Abstract - The use of web-based tutorials to deliver course content is increasing. Interface design guidelines are plentiful and easy to find. Design methodologies which focus on the pedagogical aspects of the tutorial are less available unless one has training in instructional design. This paper describes a methodology developed by the Center for Teaching and Learning at Indiana University Purdue University Indianapolis, and the experience of the authors in using the methodology to develop an interactive web-based tutorial on C# methods. The effectiveness of the resultant web-based tutorial was assessed using a pretest-posttest experimental design. A significant increase was found confirming that students learned using the web-based tutorial. While the methodology described here can easily be used by anyone to plan an effective online module, the full value comes from working through the method with an instructional designer.

Index Terms – Design methodology, Instructional design, Planning framework, Web-based tutorial.

INTRODUCTION

Each year more and more web-based tutorials are being used to deliver course work both as part of distance education courses as well as to supplement traditional classroom sessions. This may be attributed to the advantages that web-based tutorials offer: convenience, working at one’s own pace, and around-the-clock online accessibility. Guidelines for creating the user interface are easy to find. Design methodologies which focus on the pedagogical aspects of the tutorial content are less available to content experts, unless one is fortunate enough to work with an instructional designer. This paper describes a methodology developed by the Center for Teaching and Learning at Indiana University Purdue University Indianapolis, and the experience of the authors in using the methodology to develop an interactive web-based tutorial on C# methods. The results of an empirical test of the effectiveness of the tutorial are reported.

There are many ways of organizing content for delivery via the Web. Depending on the goals for the course, type of content, maturity of the students, time frame for the course and many other variables, faculty can select from a wide range of instructional strategies and frameworks grounded in learning theory. (See [1] and [2].) However, for faculty new to online learning, a straightforward method of direct instruction is often the easiest way to begin. Classic instructional design methodology suggests that authors specifically articulate key design elements so that the module provides students with the information and practice needed in order to master desired course goals. In the planning process used to develop the web-based tutorial on C# methods, the content expert, with guidance from the instructional designer, was asked to describe the major elements that needed to be included in the online module.

The methodology described here is a planning tool. It is not a prototype development tool (like Janicki and Lingle's Web-Based Tutoring Authoring System [3]) nor does it cover interface design or implementation. Its purpose is to aid instructors in integrating instructional practices with content.

The Methodology

The planning methodology consists of three phases: determining the learning objectives, planning the modules, and storyboarding.

I. Learning Objectives

As with any pedagogically sound learning object, the first step is to determine what you want the students to learn by articulating the learning objectives. Faculty often have a difficult time defining objectives. Without training, they write down a list of topics or vague statements like “Understand the general concepts of data management” rather than more specific learning objectives. This happens because they have been trained as experts in their discipline rather than in learning theory and instructional design. It is precisely their content expertise that can blind them to the incremental steps students require to master new concepts. This is where working with an instructional designer is helpful. Because instructional designers typically lack the specific subject expertise of their faculty clients, they are able to play the role of a “novice interrogator” and ask probing questions about what the faculty member intends for students to learn. Specifying learning objectives in this fashion may take two or three sessions, depending on the scope of the content and the expected length of the module.

Instruction on how to write learning objectives is beyond the scope of this paper. For more information on writing learning objectives see Felder [4], Stice [5] and Park University [6].

Eugenia Fernandez1 and Elizabeth Rubens2

1 Eugenia Fernandez, Associate Professor of Computer & Information Technology, Indiana University Purdue University Indianapolis, efernand@iupui.edu
2 Elizabeth Rubens, Director of Assessment, Office of Professional Development, Indiana University Purdue University Indianapolis, erubens@iupui.edu

0-7803-9077-6/05/$20.00 © 2005 IEEE

35th ASEE/IEEE Frontiers in Education Conference

Session F2E

October 19 – 22, 2005, Indianapolis, IN
II. Planning Framework

Once the learning objectives are identified, the next step is to group the objectives into logical sections and arrange them into modules. The 'Planning Framework for Online Modules', shown in Figure 1, is used to identify the various components which would make up each section of the module. For each objective, the planning framework requires that you identify the following components.

- **Specify Activities:** Once the objectives are defined, faculty must think about the activities that will be required in order for students to practice and eventually master the module content. Table 1 lists typical types of activities that may be part of an online module.

- **Determine the Types of Interaction that are Required:** The faculty member must stipulate what the student will be doing during the module and how the software, faculty member, or other students will respond. If there is automated feedback embedded in the software, all of the possible types of student responses must be considered, and appropriate responses must be programmed. If the tasks are more “open-ended” and faculty feedback is required, faculty will have to think through all of the possible places in the module where their feedback may be critical to student progress. Depending on the task, students may benefit from peer feedback as well. If so, faculty will have to decide where peer feedback will be helpful and what type of scaffolding or support students will need to do a credible job of providing useful responses. For instance, many faculty provide rubrics for assignments, so that students can assess each other on the basis of the criteria stated in the rubric.

- **Identify Assessment Strategies:** For each major objective, faculty should identify an assessment strategy that will allow them to determine whether an objective has been met. Typical types of strategies are to have students demonstrate a skill, submit a piece of written work, answer a multiple choice question on a quiz, write a description of a process, or create a project or a portfolio of their work. Authentic assessments (tasks that are as close as possible to what students will experience if they were to apply concepts to real-world problems) are more highly desired than quizzes and tests; but quizzes and tests can be used as an intermediate step.

<table>
<thead>
<tr>
<th>TABLE 1</th>
<th>SAMPLE ACTIVITIES FOR AN ONLINE MODULE [1]</th>
</tr>
</thead>
</table>
| 1. Participate in a question and answer session with the instructor.  
2. Participate in class discussion in an online asynchronous (delayed) discussion forum or a synchronous (real-time) chat session.  
3. Brainstorm ideas, and concepts, and solutions.  
4. Construct individual or collaborative concept maps.  
5. Read journal articles or textbooks.  
6. Search for and retrieve information. Conduct library research, participate in a web quest, or evaluate the quality of online resources.  
7. Organize, analyze, synthesize, and interpret information gathered from a variety of external sources.  
8. Engage in reflective thinking and writing.  
9. Write a research, position, or concept paper.  
10. Conduct observations or interviews and report results.  
11. Create, distribute, compile, and analyze surveys or questionnaires.  
12. Visit community resources.  
13. Develop and/or analyze case studies.  
14. Engage in individual or collaborative problem-solving.  
15. Simulate a real-world event (e.g. courtroom trial, elections, space launch).  
16. Examine and/or assess other students’ work.  
17. Participate in collaborative exam review sessions.  
18. Listen to a lecture given by the instructor or a guest expert.  
19. Watch an instructional video.  
20. Create and post a website or online presentation.  
21. Generate and manipulate a database or spreadsheet.  
22. Participate in an online debate or panel discussion.  
23. Complete an individual or group project.  
24. Participate in community service.  |

FIGURE 1
PLANNING FRAMEWORK FOR ONLINE MODULES

---

0-7803-9077-6/05/$20.00 © 2005 IEEE.

October 19 – 22, 2005, Indianapolis, IN
35th ASEE/IEEE Frontiers in Education Conference
F2E-15
**The Reality**

"All beginnings are hard."[8] Creating this tutorial was no exception. The first phase, developing the learning objectives, was the hardest part of this process. The content expert knew what content she wanted to cover and what she wanted her students to learn. What she didn't know was how to articulate that in a clear way to students who were not (yet) content experts. Working with the instructional designer forced the content expert to explain her reasoning and be specific in her choice of words. While painful at times, it was an eye-opening learning experience.

For the second phase, completing the planning framework, the content expert chose to view the tutorial as one module with multiple sections, each containing one or more learning objectives. The instructional designer's goal during this phase was to help the content expert integrate what the students were learning (the objectives) with the planned activities and the assessment. In this phase, answering the question "How will you know the student has met this objective?" was the difficult part. Figure 2 shows a portion of the completed plan for the tutorial.

The tutorial was designed as a standalone module, one which did not require any direct faculty feedback to the student. Thus all the interaction described in the framework was meant to be automated feedback embedded in the software. No extensions were identified for this module, so none are listed. This particular section of the module did not contain any documents. However when completing the framework, the content expert did not understand the intended meaning of the documents column and used it to describe what documents she had to create for the tutorial, rather than what documents the students would need to read.

The time required to complete the activities was ignored by the content expert. After this stage, of course, it can only be a rough estimate. A better estimate of the time required can be gained after a prototype of the tutorial has been created and some method of formative assessment, such as usability tests, is conducted. None were conducted with this module. Students complained that many of the pages took too long to load, a problem which would have been uncovered with testing.

Once the planning framework was completed, creating the storyboards was easy (and fun). A sample storyboard is shown in Figure 3. Each board is numbered, and labeled with the section and objective it relates to. The elements visible on the page are diagrammed inside the box, while any interaction or programming effects are described in the right hand column.
Section 2: Understanding Method Headers

<table>
<thead>
<tr>
<th>Objective</th>
<th>Activities</th>
<th>Assessment</th>
<th>Interaction Required</th>
<th>Multimedia</th>
<th>Documents</th>
<th>Extensions</th>
<th>Estimated Time to Complete Activities</th>
</tr>
</thead>
<tbody>
<tr>
<td>3 Identify the parts of a method header.</td>
<td>Read text explanation</td>
<td>Define method header and list its components.</td>
<td>Click &amp; drag component names to method header example.</td>
<td>Text document Method examples with parts identified Method examples with unlabeled parts.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4. Identify whether a method header is complete</td>
<td>Identify missing header components.</td>
<td>Given a method header, determine if it is complete.</td>
<td>Identify which part of a method header is missing.</td>
<td>Method header examples with missing parts.</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

FIGURE 2
PLANNING FRAMEWORK EXAMPLE

FIGURE 3
STORYBOARD EXAMPLE
Using this methodology, the planning phase was completed within one semester. Multiple students were recruited to implement different sections as part of an independent project course. Unfortunately, not all students were successful in completing their assigned sections. Because of this, it was two years before a complete tutorial was created. In an academic setting this may be acceptable, if you view the implementation as a learning opportunity for the students. If you are in a hurry and have the resources it is better to use professional developers.

The completed tutorial can be viewed at http://www.engr.iupui.edu/~efernand/tutorial/version2/TOC.html

The assessment

The effectiveness of the resultant web-based tutorial was assessed as part of a larger empirical study that examined the relationship between student learning styles and learning using Web-based tutorials [9]. Two versions of the tutorial were created, each with the same content, but one with narration and the other without. A pretest/posttest experimental design was used to test for differences in learning. The same 22-point test of knowledge was used for both the pretest and posttest. Subjects were college students enrolled in a 200 level C# programming course during the Fall 2004 semester. Of the 18 students in the course, 12 complete responses were obtained.

A significant difference was found (t(11) = 7.852, p < .01). Thus the post-test was 17.9 (sd = 2.18). A significant increase from the mean on the pretest was 9.58 (sd = 5.52), and the mean on the posttest. Subjects were college students enrolled in a 200 level C# programming course during the Fall 2004 semester. Of the 18 students in the course, 12 complete responses were obtained.

The change in knowledge of subjects using either of the tutorial versions was used to test the effectiveness of the Web-based tutorial. A paired-sample t-test was calculated to compare the mean pretest score to the mean posttest score. The mean on the pretest was 9.58 (sd = 5.52), and the mean on the post-test was 17.9 (sd = 2.18). A significant increase from pretest to posttest was found (t(11) = 7.852, p < .01). Thus students did learn using the web-based tutorials.

In addition, an independent samples t-test was calculated comparing the mean scores of subjects who used the narrated tutorial and subjects who used the non-narrated version. No significant difference was found (t(11) = -0.439, p > .05). The mean of the subjects using the narrated tutorial (m = 7.80, sd = 2.86, n = 5) did not significantly differ from the mean of the subjects using the non-narrated version (m = 8.71, sd = 4.35, n = 7). Thus, both versions of the tutorial were equally effective.

A classroom lecture control group was not used in this experiment based on the vast body of work showing that academic achievement using web-based tutorials is equivalent to, or better than, using traditional classroom instruction. Fernandez [10] found no significant difference in learning via a classroom lecture and using a Web-based tutorial. Melara [11] conducted a study that provides evidence that Web-based tutorials can speed up the learning process with the same level of achievement as class room lectures when tutorials are designed to allow the learner to access material in a non-sequential manner rather than the traditional step-by-step structures characteristic of classroom lectures. Day, Raven & Newman [12] found that student academic achievement was higher using net-based tutorials in contrast to those taught via the traditional classroom approach.

Conclusion

While anyone can follow this methodology to help plan an online resource, the full benefit comes from working through the method with an instructional designer. Questions from the instructional expert, who had no knowledge of the subject matter, forced the content expert to clearly explain and articulate the objectives, how the planned activities related to each objective and how the activities would be assessed. This give-and-take produced a well-structured and pedagogically sound development plan.

References