

Creating Web-Based Learning Systems: An Evolutionary Development Methodology

Said Hadjerrouit
Agder University College, Kristiansand, Norway

Said.Hadjerrouit@hia.no

Abstract

In order to become a crucial resource for learners, Web-based learning must take into consideration that education has triggered a shift from the teaching paradigm to the learning paradigm. As a result, students are becoming more independent from the teacher. Besides the evolution of the learning paradigm, educational technologies are evolving constantly. Moreover, changes caused by academic institutions, course content, ethical, legal, cultural issues need to be considered in the development of Web-based learning. Clearly, Web-based learning systems must constantly evolve in order to meet their requirements. But still, much of the development of Web-based learning is carried out without a true understanding of requirements that are proper to Web-based learning, resulting in bad analysis and design, poor quality, and maintainability. This paper focuses on the application of an evolutionary software development process model to translate Web-based learning requirements into a system that provides learners with information that supports effective learning.

Keywords: Evaluation framework, evolutionary process model, learning theories, pedagogical principles, software development, Web-based learning.

Introduction

The real value of Web-based learning lies not in accessing knowledge at any time, any place, and for anyone, but helping the right students to acquire the right skills and knowledge at the right time in order to function as active, self-reflected and collaborative participants in the information-based society (Harasim, 2000). Only then can Web-based learning become a crucial resource for learners and educational institutions. This however can be achieved only by adhering to the learning paradigm and associated pedagogical principles, and to the factors that constantly affect the development of Web-based learning (Govindasamy, 2002). Unfortunately, much of the development of Web-based learning is carried out without a true understanding of issues that are proper to Web-based learning, partly because marketing advertising and technologies still drive the construction process (Buendia-Garcia & Diaz-Perez, 2003). Clearly, there is a need is for a disciplined, systematic approach to the development process for incorporating and translating the specific requirements of Web-based learning into a system that must constantly evolve in order to ensure the relevance, correctness, and completeness of the content available on the Web.

Material published as part of this publication, either on-line or in print, is copyrighted by the Informing Science Institute. Permission to make digital or paper copy of part or all of these works for personal or classroom use is granted without fee provided that the copies are not made or distributed for profit or commercial advantage AND that copies 1) bear this notice in full and 2) give the full citation on the first page. It is permissible to abstract these works so long as credit is given. To copy in all other cases or to republish or to post on a server or to redistribute to lists requires specific permission and payment of a fee. Contact Publisher@InformingScience.org to request redistribution permission.

Clearly, there is a need is for a disciplined, systematic approach to the development process for incorporating and translating the specific requirements of Web-based learning into a system that must constantly evolve in order to ensure the relevance, correctness, and completeness of the content available on the Web.

This article is organized as follows. The first section starts with the characteristics of Web-based learning development

and the state of research. The second section presents the key principles of contemporary learning theories that form the very basis of Web-based learning. The next section describes the evolutionary character of Web-based learning construction. Then, the paper outlines the management and development processes underlying Web-based learning. Finally, the last two sections focus on the evaluation of the approach and some remarks on further work.

Development Approaches for Web-Based Learning

Characteristics of Web-Based Learning Applications

Contrary to the view of some software developers, building Web-based learning is not simply another form or a clone of traditional software development (or engineering), even if both involve analysis, design, and implementation. Basically, the development of Web-based learning is different from traditional software development in a variety of ways:

1. Learning cannot be designed as a conventional task, as though it were just another kind of work that can be implemented with conventional approaches with a number of inputs and outputs. This because learning is a by-product of understanding rather than an activity that can be supported directly (Mayes & Fowler, 1999). Therefore, progress in Web-based learning will come only from a better understanding of the learning process and not automatically from improved technology (Govindasamy, 2002; Hamid, 2002; Nocols, 2003; Tam, 2000; Watson, 2001). Clearly, unless pedagogical innovations and learners' involvement are given more attention, the development of Web-based learning practice cannot develop fully (Westera, 2005). Web-based learning is a means of implementing learning theories, and not a mode of education. Hence, learning theories must be one of the driving forces behind Web-based learning, because they are more important than the choice of the technology.
3. Web-based learning is a means of implementing education that can be applied within different educational paradigms: distance learning, blended learning, and face-to-face. First, Web-based learning may be used as online resource for distance learning at any time, any place, and for any student. Second, Web-based learning may be used for blended learning. In this case, Web-based learning does not mean to replace face-to-face learning, but to extend it with electronic means. Blended learning is a meaningful integration of Web-based learning elements with face-to-face meetings and other traditional learning settings. Third, Web-based learning may be used as supplement to classroom face-to-face teaching (Onay, 1999). It remains to be seen which solution has the potential to improve the quality of learning.
4. Web-based learning development has a broader context than traditional software engineering. It incorporates many specific issues, approaches and principles to meet the specific requirements of the stakeholders. Web-based learning development is a combination of traditional software engineering, Web and multimedia engineering, pedagogical, esthetical, human-computer interaction, cultural, and legal issues. Hence, the development process of Web-based learning is inherently multidisciplinary (Marjanovic, 2005). It must rely on knowledge from different disciplines and stakeholders with different skills, such as teachers, learners, educational researchers, graphics experts, Web administrators, and Web developers.
5. Web-based learning systems are more user-oriented than traditional software. In fact, there are multi-user systems. Thus, a significant part of any Web-based learning system concerns esthetical issues to produce look and feel of Web pages. These issues are not considered simply to enliven Web pages. They are integral to the user's experience with the system (Shiratuddin & Shahizan, 2003). Moreover, Web-based learning systems are multi-functional systems and transcend the institutional boundary. They must be developed with a user view inside and outside the academic institution (Standing, 2002).

6. Web-based learning is subject to constant change from organizational, technological, subject-specific, pedagogical, legal, and ethical reasons. Clearly, Web-based learning systems are living systems. In constantly changing environments, they must evolve rapidly in order to ensure the relevance, correctness, and completeness of the content available online (Pahl, 2003). In addition, the large number of learners with their heterogeneous needs and different learning styles makes a continuous evolution of any Web-based learning system necessary.
7. Web-based learning needs to reuse components from existing online courses, so that developers are not forced to start over again when they design Web-based learning for new courses. In addition, Web-based learning systems may be constructed by reusing and evolving standard components until the systems satisfy the problem requirements. As a result, several standardization projects have been initiated to develop standard architectures. The main focus of the standardization is to develop a format for reuse and combination of learning components and objects (Krauss & Ally, 2005; Rokou, Rokou & Rokos, 2004).

Web-Based Learning Development Approaches: State of Research

Given the specific features of Web-based learning, it is obvious that it lacks the type of specialized methodologies that exist for more traditional software development applications. There are authoring tools to produce and document Web-based learning and monitor some activities but they do not form an integrated suite of tools for the development process (Standing, 2002). Likewise, evaluation of Web-based learning cannot be done in the same manner as traditional software development, because it is embedded in a learning environment and must include issues that are proper to Web-based learning, in particular pedagogical considerations. Clearly, there are few methodologies that are appropriate for Web-based learning since most software development methodologies traditionally often focus on technical and internal aspects.

For creating Web-based learning, many software engineers apply development models that are devoted to conventional Web-based systems such as e-commerce (Balasubramaniam, 2003; Conallen, 2000; Murugesan & Ginige, 2001). However, these approaches are too general and, therefore, not adequate to be directly applied to Web-based learning, because they do not deal with teaching and learning issues that are specific to online learning.

Domain-specific approaches, which are devoted to Web-based learning (Frantiska, 2003; Horton & Lynch, 1999; McCormak & Jones, 1998; Montilva, Sandia, & Barrios, 2002; Retalis & Pappasalouros, 2005), or extensions of these approaches (Low, Low & Koo, 2003), do not explicitly address the specific characteristics that are proper to Web-based learning, such as the learning environment and pedagogical considerations based on learning theories, learner-centered design, evaluation of learning, evolution and change. In addition, few approaches concentrate on the entire development process from analysis to system evolution. Approaches that use conventional software engineering methods, such as the waterfall model (Pressman, 2000), to address the whole process are not flexible enough to be applied to Web-based learning, because they do not deal with the learning environment, evolution, change, and feedback to previous steps, and the incorporation of stakeholders from the very beginning (Hinostroza, Rehbein, Mellar, & Preston, 2000). The spiral model, which also addresses the entire development process, is complicated and difficult to manage in order to be applied to Web-based learning (Powell, 1998).

As a result, Web-based learning construction has relied heavily on ad hoc approaches rather than solid development methodology. The resulting software product is presented in descriptive formats, which do little for transferability to other institutions or even other courses (Nocols, 2003). Moreover, marketing advertising and technologies often drive the development process rather than pedagogical issues and learning philosophies (Govindasamy, 2002). In addition, Web-based

learning is often used for information transfer duplicating the objectivist techniques found in the traditional classroom based on instructor-centered instruction (Lin & Hsieh, 2001). More specifically, most course Web sites provide little support to achieve flexibility and learner-centered instruction, because they are static and rarely updated and maintained, diminishing the indisputable character of the Web (Lazarinis, 2004; Kirshner & Paas, 2001).

Learning Theories and Pedagogical Principles

The very basis of any Web-based learning system is a pedagogical foundation based on learning theories. There are many contemporary learning theories but not a single accepted one. Literature reviews suggest that theories can be related to three main commonly accepted paradigms: Behaviorism, cognitive constructivism, and social constructivism (Gros, 2002; Phye, 1997; Piaget, 1969; Skinner, 1976; Steffe & Gale, 1995; Vygotsky, 1978; Wilson, 1998). The behaviorist paradigm assumes the existence of objective knowledge independent of the learners whereas the cognitive and social constructivist paradigms view knowledge as a constructed entity made by each learner and group of learners through a learning process.

Behaviorism

Behaviorism is based on Skinners' stimulus-response theory, which means that learning is a change in the behavioral disposition of the learners that can be shaped by selective reinforcement without referring to mental processes. In terms of instruction, behaviorism assumes that the goal of learning is to efficiently transmit knowledge from the instructor to the learners. Learning is seen as largely as a passive process. In a behaviorist setting, instructors are clearly central to learning activities. However, while behaviorism promotes stability and certainty with respect to knowledge acquisition and learning outcomes, there are few opportunities for learners to express their own ideas during the course of instruction. The behaviorist model is therefore criticized for stimulating surface learning and knowledge reproduction. On the other hand, behaviorist learning is suitable for novice learners, as they need transferable knowledge from the instructor.

Cognitive Constructivism

The cognitive constructivist perspective regards learning less as the product of passive transmission than a process of active construction. It frames learning as an active construction process whereby the learners take information from the environment, and construct their own knowledge based upon prior knowledge and experience. The knowledge construction process requires cognitive skills, such analysis and reasoning skills, meta-cognitive skills, such as reflection and self-evaluation, and analogical thinking. In addition, constructivist learning takes place as learners solve authentic tasks within a meaningful, real-world environment. In a constructivist setting, teachers serve primarily as guides and facilitators of learning, not as transmitters of knowledge. Finally, assessment of learning must be embedded in the learning process and must focus on learners' individual orientations.

Social Constructivism

Social constructivism (or socially situated learning) recognizes that knowledge is partly constructed by individuals, but it is also derived from social relationships through participation in social activities with others. Accordingly, learning emerges through interaction of learners with other people, e.g. instructors, fellow learners. Learning occurs as learners exercise, test, and improve their knowledge through discussion, dialogue, collaboration, and information sharing. This means that knowledge is created as it is shared, and the more it is shared, the more it is learned. Vygotsky argued that the way learners construct knowledge, think, reason, and reflect on is uniquely shaped by their relationships with others. He argued that the guidance – scaffolding -

given by more capable others allows the learner to engage in levels of activity that could not be managed alone. This guidance occurs in the Zone of Proximal Development (ZPD) which, put simply, is the difference between what a learner can do independently and what can be accomplished cognitively with scaffolding from more knowledgeable others.

An Eclectic Approach to Learning

The literature on debate between learning theories points to the fundamental philosophical differences between them (Jonassen, Mayes & McAleese, 1993). However, in practice, a mix of learning theory is being used. Indeed, instructional designers tend to believe that what works in a learning situation is a subtle combination of behaviorism, cognitive, and social constructivism. Thus, instructional designers must allow circumstances surrounding the learning situation to help them decide which approach to learning is most appropriate (Karagiorgi & Symeou, 2005). In addition, it is necessary to realize that some learning problems require prescriptive solutions, whereas others are more suited to learner control of the environment.

An Evolutionary Development Methodology

To produce effective Web-based learning, there is a need for a systematic software development methodology, because constructing Web-based learning is a product of a creative act of development, and not a result of repetitive act of manufacturing. Thus, a rigorous, but flexible, development methodology is a crucial concern for Web-based learning in order to avoid faulty solutions, poor quality, bad analysis, design, and maintainability. A suitable development process for Web-based learning must take into consideration that education has triggered a shift from the teaching paradigm to the learning paradigm (Pahl, 2003). As a result, students are becoming more and more independent from the teacher. Change in pedagogy, in turn, affects the development process of Web-based learning. The change of the educational paradigm is a part of the changes that impact Web-based learning. Besides the evolution of pedagogy, the state of the art in educational technologies is evolving constantly. In addition, the changes inflicted by academic institutions, course content, and the subject domain need to be considered. Thus, Web-based learning systems must constantly evolve in order to meet their requirements. They must deal with change and evolution. The development process and associated modeling languages, techniques, and tools must provide support for evolution and change.

Evolution and Change

As indicated above, Web-based learning needs to evolve rapidly in order to ensure the relevance, correctness, and completeness of the content available on the Web. Thus, a continuous evolution is of crucial importance for the quality of Web-based learning. The methodology must be evolutionary and incremental since it has to deal with change and evolution of many dimensions of Web-based learning. Some of the aspects that influence a continuous evolution of any Web-based learning system are as follows:

- First, Web-based learning development calls for continuing update, change, and refinement of course content. Thus, Web-based learning content has to be constantly evaluated against what is in fact delivered online.
- Second, Web-based learning has to take into consideration the evolution of curriculum due to institutional, legal, ethical, political, and cultural changes; organization and timetabling of courses, changes related to instructors, course developers, technical staff and students.
- The third aspect of the evolutionary context is partly a direct result of the multi-user feature of Web-based learning. The large number of learners with their heterogeneous needs and different learning styles makes indeed a continuous evolution of Web-based learning necessary.

- Further, the evolutionary context of Web-based learning partially results from continuing advances in the information technology infrastructure, including hardware and software architectures, language and Web technologies, as well as mobile, pervasive and smart devices. A broad range of existing Web technologies and languages are not fully developed, and call therefore for continuing improvement.
- Finally, Web-based learning is affected by evolution of learning theory. There is indeed a growing tendency away from a behaviorist learning theory towards a theory that can be characterized as constructivist. It seems that constructivism is exerting strong influences on professional practice in Web-based learning, because it offers a springboard for pedagogical innovation. But, as indicated above, a mix of learning theory is also still being used.

Evolutionary Process Model

An ideal process model for the development of Web-based learning would help content developers, instructional designers, teachers, Web developers, and administrators address the complexity of Web-based learning, deal with evolution and change, and deliver the system as quickly as possible. However, considering that developing Web-based learning is not simply another form of software development even if both involve analysis, design, and implementation, there is not necessarily any existing software development process model that makes sense for Web-based learning.

Thus, considering that it does not exist any software development process model that fits the specifics of Web-based learning, it may be necessary to combine the advantages of the existing process models from traditional software engineering.

First, the linear sequential model called “waterfall model” is not flexible enough to be applied to Web-based learning, because it does not deal with evolution, change and feedback to previous development phases. But, this model is important from the management point of view since it can help the development team plan everything from the very beginning (Powell, 1998; Pressman, 2000; Sommerville, 2001).

Second, the spiral model is complicated and difficult to manage, but it can help the development team in particular during the analysis phase to reduce risks by focusing on what really matters.

Third, because of the evolutionary properties of Web-based learning, it appears that the evolutionary development process model is flexible enough to be applied to Web-based learning, because it modifies an early prototype, through continuous cycles of implementations, evaluation, and redesigns, until it provides all required requirements. In addition, this model involves feedback to earlier development phases and to the end users. Traditional evolutionary process models are however not without some problems for Web-based learning systems, which are subject to constant change. Thus, it would be difficult to determine when they are going to end.

Finally, considering that the reuse of previous components is a necessary option for Web-based learning systems, it is quite reasonable to rely on reusability.

As a result, it seems that the most suitable process model for Web-based learning is a modified evolutionary process model that includes some aspects of other existing process models. First, Web-based learning needs a structured process model with feedback to previous phases in order to deal with change and evolution. Second, Web-based learning needs to reuse components from existing online courses, so that developers are not forced to start over again when they design Web-based learning for new courses. Then, particular attention must be placed on the analysis of the learning environment at an early stage, since it is of crucial importance to understand the factors that affect Web-based learning. Finally, in order to deal with learning issues, which are spe-

cific to Web-based learning, the model must incorporate the evaluation of learning and underlying pedagogy.

Modeling Languages and Tools for Evolution and Change

Given the evolutionary features of Web-based learning, it appears that the object-oriented development process model provides an adequate support for constructing Web-based learning.

First, the object-oriented approach supports the underlying evolutionary process model of Web-based learning. Second, object-orientation leads to reuse of components and code. Third, it supports the maintainability of software because its structure is inherently decoupled. This leads to fewer side effects when changes have to be made. In addition, object-oriented systems are easier to adapt. Finally, object-oriented development evolves iteratively from analysis, design, to evolution with the goal of producing a software product over a series of increments (Pressman, 2000).

From the above, it follows that the very nature of Web-based learning makes the use of the object-oriented approach an essential prerequisite for an evolutionary and incremental development process. An object-oriented model is elaborated through analysis, design, coding, and evolution phases – details are added in successive iterations, changes, and refinements are introduced as needed. Development by elaboration is possible, because all object-oriented models are semantically rich and based on the same “language” – the underlying vocabulary is essentially the same (Maciaszek, 2001).

To support an object-oriented approach to Web-based learning, the development team needs an object-oriented modeling language to build visual models and discuss them with the stakeholders (Fowler, 2004; Maciaszek, 2001; Stevens, 2000). The Unified Modeling Language (UML) allows developers to express analysis and design models using a modeling notation. UML has a strong visual component and supports reusability. It is expressive enough and easy enough to understand by all people involved in the development process.

UML uses five different views that describe the system from distinctly different perspectives. Each view is defined by a set of diagrams. The following views are the most important for modeling Web-based learning:

- User model view. This view represents the system from the user’s perspective (called “actors” in UML). The use case is the modeling approach of choice for the user model view.
- Structural model view. Data and functionality is viewed inside the system. That is, static structure (classes, objects, and relationships) is modeled.

Management and Development Process

An evolutionary process model for Web-based learning encompasses two types of processes: project management process and development process.

The project management process is concerned with project planning activities that are required to manage the development process. The objective of project planning is to provide a framework that enables the project manager to make estimates of resources, costs, risks, and schedule. These estimates are made within a limited time frame at the beginning of the project and should be updated as the project progresses.

The development process is related to a software life cycle that is required to produce Web-based learning. An object-oriented software life cycle is an orderly set of activities conducted and managed for each project. The lifecycle identifies the phases along with the software product moves from the analysis of the context to delivery and evaluation. As illustrated in Figure 1 the structure

of the process model covers the complete life cycle of Web-based learning: from context analysis to evaluation. It consists of several iterative phases.

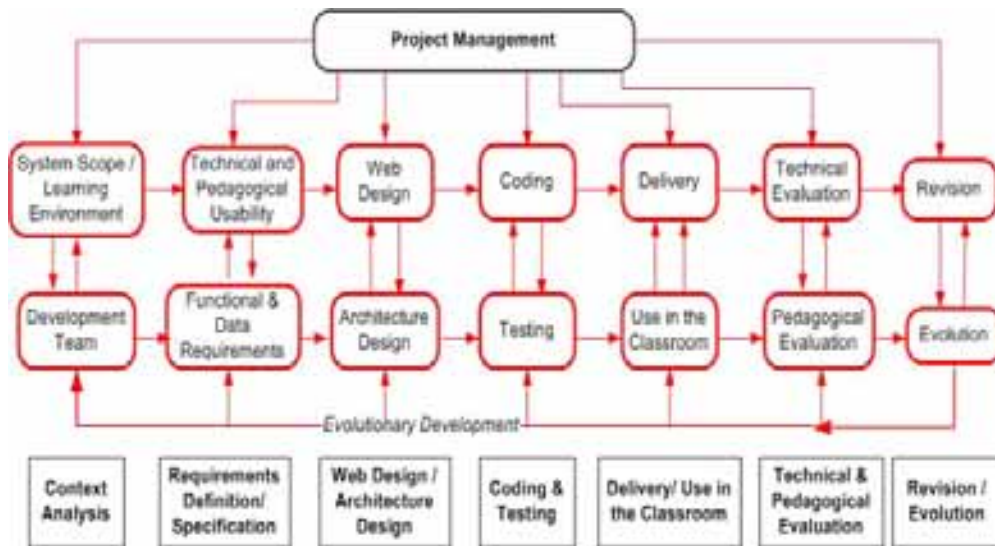


Figure 1: An evolutionary development process model for Web-based learning

Context Analysis

A crucial concern in developing Web-based learning is the scope and the learning environment of the system being developed due to ever changing requirements. Equally important in this phase are the stakeholders and the development team.

System scope and learning environment

A crucial concern in Web-based learning development is the scope of the system being developed. The system scope ensures that the requested changes do not go beyond the accepted scope. The system scope can be determined through the identification of the learning environment that directly influences Web-based learning. The system and the environment influence each other by exchanging information (Pahl, 2003). The environment can be characterized as the context of the system, which can best be described with six dimensions: the course content, learner, legal, ethical, technical, pedagogical, and usability issues. A context diagram for Web-based learning is shown in Figure 2. The rectangles designate elements that directly affect Web-based learning. The arrows depict data flows.

The *course content dimension* refers to the content of Web-based learning that is delivered online. Content is a key element since it is one of the differentiating factors that separates effective from ineffective Web-based learning (Govindasamy, 2002). Content should not be treated as self-sufficient substance that is separated and independent of the situation where it is learned. Content includes the subject matter, scope and content of the course, the knowledge domain, its topics and subtopics, the definition of learning objectives and goal of the course, as well as timetabling and syllabus of the course. This information is delivered by teachers.

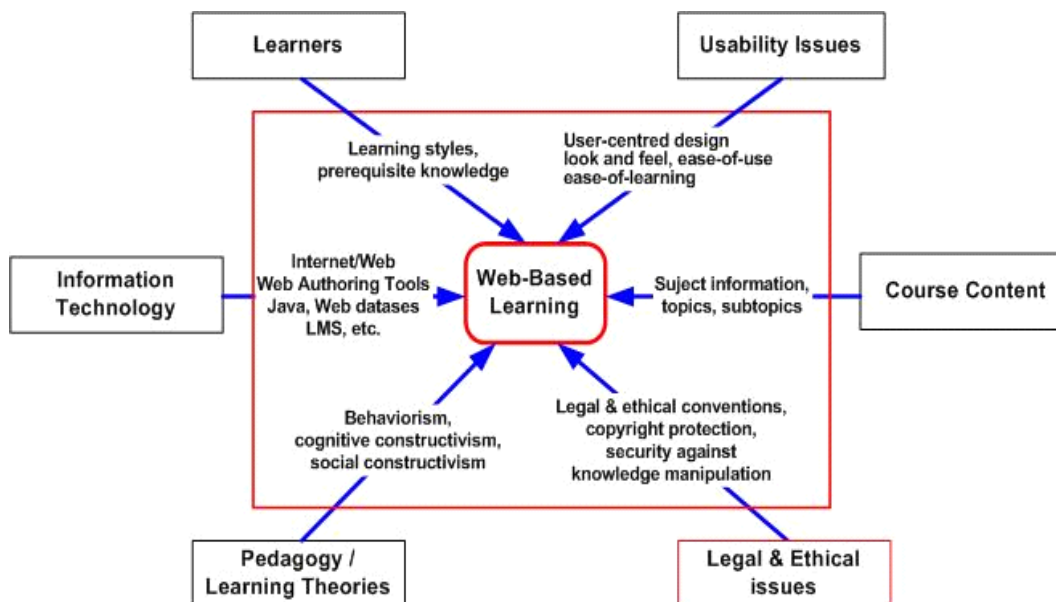


Figure 2: Learning environment and system scope

The *learner dimension* refers to the learners' characteristics that affect the development and use of Web-based learning. Learners have different knowledge backgrounds, skill levels, and learning styles. They differ in how they view the learning environment. Basically, learners within different social-cultural environments react differently to competition, authority figures, and gender differences, etc. Hence, there is no doubt that values and learning styles implicitly affect the way in which learning is undertaken (Strodher, 2003). For instance, learners may have difficulties overcoming their traditional roles as passive listeners, and they may have difficulties to adapt to the concept of Web-based learning.

The *legal and ethical dimension* refers to the legal and ethical environment of Web-based learning since any system is affected by legal constraints and ethical conventions, including copyright protection of knowledge producers, as well as security against knowledge manipulation and all forms of cheating (Graf, 2002). Finally, political issues must be considered as important factors that influence Web-based learning.

The *technical dimension* refers to the information technology infrastructure dimension, which relates to the hardware and the software environment of Web-based learning. In contrast to traditional software systems, which are built using an homogeneous technology infrastructure, Web-based learning systems run in a heterogeneous computing environment that includes multi-platforms, multi-browsers, multi-software and multimedia support. This heterogeneous environment has programming languages, automated Web authoring tools, and many other means of implementation, such as HTML, XML, JavaScript, CGI scripts, Java Servlets, Web editors and databases. In addition, wireless technologies and pervasive computing are triggering a new wave of mobile Web-based learning applications (Li, 2003).

The *pedagogical dimension* is one of the major forces behind Web-based learning, because it directly affects its implementation (Govindasamy, 2002; Hamid, 2002; Motschnig-Pitrik & Mallich, 2004; Nocols, 2003; Watson, 2001). Thus, it must form the very basis for Web-based learning development. Hence, pedagogical principles rooted in learning theories (behaviorism, cognitive and social constructivism) should exert stronger influences on Web-based learning, mostly because they offer a springboard for pedagogical innovation and far more scope for realizing possible learning benefits than Web-based learning without a pedagogical foundation.

Finally, the *usability dimension* refers to the user interface dimension, which is a central feature, because Web-based learning systems are intrinsic interactive. In fact, there are multi-user systems. Thus, a significant part of any Web-based learning system concerns esthetical issues to produce look and feel of Web pages. These issues are integral to the user's experience with the system. To translate users' requirements into a usable Web-based learning system, the construction process must be rooted in principles designed for human-computer interaction and user-centered practices, and criteria such as ease of use and learning, efficiency of use, and subjective satisfaction (Blythe, 2001; Shiratuddin, Hassan & Landoni, 2003).

Summarizing, this phase is a process of gathering data from the real environment. It includes the analysis of the course content, pedagogy, technical platform, legal, security, and esthetical issues, and the people involved in the teaching and learning environment, e.g. students, instructors, technical staff, etc. Regardless of the technical and organizational environment, the most important components are the analysis of the pedagogy, the course content, and the characteristics of the learners. Understanding learners' demographics and characteristics that impact learning is a critical issue. Learner analysis provides a better planning for a learner-centered design. Content analysis helps to identify the topics and subtopics of the course, including the prerequisite knowledge required for the students.

Stakeholders

Web-based development involves a variety of different kinds of people who have some direct or indirect influence on the system requirements. Thus, an important step in developing Web-based learning consists of identifying the stakeholders and their skills (Hadjerouit, 2005). Basically, six categories of stakeholders may influence the construction of Web-based learning:

- *Learners/students* are the users of the Web-based learning system. Thus, they must be able to operate a PC. Moreover, they need navigation skills, search engines, and file transfer. Thus, to use a Web-based learning system, they must possess some technical skills in order to browse course material.
- *Content developers* are generally teachers and instructors. They are the providers of course content, its topics and subtopics according to some pedagogical criteria. They must be able to produce, change, update, and modify the course content whenever it is necessary.
- *Educational researchers* possess substantial experiences in pedagogical research. This experience is needed for instructional design. Pedagogical knowledge is related to learning theories and philosophies, such as behaviorism, cognitive constructivism, and social constructivism. Teachers may take on the role of educational researchers if they possess sufficient knowledge in instructional and pedagogical design.
- *Web developers* are responsible for developing the system. They must be able to analyze the system's requirements, produce a design solution, implement, and test the system. They need an evolutionary process model and associated techniques and methods. Furthermore, they must be able to use Web programming languages and authoring packages, and, finally, they must possess some knowledge in human-computer interaction.
- *Graphic designers* are the constructors of the Web user interface, including the look and feel of the Web pages. Web implementation requires a subtle combination of esthetical and cognitive issues and an optimal balance between visual sensation, graphic information, text, and multimedia support.
- *Web administrators* are in charge for the total Web-based learning system, its operations, database connections, security, access rights, logging, and maintenance. Web administrators

need both knowledge in Web technologies and hardware platform in which the Web-based learning system resides.

Development team and project organization

After the stakeholders have been identified, the project management process can start. In Higher Education, it may be the university (or one of its faculties) that starts a specific project for developing a Web-based learning system (Figure 3). It nominates a project manager for organizing and managing the whole project. The project manager may be the leader of the Information Technology Department. He/she may delegate the development process to Web developers together with graphic designers and Web administrators. Web developers are then responsible for developing the system. Graphic designers focus on the graphical user interface. Web administrators are in charge for the network aspects of the system, its operation, database connections, security, access rights, logging, and maintenance.

In Higher Education, Web-based learning development can be done through HTML and Web authoring tools like FrontPage and MacroMedia Dreamweaver, Java, JavaScript, Web-databases, UML etc. Thus, developing useful Web-based learning applications does not automatically rely on information technology experts outside the institution. It may happen that the institution has Web developers, graphic designers, and Web administrators with sufficient skills and knowledge to make decisions and develop the system in cooperation with teachers and educational researchers.

Web developers, graphic designers, and Web administrators work in consultation with content developers/teachers, educational researchers, and learners in order to specify the system requirements. The development process continues with design, coding, and usability testing until the development team delivers a well-documented Web-based learning system that can be used and evaluated in the classroom. The evaluation ensures that learning issues are kept in mind, and that the decisions made throughout requirements analysis, design, and implementation are achieved.

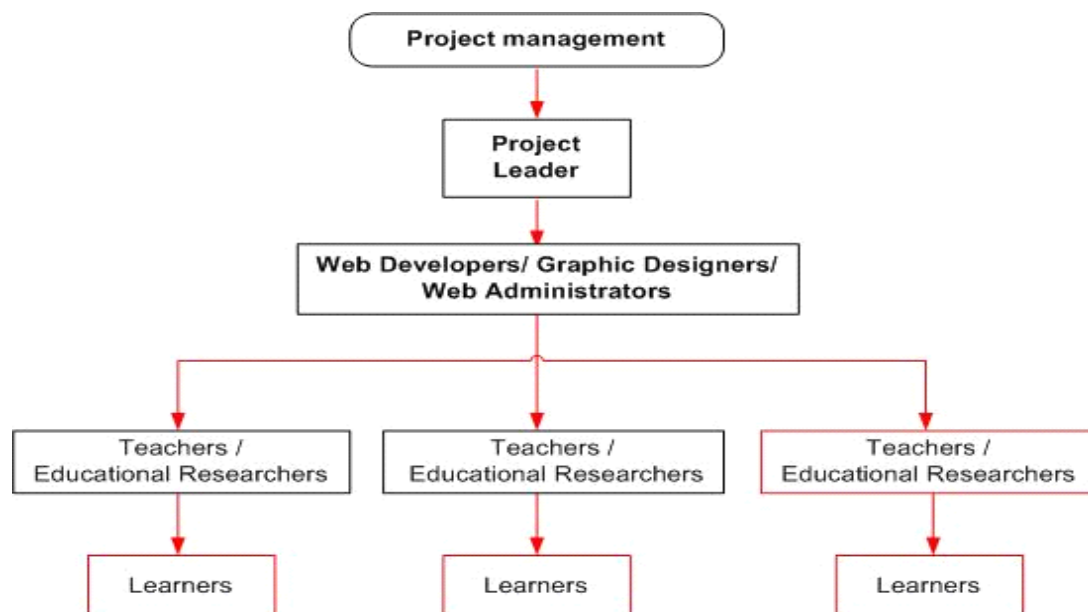


Figure 3: Web-based learning development team and project organization

Requirements Definition and Specification

The requirements definition and specification phase captures systems requirements and defines them as natural language statements with UML diagrams (Maciaszeck, 2001). The development team through consultation discovers the system requirements. The consultation involves Web developers, educational researchers, teachers (content developers), learners, graphic designers, and eventually Web administrators. This phase involves various techniques of gathering information from the involved people, such as structured and unstructured interviews, questionnaires, study of documents, observations, group meetings, etc. The product of the requirements definition phase is a requirement document. This is mostly a narrative text document with UML diagrams.

The requirements definition and specification is concerned with:

1. Technical usability criteria
2. Pedagogical usability criteria
3. System requirements
4. System constraints

Before defining the system requirements and constraints, the development team must analyze the usability attributes to determine how they are going to be designed during the design phase.

Technical usability involves techniques and methods for ensuring a trouble-free interaction with the Web-based learning system while pedagogical usability aims at supporting the learning process. Both aspects of usability are closely related to each other. The goal should be minimizing the learners' work resulting from the interaction with the system in order to free more resources for the learning process itself (Melis, Weber, Andres, 2003).

Technical usability criteria

Technical usability is related to how a Web-based learning system is convenient, practicable, and usable for the learners. Specifically, there are many factors affecting the technical usability of Web-based learning systems, such as site structure, local search, navigation and linking, screen appearance, and interactivity (Nilsen, 1993, 2000). Nilsen's factors of Web usability include page and content design. Page design has three components: cross platform, speed of page access, and page linking. Content design has two components: writing for scannability and media use. Similarly, IBM Web design guidelines focus on five major factors: structure, navigation, visual layout, textual content, and media elements (IBM, 2000).

Given these considerations, the following criteria are considered as very important for the design of the technical usability of Web-based learning. They are divided into generic usability criteria that are applicable to most educational software and usability criteria that are specific to Web-based learning:

- (a) Effectiveness, efficiency, user satisfaction, ease-of-use, and ease-of-learning
- (b) Cross platform, accessibility, navigation and linking, and content design.

Pedagogical usability criteria

The technical usability is a self-evident requirement, but it is not sufficient when designing usable Web based learning. The usability concept must be extended to capture issues that are fundamental to learning. To construct usable systems, pedagogical usability criteria must be incorporated into the development process. Important for the design of the pedagogical usability is a pedagogical foundation built on learning theory. Thus, the criteria that influence the pedagogical usability of Web-based learning are those that are associated with learning theories. The starting point for defining the criteria is to split the learning process into three types of learning: a behaviorist, a

constructivist, and, a social constructivist phase of learning. The suitable combination of the criteria produces a pedagogical usable Web-based learning.

The most important criteria that have to be considered when designing Web-based learning for behaviorism are as follows (Mayes & Fowler, 1999):

- Break down the subject matter into small learning units (lessons) and logically discrete instructional steps
- Provide a well-structured presentation of information
- Allow easy accessibility of information
- Provide powerful explanation of the information

From a constructivist point of view, the most important criteria that must influence the design of Web-based learning are the consideration of learners' individual characteristics and prior knowledge, motivation, authentic tasks, self-reflection, flexibility of learning material, multiple representation of information, and assessment of learning embedded in the learning process. Hence, Web-based learning may be designed to support the following tasks (Wilson, 1998):

- Authentic task-based activities and exercises taken from real-world situations
- Well-designed examples that students may follow when they perform task-based activities
- The presentation of knowledge from previous versions of the course that students may reuse
- The recording of learning material that is intrinsically motivating
- The multiple representation of information using various media elements
- Evaluation procedures that are embedded in the learning process
- Links to interactive educational software (WRML/animations, simulations, multimedia, etc.)
- Feedback from instructor to the learners towards solutions to authentic problems

From a social constructivist point of view, the most important criteria for Web-based learning are the consideration of the social context in which learning takes place, the concept of scaffolding based on Vygotsky's definition of Zone of Proximal Development (ZPD), social interaction, collaboration, and dialogue with fellow learners, instructors, and tutors. Hence, Web-based learning may be designed to support the following tasks (Kunz, 2004):

- Arena for collaborative assignments to produce a shared project report, or solution to a problem
- Forum for dialogue through Web-enabled discussions with the instructor and fellow students
- Online submission of compulsory project work and online instructors' feedback as well
- Shared workspaces containing resources to all students. Such workspaces would allow collaborative work on shared tasks
- Spaces that can be tailored according to the situational needs. This includes students having the rights to add, modify, customize, manage, and delete items themselves
- Links to online databases, online journals, software libraries, interest groups, etc.

System requirements

Based on usability criteria, the main part of the requirements is dedicated to the definition and specification of system requirements. System requirements are twofold: functional requirements and data requirements. Functional requirements can be modeled with use case diagrams at a high level of abstraction. Data requirements can be modeled with class diagrams at a high level of abstraction.

Functional requirements include the definition and specification of the set of operations (or use cases) that the Web-based learning system must provide to actors: teachers, students, administrators, and other users. These can be divided into teaching, learning, and administrative operations. A subset of operations is shown in Figure 4:

- Teaching operations and associated use cases are: register teacher information, teacher login, update course content (display, create, change, and delete course content), review learning activities, answer questions and requests, check course evaluations
- Learning operations and associated use cases are: register learner information, learner login, display course content, perform learning activities, establish dialogue, perform online dialogue and group interaction, and, finally, perform assessment and course evaluation.
- Administrative operations and associated use cases are: update database, produce statistics, and create new pages. These are performed by Web administrators. The operation “display course evaluations” is performed by the study leader of the institution.

Data requirements include the definition and specification of three data models:

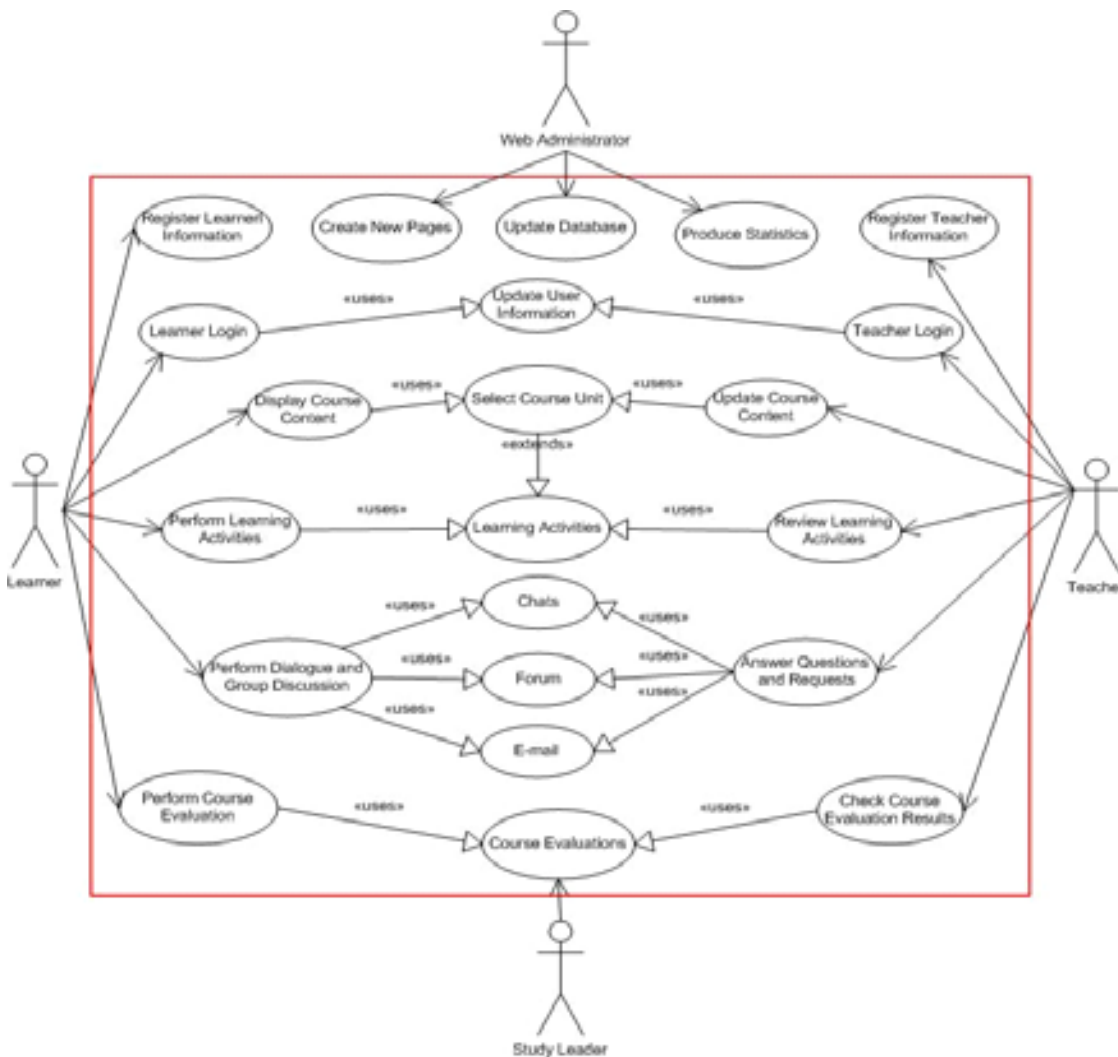


Figure 4: Functional requirements: High level use case diagram.

1. The course model
2. The pedagogical model
3. The user model

First, requirements elicited from teachers/instructors constitute the course requirements. They capture the educational goals, the learning material, the topics and subtopics, and the type of the skills that the students need to learn.

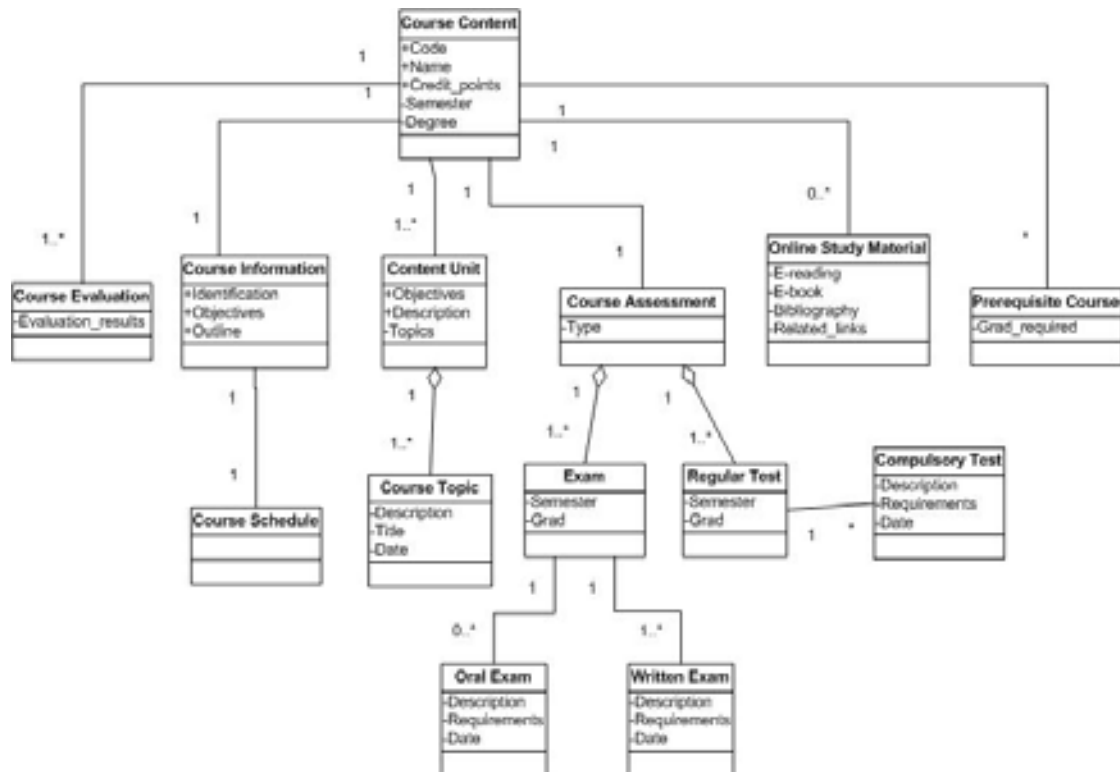


Figure 5: Course requirements: High level class diagram.

The classes that specify *course requirements* are described with primitive attributes. Only the most interesting attributes are shown. The class diagram in Figure 5 shows the most apparent relations between the classes. The class Course Content consists of six associated classes: Prerequisite Course, Course Information, Course Unit, Course Evaluation, Course Assessment, and Online Study Material. The class Course Content has one or more Content Unit, which has one or more Course Topic respectively. Similarly, the class Course Assessment has one or more Regular Test and Exam, which themselves have one (or not at all) or more Oral and Written Exam, respectively. Finally, class Regular Test may be linked to many Compulsory Test.

Requirements derived from educational researchers specify the most appropriate pedagogical strategies rooted in current learning theories. Thus, the specification of *pedagogical requirements* relies on pedagogical strategies based on learning theories with three types of learning: behaviorism, constructivism and social constructivism:

- To support behaviorist learning, Web-based learning must be designed to support the presentation of the subject matter according to pedagogical criteria defined above.
- To support cognitive constructivist learning, Web-based learning must be designed to support task-based activities and project work rather than the presentation of the subject

matter. Learner activities may use interactive software, such as simulations, VRML, multimedia, animation, etc. Appropriate links must therefore be established to interactive software tools.

- To support social constructivist learning, Web-based learning must be designed to support collaborative learning, enabling students to ask questions and discuss issues and concepts of the subject matter, as well as solutions to task-based activities. Collaborative activities are supported by online dialogue (chat, discussion forum, E-mail), online workspaces and online databases.

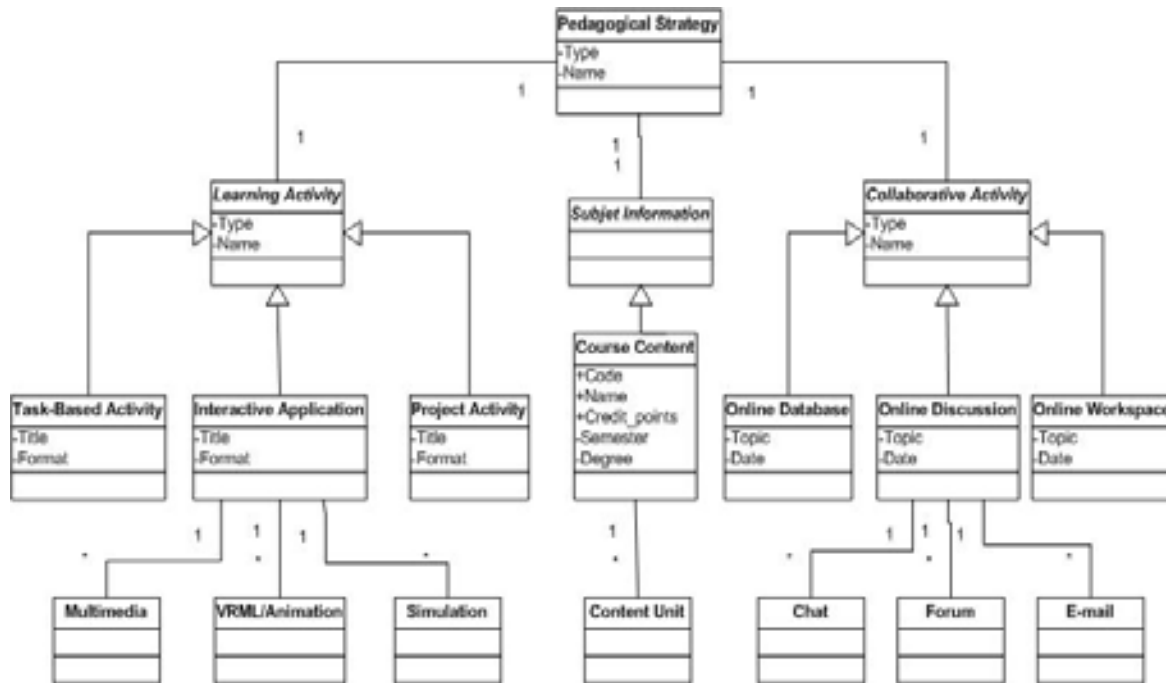


Figure 6: Pedagogical requirements: High level class diagram

To specify pedagogical requirements, the class diagram must contain a class Pedagogical Strategy that is linked to three classes (Figure 6): Subject Information, Learning Activity, and Collaborative Activity. These classes are themselves generic abstract classes for concrete subclasses (Task-Based Activity, Interactive Application (Multimedia, VRML/Animation, Simulation), Project Activity, Course Content, Online Database, Online Discussion (Chat, Forum, E-mail), Online Workspace). Class Content is the class defined in course requirements.

User requirements are specified through the classes that represent the users of Web-based learning: Teachers, learners, and Web administrators (Figure 7). Thus, the class diagram must contain a generic abstract class User for three concrete subclasses Learner, Teacher, Web Administrator. Teachers provide new study material, update the existing one, review learning activities, check course evaluations, etc. Learners use Web-based learning to satisfy their educational needs. Web administrators create new Web pages, update the database and produce statistics. Users (class User) have rights (class Rights), which allow them to login with a username and a password. Class Rights must be linked to Class Users. In addition, a class Login is needed.

System constraints

System constraints describe how the system is constrained when accomplishing its functions. Web-based learning system constraints are set with regards to:

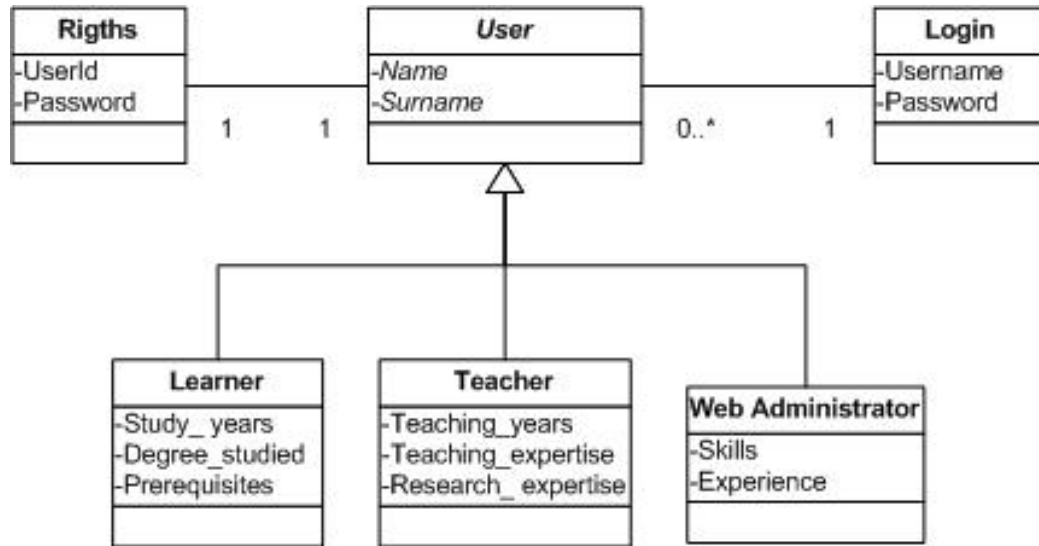


Figure 7: User requirements: High level class diagram

- Performance requirements
- Security requirements
- Operational requirements
- Political and legal requirements

Performance requirements can become quite central to the success of Web-based learning. They specify the speed (the system's response time) at which various tasks have to be accomplished. Performance requirements ensure a trouble free function of the system.

Security requirements describe user's access privileges to the information under the system's control. User can be given restricted access to the Web-based learning, including restricted access to data and/or restricted rights to execute certain operations on data. Some of these requirements are related to user requirements.

Operational requirements determine the hardware/software environment in which the system will operate. These requirements may have an impact on other aspects of the management process of Web-based learning, such as system maintenance and update.

Political requirements are frequently assumed rather than explicitly stated. These requirements are derived from the institutional environment, and specify the institutional, legal, and ethical issues. These requirements are very important because the system may be difficult or impossible to use for political, legal, and ethical reasons.

Design

This design phase is concerned with Web design and architecture design.

Web design

The structure of the Web-based learning system is usually hierarchical with the top as the home page which presents general information about the system. Then, the system must be broken down into smaller components, which themselves may consist of one or many Web pages according to the topics and subtopics of the subject matter (Figure 8).

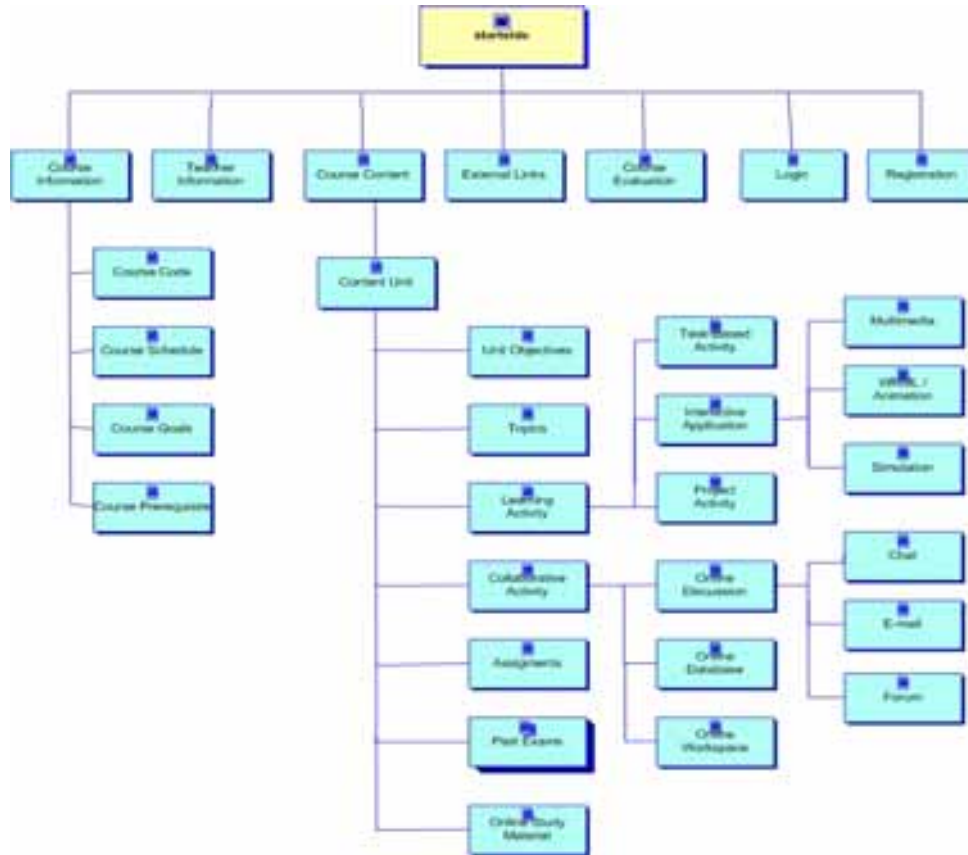


Figure 8: Web-based learning structure

Architecture design

Architecture design is done in terms of the software/hardware platform on which the Web-based learning system is going to be implemented. This platform is the three-tier client/server architecture (or BCD model) that separates data management (Entity) from presentation (Boundary) via the application logic middle tier (Control). The Boundary-Control-Entity-Database (BCED) (an extension of the BCD model) is an approach to object-oriented modeling based on four-way factoring of classes. The advantage of the BCED is its alignment with the client/server model. UML predefines four stereotypes on a class package: boundary, control, entity, and database (Maciaszek, 2001). The boundary package describes objects that represent the interface between an actor and the system. A control package describes the classes that intercept user input events and control the execution of an instructional process. An entity package describes classes that represent the semantics of entities in an educational domain. The database package describes classes that perform the extraction of data from the database. The main database package CRUD – Create-Read-Update-Delete package mediates between the entity classes and the database tables whenever the system needs to access or modify the database content. Figure 9 shows architecture design for Web-based learning. Only the most important packages are shown.

An object-oriented architecture model is elaborated through the analysis and design phases – details are added in successive iterations, changes, and refinements are introduced as needed.

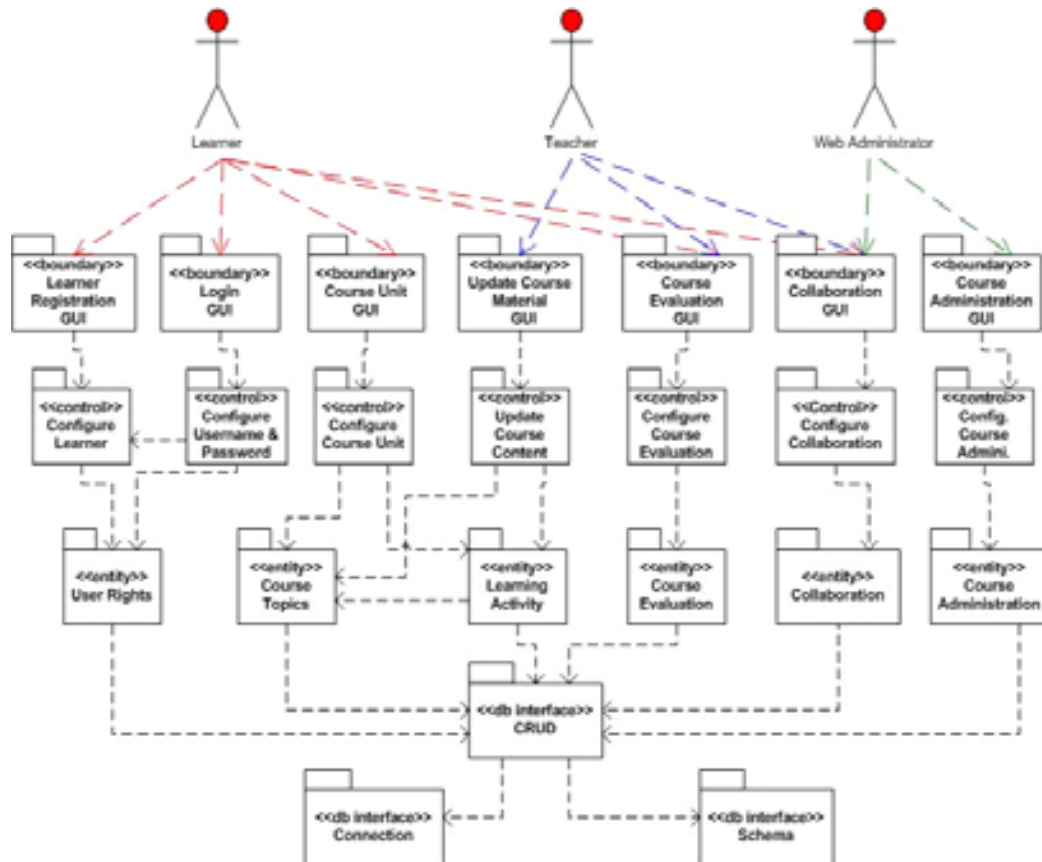


Figure 9: Architecture design

Coding and Testing

This phase enables the production of multimedia elements, e.g. text, graphic, animation, and eventually audio associated with the Web-based learning. This is followed by the implementation of the study material associated with the content pages using programming languages or specialized authoring systems or tools such as FrontPage or Macromedia Dreamweaver. It may be also possible to modify, refine, and reuse previous components, classes, and objects (Krauss & Ally, 2005; Rokou, Rokou & Rokos, 2004) that implement some of the functionalities of the system. Components may include reusable course units. Classes may contain reusable learning objects that may be lessons, exercises, assignments, learning activities, project reports, documentation units, past exams, etc.

Before delivery, the system must be tested systematically. Testing is the process of exercising the system with the intent of finding and ultimately correcting various errors, such as typographical errors, grammatical mistakes, errors in content, errors in graphical representations, cross referencing errors, navigation errors, etc. This phase includes also unit and integration testing, content review, platform, browser and operating system compatibility, user interface testing and interaction with the users, performance and reliability testing, etc.

The testing approach to Web-based learning adopts the basic principles of all software testing and applies a strategy that has been recommended for object-oriented systems (Pressman, 2000). The following steps summarize the approach:

1. The content of the Web-based system is reviewed to uncover errors

2. The design model is reviewed to uncover navigation errors
3. Selected components are unit tested
4. Integration tests are conducted
5. The assembled Web-based learning system is tested for overall functionality
6. The Web-based learning system is tested in a variety of different environment configurations and is tested for compatibility with each other
7. The Web-based learning system is tested by a controlled and monitored population of end users

Technical and Pedagogical Evaluation

After the delivery, installation, and use in the classroom, the evaluation of the Web-based learning system begins. There are many methods that may be used to evaluate Web-based learning (Benigno & Trentin, 2000; Dyson & Campello, 2003; Elissavet & Economides, 2003; Lam & McNaught, 2003; Mayes & Fowler, Nulden, 2001; Storey, Phillips, Maczewski & Wang, 2002). The evaluation ensures that the objectives are kept in mind, and that the decisions made throughout the phases of requirements definition and specification, design, coding, and testing are achieved.

Basically, the evaluation must be conducted not only for the purpose of reviewing and revising the technical quality and the quality of the learning material but also for the evaluation of students' learning and the underlying pedagogy, since these are the driving forces behind the development of Web-based learning.

Many evaluation instruments can be used to assess the pedagogical value of Web-based learning, for example individual interviews, group interviews, survey questionnaires, and video-taped observation sessions are relevant instruments to obtain data on what learners feel and think about Web-based learning. Learners' examinations and test performances are also adequate data for assessing the effectiveness of Web-based learning for acquiring skills and knowledge of the subject matter. Finally, teacher reflections and expert opinions in educational research are also good instruments to elicit the pedagogical value of Web-based learning.

After the evaluation, the system may eventually be changed, updated and improved according to the recommendations expressed by the evaluation team.

Evolution

Considering that Web-based learning systems will undergo change once they are used in the classroom and evaluated, it is necessary to plan an evolution phase in order to ensure that the content is updated and revised, and the pedagogy is improved, etc. Thus, a continuous evolution is of crucial importance for the quality of Web-based learning.

Evaluating the Methodology

A development methodology can be evaluated in a variety of ways (Standing, 2002). First, it could be evaluated according to a framework of requirements or rationale to determine if these are met with the methodology. The benefit of this approach is that it can be carried out by software developers with sufficient knowledge background in Web-based learning. The weakness is that all problems are unlikely to be identified. Second, focus groups are another method of evaluating development methodologies. This approach has the advantage of getting input from a range of practitioners and experts in the field of Web-based learning. The limitation is that all issues and problems may not be identified until the methodology is used in practice. Finally, the methodology can be evaluated by adopting it as the development methodology on a project as a form

of action research and design-based research (Barab & Squire, 2004; Baskerville, 1999). According to Standing (Standing, 2002), this is an ideal approach to evaluating a methodology.

Evaluation Requirements

This work has been evaluated according to a framework of requirements. The requirements that have been used to evaluate the methodology are based on the work of (Montilva, Sandia, & Barrios, 2002). Accordingly, a framework of requirements consists of four general evaluation principles that may be applied to any software development methodology. The first principle relies on the idea that any development methodology must have a clear specified domain. The second principle is related to the usability properties of the methodology. The third principle is concerned with the underlying product model, and the fourth principle is related to the process model of the methodology. In addition, these general principles must incorporate specific aspects that are proper to Web-based learning.

As a result, the evaluation framework consists of a set of four requirement principles: applicability domain, usability properties and user involvement, product model, and process model. Each principle can be described with a set of attributes.

The set of attributes of the applicability domain are the scope of the methodology and the pedagogical domain of Web-based learning systems. These attributes include the mode of education, the course type, and the educational level.

The usability and user involvement principle includes the applicability and completeness of the software life cycle, usability properties, and user involvement.

The attributes of the product model principle are concerned with the product model of the methodology, modeling language and notation, as well as the concepts underlying the model.

The attributes of the process model principle are related to the process model of the methodology, management processes, and development processes.

Applicability domain

In contrast to conventional approaches to Web-based learning development, the methodology presented in this paper has a broader application domain than most existing methodologies. The coverage of the methodology includes software, management, and pedagogical issues. Particularly important for Web-based learning is the consideration of the pedagogical domain, which is explicitly described with learning theories and how they can be implemented.

The approach provides an explicit methodology for developing Web-based learning. It is applicable to a wide range of situations where educational institutions are expecting to gain learning benefits in investing in Web-based learning. It is flexible enough to be adapted to the specialized conditions of the educational institutions.

More specifically, its application area includes education ranging from high schools to higher education. The methodology can be applied at different levels of educational institutions from high schools to higher education. It can be campus and distance-based or a combination of them. Further, a variety of Web-based courses may be developed using the methodology. Finally, the methodology is independent of a specific instructional approach to that it can be applied within a variety of educational situations.

Usability and user involvement

The methodology covers the whole life software cycle, including the evaluation of learning. In contrast, few approaches concentrate on the entire development process from analysis to evalua-

tion. In addition, in contrast to many existing approaches, the evaluation phase, as described in this paper, is sufficiently taken into consideration. Likewise, the living and evolutionary character of Web-based learning systems is explicitly emphasized in this approach. Hence, the methodology of this paper is more complete than existing approaches as it covers the whole life cycle, including pedagogical evaluation.

Furthermore, the methodology shows the phases of the development process and how to perform them. It uses standard techniques, is easy to use and flexible enough to be adapted to a variety of educational situations. The visibility of the methodology is improved through modularity and the three-tier client/server architecture (or BCD model) that separates data management (Entity) from presentation (Boundary) via the application logic middle tier (Control).

Finally, the methodology involves all the stakeholders in all phases of the development life cycle: content developers and instructors, learners, software developers, educational researchers, graphic designers, Web administrators. User involvement is important to ensure that all perspectives are taken into consideration: the technical, esthetical, engineering, cultural, pedagogical, etc.

Product model

The product model is concerned with the explicit representation of the Web-based learning system. This is described by the characteristics of the course being developed. The methodology clearly defines the properties of the system by describing its features and dimensions, both technical and non-technical, including esthetical and pedagogical concepts.

The methodology uses both graphical and textual notations for specifying the product model. In contrast to existing approaches, with few exceptions (Montilva, Sandia, & Barrios, 2002), the product model of the methodology is explicitly described following an object-oriented approach. Analysis and design modeling are expressed by using the Unified Modeling Language (UML). There are many advantages using an object-oriented approach with UML. First, analysis and design are connected. Second, UML increases the productivity of development teams.

Finally, the product model is described from different perspectives: learner, teacher, and administrator perspectives, etc. Existing methodologies do not sufficiently take into consideration the manifold perspectives of Web-based learning systems, that is, what the systems constitute. This limits their suitability. Basically, it is difficult for any approach to include all systems perspectives. But, the combination and use of multiple perspectives (at least the most important) would appear to be the most appropriate solution to this issue.

Process model

Existing process models for Web-based development are tied to a process which limits the applicability and flexibility. Approaches that use conventional software development approaches, or extensions of these approaches, to address the whole process are not flexible enough, or too complicated, to be applied to Web-based learning. Domain-specific approaches that are exclusively devoted to Web-based learning, do not explicitly address the specific characteristics that are proper to Web-based learning, such as learning theories, learner-centered design, multi-functionality, evolution and change. The approach described in (Montilva, Sandia, & Barrios, 2002) takes into consideration instructional aspects, such as education level, modality, course type, but not learning theories. This is clearly insufficient for implementing Web-based learning.

The methodology presented in this paper combines the advantages of existing process models. It is a modified evolutionary process model that includes some aspects of other process models.

Moreover, the methodology is not dependent on one technology or software tool. In contrast to traditional software systems, which are built using on homogeneous technology, Web-based

learning systems run in a heterogeneous computing environment that includes multi-platforms, multi-browsers, and multimedia support.

Finally, the process model encompasses both project management and development processes. Both are required to produce Web-based learning systems.

Conclusion

This work provides a framework for understanding the challenges, potentialities, difficulties, and complexity of Web-based learning. So far, the conclusions that can be drawn are as follows:

Developing Web-based learning is a complex matter. Web-based learning has a number of components and subcomponents, which include both technical and non-technical aspects. There are methodologies, but few that provide an overarching framework for developing Web-based learning. The methodology presented in this paper aims to provide such a framework. It attempts to provide pedagogical and methodological foundation as prerequisite for effective implementation of Web-based learning.

This paper highlights a number of issues for Web developers, teachers, graphic designers, educational researchers, and Web administrators.

First, to produce effective Web-based learning in real environments, there is a need for a systematic software development methodology, because constructing Web-based learning is a product of a creative act of development, and not a result of repetitive act of manufacturing. Thus, a rigorous development methodology is a crucial concern for Web-based learning in order to avoid faulty requirements definition, design solutions, poor quality, and maintainability.

Second, Web-based learning should be developed by software developers in consultation with other stakeholders, that is any person affected by the system or who has influence on system development, e.g. teachers, educational researchers, learners, graphic designers, and Web administrators. This would mean that the stakeholders should be involved throughout the development of Web-based learning to ensure that all requirements are met. This, because the main causes of software failure can be traced to the stakeholder factor.

Third, the methodology must cover not only the whole software cycle from requirements analysis, design, implementation, to evaluation, but also the complexity of the learning environment. The evolutionary process model has therefore been modified to incorporate the learning environment of the system at an early stage.

Fourth, the methodology must be evolutionary, incremental and iterative since it must deal with change and evolution of many dimensions of Web-based learning. Change and evolution are intrinsic to Web-based learning as they affect all dimensions of this technology. Changes come from new pedagogical improvements, as well as from information technology, subject matter, course content, and educational organizations and institutions.

Fifth, the methodology must explicitly incorporate learning theories and associated pedagogical principles, since the pedagogical dimension is one of the major forces behind developing and evaluating Web-based learning. Thus, the development process must be extended to include the evaluation of learning. Only such an approach ensures that pedagogical issues can be systematically integrated into Web-based learning.

Future work consists of evaluating the methodology by adopting it as the methodology on projects at the university level as a form for action research (Baskerville, 1999). Action research is an important approach for understanding how the development methodology works in practice. The improvement of the methodology will then take place through continuous cycles of practical use and experiments, evaluation, and redesigns. Action research is needed to explore the potential-

ties of the development methodology in order to improve its applicability and integration in educational settings. Finally, the approach needs to be applied in varied educational settings and courses to confirm and support the evaluation of the methodology through a framework of requirements.

References

- Balasubramaniam, R., Pries-Heje, J., & Baskerville, R. (2003). Internet software engineering: A different class of processes. *Annals of Software Engineering*, 14, 169-195.
- Barab, S., & Squire, K. (2004). Design-based research: Putting a stake in the ground. *The Journal of Learning Sciences*, 13 (1), 1-14.
- Baskerville, R.L. (1999). Investigating information systems with action research. *Communications of the Association for Information Systems*, 2, 19. Retrieved September 14, 2005, from <http://cais.isworld.org/articles/2-19/>
- Benigno, V. & Trentin, G. (2000). The evaluation of online course. *Journal of Computer Assisted Learning*, 16, 259-270.
- Blythe, S. (2001). Designing online courses: User-centered practices. *Computers and Composition*, 18, 329-346.
- Buendia-Garcia, F. & Diaz-Perez, P. (2003). A framework for the management of digital content conjugating instructional and technical issues. *Educational Technology & Society*, 6(4), 48-59.
- Conallen, J. (1999). *Modeling web application with UML*. New York: Addison-Wesley.
- Dyson, M. C. & Campello, S. B. (2003). Evaluating virtual learning environments: What are we measuring? *Electronic Journal of E-Learning*, 1 (1), 11-20.
- Duffy, T.M., Lowyck, J., & Jonassen, D.H. (1993). *Designing environments for constructive learning*. Berlin: Springer.
- Elissavet, G. & Economides, A.A. (2003). An evaluation instrument for hypermedia courseware. *Educational Technology & Society*, 6(2), 31-44.
- Fowler, M. (2004). *UML distilled third edition: A brief guide to the standard object modeling language*. New York: Addison-Wesley.
- Frantiska, J.J. (2003). Going with the flow: Applying the waterfall model of system development to educational web site creation. *Proceedings of E-LEARN 2003*, Phoenix, Nov. 7-11, 958-965.
- Graf, F. (2002). Providing security for elearning. *Computers Graphics*, 26, 355-365.
- Govindasamy, T. (2002). Successful implementation of e-learning: Pedagogical considerations. *The Internet and Higher Education*, 4, 287-299.
- Gros, B. (2002). Knowledge construction and technology. *Journal of Educational Multimedia and Hypermedia*, 11(4), 323-343.
- Hamid, A. A. (2002). E-learning: Is it the "E" or the learning that matters? *The Internet and Higher Education*, 4, 311-316.
- Hinostrroza, E., Rehbein, L. E., Mellar, H. & Preston, C. (2000). Developing educational software: A professional tool perspective. *Education and Information Technologies*, 5(2), 103-117
- Hasarim, L. (2000). Shift happens: Online learning as a new paradigm in learning. *The Internet and Higher Education*, 3, 41-61.
- Horton, S. & Lynch, P.J. (1999). *Web style guide: Basic design principles for creating web sites*. London: Yale University Press.
- Jonassen, D., Mayes, T. & McAleese, R. (1993). A manifesto for a constructivist approach to uses of technology in higher education. In T. M. Duffy, J. Lowyck, & D. H. Jonassen (Eds.), *Designing environments for constructive learning* (pp. 231-245). Berlin: Springer.
- Karagiorgi, Y. & Symeou, L. (2005). Translating constructivism into instructional design: Potential and limitations. *Educational Technology & Society*, 8(1), 17-27.

- Hadjerrouit, S. (2005). Developing web-based learning systems. A skill-based approach. *Proceedings of E-LEARN 2005*, Vancouver, Canada, October 24 – 28, 1317-1325.
- Kirschner, P.A. & Paas, F. (2001). Web-enhanced higher education: A Tower of Babel. *Computers in Human Behavior*, 17, 237-353.
- Krauss, F. & Ally, M. (2005). A study of the design and evaluation of a learning object and implications for content development. *Interdisciplinary Journal of Knowledge and Learning Objects*, 1, 1-22.
- Kunz, P. (2004). The next generation of learning management systems (LMS). Requirements from a constructivist perspective. *Proceedings of ED-MEDIA 2004*, Lugano, Switzerland. June 21-26, 300-307.
- IBM. (2000). Design Guidelines. Retrieved September 8, 2005, from <http://www-3.ibm.com/ibm/easy/eou.ext.nsf/Publish/572>
- Lazarinis, F. (2004). A template based for automatic construction of online courseware for secondary educational institutes. *Journal of Educational Technology & Society*, 7(3), 112-123.
- Lam P. & McNaught, C. (2004). Evaluating educational websites: A system for multiple websites at multiple universities. *Proceedings of ED-MEDIA 2004*, Lugano, Switzerland, June 21-26, 1066-1073.
- Lewandowski, J. (2003). Going beyond the basics: Addressing privacy, ethics, and security within a technology integration course. *Proceedings of E-LEARN 2003*. Phoenix, Nov. 7-11, 2003, 1931-1937.
- Lin, B. & Hsieh, C. (2001). Web-based teaching and learner control: A research review. *Computers & Education*, 37 (3-4), 377-386.
- Li, L. (2003). Pervasive learning: E-learning in future. *Proceedings of E-LEARN 2003*, Phoenix, Arizona (USA), Nov. 7-11, 2003, 1933-1936.
- Low, A.L.Y., Low, K.L.T. & Koo, V. C. L. (2003). Multimedia learning systems: A future interactive educational tool. *The Internet and Higher Education*, 6(1), 25-40.
- Marjanovic, O. (2005). Towards a web-based handbook of generic, process-oriented learning designs. *Educational Technology & Society*, 8(2), 66-82.
- Mayes, J.T. & Fowler, C.J. (1999). Learning technology and usability: A framework for understanding courseware. *Interacting with Computers*, 11(5), 485-497.
- McCormack, C. & Jones, D. (1998). *Building a web-based education system*. New Jersey: Wiley.
- Maciaszek, L.A. (2001). *Requirements analysis and system design: Developing information systems with UML*. New York: Addison-Wesley.
- Melis, E., Weber, M. & Andres, E. (2003). Lessons for (pedagogic) usability of elearning systems. *Proceedings of E-LEARN 2003*, Phoenix, Nov. 7-11, 2003, 281-284.
- Murugesan, S. & Ginige, A. (2001). The essence of web engineering—Managing the diversity and complexity of web application development. *IEEE Multimedia*, 8(2), 22-25.
- Montilva, J. A., Sandia, B., & Barrios, J. (2002). Developing instructional web sites – A software engineering approach. *Education and Information Technologies*, 7(3), 201-224.
- Motschnig-Pitrik, R. & Mallich, K. (2004). Effects of person-centred attitudes on professional and social competence in a blended learning paradigm. *Educational Technology & Society*, 7(4), 176-192.
- Nilsen, J. (1993). *Usability engineering*. Boston: Academic Press.
- Nilsen, J. (2000). *Designing web usability: The practice of simplicity*. New York: New Riders.
- Nocols, M. (2003). A Theory of learning. *Educational Technology & Society*, 6(2), 1-10.
- Nokelainen, P. (2004). Conceptual definition of the technical and pedagogical usability criteria for digital learning material. *Proceedings of ED-MEDIA 2004*, Lugano, Switzerland, June 21-26, 4249-4254.
- Nulden, U. (2001). E-education: Research and practice. *Journal of Computer Assisted Learning*, 17, 363-375.
- Onay, Z. (1999). Building an Internet-based learning environment in higher education: Learner informing systems and the life cycle approach. *Informing Science*, 2(2), 45-52. Available at <http://inform.nu/Articles/Vol2/v2n2p45-52.pdf>

- Powell, T. A. (1998). *Web site engineering: Beyond web page design*. London: Prentice Hall.
- Piaget, J. (1971). *Genetic epistemology*. New York: Norton.
- Pahl, C. (2003). Managing evolution and change in web-based teaching and learning environments. *Computers & Education*, 40(2), 99-114.
- Phye G. D. (1997). (Ed). *Handbook of academic learning: Construction of knowledge*. New York: Academic Press.
- Pressman, P. (2000). *Software engineering: A practitioner's approach* (5th ed.). London: McGraw-Hill.
- Retalis, S. & Papasalouros, A. (2005). Designing and generating educational hypermedia applications. *Educational Technology & Society*, 8(3), 26-35.
- Rokou, F. P., Rokou, E. & Rokos, Y. (2004). Modeling web-based educational systems: Process design teaching model. *Educational Technology & Society*, 7(1), 42-50.
- Standing, C. (2002). Methodologies for developing web applications. *Information and Software Technology*, 4, 151-159.
- Skinner, B. F. (1976). *About behaviorism*. New York: Vintage.
- Steffe, L. P. & Gale J. (1995) (Ed.). *Constructivism in education*. New Jersey: Lawrence Erlbaum.
- Sommerville, I. (2001). *Software engineering* (6th ed.). London: Addison Wesley.
- Stevens, P. & Pooley, R. (2000). *Using UML: Software development with objects and components*. London: Addison-Wesley.
- Shiratudin, N. & Shahizan, H.A. (2003). Usability study for promoting econtent in higher education. *Educational Technology & Society*, 6(4), 112-114.
- Storey, M.A., Phillips, B., Maczewski, M. & Wang, M. (2002). Evaluating the usability of web-based learning tools. *Educational Technology & Society*, 5(3), 91-100.
- Strodher, J. B. (2003). Cross-cultural issues for Asian e-learners: An analysis based on Hofstede's cultural dimensions. *Proceedings of E-LEARN 2003*, Phoenix, Arizona (USA), Nov. 7-11, 1978-1983.
- Tam, M. (2000). Constructivism, instructional design, and technology: implications for transforming distance learning. *Educational Technology & Society*, 3(2), 50-60.
- Vygotsky, L.S. (1986). *Thought and language*. Translated by A. Kozulin. Cambridge, Mass: MIT Press (Original English translation published 1962)
- Watson, D.M. (2001). Pedagogy before technology. Re-thinking the relationship between ICT and teaching. *Education and Information Technologies*, 6(4), 251-266.
- Westera, W. (2005). Beyond functionality and technocracy: Creating human involvement with educational technology. *Educational Technology & Society*, 8(1), 28-37.
- Wilson, B. G. (Ed.) (1998). *Constructivist learning environments: Case studies in instructional design*. New Jersey: Educational Technologies.

Biography



Said Hadjerrouit received the MS and PhD degrees in Software Engineering and Artificial Intelligence from the Technical University of Berlin (Germany), in 1985 and 1992, respectively. He joined Agder University College, Kristiansand (Norway) in 1991. He is currently an Associate Professor of Computer Science at the Faculty of Mathematics. He has been in the teaching profession for 23 years. He has extensive experience teaching object-oriented programming, Web design, database development, and software engineering. His research interests include computer science and software engineering education, didactics of informatics, E-Learning, Web engineering, and object-oriented software development with the Unified Modeling Language (UML). Hadjerrouit has published over 30 papers in international journals and conference proceedings.