Methodology for Formative Assessment

Wade C. Driscoll
Professor, Industrial Engineering, Youngstown State University

Abstract

This paper reports on an investigation into the development of a methodology for efficiently obtaining and analyzing formative assessment information. A primary criterion for the methodology was that it must support the efficient design of a unique questionnaire for each lesson in which each student attending that lesson would complete the custom-designed questionnaire. Additional criteria called for the instructor to be able to analyze data from the questionnaires in real time in a matter that permitted adjustment of the lesson in progress. In addition, the assessment methodology should enable the instructor to adapt subsequent lesson plans based on the feedback. The author developed Windows applications in Visual Basic to aid in meeting these criteria without an excessive investment of faculty resources. Visual Basic supports the development of Windows applications having graphical user interfaces that can be accessed over an Intranet. This made it an excellent vehicle for meeting the design criteria.

1. Introduction

Accreditation requires[6] – among other things – continuous quality improvement by assessing student outcomes [2] and using the attendant feedback to influence and impact course, curriculum, program, department and college planning. Although *some* faculty may receive additional resources to handle the extra workload required to implement the assessment of student outcomes, *all* must meet the requirements without compromising their contributions in teaching, scholarship and service. An ASEE position paper[1] addresses the concept of economical use of faculty resources by stating

> "The cost of assessment should be outweighed by the benefits to the educational program being assessed."

Many engineering professors welcome any method that can be used to provide meaningful feedback while requiring only minimal resources.

This paper reports on an investigation into the use of Visual Basic applications and computer networking capabilities for obtaining, analyzing, and applying extensive feedback on a course in progress. The feedback comprises *formative* assessment that relates to a continuous improvement process. As stated by Shaeiwitz [9] "Classroom assessment is often called formative assessment, since the feedback loop is very short term, and the specific purpose is to improve teaching and learning." *Form*ative assessment data in*form* the instructor how to conduct a course in a manner that enhances student learning. In contrast, *summative* assessment relates to determining whether higher level, longer-term outcomes are being met.

The ideal of having a custom-designed questionnaire for each lesson — completed by each student who attended class — provides a demanding scenario for the development of a methodology for obtaining feedback to be used for assessment. This paper reports on the methodology developed for questionnaire design, deployment, presentation, data logging, and data analysis. It also summarizes some of the feedback that resulted from employing the methodology, and notes salient issues that emerged as the investigation proceeded. The report should also serve engineering educators by heightening their awareness of the capabilities of Visual Basic[4, 5].

The rest of this paper is organized as follows:

- Section 2 describes the context within which the investigation was conducted.
- Section 3 presents operational details. A challenging workload is associated with questionnaire design, deployment, presentation, data logging and data analysis unless the process is expedited through the use of utilities designed for the purpose. This section describes four applications developed in support of the investigation, and summarizes the impact their use had on the process.
- Section 4 documents details on both the use of the methodology and the results from using it.
- Section 5 presents a summary, conclusions, and suggestions for future work.

2. Context of the Investigation

The investigation focused on the course *Simulation of Industrial Engineering Systems* 10 that this author taught in the Fall 2001 semester. The course, which requires extensive computations, meets in a computer laboratory. An ideal class requires students (i) to acquire new knowledge in a lecture/recitation mode of operation, (ii) to collaborate with classmates in developing mathematical models relevant to a problem scenario, (iii) to implement their models on microcomputers, (iv) to exercise those models, and (v) to interpret results from those exercises. Thus the format for classroom conduct conditions students to accept computer use as a normal part of their classroom experience. Furthermore, the fact that the computers in the classroom/laboratory are on a university Intranet made it possible to use Network Neighborhood in deploying questionnaires and logging student responses to them. These conditions were ideal for using *Simulation of Industrial Engineering Systems* for the investigation.

3. Operational Details

The ideal of having a custom-designed questionnaire for each lesson — completed by each student who attended class — provides a demanding scenario for the development of a methodology for obtaining feedback to be used for assessment. Jung et al.[7] lists the following aspects that may contribute to the workload burden that attends computer-based testing programs:

- Developing a test item bank
- Test construction
- Test administration
- Test scoring
- Interpretation and analysis of results
- Item analysis and test refinement
- Reporting test results

This section describes four applications (some of the above seven items are combined) developed to reduce the workload burden, and summarizes the impact their use had on the process.

The reader may compare the results reported herein to those reported in[7], which describes some excellent work that resulted in an application called Quizzer. Quizzer was developed using Asymetrix's Multimedia ToolBook[3] for Windows platforms and network distribution. In contrast, the applications developed as a part of the investigation reported herein function in a Windows 95 or later environment without the need for any additional authoring tools. The author implemented the applications in Visual Basic[8].

3.1 Questionnaire Design

The investigator established the following set of criteria for the application used in questionnaire development:

(a) Using existing questionnaires as resources for developing new ones;
(b) Importing a questionnaire item via cut-and-paste operations and editing text to suit the purposes of the new questionnaire;
(c) Modifying questionnaire structure, including deleting, inserting, and moving items;
(d) Importing a set of allowable responses via cut-and-paste operations, and then editing text to suit the purposes of the new questionnaire;
(e) Previewing the new questionnaire, so that the designer can respond to it as a student would; and
(f) Storing the data in a format that can be used by the application used for data analysis (described in section 3.2 below).

The above standards resulted from personal experience in using a word processor to develop questionnaires to be delivered via paper. People with similar experience may agree that much the same capabilities are appropriate for development of a questionnaire designed to be delivered electronically over the Intranet.

3.2 Questionnaire Deployment and Presentation

Once a questionnaire has been developed, its delivery to students must be implemented. A Window application developed for this purpose uses data files from section 3.1 (above) as input for presenting questionnaires to students. Students download[1] (from Network Neighborhood) the EXE file supporting questionnaire presentation so that they can use local computer resources for its delivery. The application presents the questionnaire items one at a time to each individual student, with a simple click on an option button required to indicate a response. After the student has responded to all of the questionnaire items, s/he has a chance to review — with the possibility of revising — each response. Upon the completion of response review and editing, the student uses a mouse click to initiate activities that culminate in the terminating the application. Prior to termination, the application places that student's responses in a folder with Read/Write permissions on Network Neighborhood, where it can be accessed by a companion application running on the instructor's office machine.

3.3 Logging Student Responses from the Questionnaire

A file containing the responses of a student appears momentarily in a folder of Network Neighborhood having both read and write permissions. An application called "Hoover" running on the instructor's office machine periodically scans for the presence of these files. Hoover then reads each file and appends its contents to a file residing on the office machine that is not accessible from Network Neighborhood. Then Hoover deletes the original file, which was accessible over Network Neighborhood. The system, while not misuse-proof, thus provides conditions that reduce the opportunities for misuse.

3.4 Analysis of Results From the Questionnaire

The activities described in Section 3.3 above result in a data file containing the responses of all students to all questions in a questionnaire. Another application, run at the instructor's convenience[2], analyzes the data. The analysis computes frequency counts and other statistics for the responses to each item. The application then displays each question next to its allowable responses. The user views statistics on those responses embedded into the text describing them. The application supports hard copy in an easy-to-read format.

4. Results

Conclusions relate to (i) the methodology and the workload attending implementation, and (ii) the efficacy of the method for assessment for informing class and course plans.

[1] Students thus initiate the deployment of their questionnaire via simple Windows operations.
[2] This may be done repeatedly, as logs of student responses are received.

4.1 Methodology and Its Workload

The methodology made it feasible to custom–design a questionnaire for each lesson, to present it in each class, to log the resulting data and to analyze its database. The following items compare this methodology's requirements to those for a paper questionnaire that has responses recorded on scan sheets.

(a) The time to develop a new questionnaire from old ones approximated the time required to use a word processor for the development of a paper questionnaire.

(b) The time for questionnaire deployment shrank to less than a minute. This eliminated the nuisance of distributing and retrieving paper during class. It also saved class time.

(c) The time for students to fill out the questionnaire decreased. For example, the presentation mode eliminated the need for a student to switch focus between the questionnaire and a scan sheet for recording responses.

(d) Practically no time was required to compile a database consisting of the response of each student to each question. The application described in section 3.3 above does this without human intervention. Thus this methodology saves whatever time would otherwise be required to compile results from scan sheets into a database.

(e) The time required to analyze the database diminished to that required to read and interpret the statistical summary described in section 3.4. The application requires practically no significant time (a second or two) to compute summary statistics and display them in a form suitable for interpretation.

4.2 Efficacy of the Feedback for Assessment

The term "overkill" may have occurred to the reader when s/he read about the design standard of using a custom-designed questionnaire for each lesson to be completed by each student in class. That design standard established a demanding scenario for methodology development. Having a system that meets a worst–case scenario — in terms of relief from excessive demands on limited resources — doesn't mean that the scenario best supports educational objectives.

The following observations apply to the use of questionnaires in *Simulation of Industrial Engineering Systems:*

- The extra workload for the students completing the questionnaires was not particularly bothersome to most of them; and

- The applications — developed for questionnaire design, deployment, presentation, data logging and data analysis — reduced the workload burden on the instructor to a manageable level.

Proceedings of the 2003 American Society for Engineering Education Annual Conference & Exposition.

In addition, two unanticipated favorable side effects accompanied the use of the questionnaires. First, student morale appeared to be favorably affected. They seemed to interpret the use of the questionnaires as an indication of the instructor's interest in their getting something out of the course. Secondly, the necessity of designing a questionnaire for a lesson sharpened the instructor's preparation for it. One cannot formulate questions related to the attainment of objectives without first carefully formulating those objectives.

5. Summary, Conclusions and Future Work

The author sought to develop a methodology for using daily feedback via questionnaires from students to assess and impact lesson and course objectives. The design standard required developing a methodology without excessive time requirements that made it feasible to perform questionnaire development, deployment, presentation, data logging, and data interpretation for each student in each class.

Four applications implemented in Visual Basic enabled these objectives to be achieved. A special-purpose editor expedited questionnaire design, and saved the questionnaire in a format suitable for Intranet deployment. An application that was deployed over the Intranet presented the questionnaires, and another application residing on the author's office machine compiled the responses into a database. Finally, an application for analyzing the data computed relevant statistics and displayed them in a format suitable for easy interpretation.

Experience in using the methodology led to the following observations. Most significantly, valuable feedback was obtained that helped to achieve the aims of the course. The extra workload for the students was not particularly bothersome to most of them. The extra workload for the instructor was manageable. Student morale was favorably affected as they interpreted the use of the questionnaires as an indication of the instructor's interest in their learning. Questionnaire development contributed to the instructor's preparation for lessons, since one cannot ask questions about the attainment of lesson objectives without first carefully articulating those objectives.

This investigation led to the identification of two areas for future work. First, a way to embed the data into the application presenting the questionnaire would eliminate the need for that application to make separate references for questionnaire presentation to files residing on the server. Although server response time was not a problem in the investigation reported herein, embedding the data might enable the methodology to be applied to classes with more students than is currently possible. Secondly, some conditions may mandate the use of Web pages and HTML files for questionnaire deployment and presentation. Thirdly, the investigation verified the efficacy of using Visual Basic to implement courseware for access via an Intranet. Visual Basic supports the development of applications having graphic user interfaces that are conveniently accessed via Windows' Network Neighborhood.

References

[1] Angelo, T. A., *Assessment White Paper: A Framework for the Assessment of Engineering Education,* http://www.asee.org/pubs/html/assessment.htm, American Society of Engineering Educators (1996)

[2] Angelo, T. A. and K. P. Cross, *Classroom Assessment Techniques. A Handbook for College Teachers* (2nd ed.), Jossey-Bass, San Francisco, 1993, pp. 13-23.

[3] Asymetrix Corporation, M*ultimedia ToolBook Version 3.0: Multimedia Authoring System for Windows*, Bellevue, WA.

[4] Driscoll, Wade C., *Using Visual Basic for Engineering Computations*, to be submitted to the 2003 ASEE Annual Conference Proceedings (2002)

[5] Driscoll, Wade C., *Visual Basic For Engineers*, Workbook for Engineering Computing Published by Comdoc at Youngstown State University (2000)

[6] Engineering Accreditation Commission of the Accreditation Board for Engineering and Technology, *Engineering Criteria 2000*, 111 Market Place, # 1050, Baltimore, MD 21202 (1996)

[7] Jung, I., Hosoon Ku, and D. L. Evans, *A Network-based Multimedia Computerized Testing Tool*, Proceeding of the American Society of Engineering Educators, http://www.asee.org/conferences/search/00936.pdf, Session 3530 (1997).

[8] Microsoft Corporation, *Visual Basic 6.0 Programmers Guide*, Microsoft Publications (1998)

[9] Shaeiwitz, Joseph A. C*losing the Assessment Loop*, Proceeding of the American Society of Engineering Educators, Session 2613, http://www.asee. org/conferences/search/00059.pdf, (1998)

[10] Youngstown State University, *Undergraduate Bulletin for 2002-2003*, p. 257 (2002)

Biographical Information

Dr. Wade C. Driscoll earned engineering degrees from the Pennsylvania State University, New York University, and Case Western Reserve University. His educational research interests include the development, use and evaluation of computer aids to education.