evaluator marks can be isolated by clicking on the legend (as noted at the top of the legend). A pop-up window, as shown (see Figure 3), will allow the selection of evaluator(s) to display. Also in this pop-up window, each evaluated person's average and/or median rating can be displayed. Averages are displayed with triangles and medians with diamonds. Averages and medians are given with both symbols and text.

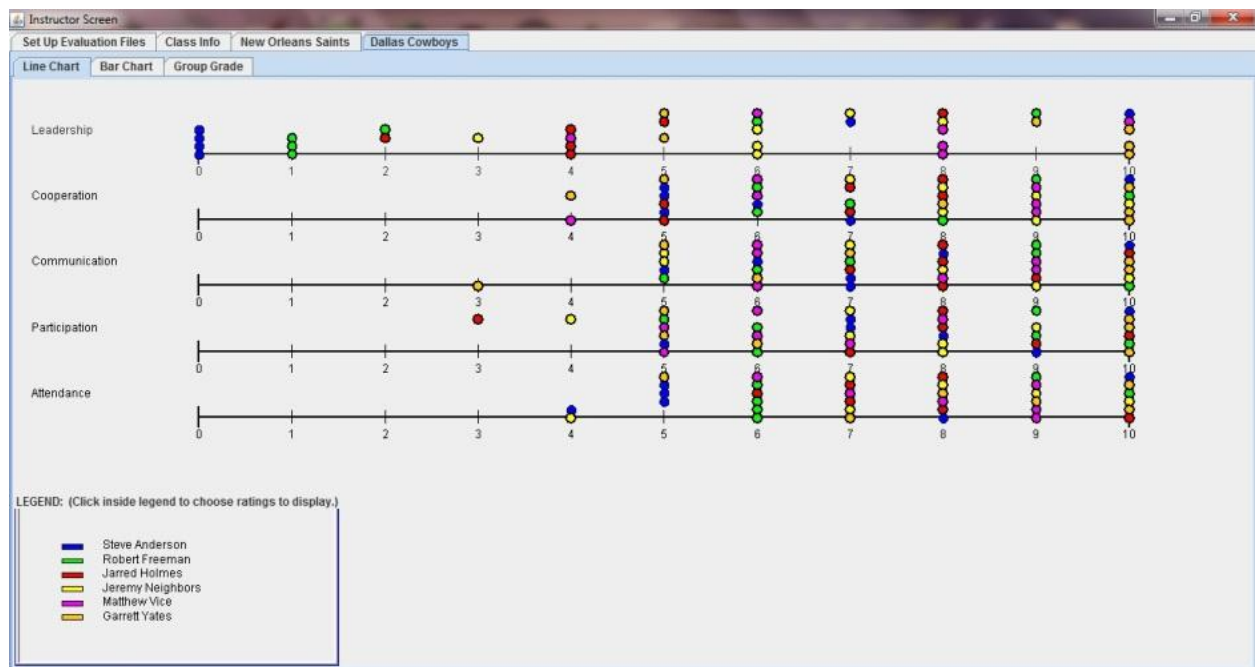


Figure 2: Line chart panel screenshot

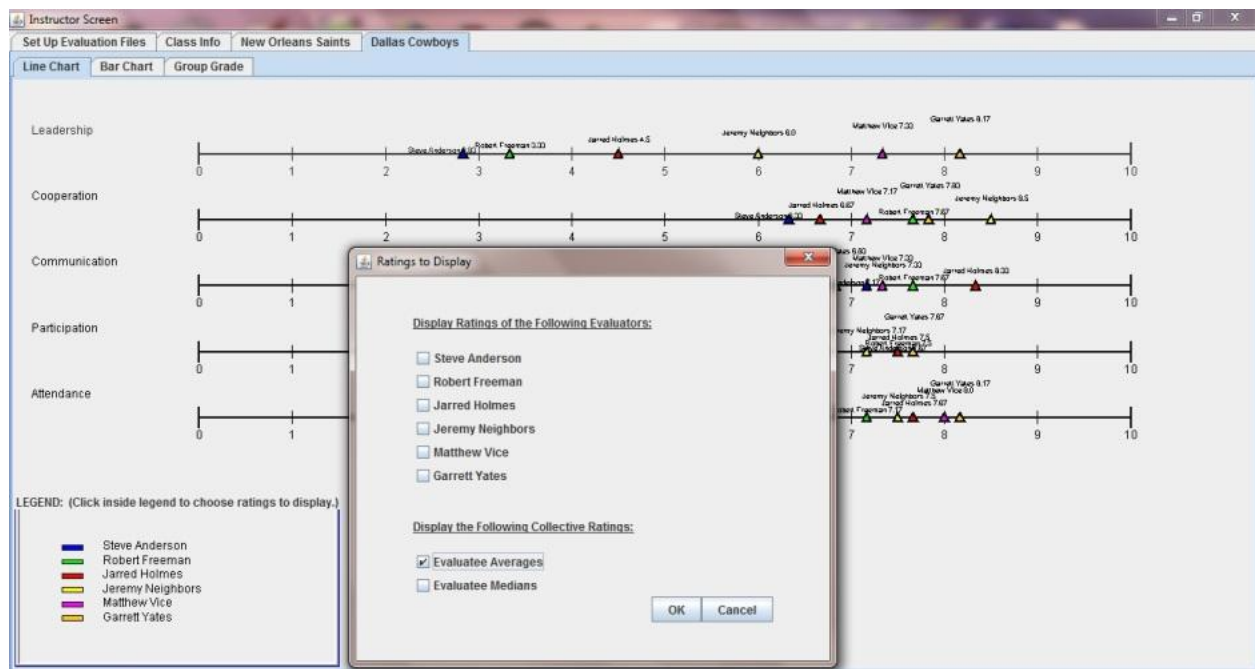


Figure 3: Evaluatee averages option screenshot.

As seen (see Figure 4) below, another next way of looking at the data is by a bar chart on the “Bar Chart” tab. Each student is listed on the abscissa. The bars represent the ratings given by each evaluator for each category. Looking at Steve Anderson and the legend, leadership is the first category – followed by cooperation, communication, participation, and attendance.

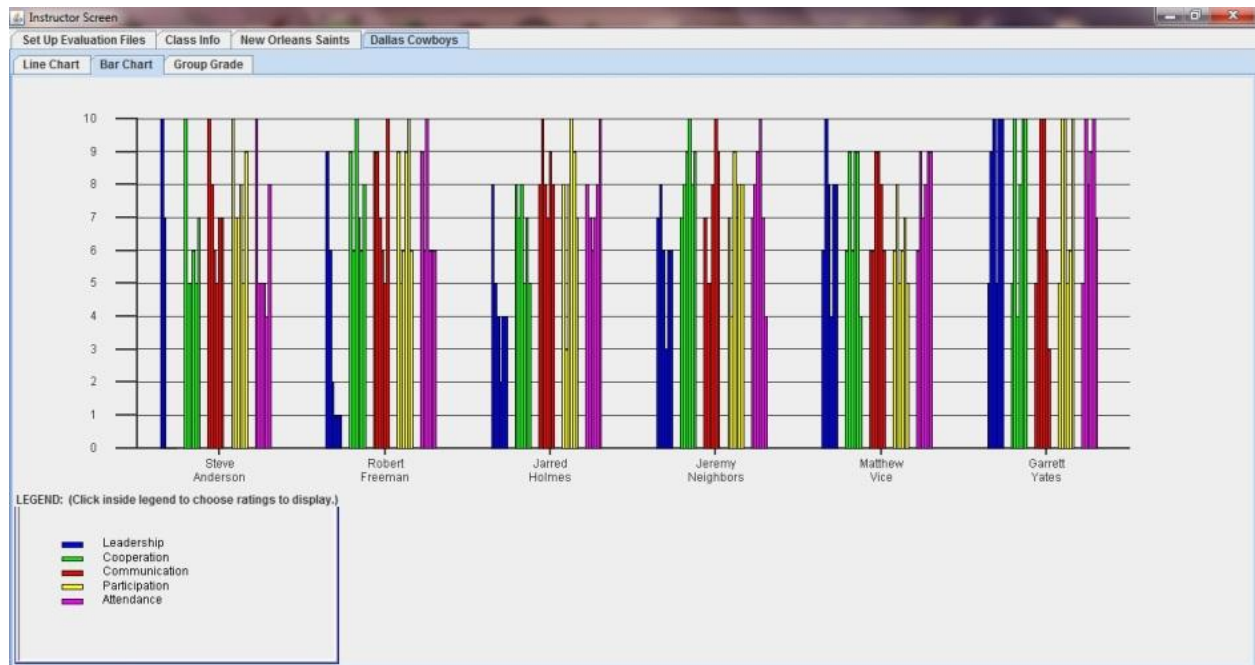


Figure 4: Bar chart panel screenshot.

The following screenshot (see Figure 5) shows two pop-up windows. The top window that allows specifying which evaluator(s) ratings to show is activated by clicking on the legend. The next window, which shows the raw data from the individual evaluation files, is activated by clicking on the bar chart.

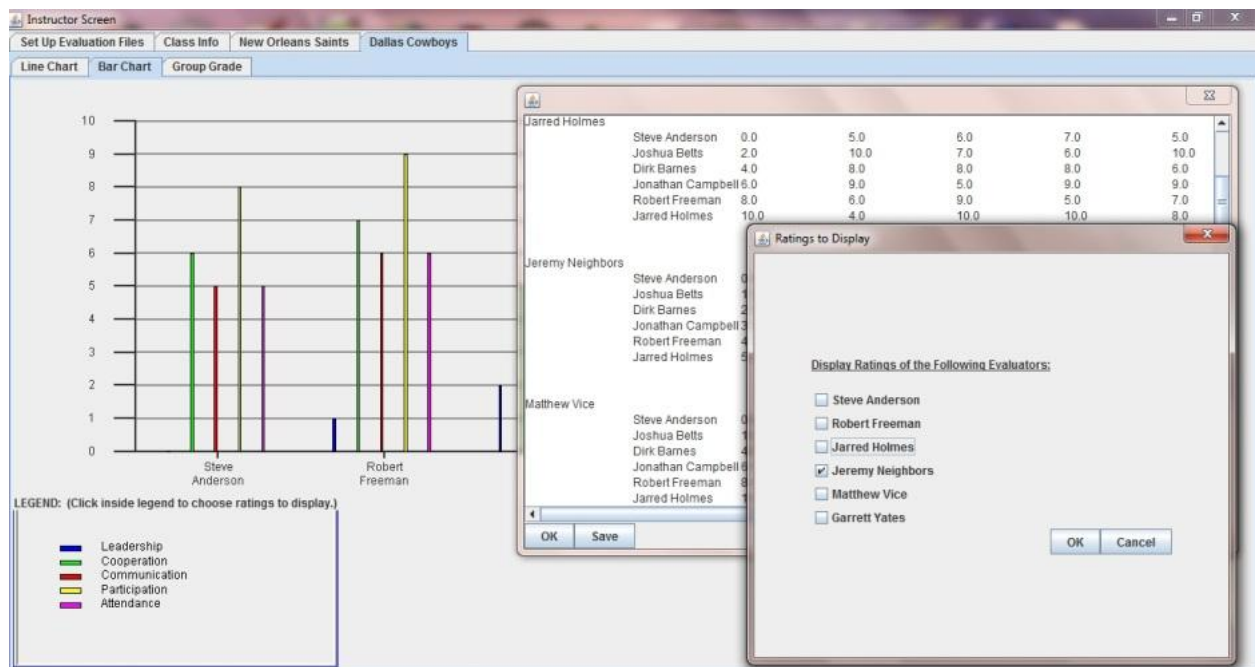


Figure 5: Evaluator display selection and raw data table screenshot.

Finally we have the "Group Grade" tab as shown (see Figure 6) below. Two items can be adjusted on this sheet – the group grade and the student influence.

The algorithms available are based on variations of a distribution method by Berryman (Berryman, 1999). Berryman includes a variable to limit the weight of the peer evaluation effect on the redistribution of the individual student's grade.

The total points (awarded by the evaluators) will remain the same as the group grade and student influence are changed. Also the IER (which is the student's points divided by the total points) will remain the same. The total final grade points (which is the group grade multiplied by the number of group members) will also remain the same. The changes will be in three areas: the group awarded points, the instructor awarded points, and the total grade.

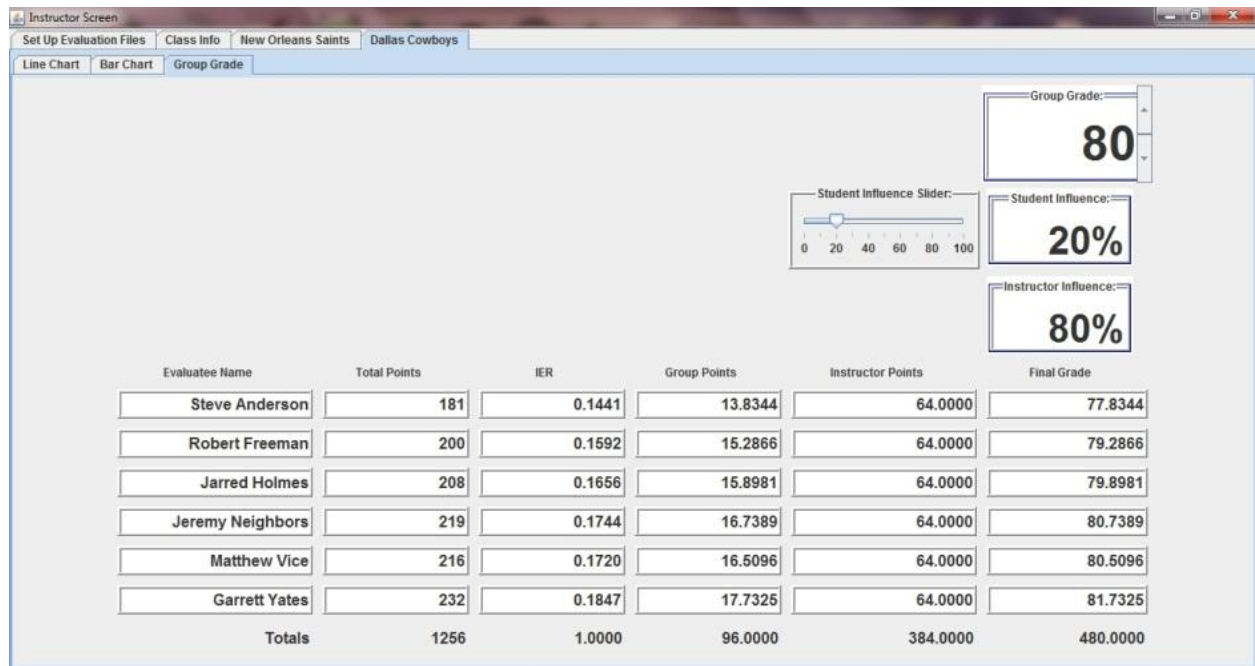


Figure 6: Group grade panel screenshot.

Conclusions

The PET developed incorporates the most desirable features of peer evaluation/grading strategies with a graphical interface and customizable tools. The rich set of features allows maximum flexibility. Table 1 summarizes the specific contributions of the peer evaluation tool with relation to construction students and instructors.

Table 1.

A unique peer evaluation tool

Features

Graphical display of results for instructor analysis
Anonymous collection of data from students
Customizable rubrics
Web deployment
Automated calculations

The PET is useful to both students and instructors, but in different ways. Instructors need an automated, systematic method for collecting peer evaluation data and performing the calculations. The PET also provides a visual means of analyzing the results of the peer evaluation, helping the instructor to determine if students are critically evaluating

the work of the group or simply assigning the same grade to all members. The PET simplifies the creation of rubrics by the instructor as well.

Finally from the perspective of the student, the PET provides anonymity and allows the student to focus on the relative effort of each group member.

References

Berryman, C. (1999). Unpublished spreadsheet.

Bray, H. & Manry, D. (2007). Active learning in a reinforced concrete design class. *Associated Schools of Construction, Proceedings of the 43rd Annual Conference*.

Bray, H. & Manry, D. (2010). Peer Evaluation Experiences. *Associated Schools of Construction, Proceedings of the 46th Annual Conference*.

Calibrated Peer Review. (2001). *Introduction*. [WWW document]. URL <http://cpr.molsci.ucla.edu/>.

Center for Teaching and Learning with Technology. (2010). *Teaching Resources*. [WWW document]. URL <http://ctl.t.jhsph.edu/resources/>.

Centers for Ocean Sciences Education Excellence. (2010). *Peer Evaluation of Group Work Rubric*. [WWW document]. URL http://cosee-central-gom.org/seascholars/lesson_plans/lesson2web/intro/peerrubric1.html.

Herreid, C. (2010). *Dolby's Delimma*. [WWW document]. URL <http://library.buffalo.edu/libraries/projects/cases/dolby.htm>.

iRubric. (2011). *Welcome to iRubric*. [WWW document]. URL <http://www.rcampus.com/indexrubric.cfm>.

Kelly & Sadowski. (2005). Peer Evaluation Within a Team Design Project. *Proceedings of the Mid-year Conference of the Engineering Design Graphics Division of the American Society for Engineering Education*

Luse, D. (2011). *Team Evaluation Form*. Unpublished manuscript, University of Louisiana at Monroe, Monroe, Louisiana.

Michaelsen & Schultheiss. (1988). Making Feedback Helpful. *The Organizational Behavior Teaching Review*, 13(1), 109-113.

The Foundation Coalition. (2002). *Peer Assessment and Peer Evaluation*. [WWW document]. URL <http://www.foundationcoalition.org>.

Sheng, Y., Phoha, V. & Rovnyak, S. (2005) A Parallel Decision Tree-Based Method for User Authentication Based on Keystroke Patterns, *IEEE Transactions on Systems, Man, and Cybernetics-Part B*, 35, 826-833.

Triveldi, A., Kar, D. & Patterson-McNeill, H. (2003). Automatic Assignment Management and Peer Evaluation. *Consortium for Computing Sciences in Small Colleges*, 18(4).

Vaidyanathan, R. (2008). Peer Evaluation FAQ. [WWW document]. URL http://www.d.umn.edu/~rvaidyan/mktg4731/peer_ass.htm.

Walker Teaching Resources Center. (2008). *Evaluating Groups – Fairly*. [WWW document]. URL <http://www.utc.edu/Administration/WalkerTeachingResourceCenter/FacultyDevelopment/EvalGroups/>.