Online Distance Learning at the University of Los Andes: A software engineering approach[1]

Jonas A. Montilva (*) & Beatriz Sandia (**)

(*) University of Los Andes, Faculty of Engineering, Computer Science Department, Mérida, Venezuela

(**) University of Los Andes, Academic Vice Rector's Office, Center for Interactive Distance Education, Mérida, Venezuela 5101

Abstract: *This paper focuses on the application of good practices to the development of instructional web sites. The development of a web site, to support a distance or campus-based online course, is a very complex process that involves many instructional and technical aspects, including the instructional purpose of the course, its content, its structure, its interface, and its teaching-learning activities. The use of a method is essential for dealing with the complexity of this process. We describe, in this paper, a software engineering method that helps course designers and teachers to develop high quality instructional web sites. The design of the method was based on the integration of well-known principles, concepts and process models borrowed from instructional design and object-oriented software engineering. The method addresses the managerial processes required to plan, organize, and control the project, as well as the instructional and technical activities involved in the development of high quality course sites. The application of the method to the design of ODL courses at the University of Los Andes – Venezuela is discussed very briefly.*

1. Introduction.

Delivering an online course via the WWW involves the use of a special kind of web site, usually called *instructional web site*, *web course site* or *course site*, for brevity. A course site is a teaching-learning environment implemented and delivered through WWW to support an online course. It can be created and maintained using either generic web editors (e.g., Netscape Communicators™) or more specialized course management software tools, such as Ariadne Authoring Tools.

For creating course sites, many developers apply general-purpose methods, such as those used for developing conventional web pages and hypermedia applications (see, for example, [1], [2]). The problem of using these methods is that they do not deal explicitly with those requirements that are proper to online courses. These methods are too general to be effectively applied to the

---

[1] Published at *E-Learning in Practice - Proceedings of the Ariadne Foundation*, Lausanne, Switzerland, Vol. 1, No. 1, November, 2001.

design and development of course sites. They do not address those aspects that are specific to education, such as the course content and the teaching-learning process. Domain-specific methods are, therefore, needed to describe and guide effectively and efficiently the process of developing course sites.

Several methods and design principles, devoted exclusively to the development of online courses, have been published in the literature during the last five years (see, for example, [3] – [9]). Hall [4] and O'Sullivan [6], in particular, focus on the definition of relevant issues and design principles for developing course sites, study guides, online syllabus and other kinds of online instructional material. Boling & Frick [9] and Pernici & Casati [7] concentrate exclusively on the design activities, which are part of the whole development process. But only a few of the cited authors describe the entire process of developing an online course from its planning to its delivery. Pallof & Pratt [8], for example, discuss the whole process, but they concentrate only on the course syllabus. McCormak & Jones [5] describe the process of developing course sites using a conventional approach known as the waterfall model. This approach has, however, several weaknesses that are well documented in the software engineering literature [10].

An object-oriented software engineering approach may provide a more comprehensive view of the course site development process. By centering on the objects of a course site, this approach may provide a more natural and complete definition of the development process that includes not only the design, but also other important development activities, such as the analysis of the course, the requirements specification, the implementation and the evaluation of the course site. Additional benefits are a better structure and visibility of the process, which are essential for planning and controlling the project; a better quality of the instructional products; and a better managerial control of time, effort, and resources required to produce a course site.

We describe, in this paper, an object-oriented software engineering method for guiding instructors and course designers during the process of developing course sites to support online courses. We first define the general requirements that a method for developing course sites should comply. Then, we present a conceptual model that captures the main objects of a course site, as well as its functionality and technology. This model is essential for gaining an understanding of the purpose, general characteristics, and structure of a course site. It provides the conceptual framework needed to describe and use the method. The method structure is then analyzed in detail, followed by a brief description of its phases and steps. Finally, an evaluation of a course site developed using the method is given before the concluding remarks.

2. Requirements for course site development methods.

Developing course sites to support web-based courses demands specific requirements that made this process very different from the development of any other type of WWW sites, such as corporate home pages and e-commerce sites. Some of the most important and general properties of a course site, that are not present in other type of web applications, are the instructional objectives of the course; the structure and content of its study guide; the mode and media used for the interaction among instructors, students, and contents; and the teaching-learning process.

In addition to deal explicitly with these properties, a method for developing course sites should also comply with some general requirements. The following list is an extension of the requirements specified in [11]:

1. Systematic: The method should describe what to do and provide guidelines for how to do the activities required to produce the components of a course site.
2. Complete: It must cover the whole life cycle of a course site, which involves the typical phases of the software engineering process: requirement analysis and specification, design, implementation, testing, and delivery [10].

3. Adaptable: It must be modular and modifiable to allow its adaptability to different educational disciplines, pedagogical orientations, and development environments.
4. Technology independent: It must be independent of the software tools used to create, operate, and maintain the course site.
5. Quality Assurance: To ensure the high quality of a web study guide, the method must provide some form of quality assurance and the verification and validation of its intermediate and final products.

The purpose of the method is also a crucial aspect to be considered. A course site development method must guide the team of instructors and designers into the process of developing a course site. It must provide visibility, orientation and guidance to the team involved in this process.

3. A course site conceptual model.

An essential element for designing any software development method is the conceptual model of the objects that the method is intended to produce. A conceptual model describes the relevant concepts and the common components of the objects produced by the method. The purpose of this section is to describe the conceptual model that was used for designing our method. A good understanding of this model is a necessary condition to apply the method.

A course site is a virtual teaching-learning environment that can be seen and understood from, at least, four different perspectives or views: instructional, technological, structural, and functional.

1) The instructional perspective: A course site can be seen as an instructional and communicational medium that provides an environment to facilitate the teaching-learning process of a web-based course. It must support different kinds and modes of interactions. The interaction mode, on the other hand, determines the media (e.g., e-mail, videoconferences, bulletin board, discussion lists, and chats) that are used for establishing the communication between teachers and learners.

2) *The technological perspective*: Under this perspective, the course site is seen as a collection of interlinked web pages stored in a web server and accessed from any client computer connected to Internet.

3) The structural perspective: This view focuses on the components of a course site. A course site is made of three different components: the site management tool, the set of interaction tools, and the web study guide.

The site management tool is a web-based software system that provides the functionality needed to create, manage, update, and maintain a course site. It could be a web general-purpose tool (e.g., an HTML editor or an integrated environment, such as Netscape Communicator™), or a more specialized web tool, such as a course management tool (e.g., Ariadne Authoring Tools, Virtual-U [12], WebCT™, CourseInfo™, and TopClass™).

The set of interaction tools is a collection of media that facilitates the communication between instructors and students. E-mail editors, videoconference viewers, bulletin boards, digital disks, web browsers, and virtual chat tools are some of the most common media used in a course site to support interaction.

The most important component of a course site is, however, the web-based study guide (or *web study guide*, for short). The study guide is widely recognized as the central component of a course. It is considered as "the foundation of every good distance education course" [13]. Its purpose is to communicate teaching about a given subject to a group of students [14]. It describes the structure, the content, the interaction and the media needed to support the teaching-learning process.

Figure 1 presents an object-oriented structural model of a typical web study guide. This generic model captures the main components of a web study guide and their relationships using the widely known UML class notation [15].

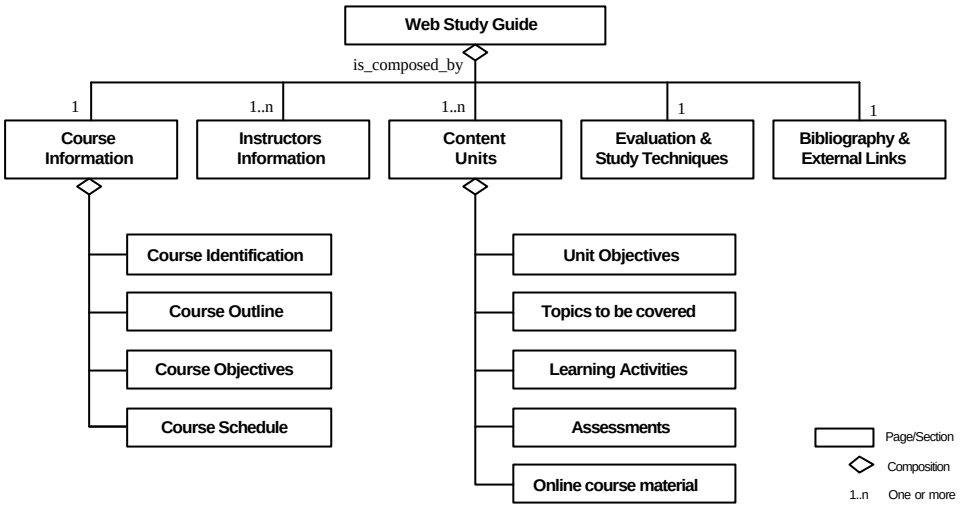

Figure 1. Typical structure of a web study guide

According to this model, a web study guide is composed by several groups of web pages that provide complete description of the course, its instructors, the course content, the evaluation procedures, the study techniques, the bibliography and other instructional materials to be used by the students.

4) The functional perspective: It concentrates on the behavior of a course site, i.e., on the operations that are available to the developers, instructors, and students. The course developers must have available, through the site management tool, the operations needed to create, administer, and maintain the site. The instructors must be able to create and update the course content of the web study guide. Other operations that are commonly used by the instructor are the monitoring of students, online grading, and the publishing of online material, guidance information, and instructions (e.g., announcements). The functionality available to the students must include additional activities other than reading the web pages, such as writing, viewing, listening, practicing and testing. The site must also provide the functionality needed to establish remote communication between students and instructors.

The course site conceptual model, as described in this section, highlights the most important elements that are present in any course site. This model provided the concepts needed to design the method described next.

4. The method for developing course sites.

The method was created through the integration of principles and process models used in Object-Oriented Software Engineering, Hypermedia, and Instructional Design. The structure of the method was designed based on an object-oriented process model for developing business applications [16]. The Instructional Systems Design model [17] was used a reference framework to outline the steps needed to deal with the development of instructional material. For dealing with the process of modeling, which is crucial in any software development method, we chose the Unified Modeling Language UML [15]. This language is widely known and has been broadly used by the software industry and academic community since its standardization in 1997.

The proposed method is structured into two processes. The first of them, called the management process, is concerned with the activities that are required to manage the development of a course

site. The second one, called the development process, is related to the instructional and technological activities that are required to produce and deliver a course site. As illustrated in Figure 2, the structure of the method covers the complete life cycle of a course site: from the analysis of the course to the delivery of the course site as a final product. It resembles the structure of a watch. The outer part of the cycle corresponds to the development process. It is made of five sequential phases that are executed in a clockwise order. The management process is at the center of the cycle to control the progress of the development phases.

The application of the method starts at the managerial process and continues through the sequential execution of the development phases. There is, however, a great deal of iteration between phases, which is quite common in the development of any object-oriented software application. The managerial process determines the flow of execution and the iteration among the development phases.

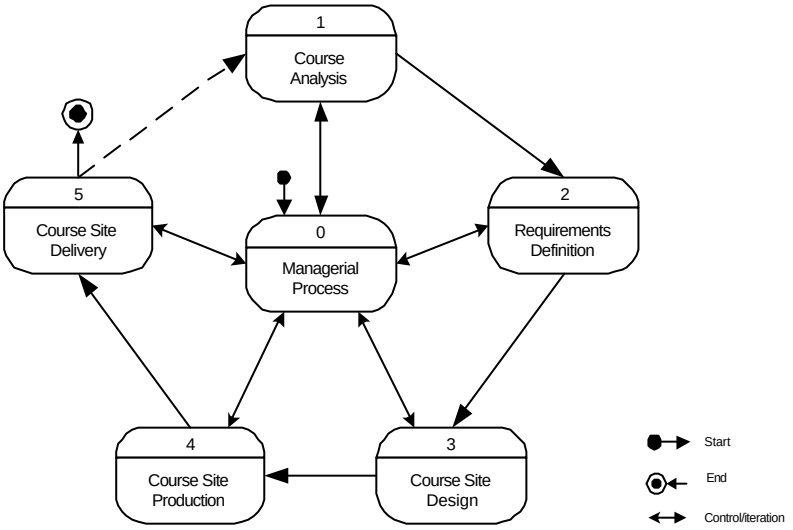

Figure 2. The structure of the method

The managerial process involves several activities, such as project management, quality assurance, tools selection, and training. These activities are described below. Table I presents a summary of these activities, including an indication of the techniques and tools that can be applied for performing each activity.

Table I. The management process

Activities	**Sub-activity**	**Techniques and tools**
Project Management	Project Planning	Activity graphs, Gantt charts, cost estimation
	Team Organization & Staffing	Organizational structures, job description
	Team Direction	Motivation, leadership, coordination
	Project Control	PERT, CPM
Quality Assurance	Verification & Validation	Technical reviews (e.g., Design Inspections and Walkthroughs) Web site design principles and standards
Tools Selection	Selection of the course-creation software tools	Software Evaluation and Selection
Training	Training the development team	Evaluation of needs and training selection
	Training the users	User's guide production

Project planning and control: Project planning is the first activity to be executed in the development of a project. It produces a schedule and a budget.

Team organization & team staffing: These activities are related to the structure and composition of the development team. A team integrated by a webmaster, one or more content experts (e.g., the course instructors), and one or more media specialists may be sufficient for developing most of the course sites. Moore and Kearsley [13] discuss alternative team structures.

Quality Assurance: The quality of a course site can be ensured through the application of verification and validation techniques [10]. Technical reviews, such as design inspections and walkthroughs, must be conducted at the end of each development phase with the participation of selected groups of students. The technical reviews ensure that the instructional products comply with quality standards and well-established design principles, such as those indicated in [1] and [4].

Tools selection: It is concerned with the evaluation and acquisition of the course management software tool. O'Sullivan [6] discusses some of these tools and provides an extensive list of tool comparison sites.

Training: Team training is performed through the development process to ensure that the development team has the skills and knowledge needed to use appropriately the method, the software tool and the development techniques. User training, on the other hand, is performed at the end of the development process to train students and instructors on the proper use of the course site.

Once that the project is planned and the team is organized, the development process is initiated. This process consists of five phases, which are explained in the next sections.

Phase 1: Course Analysis

The initial development phase focuses on gaining a good understanding of the course that will be supported by the course site. Two aspects of a course that are critical to get this understanding are its domain and its objects.

Step 1.1: Analyzing the course domain. This step is concerned with the analysis of the context or environment of the course, i.e., the educational system or program of study to which the course belongs. Understanding this program is critical for a proper definition of objectives, needs and requirements of the course.

Step 1.2: Analyzing the course objects. The main objects or entities involved in a course are the content, the student, the instructor, and the learning environment [18]. The tasks of this step are concerned with the analysis of these objects, as explained next:

- *Analyze the content*: It involves the identification and analysis of the subject of the course, the search of relevant bibliography on the subject, the definition of the course content and its scope, the organization of this content in themes, and the definition of the learning objectives and goals of the course.
- *Analyze the students*: The main variables to be considered about the student are: the knowledge, skills and abilities to be learned; prior knowledge on the subject; skills required before taking the course; students profile; and student's motivation to take the course.
- *Analyze the instructors*: Some of the most important variables or attributes to be considered about the instructors are the following: the subject-matter knowledge; their distance teaching experience and attitude; computer proficiency; their knowledge and experience on Internet services (WWW, FTP, E-Mail, News, etc.); and their pedagogical profile.
- *Analyze the learning environment*: It refers to the following aspects: the location of the students; the telecommunication technologies and hardware-software platform that they will have available at their learning sites; the social and physical environment; and the time

availability of the students for taking the course.

Phase 2: Requirements Definition

This phase is concerned with definition of quality attributes, development restrictions, and functional requirements to be satisfied by the course site. Defining requirements are needed for two purposes: (1) to guide the development team about the most important features to be considered during the design phase; and (2) to ensure the quality of the course site through the verification and validation activities.

Step 2.1: Defining the type of the course site. Depending on the objectives of the course, the development team must define, in this step, the type of course site to be developed. Knowing the type is important to determine the scope of the course site, the selection of tools, and the effort required to design and implement the course site. A course site can be classified as: instructional, supplemental, dependent, and fully online [6].

+Step 2.2: Defining the functional requirements. Based on the functional view of the conceptual model, this step defines the functional requirements of the course site, that is, the set of operations that the course site must provide to instructors and students. They can be divided into teaching operations and learning operations. Some of the most common teaching operations are the ability to create and update the course content of the web study guide, the monitoring of the students, the evaluation of students, and the publishing of online material, announcements and special instructions for the students. The learning operations include reading course information and online material, writing and submitting reports and assignments, viewing videoconferences, listening audio-conferences, group interaction, and online testing.

Step 2.3: Defining the interaction requirements. Interaction requirements are concerned with the ways the students and instructors communicate each other through the course site and how they interact with the content provided by the web study guide. The following tasks are needed to identify and define these requirements:

- Define the kinds of interaction: The team must specify and describe here the kinds of interaction (i.e., instructor-content, student-content, student-instructor, and student-student) that the site will support.
- Define the interaction mode: Once the interaction is defined, the team must determine the media that are required to support each kind of interaction. E-mail, discussion lists, chats, videoconferences, student and group web pages, newsgroups, and file transfer are some of the most common types of media used in a course site.

Step 2.4: Defining the development and operational requirements. The design and production of a course site depends heavily upon the availability of limited resources, such as time, hardware, software, people and financial support. The purpose of this step is, therefore, to define the restrictions of time, cost, people, and computing resources under which the course site will be designed and produced in the remaining phases of the development process.

Step 2.5: Defining the quality attributes. The quality of a course site is as important as its content. Achieving quality requires a commitment to well established web style rules and design criteria, such as those given in [1], [4], and [5]. The quality of a course site is mainly determined by the quality of its study guide. These attributes can be grouped as follows:

- *Content attributes*: The content is at the heart of a study guide. Some of its most important quality attributes are: the scope of the content; the logical sequence and organization of the content; its completeness; the way of stimulating or motivating the student; the feedback on assignments; the methods used for evaluating the content; and the repetition and summary of the most important ideas. Furnell et al. [19] discuss some of these requirements in detail.
- *Structural attributes*: The structure of a web study guide must be modular, visible, well

balanced, easy to modify, and easy to navigate.

- *Interface attributes*: Students interact with the course content through the pages of the web study guide. Specific attributes of the interface are the following: page length, background color and texture, design grids, size and resolution of graphics and images, and typographic design [1].

Phase 2 concludes with the verification and validation of requirements and the selection of tools, as indicated by the management process (see Table I).

Phase 3: Course Site Design

Designing the course site is the most complex and time-consuming phase. The central component of the course site - the web study guide - is designed in this phase. The four perspectives or views of our course site conceptual model - instructional, technological, structural, and functional - are used here to define the main aspects to be considered during the design of the web study guide. These aspects are the content, the user-interface, the structure, and the interaction of the web study guide.

Step 3.1: Designing the structure of the web study guide. The structure of the study guide is usually hierarchical. A possible design of this structure is exemplified in Figure 1. At the top of the hierarchy is the host page or Web Study Guide main page, i.e., the first page that is displayed on a client computer when users connect to the course site. This page must include an index or framework of links to establish the navigation to the pages of the lower level of the hierarchy, as shown by the UML statechart diagram given in Figure 3.

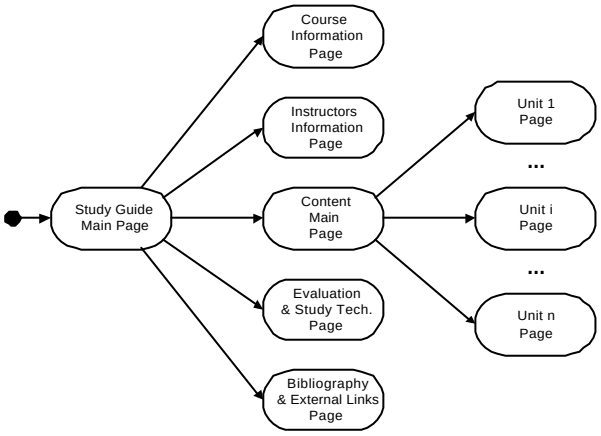

Figure 3. Navigational graph of a typical web study guide

The Course Information Page contains the course syllabus. It describes the learning objectives; gives an overview of the course content; and presents the schedule of classes. The Content Main Page describes the course content in detail and provides a link to each of the lessons or content units that conform the course. The rest of the pages are self-explanatory.

Step 3.2: Designing the content of the web study guide. The course content must be divided into a sequence of units, lessons or themes based on a pre-defined criteria, such as the conventional structure of the subject, as exhibited by the main textbooks used to support the course. The course content is then distributed over a set of pages, called *unit pages*, which are accessed through the Content Page, as shown in Figure 3. Each unit page contains a unit or lesson that covers a specific content. For each unit page, the team must perform the following tasks:

- *Define the structure of the unit*: A unit may be structured into a set of sections, including the learning objectives, the specific content, the learning activities, and the assessments. Figure 4 models the structure of a unit page.

- *Design the presentation of the unit:* The information of a unit page may be presented in many different ways. Lists and tabular organizations are two common ways of presenting the sections of a unit. Table 2 shows an example.

Table 2. Tabular organization of a unit page

Unit Number/ & Schedule	Learning Objectives	Topics	Learning Activities	Evaluation
5 Duration: 1 week From: 03/26/2001 To: 03/30/2001	1. To familiarize with well-known software process models. 2. To learn how to select the best model for a given project.	Software Process Models: 1. The Waterfall Model. 2. The Prototype Model. 3. The Object-Oriented Models.	Reading assignments: • Textbook : Read Chapter 12. • Read Unit 5: Web Material. Written Assignment: • Comparison of Models (see Assignment Description)	• Written assignment #5 (due on 02/04/2001) • Quiz #5: (06/04/01)

Step 3.3: Linking the learning activities to the interaction tools. Some of the learning activities, specified in the previous step, involve the use of media or interaction tools. Appropriate links must, therefore, be established from the learning activity section, in each unit page, to the media tools that will be used by the student. The statechart diagram in Figure 4 illustrates the associations between a unit page and the interaction tools required by the learning activities.

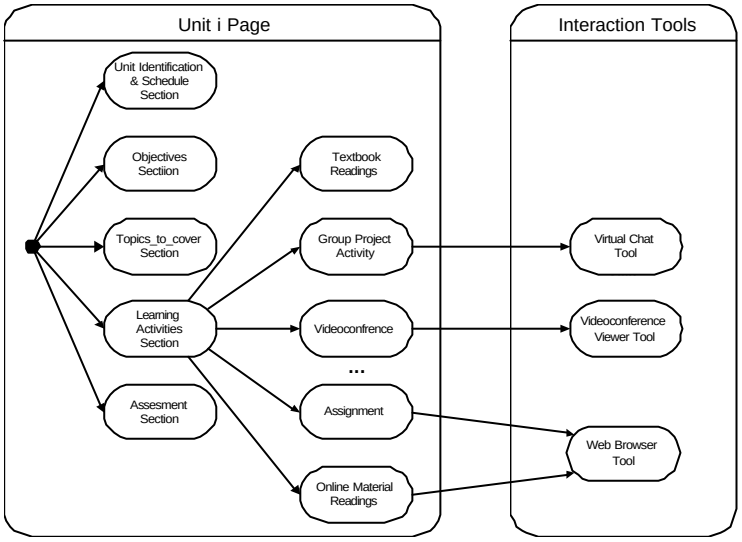

Figure 4. Structure of a unit and its association with the interaction tools

Step 3.4: Designing the user interface. The last aspect of the web study guide to be designed is its user interface. The purpose of this step is to define the aesthetics and visual characteristics of the web study guide. These details may be defined through the following tasks:

- *Design the visual structure and the navigation capability of the web pages:* The team must design here the visual structure or layout of each one of the different types of web pages that compose the study guide. A page layout indicates the location of items and buttons on the rectangular area or space occupied by the page.

- *Design the multimedia items and buttons*: The purpose of this task is to design in detail all the multimedia items and buttons to be included in the different pages of the web study guide. The content, art and visual details of each multimedia item and type of button should be outlined depending on its type.

Step 3.5: Building a prototype of the Web study guide. A prototype is a skeleton or framework that implements the structure of the study guide, which is made of pages, links and navigational aids. The purpose of the prototype is dual. Firstly, it is used for verifying and validating the design. Secondly, it is used as a framework for assembling the different items that are produced in Phase 4. The prototype will evolve into a finished product through the assembling of the multimedia items and its verification and validation, as explained in Phase 4.

The prototype must be constructed using the site management tool chosen by the Tool Selection management process. Building the prototype involves two tasks:

- *Create the structure of the prototype:* If the site management tool is a generic one, the structure of the prototype may be created just as it was designed in the steps 3.1 – 3.3. However, if a specialized course management tool is used, a mapping must be done between the designed structure of the study guide and the structure used by the tool. The pages and sections of the study guide structure should be accommodated according to the structure imposed by the course management tool.
- *Place the textual content into the prototype:* Most of the textual information associated with each page of the web study guide can be placed in the prototype using the site management tool.

Once the prototype is created, it is used by selected groups of students to verify and validate the design of the web study guide, as defined by the quality assurance activity of the management process.

Phase 4: Course Site Production

The production phase is concerned with the elaboration of the items and the study material that were designed in Phase 3, as well as their assembling into the prototype. This phase involves the following steps:

Step 4.1: Producing the multimedia items and the study material. Multimedia experts may be needed, in this step, for producing image, audio, animation and video items. Content expert or instructors must write the textual content of the study material and work with multimedia experts to produce the items required by the study material.

Step 4.2: Assembling the items and the study material into the prototype. The produced items and study material must be assembled into the prototype using the linking capabilities of the site management tool.

Step 4.3: Integrating the components of the course site. The prototype can now evolve into a course site by integrating or linking the interaction tools and publishing the web study guide into a web server.

Step 4.4: Validating the course site. As soon as the prototype evolves into a course site, it is tested again by a group of students to complete the validation process. The course content, the user-interface, the course site structure and the interaction media are evaluated by the students to determine if they comply with the requirements established during the execution of Phase 2. The results of this process are used for adjusting and refining the course site before its delivery.

Phase 5: Course Site Delivery

Once the course site has been validated, it is ready to be delivered and enter into operation. Some of the activities that must be conducted, in this step, are: producing the user's guide, training the users, and creating the student's accounts for using the course site.

This step signals the end of the development process and indicates the beginning of the operation and maintenance stage of the course site.

5. Evaluating the method.

Our method has been used for developing five online courses. Two of them are ODL courses in Software Engineering (http://www.centauro.ing.ula.ve/isoo/) and Logic & Mathematics for CS (http://www.pgcomp.ing.ula.ve/DISTANCIA/LOGICA). Both of them are part of a graduate program in Computer Science taught at the University of Los Andes (ULA) in Venezuela. The others are two web course sites for supporting campus-based education on Software Engineering. They were developed by the main author at the Department of Computer Science & Engineering at the University of South Florida, Tampa (http://www.csee.usf.edu/~montilva).

These experiences were used to provide an indirect evaluation of the method. The Software Engineering course conducted at USF was used for this purpose. During the Fall 1999, the course was taught without a course site support. Only a course page was used to describe the course syllabus. Except for e-mail, no other media was used to provide a distance interaction between the instructor and its students. During the Spring 2000, the same course was taught by the same instructor, but this time we used a course site that was developed by following the method. The course site was the only way to provide instructor-students interaction outside the lectures. The course content, structure and evaluation were kept identical for both courses. The student assessment of instruction provided the evaluation instrument. Table 3 shows the results of this evaluation.

Table 3. Evaluating the use of a course site

Evaluation Variable	Course Evaluation without Course Site Support [1]	Course Evaluation with Course Site Support [2]
Description of course objectives and assignments	3.43	4.65
Communication of ideas and information	3.36	4.41
Student assistance in or out class	3.91	4.59
Stimulation of interest in the course	3.82	4.24
Facilitation of learning	3.77	4.33

Point averages range: [1: Poor – 5: Excellent]
[1] Students enrolled: 48; Students responding: 22
[2] Students enrolled: 19; Students responding: 17

Even when these results are not conclusive about the method, they show that the use of a well-designed course site is an important factor for improving the overall quality of a course. A more formal and direct evaluation of the method is actually underway using the ODL courses at ULA. This evaluation will take into consideration the method itself, as well as other aspects of the quality of a course site such as, its user-interface, its content, and its interaction media.

Conclusions.

Developing a course site to support an online course is not a simple process. Many features, such as the instructional, structural, functional, and aesthetical properties of the site, must be considered. The method applied to develop the course site is also crucial, because it allows the development team to deal with the complexity of the process. The method, presented in this paper, is complete, modular, adaptable, technology independent, and simple enough to provide insight into the complexity of the course site development process. Its emphasis on quality assurance and its ability to integrate instructional, managerial and technical activities into a single methodological framework are also two important features of the method.

Our method differs from those described in [7], [8], and [9] in at least two important aspects. Firstly, our method is more complete. It covers the whole life cycle of a course site and describes not only instructional and technical activities, but also the managerial ones. Secondly, our method uses the object-oriented approach, which has proved to increase the productivity of

development teams, as well as the quality and adaptability of the products [20]. Our method is similar to that one described in [5]. The main differences are based on the scope and approach used by the methods. Their method is more comprehensive than ours in the sense that it describes in more detail the development phases and includes a maintenance phase. Our method, however, gives more emphasis to the managerial activities and the quality assurance, which are essential to develop efficiently and effectively good quality course sites.

Acknowledgements

This research has been sponsored by CONICIT - the Venezuelan national research council - under project No. G-97000823. The first author is also grateful to the Software Testing Center of the Department of Computer Science & Engineering of the University of South Florida at Tampa, which provided a nice environment for completing the research.

References

[1] Lynch, P. J. and Horton, S. (1999) Web Style Guide: Basic Design Principles for Creating Web Sites. Yale University Press, New Haven.
[2] Conallen J. (2000) Building Web Applications with UML. Addison-Wesley, Reading.
[3] Khan, B. H. (1997). Web-Based Instruction. Educational Technology Publications, Englewood Cliffs.
[4] Hall, R. H. (1999) Instructional Web Site Design Principles: A Literature Review and Synthesis. The Virtual University Journal, 2, 1--12.
[5] McCormack, C. and Jones, D. (1998) Building a Web-Based Education System. Wiley, New York.
[6] O'Sullivan, M. F. (1999) Worlds within Which We Teach: Issues for Designing World Wide Web Course Material. Technical Communication Quarterly, 8(1), 61--72.
[7] Pernici, B. and Casati, F. (1997) The Design of Distance Education Applications Based on the World Wide Web. In Web-Based Instruction, B.H. Khan (ed), Educational Technology Publications, Englewood Cliffs, pp. 245--254.
[8] Pallof, R.M. and Pratt, K. (1999) Building Learning Communities in Cyberspace. Jossey-Bass Publishers, San Francisco.
[9] Boling, E. and Frick, T. W. (1997) Holistic Rapid Prototyping for Web Design: Early Usability Testing is Essential. In Web-Based Instruction, B.H. Khan (ed), Educational Technology Publications, Englewood Cliffs, pp. 319--328.

[10] Bruegge, B. and Dutoit, A. H. (2000) Object-Oriented Software Engineering. Prentice Hall, New Jersey.
[11] Welsh, T. M. (1997) An Event-Oriented Design Model for Web-Based Instruction. In Web-Based Instruction, B.H. Khan (ed), Educational Technology Publications, Englewood Cliffs, pp. 159--165.
[12] Harasim, L. (1999) A Framework for Online Learning: The Virtual-U. IEEE Computer, 32 (9), 44--49.
[13] Moore, M. G. (1997) The Study Guide: Foundation of the Course. Editorial note. The American Journal of Distance Education, 11(2), 1--2.
[14] Moore, M.G. and Kearsley, G. (1996) Distance Education: A Systems View. Wadsworth Publishing Company, Belmont.
[15] Booch, G., Jacobson, I. and Rumbaugh, J. (1998) The Unified Modeling Language User Guide. Addison Wesley, Reading.

[16] Montilva, J.A. (2000) The Watch Process Model for Developing Business Software in Small and Mid-size Organizations. In Proceedings of the IV World Multiconference on Systemics, Cybernetics, and Informatics. Edited by the International Institute of Informatics and Systemics. Orlando, July, (1).
[17] Dick, W. and Carey, L. (1985) The Systematic Design of Instruction. Scott, Foresman & Co, Glenview.
[18] Dean, R., Biner, P. and Coenen, M. (1996) Distance Education Effectiveness: A System Approach to Assessing the Effectiveness of Distance Education. Education at Distance Journal, 10(3), J1--J4.
[19] Furnell, S. M., Evans, M.P., Phippen, A.D. and Abu-Rgheff, M.N. (1999) Online Distance Learning: Expectations, Requirements and Barriers. Virtual University Journal, 2, 34--48.
[20] Taylor, D. A. (1999) The Keys to Object Technology. In Handbook of Object Technology. S. Zamir (ed), CRC Press, Boca Raton, pp. 1.1 -- 1.12.