

Writing Effective Software Tutorials for the Delivery through the World Wide Web

Alice Y. Scales¹

Abstract

As software has become essential in engineering/technical graphics classes, the need to find efficient ways of teaching the software has become important. One method used for software instruction has been software tutorials. Although tutorials are frequently available from software manufacturers, they do not always meet the needs of a particular course or adhere to good pedagogical practices; therefore, instructors often resort to creating tutorials to use in their courses.

Since students are frequently asked to complete tutorials outside of class, tutorials need to be efficient and effective as teaching tools. This requires the creator of the tutorials to take in to consideration several issues, in particular student learning styles and background experiences. As tutorials are increasing delivered over the Internet, additional issues become important in construction of tutorials delivered this way. This paper will discuss issues and some appropriate practices when creating software tutorials that are delivered through the web. The paper will also discuss some of the lessons learned by the author when she created two web-based tutorials for AutoCAD Release 14 and 2000i.

Introduction

"Three fundamental developments have rocked the foundations of education in recent years: information overload, student diversity, and explosion of the Web as a medium of instructional delivery" (Gillani, 2000, p. 161). In engineering/technical graphics courses, instruction in CAD and modeling software has also created fundamental changes in the way courses are taught. Instruction in these software programs has lead graphics instructors to look for efficient ways to teach them. Web-based tutorials are one obvious solution to this problem. As a means of instruction, they have several advantages and disadvantages. A web-based tutorial can be extremely effective or extremely ineffective as a tool to instruct students. A large number of the web tutorials currently available do not take into consideration the nature of the learners who will utilize them and focuses instead on presenting the content rather than the most appropriate methods of presenting the content to learners. Research in pedagogy for web-based instruction has been at best sporadic due to the newness of the technology, and at present there is not a sufficient body of research to provide any foundation for practices that are appropriate for this media (Jackson & Anagnostopoulou, 2001). Some individuals would argue that educational pedagogy is not specific to the media used; however, for web-based instruction to be successful there must be a shift from teacher-centered instruction to student-centered instruction (Mayes, 2001).

Students come not only in all sizes, races, and genders, but also in all learning styles and with diverse backgrounds and experiences. The tutorial author must place the student needs at the heart of the design process and take their backgrounds into consideration. The tutorial designer who fails to recognize that different learners approach and absorb material differently has failed a good portion of the tutorial's users from the start. This is particularly true if the material is to be used without the assistance of an instructor, as many web-based tutorials are (Coomey & Sephenson, 2001). The more relevant the education is to the individual, the more meaningful the learning and retention process (Jolliffe, Ritter & Stevens, 2001). The author of this paper has created several web-based tutorials, and will discuss the advantages and disadvantage that web-based tutorials offers, some strategies for creating web-based tutorials for software, and provide some insight from her experiences.

¹ North Carolina State University

Advantages and Disadvantages of Web Tutorials

Like any instructional media, there are both advantages and disadvantages in using web-based tutorials. The World Wide Web has become part of every day life for many people, but designing instructional material delivered online requires that you understand both its strengths and limitations.

Advantages of Tutorials Delivered on the Web

The advantages of using the web as a means of delivering instruction are many. Material delivered through the web can be viewed virtually any place and at any time. Web-based instructional materials are relatively easy to update, and with the rapid changes in software, frequent updates to software tutorials are a necessity. Web-based tutorials provide an efficient way to keep material current without having to depend on professional publishers and other instructional material designers to generate them. It also allows tutorials to be tailored to the content of a particular course. Web-based instruction can also use resources already available on the web. It has the ability to include multiple media such as text, graphics, audio, video, and animation. Learners are now largely comfortable with learning on the web, and the web allows learner-centered delivery systems that takes into account the learner's own pace of learning. An additional advantage of web tutorials in a higher education setting is they can be used as instruction in or out of the classroom. The inclusion of reference material in the web tutorial places the learning more directly in the hands of the students and allows them to be their own problem-solvers and troubleshooters, major goals of an institution of learning. It also saves on cost by eliminating printing expenses (Jolliffe, Ritter & Stevens, 2001).

Disadvantages of Tutorials Delivered on the Web

There are also disadvantages to creating tutorials for the web. One major disadvantage is the technical limits of some learning environments. In order to utilize some features of sophisticated tutorials, the equipment and software on the computer used by the learner must be able to support them. Bandwidth can also be a limiting factor when downloading certain types of material. Likewise, there is the expense associated with the purchase of the equipment and software as well as its maintenance.

Creation of the tutorials, likewise, depends on the knowledge base of the web designer in terms of appropriate pedagogy and functionality. The learners may also require training to use web tutorials. However, as this form of instruction becomes more of a norm, the individuals creating tutorials will find them a natural part of their teaching activities and easier to create. Authoring programs, such as Dreamweaver and PageMill, have already taken much of the drudgery out of creating tutorials and allow the designer to easily incorporate many advanced web features. However, the designer must first invest time in learning the authoring program. Likewise, the majority of students now entering institutions of higher education are already prepared to interact with the computer systems and the Internet or receive training at the institution that prepares them to interact with the computer systems and the Internet. This makes the use of web instruction less problematic.

Probably the largest drawback to creating web-based tutorials is the time it takes to create and test them for usability and browser compatibility. Structuring tutorials that function well for students requires thought and planning. It also requires that the tutorials be tested to eliminate problem areas and improve their compatibility with the individuals who will use them. Administrators who are pushing this mode of instructional delivery frequently do not have the background to understand the time it takes to write, proof, and troubleshoot instructional material placed on the World Wide Web (Jolliffe, Ritter & Stevens, 2001).

Steps in Designing Tutorials for the Web

Designing a web tutorial involves several steps from initial planning to its launching stage. Like any instructional material, taking the time to think through the process and decide on appropriate goals, objectives, and instructional strategies produces a superior product that is appropriate for the intended users. Several texts on the subject suggest a similar procedure be followed in designing instructional web sites, which are largely based on tried and true methods that are media independent.

Identify Instructional Goals.

The place that any web designer must start is to decide what material should be included in the tutorials. This should begin by defining on a set of instructional goals for the tutorial. The identification of the instructional goals is a systematic way to decide what you wish learners to know or be able to do after completing the tutorial. To help choose these goals, a couple of questions should be answered. First, what level of expertise should the students achieve? Second, are the tutorials part of a course that will provide students with additional assistance, or are they stand-alone? The answers to these questions influence choices made in the later stages of the tutorial design (Good, 2001; Coomey & Stephenson, 2001).

Conduct Instructional Analysis

Once the instructional goals are defined, the skills to be taught should be listed and then broken down into subsequent subskills. The subskills are then written as performance objectives, which are the specific outcomes for a learning event and are written in the form of measurable outcomes. Here the tutorial author determines the rules, concepts, and information that learners must know to fulfill the tutorial's goals. Next, the tutorial author determines the order in which these skills should be taught and clusters them into manageable groups. The tutorial author needs to limit the number of objectives included in a single lesson while endeavoring to include objectives that form a natural grouping in each lesson. Skills that have some relationship to each other, or to a process, allow learners to associate and remember commands and software functions easier. However, including too many objectives in a single lesson reduces the students' ability to absorb a particular lesson's content. Although a student interacting with a computer is frequently more engaged than the same student in a lecture setting, there are still limits to the amount of material that can be absorbed in a single session.

Identify Entry Behaviors

The skills that learners must bring with them to the tutorial are referred to as entry behaviors. The entry behaviors of the learners must be considered when designing tutorials. If a range of entry behaviors is anticipated, greater detail must be included in the tutorial to help learners with less experience. By creating a profile of the users, the tutorial designer can anticipate the level of expertise the users bring to the tutorials and design for it.

Develop the Instructional Strategy

This step involves the creation of the strategies that will be used for students to achieve the objectives. Determining the means of structuring the tutorials involves thinking about how students use the tutorials, in what setting, and with what kind of assistance (if any) they are likely to have.

Select and Develop Instructional Materials

This is obviously the point in which the writing of the actual tutorials begins. In designing the instruction, several things should be taken into consideration. What layout will you use? What type of interface is appropriate? How will you organize the site? What kinds of media best represent the students' backgrounds? What linguistic style is appropriate? What kinds of graphics are appropriate? If the tutorials are stand-alone and will to be used outside of a class, the tutorials must be highly detailed so that learners do not become frustrated. If the tutorials are supplemental to a class and are completed during a class or lab session, then the instructional support may not have to be as in-depth.

Design and conduct formative evaluation

Once the tutorial is written, formative evaluations are necessary to determine where changes must be made. Generally, this involves both subject-matter experts and novice users. Feedback is necessary to both eliminate errors and refine the tutorials to better suite the learners. The evaluation information can be gathered through several means: questionnaires, interviews, observations, or tests. This is the step that frequently is overlooked by individuals writing their first web tutorials. However, it is crucial to the success of the tutorials as means of instruction.

Suggested Features for Web-based Software Tutorials

The author's experiences in writing tutorials for computer aided drawing software and has lead her to develop some practical suggestions that are not always covered by the manuals on writing this type of instruction. Some of these may be obvious, but frequently are not until students begin using the tutorials. Others have been learned by observing and talking to students who have used tutorials she has written.

Highlighting Software Patterns

To easily interact with any piece of software, individuals must form a mental model of the program. According to Jolliffe, Ritter and Stevens (2001), "Mental models are a conceptual and operational representations that people develop in order to interact with complex systems" (p.22). A mental model of a software package allows students to problem solve, troubleshoot, and predict outcomes during the program's use. All software programs have an underlying structure and pattern in the way they function and users interact with them. Helping students to see the patterns in the software allows them to more quickly form a mental model of the software's structure and procedures. Whenever it is appropriate, the tutorial author should point out these patterns to learners. Indeed, some learners may never identify these patterns unless they are pointed out.

Anticipating Holes

In examining software web tutorials, one of the most common mistakes observed by the author is that a tutorial writer has failed to anticipate the "holes" that the learner might fall into; therefore, little information is included on how to extricate the learner when he or she makes a mistake. Unfortunately, the tutorial writer's familiarity with the software actually reduces his or her awareness of the possible mistakes that learners can make when using the software and the need for strategies to recover from them. Even professionally produced tutorials provided by software companies frequently fail to anticipate the need for this type of material. Information on how to back out of a command, warnings that a step is difficult if performed incorrectly, and discussions on how to analyze errors and correct them should be part of a well-designed tutorial. Information of this type should be provided when a student uses a command or feature of a program for the first time. Thereafter, this information should be accessible in some fashion, either through a link to a frame, a link to a pop-up window, or a link to an earlier part of the tutorial. If not included, the user of the tutorial can quickly become frustrated, especially if the tutorial is used when an instructor is not present. A frustrated learner may also develop a poor attitude towards the program, which can affect their success in learning the software. This type of material should be highlighted for easy retrieval. Figure 1 provides an example of this type of information displayed in a box so students can easily locate it.

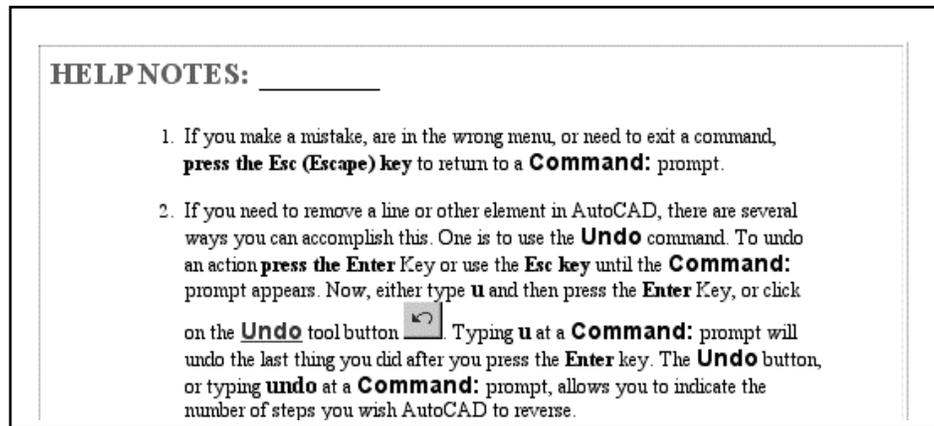


Figure 1

Testing the Tutorials

The only sure way to determine where the students will get into trouble with a tutorial or software is have 20 to 30 novice users of the software test the tutorial and provide feedback to the tutorial author. In selecting the tutorial testers, individuals with similar characteristics to the learners who will be using the tutorial should be used. Individuals who are expert users of the software should also review the tutorials. They can provide input on additional concepts that should be included and ways that instructions can be refined.

Beyond testing the site for errors in texts and instructional concepts, the site also needs to be tested on different computer platforms and in different browser programs. This is frequently not understood by novice creators of web sites. The look, font size, and alignment of graphics may vary significantly when viewed on different computers and through different browsers. For example, font sizes tend to look bigger on a PC than a Macintosh because of the difference in the way the platforms define font sizes. Other platform issues relate to color and gamma settings. Macs, PCs, SGIs, and Suns all use different color cards and none of them are calibrated to match each other. Gamma settings on computers dictate the brightness and contrast of a computer's display. Images on a Macintosh, for instance, are brighter and have more contrast than the same images displayed on a PC. Gamma and color differences can affect the readability of some font colors against certain backgrounds and element discrimination in illustrations where adjacent items are close to the same color (Weinman, 1999).

Browser differences also have a profound effect on the way a web site looks. For that reason, the web designer should view his or her site in several browsers. The best a web designer can sometimes manage, when more than one browser program is used to view a site, is a compromise in the way the site looks. Feature support in older browsers is also an issue. If the web tutorial is being designed for a wide audience with diverse platforms and browsers of various types and age, the designer should only utilize features supported by older browsers, resisting the impulse to use the newest features available. If, however, the tutorial author knows that the platforms and browsers used to display the tutorials will be limited, as in a computer lab on a college campus, then he or she is free to design for the features supported by the browsers loaded on those machines. However, the web tutorial designer needs to be aware that a tutorial that can be accessed by the general public will be used by them. In this case, the web designer may find that individuals outside of the intended audience will be upset if the tutorials are not compatible with their systems, are moved to a new location, or are later taken down or restricted to a specific population. If the web designer wishes to avoid this, they should probably limit access to the students the tutorials were written for.

Redundancy and Review

The tutorial design should include both practice and redundancy in activities so students practice each procedure several times. As is understood by all instructors, repetition is the means through which students learn software commands and procedures. It is also through repetition that students begin forming a mental model of the software's underlying structure. Tutorials must provide students an opportunity to practice the procedures either by providing additional work outside of the tutorial, opportunities to reuse the procedures several times in the same lesson, or opportunities to use a procedures in later lessons of the tutorial. This may seem obvious, but a number of tutorials that the author has dealt with a procedure only once in a lesson and then never repeat its use. To further reinforce learning, the tutorial should also include reviews of complex or important concepts. Since most tutorials contain several lessons and are not completed in a single sitting; reviews at the beginning of each lesson help bring the learners “back up to speed” before tackling new material. Figure 2 provides an illustration of a review panel included in a tutorial that covered several important concepts taught previously.

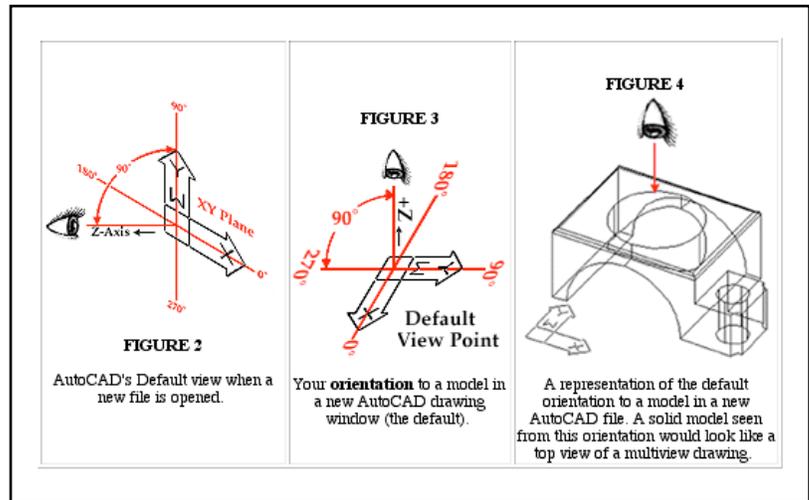


Figure 2

Isolating Command Procedure Information

For some students, a major problem with software tutorials is that command descriptions and procedures are integrated into the tutorial activity instructions. These learners have difficulty isolating the command procedures from this surrounding information. By showing a command procedure separately, they can assimilate the procedure easier. With the flexibility of the web, providing this is not difficult. The procedure can be displayed in a box on the tutorial page, on a separate web page, in a frame, or in a pop-up window.

This is particularly an issue for individuals with specific learning styles, for instance field-dependent learners. Field-dependent and field-independent are learning styles that were defined by the work of Witkins, et al. in the 1950s. According to Witkins, field-dependent learners are individuals who cannot isolate elements and patterns that are part of a more complex structure, and field-independent are individuals who can (Witkins, et al., 1977). As learners, field-dependent students require more help seeing command structures and procedures because they cannot isolate them from the discussion in the tutorials for themselves. In this situation they can follow the instructions and perform tasks when they are given step-by-step instructions, but frequently fail to be able to utilize the command again in a new situation. For this reason, displaying the commands separately from the tutorial information helps these learners to form a picture of the command procedures. Figure 3 is an illustration of a command procedure displayed in a side frame. Students can use links in the tutorial to access this information.

Quick Reference Material

The side frame, described above, also doubles in the author's tutorials as a quick reference system when students need to review procedures. Although software programs provide "Help" menus, students frequently find it difficult to locate the specific information they need in the melange of material on these menus. By providing references for the commands the users will need for the tutorial, the author reinforces the concepts of those commands and limits students' frustration. A guide of this fashion can also be designed so that it is accessible through links to a new page or a pop-up window. There are advantages and disadvantages to each of the systems. Links to other pages allow you to include larger reference graphics and diagrams; however, the learner cannot see the instructions in the main page at the same time and may have to use the back button to refer to it occasionally while reading the reference information. Pop-up windows are usually smaller, but still cover some of the main page. Displaying the information in a side frame on the same page with the tutorial directions eliminates overlap and allows the user to refer to the reference in conjunction with the tutorial instructions. The main disadvantage of using a side frame is its limited size, especially if graphics are included. To keep the side frame from overly reducing the size of the main tutorial window, it must be kept fairly narrow. However, the author elected to use this system because of the other advantages it offered. No matter what method is used, the tutorials should include a ready list of reference information that can be accessed easily as needed without having to locate a link in the body of the instructions. This is particularly important if the tutorials are to be used outside of a class setting. Figure 4 illustrates a side frame with reference links.

Letting Go of the Learner's Hand

Although a quick reference feature should be included in a tutorial, eventually the writer must let go of the learner's hand. Students, by nature, will take the easiest way out by progressing through a tutorial looking for the directions that tell them what keys need to be pressed, what parameters need to be entered, and what buttons need to be used to complete a particular task. In this fashion, learners can complete a

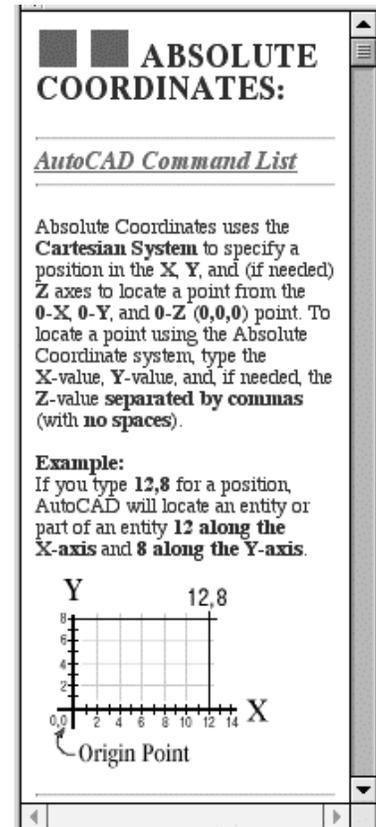


Figure 3

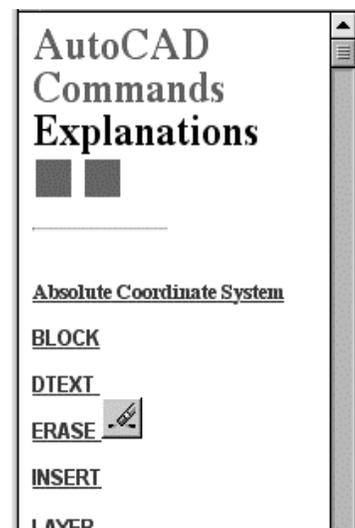


Figure 4

tutorial without ever actively reading the explanations and other pertinent information in the instructions. This "follow-the-yellow-brick-road" approach to getting through a tutorial allows students to reach "Oz," but they are unable to apply the commands to a new situation, the point of any tutorial. Telling them that they will have to use the procedure later without help sometimes gets their attention, but they will have to eventually read the material when they are on longer guided step-by-step. Another way to bring this point home to students is to begin eliminating all help and reference material in later lessons after a command procedure has been covered and practiced several times. They can still return to the earlier lesson that discusses the procedure, but that requires that they do some work to get there and demonstrates the need to learn the procedure when it is presented.

Creating a Friendly Tone

Tutorials written for student consumption tend to be rather dry. Since students spend several hours reading the tutorials, adding humor and a human touch keeps them from becoming boring. A conversational tone works best to provide this and also gives the tutorials personality. Students in higher education seem to appreciate this lighter touch and prefer tutorials where the author writes in this style.

Graphics as a Teaching Tool

Selection of appropriate graphics for tutorials is extremely important. Some learners learn best through written material and others by seeing it as a graphic. In any software instruction, screen dumps of the software windows and dialogue boxes are necessary, but graphics should not stop there. Graphics should be clean and free of unnecessary distractions. Explanations should be illustrated as often as possible and each step in a lesson should be accompanied by an illustration of the completed step so learners have something to compare their work to at each stage. This provides feedback to students on the success of their work and allows them to instantly see when they have made a mistake. It is also important that the learners are given a preview of the project, drawing, or model they are going to create during the lesson prior to creating it. This provides students with an advanced organizer, thus giving them a frame of reference and readies them for the tasks they will be performing. Figure 5 is an example of a preview of model that learners will complete during a tutorial. Notice that parts of the model are labeled to provide students with a frame of reference and improve their ability to follow the description in the written part of the lesson.

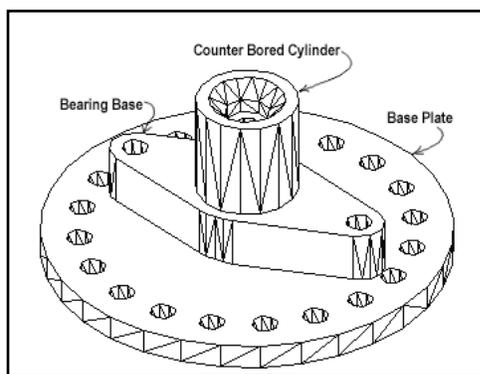


Figure 5

Graphics and File Formats

Graphic file formats selected for web tutorial, and other web sites, should be appropriate for the type of graphic used. Generally, graphics can be broken down into two categories: line drawings (contain large blocks of single colors) or continuous tone graphics (like a photograph). Determining which type of graphic you are using allows you to decide on the best file format to use. The two most common utilized formats for web sites are JPEG (Joint Photographic Experts Group) and GIF (CompuServe Graphics Interchange). Each graphic format has advantages and disadvantages as well as appropriate uses (Weinman, 1999).

GIF file formats are cross platform compatible and are the best choice for line drawings. Although the GIF format only supports 256 colors, they are always compressed and provide relatively small files. GIF compression is referred to as "looseless" compression, which means that it compresses files without removing data from the file. GIF also supports transparent images (images where one color is transparent), and are appropriate for screen dumps of software screens and windows, since program interfaces are limited to 256 colors.

JPEG files are also cross platform compatible, but are more appropriate for continuous tone images, like photographs. The JPEG format supports 16.7 million colors and also provides multiple levels of compression. Unlike GIF files, which are "looseless," the JPEG format compress files by removing data from the file. Therefore, this type of compression is referred to as "loosy" compression. The major drawback to this format is that the more compressed the file size, the poorer the quality

of the image. Another problem with JPEGs occur when they are saved over and over again in JPEG format. Each time the file is saved, more data is removed, causing a deterioration of the image quality. The rule of thumb for saving JPEG files is that they are left in a non-compression format (such as PhotoShop's .psd) until all the modifications to the image are made, and then saved as a JPEG (Weinman, 1999).

Text Size and Legibility Issues

One issue that is rarely addressed when creating a web site is text legibility, readability, and font size. Reading material on a computer monitor is not the same as reading from a printed page. Several studies have demonstrated that font sizes between 12 and 16 points works better for body text displayed on a computer monitor, although in the author's opinion, 12 point is the optimal size. The choice of font style can also affect readability. Websites can be designed so that they use the default font specified by the user's browser or can be locked down so that the tutorial is viewed in a specific font type, serif or sans serif. For body text, research demonstrates that a serif style font, such as Times New Roman and New Century Schoolbook, is easier to read than a sans serif font face, like Arial or Helvetica. The serifs on a serif style font (decorative elements on the letters) provide more visual cues to the reader and help link letters in a word so that it stands out from the space around it. It is fairly simple to limit the fonts that can be displayed in a site either through an authoring program or by html hand coding. However, more than one font face, which is similar in look, should be specified because computers do not have the same fonts. This assures that the font will look fairly similar from computer to computer. Another issue that effects readability is the choice of background color or background pattern for a site. For optimum readability, choose fairly plain backgrounds and contrasting colors for text. The optimum color combinations for readability is black letters on a white background (Spiekermann & Ginger, 2003; Weinman, 1999).

Conclusions

Creating software tutorials for the web is an efficient ways to deliver software instruction to students. It allows students to work at their own pace and complete lessons outside of class. However, it takes considerable planning to write tutorials if they are successful. The media used to deliver the tutorials has little to do with the success of the tutorials as a means of instruction, but issues related to the media also must be considered. Good instructional material design and the application of appropriate pedagogical practices are the elements that make the tutorials work. The graphics and the tutorials layouts can be both enhanced and limited by the nature of the equipment used to provide information over the World Wide Web. Consideration, however, must first be paid to the learner and then to the media on which the tutorial is displayed.

References

- Coomey, M. &, Stephenson, J. (2001) Online learning: it is all about dialogue, involvement, support and control - according to the research. *Teaching and Learning Online: Pedagogies for New Technologies*. J. Stephenson (Ed.) Stylus Publishing, Sterling, VA.
- Gillani, B. B. (2000). Using the web to create student-centered curriculum. *Issues in Web-based Pedagogy: A Critical Primer*. Robert A. Cole (Ed.) Greenwood Press, Westport, Connecticut. Pg. 161-181.
- Good, M. (2001). On the way to online pedagogy. In J. Stephenson (Ed.) *Teaching and Learning Online: Pedagogies for New Technologies*. (pp. 165-174). Stylus Publishing, Sterling, VA.
- Jackson, B. & Anagnostopoulou, J. (2001). Making the right connections: improving quality in online learning. In J. Stephenson (Ed.). *Teaching and Learning Online: Pedagogies for New Technologies*. (pp. 53-64). Stylus Publishing, Sterling, VA.
- Jolliffe, A., Ritter, J. and Stevens, D. (2001). *The On-line Learning Handbook*. Stylus Publishing, Sterling, VA.
- Mayes, T. (2001). Learning technology and learning relationships. In J. Stephenson (Ed.) *Teaching and Learning Online: Pedagogies for New Technologies*. (pp. 16-26). Stylus Publishing, Sterling, VA.
- Spiekermann, E. and Ginger, E.M. (2003). *Stop Stealing Sheep & Find Out How Type Works* (2nd ed.). Peachpit Press, Berkeley, CA.
- Weinman, L. (1999). *Designing web graphics*.3. New Riders Publishing, Indianapolis.
- Witkins, H. A., Moore, C. A., Goodenough, D. R. & Cox, P. W. (1977). Field dependent and field-independent cognitive styles and their educational implications. *Review of Education Research*, 47 (1), 1-64.