EVALUATION AND EFFICIENCY OF E-LEARNING SYSTEMS

BURCU KÖR

BOĞAZİÇİ UNIVERSITY

2008

EVALUATION AND EFFICIENCY OF E-LEARNING SYSTEMS

Thesis submitted to the

Institute for Graduate Studies in the Social Sciences

in partial fulfillment of the requirements for the degree of

Master of Arts

in

Management Information Systems

by

BURCU KÖR

Boğaziçi University

2008

Evaluation and Efficiency of E-learning Systems

The thesis of Burcu Kör

has been approved by

Assoc. Prof. Zuhal Tanrıkulu ________________________________
(Thesis advisor)

Prof. Dr. Meltem Özturan ________________________________

Assoc. Prof. Sevinç Gülseçen ________________________________

August 2008

Thesis Abstract

Burcu Kör, “Evaluation and Efficiency of E-learning Systems”

E- learning system is a complex issue not only because of its success and effectiveness but also because of its evaluation. Reliable ways to measure the success and effectiveness of the e-learning system are required for e-learning applications to be used efficiently. The purpose of this thesis is to provide a prototype to evaluate e-learning systems’ success and effectiveness by addressing the success criteria. E-learning systems and testing tools used in this thesis were especially selected from among non-commercial and open source software.

In order for a successful e-learning system assessment, different dimensions of the system such as standards, quality attributes and several other criteria should be taken into consideration. In this thesis study, some automated software testing tools are examined and then selected to execute to test the e-learning systems’ software. Open source Learning Management Systems were selected to evaluate, namely Moodle, Ilias, Dokeos, Docebo, Claroline and Efront. Assessment of e-learning systems success was focused on software testing, the whole e-learning process was not considered. It is assumed that performance, accessibility, security and standard compliance, and functionality comparisons of the systems can be indicators of the whole e-learning system success. Some non-commercial testing tools were used to evaluate e-learning systems according to the previously defined criteria except functionality. In order for the functionality testing of the systems, functionality comparison matrix was developed. Each criterion in the matrix weighted according to the survey results, which was answered by 22 e-learning specialists.

Weighted sum of the standardized scores of tests were used to evaluate systems overall success according to the defined criteria. Weights were calculated according to the third part of survey. LMSs’ scores according to these calculations were figured out in the order of Docebo, Dokeos, Moodle, Claroline, Ilias and Efront, whereas the scores of Dokeos and Docebo were very close to each other.

Tez Özeti

Burcu Kör, “E-öğrenme Sistemlerinin Verimliliği ve Değerlendirilmesi”

E-öğrenme sistemlerinin sadece başarı ve etkinliği değil değerlendirilmesi de karmaşık bir konudur. E-öğrenme uygulamalarının verimli olarak kullanılması için e-öğrenme sistemlerinin başarı ve etkinliğini ölçmek için güvenilir metotlar gereklidir. Bu tezin amacı başarı kriterlerinin belirlenerek e-öğrenme sistemlerinin başarı ve etkinliğinin ölçülmesini sağlayacak bir prototip sağlamaktır. Tezde kullanılan e-öğrenme sitemleri ve test araçları ticari olmayan, açık kaynak kodlu yazılımlar arasından seçilmiştir.

Bir e-öğrenme sistemini başarılı bir şekilde değerlendirmek için sistemin değişik boyutları hesaba katılmalıdır, örneğin standartlar, kalite nitelikleri ve başka kriterler. Bu tez çalışmasında, e-öğrenme sistem yazılımlarını test etmek için bazı otomatik yazılım test araçları incelenmiş ve seçilenler kullanılmıştır. Açık kaynak kodlu Ilias, Dokeos, Docebo, Claroline ve Efront öğrenme yönetim sistemleri değerlendirilmek için seçilmiştir. E-öğrenme sistemleri değerlendirilirken yazılım testine odaklanılmış, tüm e-öğrenme süreci göz önüne alınmamıştır. Performans, işlevsellik, erişilebilirlik, güvenlik ve standart uyumluğu e-öğrenme sistemlerinin başarısı için gösterge olarak varsayılmıştır. İşlevsellik dışında diğer kriterler ticari olmayan test araçları ile değerlendirilmiştir. Sistemlerin işlevsellik testi için, işlevsellik karşılaştırma matrisi geliştirilmiştir. Matristeki her bir kritere 22 e-öğrenme uzmanının cevapladığı anket sonucuna göre ağırlık verilmiştir.

Belirlenen kriterlere göre sistemlerin genel değerlendirmesi, testlerden alınan standart puanların ağırlıklı toplamları kullanılarak yapılmıştır. Kriterlerin ağırlıkları anketten alınan sonuçlar kullanılarak hesaplanmıştır. Hesaplamalar sonucunda öğrenme yönetim sistemlerinin sıralaması Docebo, Dokeos, Moodle, Claroline, Ilias ve Efront şeklinde oluşmuştur. Docebo ve Dokeos’un puanları birbirine çok yakındır.

ACKNOWLEDGEMENTS

The first, I am grateful to my thesis supervisor Assoc. Prof. Zuhal Tanrıkulu for her guidance and patience. I also wish to thank Prof. Dr. Meltem Özturan and Assoc. Prof. Sevinç Gülseçen, my thesis committee members, who devoted their valuable time and energy to this study.

I am also indebted to my parents Nurdane, Fuat and my dear sister Füsun for their support, understanding and tolerance throughout my all life.

I will also never forget the encouragements, helps and supports of all of my friends who are always with me whenever I need them.

CONTENTS

TABLES

FIGURES

CHAPTER I

INTRODUCTION

Information systems are increasingly important for improving business processes and competition throughout the world. Information systems are widely used in both public and private organizations. Moreover, assessing the success of information systems is critical issue for organizations since information systems help to solve organizational problems and respond to a changing environment, and information systems can be expensive (Laudon, K. and Laudon,J., 1991). In order to evaluate the success of information systems, the study was performed on e-learning systems. In this study, e-learning systems were chosen as an example of information systems.

Clarke and Hermes (2001) explained that increasing demand on lifelong and flexible education through with the increasing capacity and availability of communication technologies are stimuli for development of e-learning (as cited in Luther, 2005, p.7). Besides, e-learning is becoming a dominant delivery method in workplace-learning settings at various sectors and varying sizes (Kim, Bonk and Zeng, 2005). Various sectors with different sizes not only work on producing more economical and productive systems, but also concentrate on learning with technology to improve the effectiveness and quality of education. However, measuring the effectiveness and quality of e-learning is a sophisticated issue. There is no one exact way to measure the success and quality of e-learning. In addition, defining the effectiveness or quality of e-learning is a complicated issue on its own. To make e-learning successful, the technology must have several characteristics (Clarke and Hermes, 2001).

The success of e-learning systems cannot be assessed with a single criterion; therefore different dimensions should be taken into account. According to Alotaiby (2005), e-learning success depends on factors like data reusability and interoperability by containing proper functionality and these factors have been addressed in many existing standards such as SCORM. Moreover, the standardization provides accessibility and reusability of learning content from various systems that follow the same standards. Standards' compliancy of e-learning authoring tools and systems is presented with a view of assessing how their functionality meets the requirements for robust development of e-learning content that complies with the existing e-learning standards and specifications (Ganchev, O'Dromal and Andreev, 2007). Thus, standardization efforts increase the life span of the developed content by providing the success or quality criteria of e-learning systems.

For e-learning applications to be used efficiently, reliable ways are needed to measure the success, quality and/or effectiveness of the e-learning system. Learning Management Systems (LMSs) were chosen to assess e-learning systems' success and efficiency by taking into consideration all aspects of organizational learning and benefits for all users. Open source and non-commercial LMSs were preferred in the study since these software are free to run, to study and to modify. Moreover, Wheeler (2005) claimed that information that would not be available in a proprietary setting could be used when evaluating Open Source software to give a better picture of the software and the project that brings it forth and gives a better idea of the potential continuity of the project (as cited in Karin van den Berg, 2005, p.3).

Some commercial and non-commercial testing tools can be used to evaluate these various dimensions. However, while evaluating these dimensions, software or technical properties of e-learning systems are taken into account by ignoring users' perspective, social interaction, collaboration, teaching, learning and support of students' interactivity.   Software must be tested in order to achieve quality that meets expectations, and software testing is a necessity to help attain any desired level of software quality (Bell, 2006). According to IEEE/ANSI standard, testing is the process of operating a system or component under specific conditions, observing or recording the results, and making an evaluation of some aspects of the system or component. Furthermore, Nagappan (2005) defined software testing as verification and validation of software practice and as a software quality assurance practice. Thus, the purpose of this study is to provide a prototype to evaluate e-learning systems success and/or effectiveness by addressing the success criteria and measuring these criteria with some tools. In this study, assessment of e-learning systems success is limited to pre-defined criteria, the whole e-learning process is not considered.

CHAPTER 2

LITERATURE REVIEW

Information Systems

Information system (IS) can be defined as a set of interrelated components working together to collect, retrieve, process, store, and disseminate information for the purpose of facilitating planning, control, coordination, decision making in businesses and other organizations through including information on significant people, places, and things in a business organization's surrounding environment and within the business itself. IS doesn't consist of just software or a computer or other technical artifacts. ISs can also be defined as a system in operation. ISs essentially transform information into a form usable to coordinate the flow of work in a firm, help employees or managers make decisions, and solve other kinds of problems (Laudon, K. and Laudon,J., 1991).

IS is a particular type of work system whose internal functions are limited to processing information by performing six types of operations: capturing, transmitting, storing, retrieving, manipulating, and displaying information. IS not only produces information but also supports or automates the work performed by other work systems (Seddon, Staples, Patnayakuni and Bowtell, 1999).

According to Alter (1996), ISs, which perform work related to information as subsystems of an organization, is a combination of work practices, information, people, and information technologies organized to accomplish goals in an organization. Work practices are defined as the methods used by people and technology to perform work. Work practices indicate both procedure-oriented and tool-oriented systems in information systems. Furthermore, procedure-oriented systems are used repetitive tasks and tool-oriented systems help people communicate and make decisions. At ISs, the centrality of work practices explains a great deal about how they operate and the sources of their successes or failures. Additionally, information is a collection of related data, which is appropriate for a particular use. People are the other components of ISs when the systems aren't totally automated since people enter, process, or use data. Work practices affect the people and that the characteristics of the people in the systems determine what work practices are feasible. People have a crucial impact on ISs at developing and implementing phases. Furthermore, information technology is defined as combination of hardware and software that performs processing tasks like capturing, transmitting, storing, retrieving, manipulating, or displaying data. Information technology is useful only as part of information systems that contains work practices, people, and information so understanding information technology is not understanding of total information systems concept (Alter, 1996).

According to Laudon and Laudon (1991, p. 40), "the most powerful explanation of why the businesses build systems is to solve organizational problems and to respond to a changing environment. Businesses build systems to respond to competitors, customers, and vendors in a dynamic and fluid environment."

In today's increasingly dynamic and competitive environment, ISs have been known to improve an organization's competitive advantage and value of the organizations by increasing the firm's bargaining power and comparative efficiency (as cited in Bakos and Treacy, 1986). ISs have an important impact on business development and survivability of business. Because of this importance, firms have made large investments in developing and implementing IS. However, studies on IS development have reported that there are a large number of reported failures and IS development was not completed on time or on budget (Gibbs, 1990; The Standish Group, 1999). Markus (2000) claimed that developing a successful IS becomes more important when failure in one system has an important effect on other information systems, people, and organizational processes in an integrated environment (as cited in Kwun, 2004, p.1).

Information Systems' quality

Increasing competition throughout the world has forced most businesses to look at the quality of their products and services and quality is a crucial competitive issue that adds value to the business. Harrington (1991) claimed that quality is doing the job every time.

Quality is not only conformance to the technology measured as deviations from specifications or as defect rates but also customer satisfaction and that quality can be measured only in terms of the customers' perception. Quality is the degree to which information has content, form, and time characteristics that gives it value to specific end users. Even with these definitions about quality, many companies are taking total quality management into consideration as part of their competitive strategy, which is based on three general principles: customer focus, process improvement, and total involvement by identification, analysis, and improvement of the processes of creating value for the customer (Alter, 1996).

Information systems quality is important aspects of the process of realizing a new product, service or project outcome. Therefore, quality assessments are an essential element of the project life cycle, including the product, project and project management processes. Organizations are still struggling to improve the quality of information systems despite numerous research efforts and experience. According to DeLone and McLean (D&M) Model of IS Success, IS' quality affects both the use of the information system and how satisfied the intended users are with this use. Moreover, the aim of the information systems quality is evaluation of components such as system quality, data quality, information quality as well as model quality and method quality. System quality uses features of the systems themselves to assess quality (Almutairi, 2001).

ISs development and use do not differ from business processes of an organization. The quality of product is related with the inputs of the process as well as on the process itself. Salmela argued that, ISs' quality depends on business value of information systems to be dependent on IS work quality, IS user quality and business integration quality, and the organizational costs and benefits that are affected by the IS work quality and IS user quality. IS work quality comprises those aspects of IS processes and products, which aim at ensuring efficient delivery and maintenance of IS products according to user requests. IS work quality criteria contains attributes such as maintainability, flexibility (in modification), readability of code, availability of documentation, reusability, and testability. IS user quality views the quality of IS processes and products from the perspective of users, and IS user quality criteria contains attributes such as ease of use, ease of learning, flexibility in use, and security (Andersson and Eriksson, 1996).

ISs can be regarded as products of a development process, and their quality can be assessed against the high level quality attributes of International Organization for Standardization-9126 (IS0) standards for product quality: functionality, reliability, efficiency, usability, maintainability, portability and reusability. As some of the elements of IS0 are thus expressed in more objective terms, the content of information systems quality becomes more tangible than otherwise would be possible.

Helletd proposed that, some apparent problems of assessing software and information can be in the organization; such as there are well-established objective measurements for hardware quality, which can be fault tolerance, backup procedures, and spare capacity. Furthermore, the quality of the data within databases can be assessed by objective measures, which can be error ratios, number of data validation errors observed and number of backup files maintained. Additions to these, process assessment techniques are used to assure the effectiveness of the procedures, as well as audit and inspection techniques (Hellens, 1997).

Quantitative measurements can be used for evaluation of various IS components, such as the number of data validation errors observed and corrected. However, the users` overseeing the entire system as a whole, and using their common sense to assure that all the components and the quality control provision are harmoniously inter-linked. Besides, from the organizational viewpoint it is clear that both quantitative and qualitative assessments are necessary for the evaluation of the information and software systems. Measurement, by definition is quantitative, providing means to compare one item or situation with another. Assessment, which may appear to be more objective, such as cost of acquiring software, provides no firmer foundation for quality assessment. Considering the assessment of both information systems quality and software quality from a more technical viewpoint, the difference between quantitative and qualitative properties is not very dramatic. Additionally, the engineering approach tends to reduce the question to that of the choice of the scale to be used for comparison. Nominal or ordinal scales can be used for qualitative evaluations whereas with quantitative metrics, interval or ratio scales can be employed. The choice between quantitative or qualitative evaluations relates to how much is known about the attribute concerned. Qualitative analyses should be used to assess the more ill-structured quality factors whose specific details are uncertain and cannot be expressed in exact figures. Overall, information systems quality cannot be improved in a solution, although the initiative can come from inside the IS function. The initiative for the improvement can come from either the IS department or from elsewhere in the organization, but it is not likely to be achieved except through collaboration among IS staff and users (Hellens, 1997).

Additionally, according to European Quality Observatory, quality approaches in the context of e-learning are any policies, procedures, rules, criteria, tools, checklists or any other verification instruments and mechanisms that have the purpose of ensuring and enhancing the quality of any e-learning offering (http://www.eqo.info/files/EQO-Model-1.2a.pdf).

Information Systems' success

Information systems success is a multi-dimensional issue and essentially is about the delivery of business value to the organization. Besides that there are many measures to evaluate the success of ISs (Sabherwal, Jeyaraj and Chowa, 2006). For Almutairi (2001), most of the researchers have focused on both identifying factors that influence the success of the IS, and investigating how to measure IS's success. Some studies have focused on internal users and impacts of information systems without taking into account external users and their impacts on these systems, although DeLone and McLean (1992) focused on the dependent variable that is IS's success. Delone and McLean (1992, p.87) initially found through extensive literature review that IS success could be explained by six dimensions: information quality, system quality, information use, user satisfaction, individual and organizational impact (Figure 1).

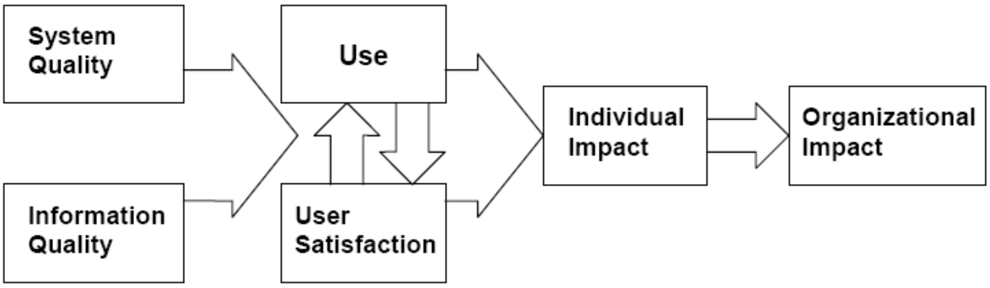


Figure 1. Delone and MecLean’s model of IS success

Based on DeLone and McLean’s (1992) comprehensive survey of prior literature on information systems success, system quality comprises creating and testing a productivity model for computer systems, including such performance measures as resource utilization and investment utilization. Systems quality criteria also include the reliability of the computer system, on-line response time, the ease of terminal use. Emery (1971) also suggested measuring system characteristics such as the content of the database, aggregation of details, human factors, response time, and system accuracy (as cited in DeLone and McLean, 1992, p.64). Hamilton and Chervany (1981) proposed data currency, response time, turnaround time, data accuracy, reliability, completeness, system flexibility, and ease of use among others as part of a "formative evaluation" scheme to measure system quality (as cited in DeLone and McLean, 1992, p.64). Moreover, ISs’ researchers also focused on the quality of the information system output, namely, the quality of the information that the system produces, primarily in the form of reports. According to Delone and McLean (1992, p.64) research, Ahituv (1980) incorporated five information characteristics into a multi-attribute utility measure of information value: accuracy, timeliness, relevance, aggregation, and formatting. Gallagher (1974) developed measurement that included measures of relevance, informativeness, usefulness, and importance. Additional information characteristics developed by Swanson (1974) to measure Management Information Systems (MIS) appreciation among user managers included uniqueness, conciseness, clarity, and readability measures. Researchers studying the quality of the information dimension define measurement of accuracy, precision, currency, timeliness, reliability, completeness, conciseness, format, relevance, sufficiency, understandability, freedom from bias, comparability and quantitativeness. Furthermore, the use of IS reports is one of the most frequently reported measures of the success of an IS. Use of information system includes measurement of use or non-use of computer-based decision aids, use of IS to support production, use of numerical vs. nonnumerical information, frequency of requests for specific reports, use of chargeback information, acceptance of report, direct use of IS vs., number of requests for information, frequency of use, use vs. non-use of data sets, motivation to use, number of decision support systems features used, number of minutes, number of sessions, number of functions used, expenditures/charges for computing use, voluntariness of use, number of queries, nature of queries, extent of use, regularity of use, and use in support of cost reduction, management, strategy planning and competitive thrust. Additionally, user satisfaction is the measure of the successful interaction between the information system itself and its users. Researchers, who are studying on user satisfaction, measure the criteria of successful interaction by management of the information system. According to Delone and McLean (1992) research, studies have found that user satisfaction is associated with user attitudes toward computer systems so that user-satisfaction measures may be biased by user computer attitudes (Igerhseim, 1976; Lucas, 1978, p.69). Therefore, studies, which include user satisfaction measurement, should ideally also include measures of user attitudes so that the potentially biasing effects of those attitudes can be controlled in the analysis. According to research, user satisfaction is also included in overall satisfaction like decision support systems, user information satisfaction, both top management and personal management satisfaction, software and hardware satisfaction, enjoyment, satisfaction with the development project, information satisfaction, difference between information needed and information received controller satisfaction and decision making satisfaction. Individual impact is the other dependent variable of DeLone and McLean (1992). Individual impact refers to the effect of information on the behavior of the recipient of the information, and DeLone and McLean (1992) indicated that performance of users of information system and individual impact are closely related. According to research, individual impact includes user confidence, quality of decision analysis, efficient decisions, time to arrive at a decision, time taken to complete a task, time to make pricing decisions, decision quality, forecast accuracy, decision-making efficiency and effectiveness, interpretation accuracy, computer awareness, cost awareness, and so on. The last dependent variable of DeLone and McLean (1992) is organizational impact, which includes measurement criteria like profit performance, pre-tax return on assets, return on net worth, pre-tax profits (% of sales), average 5-year sales growth, number of computer applications, economic performance, marketing achievements, productivity in production, innovations, product and management quality, overall cost-effectiveness of IS, field Organizational effectiveness and so on.

Seddon (1997), who uses theoretical considerations to modify DeLone and McLean's (1992) model, viewed system use as a behavior that reflects an expectation of net benefits upon using the system. Overall, Seddon's model included three types of constructs: measures of information and system quality, system use as behavior and general measures of net benefits from system use. Moreover, Rai et al. (2002) further built on DeLone and McLean (1992) and Seddon (1997). They viewed perceived usefulness as being related to individual impacts because it is based on several of the constructs DeLone and McLean (1992) had linked to individual impacts, such as improved individual productivity. Rai et al. (2002) focused on five constructs, which are system quality, information quality, perceived usefulness, user satisfaction, and system use, and they represented system quality and system use in terms of ease of use and system dependence, respectively (Sabherwal, Jeyaraj and Chowa, 2006).

Alter (1996) claimed that a major issue in evaluating information system effectiveness or success is the fact that information systems typically exist to support other work systems, which may be other information systems. So performance of the information system is important evaluation criteria for the system, but the systems may include a work system and an information system. According to Alter (1996), the work system and information system may overlap to some degree to indicate that some aspects of the work system are not included in the information system and some aspects of the information system are not included in the work system. The partial overlap between work system and the information system cause a number of difficulties for the observer trying to assess the effectiveness of the information system. The performance of the systems can be divided into internal and external performance. Internal performance is how well the system operates business process measures such as productivity, cycle time, consistency (of the work that is done), and rate of output whereas external performance is how well the system achieves its purpose, which include cost, quality, reliability, responsiveness, and conformance to standards as viewed by the customer. Apart from these measures, performance also applies to the other elements such as work system, the technology, information, and human participants. Measures of performance for the technology are related to various aspects of its functional capabilities (capacity, speed, etc.), ease of use, compatibility, and maintainability. Measures of performance for information are related to information quality (accuracy, timeliness, etc.), accessibility, presentation, and security. Measures of performance focusing on participants include measures of the impact of the work system on them (related to stress, variety, and social connection, personal growth).

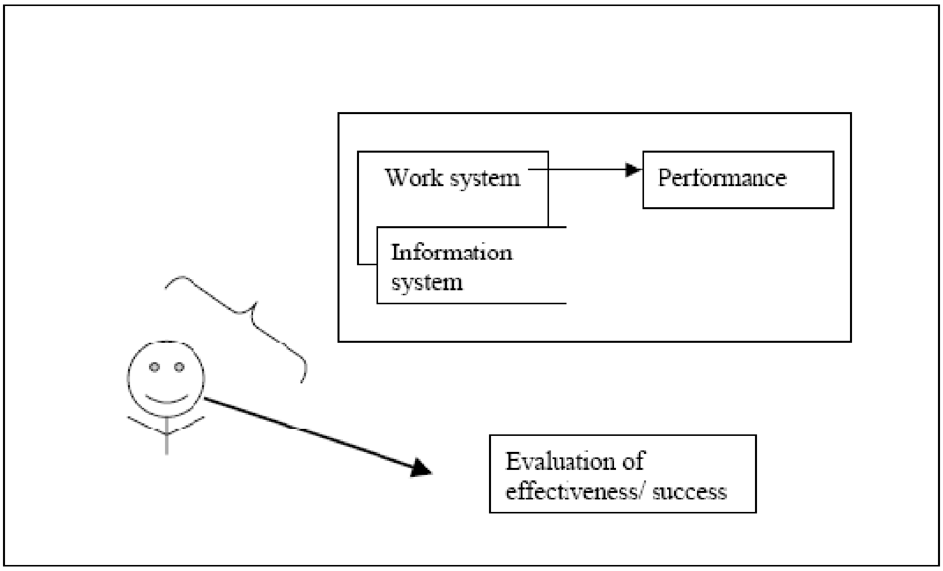


Figure 2. Evaluating an information system that supports a work system

Overall, information system (IS) success and its determinants have long been considered critical to the field of information systems. The research provides insights into the success of information systems and the researchers determined or adopted reliable variable by empirically integrating prior research in the area.

Types of Information System

Information Systems perform important operational and managerial support roles in businesses and other organizations. Generally ISs are classified into six categories: office information systems, transaction processing systems, management information systems, decision support systems, executive information systems and expert systems (Alter, 1996).

The main kinds of information systems in business are described briefly below:

Transaction Processing Systems

A transaction processing system (TPS) collects and stores data about transaction and sometimes controls decisions that are made as part of a transaction. A transaction is a business event that generates or modifies data stored in information systems. TPS is an information system that captures and processes data generated during an organization's day-to-day transactions. A transaction is a business activity such as a deposit, payment, order or reservation. Example of TPS in different functional areas can be point of sale system for sales transactions, tracking of movement of work in process and processing of credit card payments (Alter, 1996).

Transaction processing systems were among the first computerized systems developed to process business data – a function originally called data processing. Usually, the TPS computerized an existing manual system to allow for faster processing, reduced clerical costs and improved customer service. The first transaction processing systems usually used batch processing. With batch processing, the data for individual transactions are gathered and stored whereas the transactions aren't entered into the system immediately. Transaction data is collected over a period of time and all transactions are processed later, as a group. As computers became more powerful, system developers built online transaction processing systems. With online transaction processing (OLTP) the computer processes transactions as they are entered (Shelly, Cashman and Vermaat, 1999).

Management Information Systems

Management information systems (MISs) are the most common form of management support systems. Management information system (MIS) provides information for managing an organization. MIS is an information system that generates accurate, timely and organized information so managers and other users can make decisions, solve problems, supervise activities, and track progress. In addition, MIS converts data from transaction processing systems into information for managing an organization and monitoring performance. MIS focuses on generating information that management and other users need to perform their jobs. MIS often is integrated with transaction processing systems. Computerized MIS typically summarizes data from TPS to allow managers to monitor and direct the organization and to allow employees to receive meaningful but easy-to-assess feedback about their work. Example of MIS in different functional areas can be weekly sales report by product and region, weekly production report by product and operation, and receivables report showing invoices and payments (Alter, 1996).

An MIS generates three basic types of information: detailed, summary and exception. Detailed information typically confirms transaction processing activities. A Detailed Order Report is an example of a detail report. Summary information consolidates data into a format that an individual can review quickly and easily. To help synopsize information, a summary report typically contains totals, tables, or graphs. An Inventory Summary Report is an example of a summary report (Shelly, Cashman and Vermaat, 1999).

Decision Support Systems

Transaction processing and management information systems provide information on a regular basis. Decision support system (DSS) is an interactive system that helps people make decisions, use judgment, and work in areas where no one knows exactly how the task should be done in all cases. DSS gives users direction in using the system and decision- making, and may provide methods and formats for portions of a decision process. DSS may support both repetitive and nonrepetitive decision making. DSS supports repetitive decision- making by defining procedures and formats whereas still permit user to decide how and when to use the system's capabilities. At the nonrepetitive decision making, DSS provides data, models, and interface methods that can be used however the user wants (Alter, 1996).

Frequently, however, users need information not provided in these reports to help them make decisions. A sales manager, for example, might need to determine how high to set yearly sales quotas based on increased sales and decreased product costs. Decision Support Systems help provide information to support such decisions.

A decision support system uses data from internal and/or external sources. Internal sources of data might include sales, manufacturing, inventory, or financial data from an organization's database. Data from external sources could include interest rates, population trends, and costs of new housing construction or raw material pricing. Users of a DSS, often managers, can manipulate the data used in the DSS to help with decisions (Shelly, Cashman and Vermaat, 1999).

Example of DSS in different functional areas can be system helping insurance salespeople test alternatives, system displaying current aspect to machine operator, and system displaying portfolio breakdown to stock broker.

Executive Information System

Executive information system (EIS) is highly interactive system providing managers and executives with flexible access to information for monitoring operating results and general business conditions. EIS provides information in a readily accessible, interactive format without forcing executives to convert data into information. Information in an EIS is presented in charts and tables that show trends, ratios, and other managerial statistics. Example of EIS in different functional areas can be flexible access to sales data by region and product, flexible access to production data by product and operation, and flexible access to corporate financial plan by line item (Alter, 1996).

Expert Systems

An expert system supports the intellectual work of professionals engaged in design, diagnosis, or evaluation of complex situations that require expert knowledge in a well-defined area. An expert system makes the knowledge of experts available to others, and assists problem solving in areas where expert knowledge is needed (Alter, 1996).

Expert systems are composed of two main components: a knowledge base and inference rules. The former one is the combined subject knowledge and experiences of the human experts. The latter one is a set of logical judgments applied to the knowledge base each time a user describes a situation to the expert system. In addition, expert systems can help decision-making at any level in an organization, nonmanagement employees are the primary users who utilize them to help with job-related decisions (Shelly, Cashman and Vermaat, 1999).

Example of expert systems in different functional areas can be system to develop sales strategy against competition, system for pricing competitive bids, system to diagnose machine failures, system to decide which lot to do next, system to support credit-granting decisions, and system to identify changes in trends.

Office Information Systems

Office information systems facilitate everyday communication and information processing tasks in office and business organizations by trying to improve the productivity of employees who need to process data and information. Office information systems help people to process documents and messages, and provide tools that make general office work more efficient and effective. These systems include a wide range of tools such as word processors, spreadsheets, and telephone systems, with a few exceptions, that are used in unstructured situations and are oriented primarily toward data rather than models. Office information systems example can be spreadsheets, electronic mail, electronic calendar, word processors, phone systems, conferencing systems, desktop publishing systems and so on (Alter, 1996)

Licensing Software

A software license is a legal instrument governing the usage or redistribution of copyright protected software (http://en.wikipedia.org/wiki/Software_license). Bretthauer (2002) stated that license terms allow anyone to revise source code and there is usually one person (or a very small group of volunteers) who maintains control of the software and incorporates patches, bug fixes, and added features contributed by others as new releases. Madison (2005) suggested that software licensing and licensing of digital information in general create a rule of information governance for the Internet and beyond. Licenses define the circumstances under which those who work with copyrighted material can do so without hesitation. For all intents and purposes, according to software licenses themselves, copies of computer programs are never sold outright at the software licensing level. Moreover, license for a given program governs not only the relationship between the copyright owner and a particular licensee but also the relationship between the owner and all users of that program. Furthermore, to the extent that all computer programs are subject to licenses and to the extent that those licenses are effectively identical in relevant respects, the world of software is effectively governed by the very concept of the license. If there is no ability to choose an "unlicensed" version of the copyrighted work, the licensing norm displaces the Copyright Act as the relevant law (Madison, 2005).

Open Source vs Commercial

Accourding to Fuggetta, there is a debate about the definition of open source. There are two different interpretations that are currently used: ‘‘free software’’ and ‘‘open source’’. The term open source can be defined as weaker forms of distribution of the source code. The term free software originates from the GNU project and can be defined as a matter of liberty, not price. Free software is a matter of the users' freedom to run, copy, distribute, study, change and improve the software (as defined in Free Software Foundation, 2008). Moreover, ‘free’ is also used for software that is available at no cost; however the source code is not available. This type of software is often labeled as ‘freeware’ (Berg, 2005).

According to Henley and Kemp (2008), the Open Source is defined as:

- Free redistribution: software to be available for redistribution without payment.
- Source code: the software to be distributed with the source code or well-publicized access to it.

- Derived works: license to allow modification of the software and distribution of resulting derived works.
- Integrity of the author’s source code: distribution of ‘‘patch files’’ used to recreate the derived work (rather than full source code) to be permitted.
- No discrimination against persons or groups
- No discrimination against fields of endeavor; for example, limiting use to non-commercial purposes is not permitted.
- Distribution of license: must be no need to execute extra licenses for redistributed software.
- License must not be specific to a product: license rights not to depend on the software being distributed with other specified software.
- License must not restrict other software: the license must not place restrictions on software distributed together with the licensed software.
- License must be technology-neutral.

Open source software (OSS) is copyrighted and distributed with license terms designed to ensure that the source code will always be available (Bretthauer, 2002). Moreover, as Henley and Kemp argued, OSS is software provided under license granting certain freedoms to a licensee and should properly be seen as a range of associated licensing techniques. Open source software licenses are ranging from the intrusive, ‘copyleft’ General Public License (GPL) to short licenses containing virtually no express terms. Furthermore, open source software has two distinct features. First, open source software comes equipped with licenses that provide existing and future users the right to use, inspect, modify, and distribute modified and unmodified software to others. Second, while software can be classified as ‘‘open source’’ independently of how it was developed, years of development have given rise to a new practice of innovation associated with open source software (Krogh and Spaeth, 2007).

Krogh and Spaeth (2007) claimed that the open source software phenomenon has had a ubiquitous impact on society and the economy. For instance, open source software has an impact on a massive social movement in which contributors, developers, governments, and firms collaborate to create a public good that shapes society. Moreover, open source software has altered global competition in the computer software and hardware industries inasmuch as firms that develop and sell proprietary software products have started to adopt open source software solutions in their own product portfolio. Furthermore, in many countries the government has adopted explicit policies towards open source software because of the impact on reduction of procurement cost, better bargaining positions, the need to support local software and service firms, the adaptability of the software to the government's needs, transparency of the software, and security issues. Additionally, open source software has been advocated by many as a solution for closing the ‘‘digital divide’’ by assisting developing countries in their efforts to apply information technology as it is free and easily accessible online.

Fuggetta (2003) explained that the open source model also means increased security; because code is in the public view so that problems being found and fixed instead of being kept secret until the wrong person discovers them.

Additionally, Fuggetta (2003) focused on commercial software in order to properly frame the notion of open source. Because, Fuggetta (2003) claimed that it is important to clarify the relationship between open source software and commercial software. The definition of commercial software is: (as defined in Free Software Foundation, 2008) “Commercial software is software being developed by a business that aims to make money from the use of the software. ‘Commercial’ and ‘proprietary' are not the same thing! Most commercial software is proprietary, but there is commercial free software, and there is non-commercial non-free software.”

At Table 1, there is a comparison of some software licensing in the aspect of price, distribution, users and usage limitation, available source code, modifiable source code.

Table 1. A Software Licensing Taxonomy

Software Type License Feature	Zero Price	Redistributable	Unlimited Users and Usage	Source Code Available	Source Code Modifiable
Commercial (e.g., typical Microsoft products)					
Trial Software (e.g., time-bombed evaluation products)	X	X			
Royalty-free binaries (e.g., Microsoft’s Internet Explorer and NetMeeting, distributed in binary form only)	X	X	X		
Freeware (e.g., Leap Frog, released in binary form only and in the public domain)	X	X	X		
Open Source (e.g., Linux, Apache)	X	X	X	X	X

(Adapted from Feller and Fitzgerald, 2000, Table 2, p.60)

E-learning

Information Systems (IS) researchers have become more and more interested in studying not only how to make e-learning effective but also investigating the business benefits of e-learning. Clarke and Hermes (2001) explained that increasing demand on lifelong and flexible education through with the increasing capacity and availability of communication technologies are stimuli for development of e-learning (as cited in Luther, 2005, p.7). Additionally, e-learning is becoming a dominant delivery method in workplace-learning settings at various sectors and varying sizes (Kim, Bonk & Zeng, 2005) E-learning is taking place in every different type of industry and business as well as universities (Jun, 2002). Pantazis defined that technology-enabled learning designed to increase workers' knowledge and skills so they can be more productive, find and keep high-quality jobs, advance in their careers, and contribute to the success of their employers, families, and communities. It is also vital to increase employee perceptions of e-learning's usefulness to improve job performance, which is the objective of most information systems (Cao, 2005).

There are many different definitions of e-learning. According to Leiserson (n.d.), e-learning term covers a wide set of applications and processes, such as Web-based learning, computer-based learning, virtual classrooms, and digital collaboration. It includes the delivery of content by electronic technology, audio- and videotape, satellite broadcast, interactive TV, CD-ROM, and more.

Further, as Zemsky and Massy (2004) discussed in their report, three main domains define e-learning's basic market niches. These are:

> First, e-learning is considered as a kind of distance education which serves online courses on the web. Online certificate or graduate programs are examples of e-learning in this domain.
> Secondly, e-learning is seen as electronic learning materials as standardized tests, flash animations, simulations, interactive CDs etc. Such materials are in digital format and differ from the first group products since these are not necessarily remote.
> Finally, the third includes the course management systems by which courses, schedules, assignments, grades, and any kind of learning materials are served. While the authors limited this category of e-learning as course management systems, it would be better to expand it as e-learning management systems since there are other virtual

> platforms serving various kinds of functions other than course management systems. (as cited in Hancı, 2007, p.18)

According to research of Tai (2005), e-learning is defined as education and training delivered by an instructor or self-paced from a curriculum database stored on the enterprise local area network (Berry 2000). It refers to anything delivered, enabled or mediated by electronic technology for the explicit purpose of learning (Hicks 2000). It offers the possibility of learning from information delivered to us electronically (Honey 2001). It is a web based personalized learning experience and provides measurable results.

Advantages

For some organizations, the purpose of adopting an e-learning system is to increase return on investments, reduce travel costs, assist with workforce planning, and deliver content without having to sacrifice quality (Driscoll, 1999). By the help of e-learning systems, business or employees can be more productive, find and keep high-quality jobs, advance in their careers. Additionally, e-learning is more flexible since e-learning provides learners to complete courses anytime at their own pace, without having to travel anywhere by opportunities to meet electronic modes of delivery such as chat rooms, discussion boards, and instant messaging. Kruse (2006) stated that e-learning is becoming a standard method for delivering course content and for lowering training costs (as cited in Womble, 2007). According to Rummler and Brache, (1990) organizations are implementing online learning systems at a rapid rate, and e-learning could make the difference in an organization's overall productivity and ultimately affect company-wide performance (as cited in Womble, 2007).

However, e-learning has also some barriers like other implementations of new ideas in businesses. According to Yaw (2005), the barriers can be in the side of cost - there will probably be need to purchase new hardware and software investment for the e-learning environment- and users- the instructor may have to create e-learning courses, and that will take more thought, time, and investment. Additionally, e-learning has ongoing expenses like helpdesk support, maintenance, connect time and telephone charges could be an issue for instance for an international audience. At the learner side, unfamiliarity of e-learning platform, lack of self-motivation and natural resistance to change are some examples of the barriers (Yaw, 2005).

Synchronous and Asynchronous Learning

Chambers and Lee (2004) claimed that asynchronous learning is "Learning in which the instructor and learners communicate through the Web at 'uninterrupted times.' The learner can choose to post or respond to a message at his or her leisure, for example by email or electronic bulletin board" (as cited in Womble, 2007).

Henderson (2003) indicated that asynchronous collaboration is when the instructor posts assignments and/or lectures and the students answer individually or through a discussion board (as cited in Yaw, 2005, p. 41).

Chambers and Lee (2004) claimed that synchronous learning is

> "An instantaneous form of learning that allows the instructor and learners to interact via the Web in a 'real-time' context. This can be accomplished through on-line discussions, real-time audio, videoconferencing, and application sharing whereby two or more people can work on the same file, such as a shared electronic whiteboard or a spreadsheet, simultaneously" (as cited in Womble, 2007).

E-meetings, e-conferences, and virtual classrooms can be synchronous learning.

According to Henderson (2003), with synchronous collaboration the instructor and all of the students communicate simultaneously and are connected by technology that acts like real-time interactive audio/video (Yaw, 2005).

Standards of E-Learning

ISO defined standards as "documented agreements containing technical specifications or other precise criteria to be used consistently as rules, guidelines, or definitions of characteristics, to ensure that materials, products, processes and services are fit for their purpose" (as cited in Friesen, 2005).

Standards' compliancy of e-learning authoring or assembling tools and systems is presented with a view of assessing how their functionality meets the requirements for robust development of e-learning content that complies with the existing e-learning standards and specifications (Ganchev, et al., 2007). The standardization provides accessibility, interoperability and reusability of learning content from various systems that follow the same standards (Alotaiby, 2005). Interoperability helps systems or products to work with other systems. Interoperability refers mainly to the interactions between learning objects and learning management systems but is moving towards interactions between learning objects as well (Gallagher, 2007). According to Brown and Fallon (2003), learning object (LO) is the conceptual building blocks of e-learning and it is the smallest part of content. LO can be shared by and reused in multiple lessons or courses, and their actual size and scope is adjusted by the authors. LOs are often compared with LEGO blocks inasmuch as LOs are the smallest unit of learning that can be automatically managed and tracked. LOs need to be designed and implemented with affordances for those with different types of disabilities in order to provide accessibility (Gallagher, 1998). Moreover, reusability is the ability of a learning object to achieve multiple outcomes across multiple contexts. All in all, standardization efforts increase the life span of the developed content by providing the success or quality criteria of e-learning systems.

Furthermore, there are three major organizations that contribute to the development of e-learning standards: The IMS Global Consortium, the IEEE LTSC (Institute of Electrical and Electronics Engineers, Inc., Learning Technology Standards Committee), and the ISO/IEC (International Standards Organization/International Electrotechnical Commission). There are other standards organizations (many of them national or regional standards bodies) that may make significant contributions to international e-learning standards development, but which fall outside of the scope of this introduction. These include ANSI (American National Standards Institute), CEN-ISSS (European Committee for Standardization-Information Society Standardization System), DIN (Deutsches Institut für Normung), BSI (British Standards Institute) and the CSA (Canadian Standards Association) (Friesen, 2004).

The standard reference model used as a basis for the comparative analysis is the Sharable Content Object Reference Model (SCORM) of the Advanced Distributed Learning (ADL) Initiative. SCORM is a collection of standards and specifications adapted from multiple sources to provide a comprehensive suite of e-learning capabilities that enable interoperability, accessibility and reusability of Web-based learning content (ADL, 2008). Jones (2002) explained that SCORM is widely acceptable set of standards and specifications for developing, packaging and delivering high quality education and training materials whenever and wherever they are needed. SCORM adapts the object properties listed above into high-level functional requirements, which are known as ADL's "ilities". These "ilities" are as:

> Accessibility: The ability to locate and access instructional components from one remote location and deliver them to many other locations.
> Adaptability: The ability to tailor instruction to individual and organizational needs.
> Affordability: The ability to increase efficiency and productivity by reducing the time and costs involved in delivering instruction.
> Durability: The ability to withstand technology evolution and changes without costly redesign, reconfiguration or recoding.
> Interoperability: The ability to take instructional components developed in one location with one set of tools or platform and uses them in another location with a different set of tools or platform.
> Reusability: The flexibility to incorporate instructional components in multiple applications and contexts (ADL, 2008, p.1-6).

Moreover, the goal of SCORM is to resolve confusion, inconsistencies, and overlaps between current standards and specifications (Alotaiby, 2005). SCORM specifications are a composite of several specifications developed by international standards organizations, including the IEEE, IMS, AICC and ARIADNE (IEEE Computer Society, 2001), (ISO/IEC FCD 24751-1), (ISO/IEC FCD 24751-2), (ISO/IEC 24763). Furthermore, according to Alotaiby (2005), SCORM defines the requirements for success content reusability between different learning systems by supporting learning content composed from relatively small, reusable content objects (SCOs) aggregated together to form units of instructions. SCO is a collection of one or more assets that represents a single launchable resource that can communicate with an LMS using the SCORM RTE, and SCO represents the lowest level of granularity of learning resources that can communicate with an LMS using the SCORM Run-time Environment (ADL, 2008). SCORM is described in terms of the following three components, which are content packaging, runtime communications, and course metadata. Content packaging refers to the packaging of all resources needed to deliver a course into a single zip file. The runtime communications are conducted using runtime commands for communicating student information to and from the LMS, and student metadata for storing information on individual students. Course metadata are data packaged with a course when it is archived in a SCORM^{TM} repository (Jones, 2002). Parmar, Anane and Hendley (2007) also claimed that SCORM has two significant components of e-learning systems, which are Content Aggregation Model (CAM) and Run-time Environment (RTE). CAM describes contents and how to package them to be exchanged from one system to another and how to be discovered and searched by supporting comment model content packaging, metadata, sequencing and navigation between activities. The CAM enables consistent labeling, packaging, storing, exchange and discovery of content objects (ADL, 2008). RTE provides interoperable across multiple systems by modeling the selection, tracking, and the interactions with learning content. The SCORM Sample RTE allows the ADL community developers to evaluate the content in a scaled-down LMS environment without the cost of a commercial LMS (ADL, 2008).

ADL Community use the terms "compliant," "conformant" and "certified" in different contexts. ADL avoids using the term "SCORM compliant" in favor of "SCORM conformant," even though some often use the terms "SCORM conformant" and "SCORM compliant" interchangeably. The term "conformant" should be used when describing a product that follows the SCORM 2004 specifications. A "SCORM Conformant Product" is defined as a product that has passed the SCORM Conformance Test Suite (Self Test), which indicates that the product conforms to the latest version of SCORM Conformance Requirements. A "SCORM Certified Product" has been independently tested by one of the ADL Certification Testing Centers and has passed (ADL, 2008).

Learning Management System

In SCORM, Learning Management System (LMS), which implies a server-based environment, determines what to deliver and when, and tracks progress and performance as the learner moves through the instructional program (ADL, 2008). LMS is used for formal learning in terms of registration and tracking of the students' records (Hancı, 2007). LMSs are applications that automate many of the processes associated with e-learning, which is more than just the administrative part of an e-learning deployment. LMS, which is used to send, trace, report and manage knowledge object, study progress and so on, is the basic framework of e-learning (Wang, Niu, Song and Liu, 2007). Most of LMSs serve online courses, while some of them have other special features according to the context, and LMSs facilitate "anytime, anywhere" access to learning content with administration through a web browser (Yaw, 2005). Moreover, Islas, Pérez, Rodriguez, Paredes, Ávila and Mendoza claimed that LMS is a high-level, strategic solution for planning, delivering, and managing all learning events within an organization, including virtual classrooms and instructor-guided courses by providing assessing and raising competency and performance levels throughout the organization. According to Oakes (2002), LMS is focused on both learner and organization. LMS is related with the logistics of managing learners, learning activities, and the competency mapping of an organization. LMS usually provides keeping track of individual skills of learners and competencies, and helps locate and register learners for relevant learning activities that enable them to acquire new skills or improve existing skills. Besides, LMS helps administrators manage and track the relationship between the users and learning activities, including progress on different activities, and the competencies and skill levels acquired.

LMS consists of seven parts: a tracking service; a delivery service; a learner profile service; a course management service; a content management service; a test/assessment service; and a sequencing service (Chu, Chang, Yeh,C and Yeh,Y., 2004). The specific functions of LMS vary according to the basis of the services. In general, LMS provides basic functions of managing educational resource objects, evaluating learners' ability and suggesting to learners what courses to study according to evaluation information as well as managing students' study progress, sending evaluation information and test reports of learners, and supplying functions of tracing and reporting, including tracing completed online/offline courses (Wang, Niu, Song and Liu,2007).

Morrison (2003) also indicated that LMS comprises a suite of tools that centralizes and automates aspects of the learning process through the functions, which are to register learners, maintain learner profiles, maintain a catalogue of courses, store and deliver self-paced e-learning courses, download e-learning modules and tools, track and record the progress of learners, assess learners, track and record assessment results, and provide reports to management (as cited in Yaw, 2005, p. 31). A highly generalized model showing potential components or services of a LMS is shown in Figure 1 (ADL, 2008, p1-7).

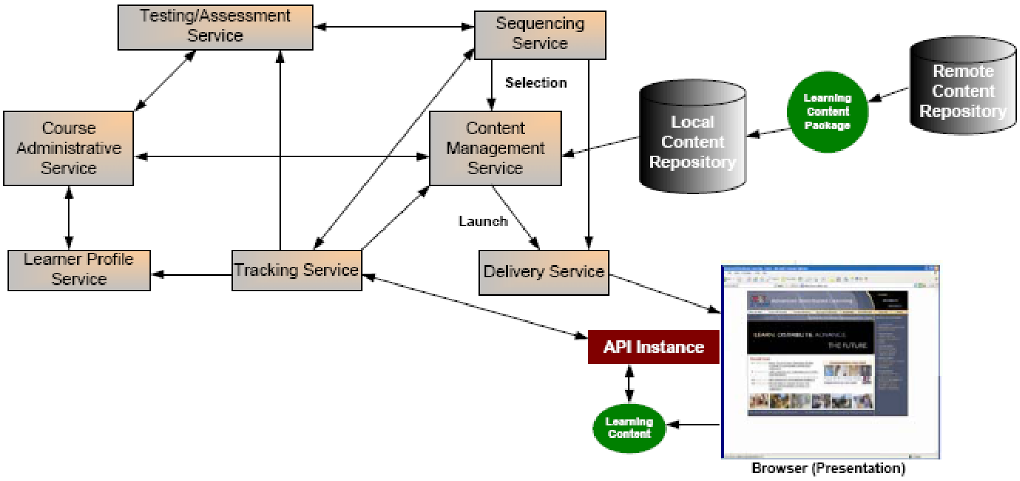


Figure 3. Potential components or services of a LMS

<u>Learning Content Management System</u>

Learning Content Management Systems (LCMSs) are very closely related to LMS with the addition of content authoring. LCMS is a multi-user environment where learning developers can create, store, reuse, manage, and deliver digital learning content from a central object repository (Islas et al., 2007). Ismail (2002) explained that the primary role of an LCMS is to provide a collaborative authoring environment for creating and maintaining learning content, while activating workflow processes to coordinate collaborative authoring of learning content. Moreover, the goal of an LCMS is the creation of instructionally effective learning content that is on time and within the budget by providing the developer with the tools and functionality required to produce and manage effective learning content (Ismail, 2002).

According to Oakes (2002), a good LCMS, which is all about the content, needs the right mix of

- authoring and content-creation capabilities
- support for a wide variety of content formats
- robust model for creating and managing learning objects

- scalable object repository (the database where everything gets stored)
- good search-and-browse capabilities
- ability to personalize delivery of content
- detailed tracking and reporting capabilities.

Additionally, Ismail stated that the LCMS allows organizations to:

- Capture the knowledge within their organization.
- Structure the knowledge into focused, directed learning programs.
- Incorporate third party content.
- Achieve rapid updates, dissemination, management, and utilization of that knowledge throughout the organization.

Oakes (2002) also claimed that LCMSs and LMSs are distinct in focus but complementary. LCMS obtains that information from the LMS in real time to offer successful personalized delivery is another important consideration for the smooth functioning of the combined systems.

According to Chapman and Hall (2004), the e-learning systems of LMS and LCMS are very different systems in terms of serving different masters and addressing unique business challenges. LMS is a useful organization on the whole and has direct impact on all learners, whereas LCMS is limited by the content developers and learners who need personalized content. The other differences of LMS and LCMS are given in Table 2.

Table 2. LMS and LCMS Difference

	LMS	LCMS
Who benefits?	All learners; organization	Content developers; learners who need personalized content
Provides primary management of	Learner performance; learning requirements; learning programs and planning	Learning content
Manages e-learning	Yes	Yes
Manages traditional forms of training, such as instructor-led	Yes	No
Tracks results	Yes	Yes
Supports learner collaboration	Yes	Yes
Includes learner profile management	Yes	No
Allows HR and ERP systems to share learner data	Yes	No
Schedules events	Yes	No
Offers competency mapping/skill gap analysis	Yes	No
Includes registration, prerequisite screening, and cancellation notification	Yes	No
Creates test questions and test administration	Yes	Yes
Supports dynamic pretesting and adaptive learning	No	Yes
Supports content creation	No	Yes
Organizes reusable content	Yes	Yes
Includes workflow tools to manage content creation process	No	Yes
Develops content navigation controls and user interface	No	Yes

Evaluation

Kramer (2007) stated that determining what to measure is important as without evaluation it is not possible to know whether one's objectives are achieved. There are different methods of training evaluation. Table 3 compares the evaluation models (Kramer, 2007).

Table 3. Compares the Evaluation Models

Methodology	Evaluation Elements	Objective
Kirkpatrick	Level 1 – Reaction Level 2 – Learning Level 3 – Behavior Level 4 – Business Results	Provides training data in four areas
Training for Impact	Identify Business Need and Client Form a collaborative relationship Conduct Initial Project Meeting Assessment Conduct Training Collect and Interpret Data Report to Client	Measure results of training in business
Success Case Method	Focus and Plan Study Create an Impact Model Design & Distribute a Survey Interview Prepare Report of Findings	Measure results of training in business to ensure alignment with organizational strategy
Kirkpatrick-Phillips	Level 1 – Reaction Level 2 – Learning Level 3 – Behavior Level 4 – Business Results Level 5 – Return on Investment	Adds a monetary value added verses cost comparison, called Return on Investment (ROI)

At Kirkpatrick methodology each level adds information to create a comprehensive view of the value of the training program. Reaction includes satisfaction of the trainee and assessment of training participants' reaction to the training program. Learning measures are quantifiable indicators of the learning that has taken place during the course of the training. Behavior outcomes address either the extent to which knowledge and skills gained in training are applied on the job or result in exceptional job-related performance. Business results are intended to provide some measure of the impact that training has had on broader organizational goals and objectives (Bates, 2004).

E-learning can also be evaluated by Kirkpatrick methodology. Level 1 evaluations can help monitor learners' emotional acceptance of e-learning. E-learning greatly simplifies Level 2 evaluations by automatical administration of test or assessments. Since a change in behavior occurs outside the e-learning experience, Level 3 evaluation is less associated with the e-learning or to the technologies needed for e-learning. However, some functionality of e-learning systems like e-mail, online forms, and discussion forums can also be used to ask supervisors to appraise employees' progress on specific performance goals, and thereby measure whether distant learners have achieved these goals. Monetary value of a change resulting in part from e-learning can be part of Evaluating Level 4 (Kramer, 2007).

According to Kramer (2007), training for impact model associate training programs with business needs, problems, or opportunities. The training model consists of the following twelve steps:

- Identify Business Need and Client
- Form a Collaborative Relationship
- Conduct Initial Project Meeting
- Conduct Performance Assessment
- Conduct Cause Analysis
- Tabulate and Interpret Data
- Report Results to Client
- Design Evaluation System
- Design Tracking System
- Conduct Training
- Collect, Tabulate, Interpret Data
- Report to Client

Kramer (2007) stated that one of the steps of Success Case Method (SCM) can be focusing and planning the study by creating an impact model, which defines what success, would look like if the initiative were met. The others steps can be design and distribute a survey to search for best and worst cases of program success. Later, the participants should be interviewed, and finally a report of the findings, conclusions, and recommendations should be made. Additionally, Kramer (2007) cited that the Kirkpatrick-Phillips model (also known as The Phillips Five-Level ROI Framework) takes Kirkpatrick's four level framework and adds a fifth level: ROI, which measures the monetary value of the results and costs for the program and is usually presented as a percentage or benefit-cost ratio, in which monetary benefits are compared to determine whether training costs were excessive.

Software Testing

According to the IEEE/ANSI standard, testing is the process of operating a system or component under specified conditions, observing or recording the results, and making an evaluation of some aspects of the system or component (Chen, 2004). Moreover, Misra (2000) defined some common practitioners' views of what is meant by software testing as follows:

- Checking programs against specifications.
- Finding bugs in programs.
- Determining user acceptability.
- Insuring that the system is ready for use.
- Gaining confidence that it works.
- Showing that a program performs correctly.
- Demonstrating that errors are not present.
- Understanding the limits of performance.
- Learning what a system is not able to do.
- Evaluating the capabilities of a system.

- Verifying documentation. Convincing that a job is finished.

Chen (2004) claimed that software testing should be able to find weak points in the software and undiscovered errors, and the testing procedure should be quick, cheap and as efficient as possible in order to improve software reliability. Moreover, according to Bell (2006), software testing is a necessity to help attain any desired level of software quality. There are two main forms of testing, which are verification and validation. In the IEEE/ANSI definition, verification is the process of evaluating a system or component to determine whether the product of a given development phase satisfies the conditions imposed at the start of that phase by including evaluating, reviewing, inspecting, and performing some static checks of requirement specifications, design specifications, and code. Moreover, validation form of testing, as defined by IEEE/ANSI, is the process of evaluating a system or component during or at the end of the development process to determine whether it satisfies specified requirements. There are two levels at validation activities, which are low-level testing and high-level testing. The low-level testing consists of unit testing and integration testing. The high-level testing consists of usability testing, function testing, system testing and acceptance testing (Chen, 2004)

Xie (2005) claimed that, software testing is intensively arduous and takes approximately half of software development effort. Additionally, to facilitate verification and validation periods and removing the burden of manual testing of the software, automated software testing is made by using tools. Automated software testing not only reduces the effort for the whole testing process but also increases the quality of software (Chen, 2004). A sequence of actions in software testing can be saved as testing script, which can be executed in other testing. Additionally, with automated software testing tools performance of the software can be easily managed by comparing manual testing. With the help of automated tools, data processing sequence are the same every time, which gives consistency to the testing (Chen, 2004).

According to Whittaker (2000), software testing is classified into two phases. The scope of the first phase is modeling the software's environment, which includes unit testing, integration testing and system testing. The scope of second phase is test selection, determines what type of testing is being done. There are two main types, which are structural (white-box testing) and functional testing (black-box testing).

Testing Strategies

Unit testing

Unit testing tests, which often execute the test in a debugger, individual software components or a collection of components sometimes require the construction of throwaway driver code and stubs. At unit test, testers define the input domain for the units in question and ignore the rest of the system (Whittaker, 2000). Hwang (2007) defined unit test as tests performed for each program module and run in isolation from the rest of the program. The unit test is usually conducted by programmers while the later stages of testing may be done by an independent testing group. According to Nagappan(2005), the primary objectives of unit testing are:

- to verify the code against the component, i.e. to see if the code does what the component is expected to do with respect to the overall system,
- to execute all new and changed code to ensure all branches are executed in all directions,
- to check for the correctness of logic and data paths,
- to exercise all error messages, return codes and response options.

Integration testing

Integration testing tests multiple components such as hardware and software which receive prior and separate unit testing by a focus on the subset of the domain, which represents communication between the components (Whittaker, 2000). According to Hwang (2007), integration test includes subsystem and system test, which involves testing for the interactions of subsets of the system modules.

System testing

Whittaker (2000) defined system testing as a collection of components that constitutes a deliverable product by providing the satisfaction criteria for the entire domain. According to Hwang (2007), system test determines whether all requirements have been satisfied and are performed in accordance with the reviewed software verification and validation plan. Moreover, system test is also used to verify the correctness and reliability of the system in its predicted operational environment, i.e., the software will be tested based on how users will employ it.

Functional Testing/ Black-Box Testing

Functional testing requires the selection of test scenarios without regard to source code structure or the internal mechanism of a system or component. Functional testing behaves on complete or integrated systems by focusing on the outputs of execution conditions. Hwang (2007) claimed that black-box testing is typically performed when the components of the system are integrated sufficiently so as to demonstrate that all requirements are fulfilled. According to Chen (2004), the activity for functional testing is performed by selecting inputs and executing them to verify their functional correctness with regards to the requirement specification. The inputs used for testing, especially functional testing, is called test case. The standard IEEE computer dictionary has defined the test case as "A set of inputs, execution preconditions, and expected outcomes developed for a particular objective, such as to exercise a particular program path or to verify compliance with a specific requirement."

In "black box" testing the tester does not require an understanding of the internal structural organization and behavior of the software. The test cases are derived from input conditions that will fully execute all functional requirements of a program (Misra, 2000)

Structural testing/ White-Box Testing

Bell (2007) explained that white-box testing focuses on the internal mechanism of a system, for instance examining all possible branches or all assignment statements found in source code. Hwang (2007) claimed that white-box testing is often associated with test coverage metrics, which measure the percentage of paths of selected types that are exercised by test cases. According to Misra (2000), test cases are derived to exercise all independent paths, all logical conditions and their possibilities, all loops and data structures involved within a piece of code in order to understand the logic of the program, the data structures, file structures and different control structures used.

Security

McGraw (2003) defined software security as the idea of engineering software that continues to function correctly under malicious attack by understanding software-induced security risks and how to manage them. Potter and McGraw (2004) also explained that software security is about making software behave correctly in the presence of a malicious attack, and software security testing goes deeper than simple black-box probing on the presentation layer and even beyond the functional testing of security apparatus. Building security into every process especially online deployment should be an integral part of the business process for software and systems delivery within a well-governed organization. Security vulnerabilities can cause sufficient opportunity for hackers to access the products details or personal data, and risks for the entire business. In addition, a software security tester can properly focus on areas of code in which an attack is likely to succeed and provides high assurance. McGraw (2003) also stated that central and critical aspect of the computer security problem resides in software. According to Cowan (2003), software security is fundamentally simple: just runs perfect software. A perfect software is especially necessary for large, complex systems.

According to Kals, Kirda, Kruegel, and Jovanovic (2006), security vulnerabilities in software systems can be scanned by black-box vulnerability scanners. Moreover, there is the need for a scanner that covers a broad range of general classes of vulnerabilities to have secure systems or web applications, without specific knowledge of bugs in particular versions of web applications. Web application vulnerabilities can be performed both manually and automatically. Automation process included scanning tools and static analysis, while manual process included penetration testing and code review (as cited in Kolat, 2006)

Various automated security testing tools are available. These web applications security software exist to ensure the security and compliance of websites. Users can focus on the more challenging issue of securing the web applications from any exploitable vulnerability by the help of automated vulnerability scanning.

Security Testing Tools

IBM Watchfire AppScan is Web application security software. IBM Watchfire AppScan automatically scans web applications looking for security vulnerabilities.

IBM Watchfire AppScan provides scanning, reporting and fix recommendations, and is suitable for all types of security testing by a variety of users, including application developers, quality assurance teams, penetration testers, security auditors and senior managers (ftp://ftp.software.ibm.com/software/rational/web/brochures/ r_appscan_lifecycle.pdf). Additionally, the tool scans through complicated login forms and other technologies that may cause problems for automated crawlers. IBM Rational AppScan security issues view can be seen at Figure 4. IBM Watchfire AppScan simulates like hacker attacks such as Cross-Site Scripting; HTTP Response Splitting; Parameter Tampering; Hidden Field Manipulation; Backdoors/Debug Options; Stealth Commanding; Forceful Browsing; Application Buffer Overflow; Cookie Poisoning; Third-Party Misconfiguration; Known Vulnerabilities; HTTP Attacks; SQL Injections; Suspicious Content; XML/SOAP Tests; Content Spoofing; LDAP Injection; XPath Injection; Session Fixation(http://www.spectrum-systems.com/vendors/watchfire/appscansix-overview.pdf) . Furthermore, there are delta analysis reports, which tell users what changes have occurred from one scan to the next. The reported information includes what has been fixed, what has not and what new security issues have been introduced since the initial scan (ftp://ftp.software.ibm.com/software/rational/web/datasheets/ watchfire_appscan_ds.pdf). Rational AppScan core features for scanning efficiency is:

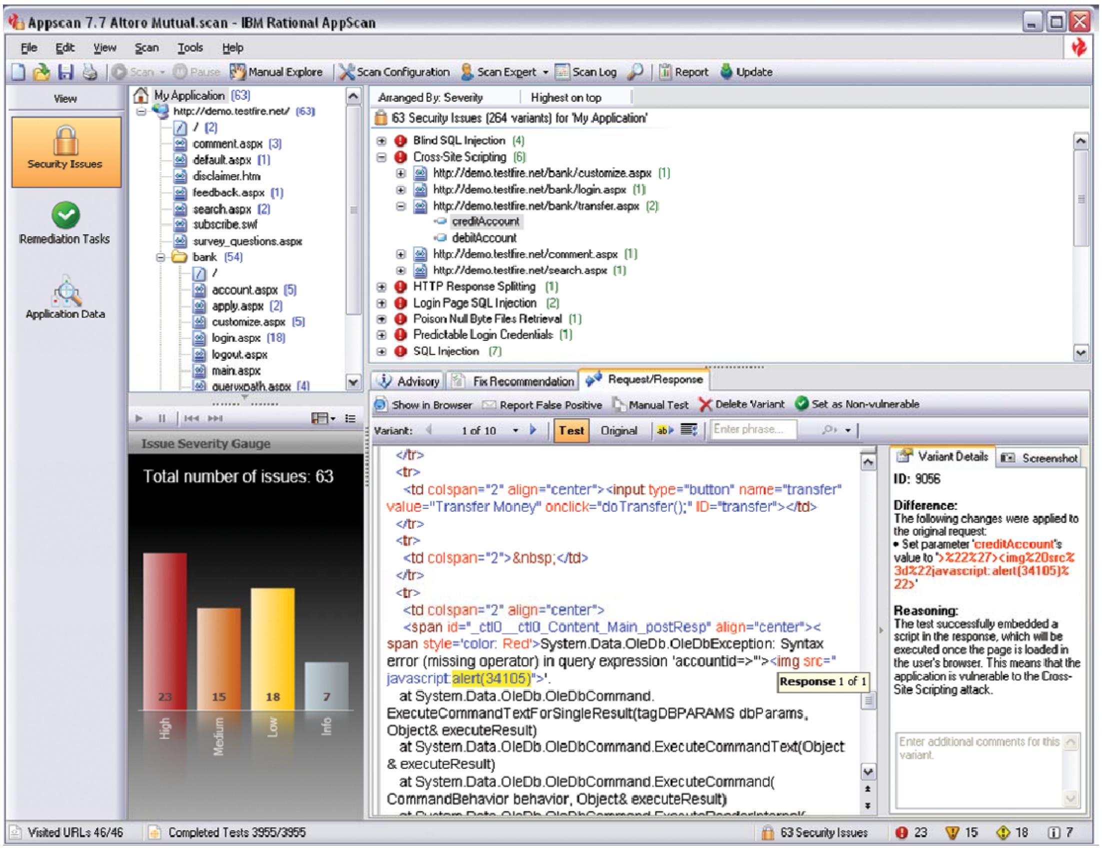
- A user interface with a view selector for the application tree, hierarchical security issues results lists, developer remediation views and details panes.

- An adaptive test process that enables users to analyze application parameters and select only relevant tests that do not impede the development process.
- Complex authentication support that enables testing for multistep authentication procedures in Web applications, including stepped authentication, multifactor authentication, one-time passwords, Universal Serial Bus (USB) keys, smart cards and mutual authentication.
- Advanced session management that performs automatic relogins when required.
- Realtime results views that enable users to act on issues before a scan is complete.
- Pattern search rules that facilitate security testing around credit card, social security or other numerical sequences.

System requirements for Watchfire AppScan V7.6 software are given at Table 4.

Table 4. System Requirements for Watchfire AppScan Version 7.6

Memory 512MB RAM (1GB recommended for scanning large sites)
Free disk space 1GB (10GB recommended for scanning large sites)
Operating system Microsoft Windows® XP, Windows 2000, Windows 2003 Enterprise Edition, Windows Vista
Browser Microsoft Internet Explorer 5.5 or higher (IE 6.0 or higher recommended)


Appscan 7.7 Altoro Mutual.scan - IBM Rational AppScan
File Edit View Scan Tools Help
Scan Pause Manual Explore Scan Configuration Scan Expert Scan Log Report Update
View
Security Issues
Remediation Tasks
Application Data
My Application (63)
http://demo.testfire.net/ (63)
/ (2)
comment.aspx (3)
default.aspx (1)
disclaimer.htm
feedback.aspx (1)
search.aspx (2)
subscribe.swf
survey_questions.aspx
bank (54)
/
account.aspx (5)
apply.aspx (2)
customize.aspx (5)
login.aspx (18)
logout.aspx
main.aspx
queryxpath.aspx (4)
Issue Severity Gauge
Total number of issues: 63
23 High
15 Medium
18 Low
7 Info
Arranged By: Severity Highest on top
63 Security Issues (264 variants) for 'My Application'
Blind SQL Injection (4)
Cross-Site Scripting (6)
http://demo.testfire.net/bank/customize.aspx (1)
http://demo.testfire.net/bank/login.aspx (1)
http://demo.testfire.net/bank/transfer.aspx (2)
creditAccount
debitAccount
http://demo.testfire.net/comment.aspx (1)
http://demo.testfire.net/search.aspx (1)
HTTP Response Splitting (1)
Login Page SQL Injection (2)
Poison Null Byte Files Retrieval (1)
Predictable Login Credentials (1)
SQL Injection (7)
Advisory Fix Recommendation Request/Response
Show in Browser Report False Positive Manual Test Delete Variant Set as Non-vulnerable
Variant: 1 of 10 Test Original Enter phrase...
Response 1 of 1
Variant Details Screenshot
Enter additional comments for this variant.
Visited URLs 46/46 Completed Tests 3955/3955 63 Security Issues 23 15 18 7

```
  </tr>
  <tr>
   <td colspan="2" align="center"><input type="button" name="transfer"
value="Transfer Money" onclick="doTransfer();" ID="transfer"></td>
  </tr>
  <tr>
   <td colspan="2">&nbsp;</td>
  </tr>
  <tr>
   <td colspan="2" align="center">
   <span id="_ctl0__ctl0_Content_Main_postResp" align="center"><
span style='color: Red'>System.Data.OleDb.OleDbException: Syntax
error (missing operator) in query expression 'accountid=>"'><img src="
javascript:alert(34105)">'.
   at System.Data.OleDb.OleDbCommand.
ExecuteCommandTextForSingleResult(tagDBPARAMS dbParams,
Object& executeResult)
   at System.Data.OleDb.OleDbCommand.ExecuteCommandText(Object
& executeResult)
   at System.Data.OleDb.OleDbCommand.ExecuteCommand(
CommandBehavior behavior, Object& executeResult)
```

ID: 9056

Difference:
The following changes were applied to the original request:
- Set parameter 'creditAccount''s value to '">%22%27><img%20src%3d%22javascript:alert(34105)%22>'

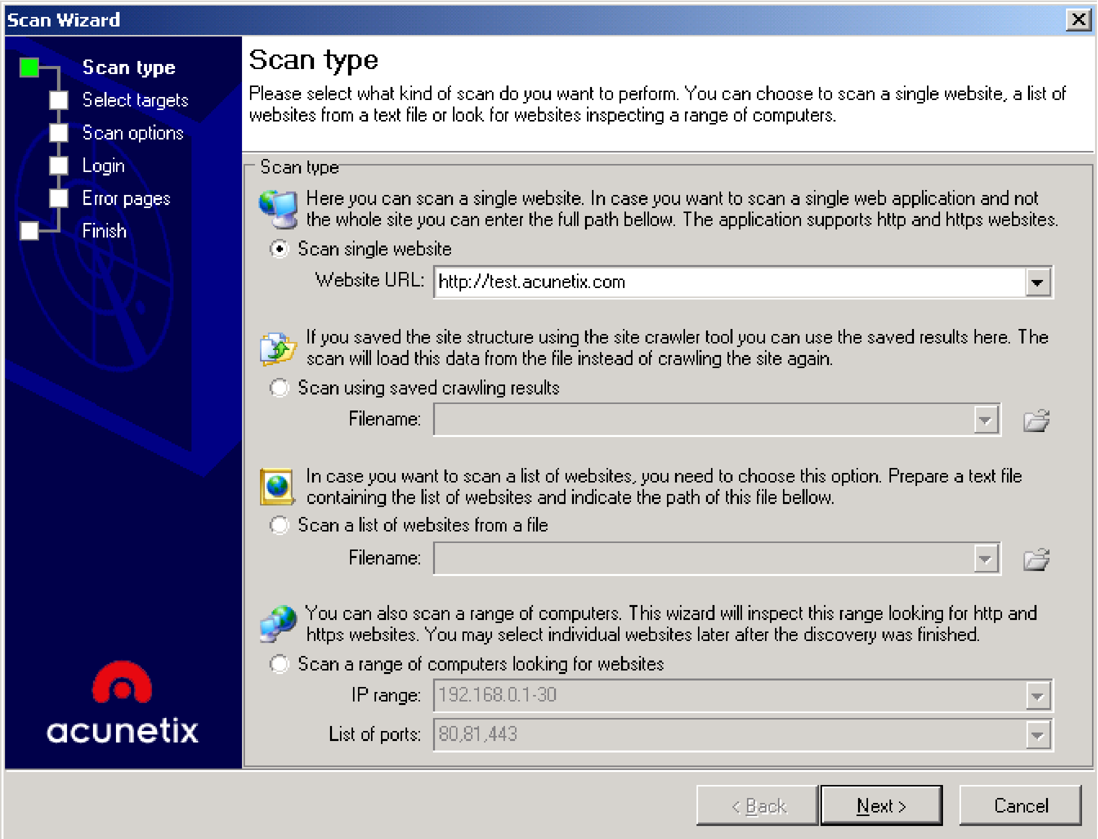
Reasoning:
The test successfully embedded a script in the response, which will be executed once the page is loaded in the user's browser. This means that the application is vulnerable to the Cross-Site Scripting attack.

Figure 4. IBM Rational AppScan security issues view

Acunetix Web Vulnerability Scanner is other Web application security software. Acunetix Web Vulnerability Scanner crawls web sites, automatically analyzes the web applications and finds perilous SQL injection, cross site scripting and other vulnerabilities that expose on line business. Acunetix Web Vulnerability Scanner reports identify where web applications need to be fixed, thus enabling users to protect their business from impending hacker attacks. Moreover, there are four different scan types (Figure 5). The default one offers a normal procedure where one web site gets all the attention. There is also multiple sites scan options by selecting a file that contains the list of URIs. If software's built-in crawler module was used, users can also act upon its results. The final scan type offers scanning of a range of

IP addresses with web servers running on ports specified by the user.

(http://www.acunetix.com/vulnerability-scanner/wvs4manual.pdf)



Scan type

Please select what kind of scan do you want to perform. You can choose to scan a single website, a list of websites from a text file or look for websites inspecting a range of computers.

Scan type

Here you can scan a single website. In case you want to scan a single web application and not the whole site you can enter the full path bellow. The application supports http and https websites.

● Scan single website

Website URL: http://test.acunetix.com

If you saved the site structure using the site crawler tool you can use the saved results here. The scan will load this data from the file instead of crawling the site again.

○ Scan using saved crawling results

Filename:

In case you want to scan a list of websites, you need to choose this option. Prepare a text file containing the list of websites and indicate the path of this file bellow.

○ Scan a list of websites from a file

Filename:

You can also scan a range of computers. This wizard will inspect this range looking for http and https websites. You may select individual websites later after the discovery was finished.

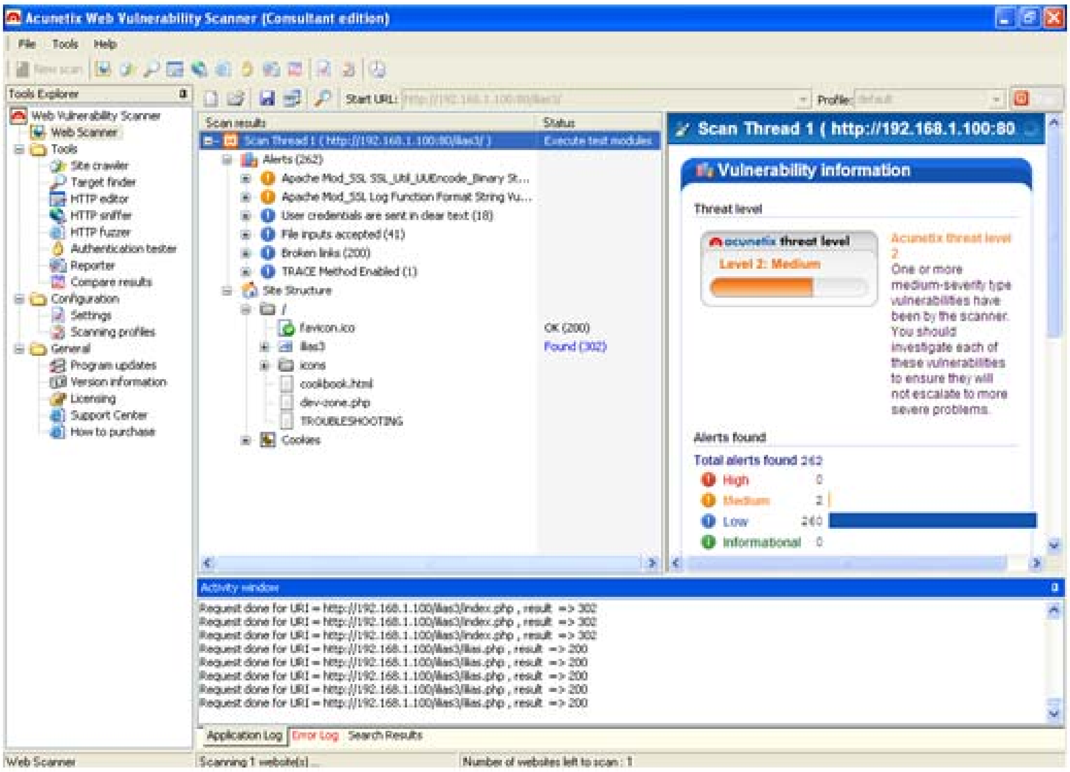
○ Scan a range of computers looking for websites

IP range: 192.168.0.1-30

List of ports: 80,81,443

Figure 5. Scan type options of Acunetix Web Vulnerability Scanner

Acunetix Web Vulnerability Scanner includes features of an automatic JavaScript analyzer allowing for security testing of Ajax and Web 2.0 applications, sophisticated SQL injection and Cross site scripting testing, visual macro recorder makes testing web forms and password protected areas easy, extensive reporting facilities including VISA PCI compliance reports, multi-threaded and lightning fast scanner crawls hundreds of thousands of pages with ease, intelligent crawler detects web server type and application language, and Acunetix crawls and analyzes websites including flash content, SOAP and AJAX. (http://www.acunetix.com/ vulnerability-scanner/)



Acunetix threat level 2
One or more medium-severity type vulnerabilities have been by the scanner. You should investigate each of these vulnerabilities to ensure they will not escalate to more severe problems.

Total alerts found 262

High	0
Medium	2
Low	260
Informational	0

```
Request done for URI = http://192.168.1.100/ilias3/index.php , result  => 302
Request done for URI = http://192.168.1.100/ilias3/index.php , result  => 302
Request done for URI = http://192.168.1.100/ilias3/index.php , result  => 302
Request done for URI = http://192.168.1.100/ilias3/ilias.php , result  => 200
Request done for URI = http://192.168.1.100/ilias3/ilias.php , result  => 200
Request done for URI = http://192.168.1.100/ilias3/ilias.php , result  => 200
Request done for URI = http://192.168.1.100/ilias3/ilias.php , result  => 200
Request done for URI = http://192.168.1.100/ilias3/ilias.php , result  => 200
```

Scanning 1 website(s) ...

Number of websites left to scan : 1

Figure 6. The site crawler of Acunetix Web Vulnerability Scanner

The scan results present the scan results in a printable format. Both summary and detailed reports can be created. Moreover, the scan results contain alerts. Alerts are classified as Informational only, Low, Medium or High (Figure 6). Alerts can be viewed under the Alerts node. Site Structure node shows the directories and files that the crawler discovered (including those discovered by a manual crawl) and their structure within the file system. Activity Window logs all of the activities performed by WVS. The Scan Results window will show users the number of each vulnerability found. By choosing Compare Results in the Tools menu, the results of a current scan and previous (saved) scan can be compared. Moreover, the report includes sections of scan groups, scan summary, alerts summary and alerts details.

System requirements for Acunetix Web Vulnerability Scanner Version 4.0 are given at Table 5.

Table 5. System Requirements for Acunetix Web Vulnerability Scanner v.4.0

Memory: 128 MB of RAM (256MB or higher recommended).
Free disk space: 200 MB of available hard-disk space.
Operating system: Microsoft Windows XP Professional or Home Edition, Windows 2000, Windows Server 2003.
Browser: Microsoft Internet Explorer 5.1 (or higher).

Accessibility

Boldyreff explained that while web is growing rapidly and is being used more and more, web accessibility is become more crucial. Accessibility is a measure of how easy it is to access, read, and understand the content of a Website. Berners-Lee also indicates the following quote on www.w3.org/WAI/ :“The power of the Web is in its universality. Access by everyone regardless of disability is an essential aspect." (as cited in Boldyreff, 2002).

Additionally, ISO/IEC FCD 24751-1 Individualized Adaptability and Accessibility in E-learning, Education and Training Part 1 standard describes how learning systems can be more accessible. Moreover, while designing web page, developers should consider different situations or problems like difficulty reading or comprehending text, a slow Internet connection, an early version of a browser, a different browser entirely, a voice browser, or a different operating system, difficulties in seeing, hearing, moving or process some types of information easily or at all, and so on. W3C/WAI Web Content Accessibility Guidelines describes how to create accessible content (http://www.w3.org/TR/WAI-WEBCONTENT/). Web Content Accessibility Guidelines 1.0 explains the ways of both making Web Content accessible to people with disabilities and promoting accessibility. Thus, people can find information on the Web more quickly by following these guidelines. That document also includes an appendix that organizes all of the checkpoints by topic and priority, which are identified in the appendix include images, multimedia, tables, frames, forms, and scripts, therefore checkpoints directly improve the performance of the Web services while reducing the maintenance effort required (http://www.w3.org/TR/WCAG10/). The guidelines for accessibility are:

- Provide equivalent alternatives to auditory and visual content.
- Don't rely on color alone.
- Use markup and style sheets and do so properly.
- Clarify natural language usage
- Create tables that transform gracefully.
- Ensure that pages featuring new technologies transform gracefully.
- Ensure user control of time-sensitive content changes.
- Ensure direct accessibility of embedded user interfaces.
- Design for device-independence.
- Use interim solutions.
- Use W3C technologies and guidelines.
- Provide context and orientation information.
- Provide clear navigation mechanisms.
- Ensure that documents are clear and simple.

W3C primarily pursues its mission through the creation of Web standards and guidelines for providing “Web interoperability”. Web interoperability term is related with the compatible Web technologies and allowing any hardware and software used to access the Web to work together.W3C also engages in education and outreach, develops software, and serves as an open forum for discussion about the Web in order for the Web to reach its full potential.

W3C working group has stated that in Determination and evaluation of Web accessibility:

> Conformance with the WCAG 1.0 (and other W3C) guidelines will enhance the market share and audience reach of your Web site by increasing its general usability. Adoption of WCAG 1.0 recommendations also demonstrates your commitment to social responsibility and equity of access to information and services. In addition, many of the WCAG 1.0 checkpoints will directly improve the performance of your Web services and reduce the maintenance effort required.

There are various other tools exist to determine whether or not a web site adheres to various accessibility standards by providing useful feedback to Web developers and maintainers, and many assist with the repair of the site.

Functional Accessibility Evaluator (FAE) free online service, which is developed by University of Illinois at Urbana-Champaign, is used for accessibility criteria measurement. FAE is useful for development of functionally accessible web resources and analyzes online systems for markup that is consistent with the use of CITES/DRES HTML Best Practices, which is statement of techniques for implementation of the W3C The CITES/DRES HTML Best Practices are not a new standard, but rather a statement of techniques for implementation of the W3C and United States Federal Government Section 508 standards. (http://fae.cita.uiuc.edu/about.php). Section 508 requires that individuals with disabilities, who are members of the public seeking information or services from a Federal agency, have access to and use of information and data that is comparable to that provided to the public who are not individuals with disabilities, unless an undue burden would be imposed on the agency (http://www.section508.gov/index.cfm?FuseAction=Content&ID=12).

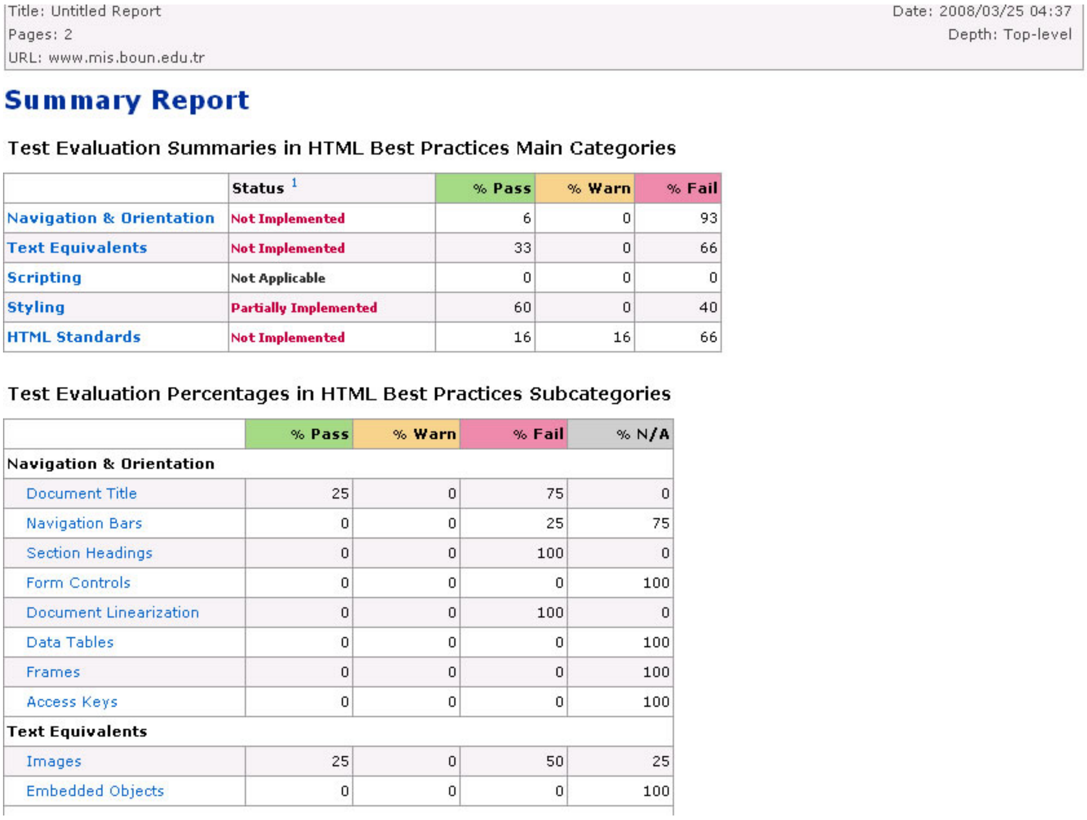
The “Run FAE” page provides the analysis parameters, which are URL(s), report title, depth of evaluation and follow links in. URL(s) parameter specifies the page(s) to be evaluated. When there is more than one URL, that page will serve as the starting point of the analysis, along with additional pages, depending on the “Depth of Evaluation” and “Follow Links in” settings. Moreover, Depth of Evaluation chooses are at below.

> When “Top-level page only” is selected, only the page specified by the URL will be analyzed.
> “Include all second-level pages” will cause all pages linked from the top-level page (with domain restrictions as explained below) to be included in the analysis.
> “Include all third-level pages” will cause all pages linked from the top- and second-level pages (with domain restrictions as explained below) to be included in the analysis.
> (http://fae.cita.uiuc.edu/about.php?page=overview)

Furthermore, at Follow Links in:

> In cases where “Depth of Evaluation” is set to either “Include all second-level pages” or “Include all third-level pages” the web crawler used by FAE can follow links in two different ways:
> When “Specified domain only” is selected, link following is restricted to the same domain as the specified URL.
> When “Next-level subdomains” is selected, links that are in subdomains of the next-level domain (relative to the domain specified by the URL) will also be followed.
> (http://fae.cita.uiuc.edu/about.php?page=overview)

The analysis of documents based on the categories of navigation & orientation, text equivalents, scripting, styling and standards (Figure 7). According to FAE, Navigation & Orientation is Inclusion of structural markup that facilitates navigation and contextual orientation. Text Equivalents is proper use of images for interoperability and the provision of text descriptions for non-text content. Scripting is avoidance of scripting techniques that compromise accessibility and interoperability. Styling is use of CSS styling techniques to separate content and structural information from styling and presentation. Finally, standards improve interoperability and provide more choices in the use of technologies for rendering web content. (http://fae.cita.uiuc.edu/about.php?page=overview)



Title: Untitled Report
Pages: 2
URL: www.mis.boun.edu.tr

Date: 2008/03/25 04:37
Depth: Top-level

Summary Report

Test Evaluation Summaries in HTML Best Practices Main Categories

	Status [1]	% Pass	% Warn	% Fail
Navigation & Orientation	Not Implemented	6	0	93
Text Equivalents	Not Implemented	33	0	66
Scripting	Not Applicable	0	0	0
Styling	Partially Implemented	60	0	40
HTML Standards	Not Implemented	16	16	66

Test Evaluation Percentages in HTML Best Practices Subcategories

	% Pass	% Warn	% Fail	% N/A
Navigation & Orientation				
Document Title	25	0	75	0
Navigation Bars	0	0	25	75
Section Headings	0	0	100	0
Form Controls	0	0	0	100
Document Linearization	0	0	100	0
Data Tables	0	0	0	100
Frames	0	0	0	100
Access Keys	0	0	0	100
Text Equivalents				
Images	25	0	50	25
Embedded Objects	0	0	0	100

Figure 7. Summary report of an accessibility test

Functional Accessibility Evaluator Version 0.9.3 properties are at below.

(http://fae.cita.uiuc.edu/about.php?page=versions)

> The Sitewide Report now provides information for each test result indicating which pages passed or failed.
> In both the Page and Sitewide Reports, rules are stated more succinctly and independently of numerical results.
> A 'Rules Summary' page is now provided in 'About FAE', which lists the Best Practices and associated criteria used in FAE tests.
> The HTML Standards test for the character encoding declaration now evaluates only to 'pass' or 'warn'.
> A problem was fixed with the test for data tables relating to the number of rows and columns required.
> The evaluation criteria associated with the test for proper nesting of heading elements was updated: the null result is now triggered simply when there are no 'h1' elements.

Web Accessibility Checker is a link to the complete list of accessibility tests, and an Open Source software program. This accessibility checker supports the Evaluation and Repair Language (EARL) standard (http://checker.atrc.utoronto.ca/index.html). EARL standard method for expressing test results is used to express accessibility errors and compliance to accessibility standards. It is only useful for exchanging information between machines (http://checker.atrc.utoronto.ca/servlet/Earlpage).

While the accessibility checker evaluates Web page, the checker identifies three types of problems:

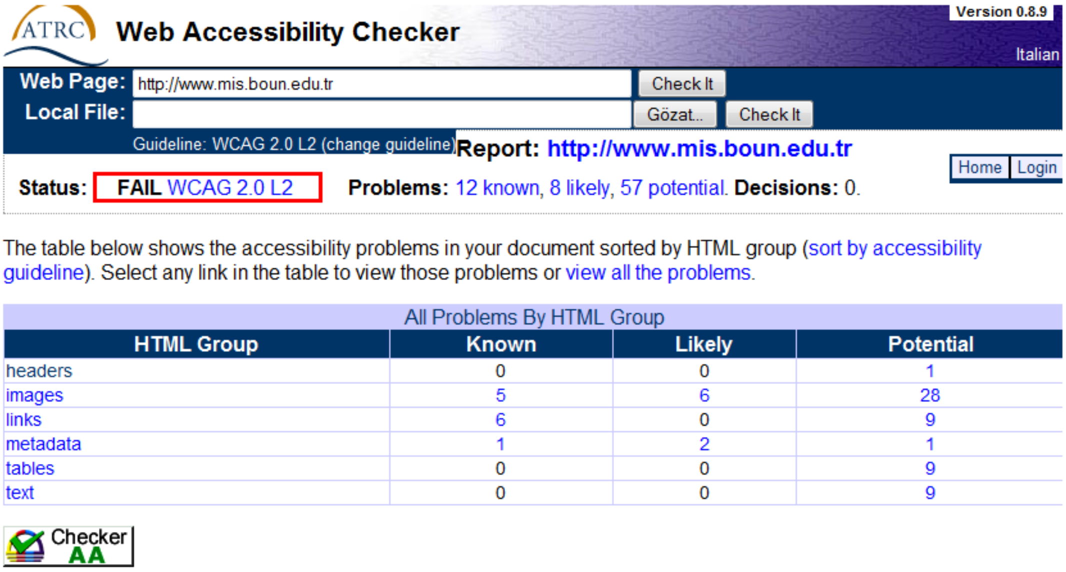
- Known problems (you must modify your page to fix these problems)
- Likely problems (you likely must modify your page to fix these problems)
- Potential problems (you may not have to modify your page for these problems)

Accessibility checker terms are:

> Conditional Pass: A Conditional Pass is given to a web page when it passes all accessibility checks for known problems. The web page still contains potential accessibility problems that require a human to make decisions and resolve them.
> Known problems are things that the checker can detect with certainty. An example of this problem is when an image is missing alternate text (ALT text). When a known problem is encountered, the checker displays the item that is causing the problem and suggests a way of fixing the problem.
> You must modify your page to resolve known problems.
> Likely problems are things that the checker thinks are a problem but is unsure of. You must view the problem and decide if it really is a problem. An example of this problem occurs when the alt text for an image is the same as the filename for the image. Example: <img src="rex.jpg" alt="rex.jpg"/>
> You must likely modify your page to resolve one of these problems.
> Potential problems are things that the checker cannot detect for certainty. You must view the problem and decide if it really is a problem. Most potential problems can be resolved simply by having authors make decisions. For example, a user could decide that an

image does not require a long text description and have the issue removed from file's list of potential problems. (http://checker.atrc.utoronto.ca/terms.html#knownproblem)

Web Content Accessibility Guidelines (WCAG), Version 2.0 includes html groups of applets, forms, frames, general, headers, images, links, metadata, objects, scripts, tables and text (Figure 8).



Web Page: http://www.mis.boun.edu.tr

Local File:

Guideline: WCAG 2.0 L2 (change guideline)

Report: http://www.mis.boun.edu.tr

Status: FAIL WCAG 2.0 L2 Problems: 12 known, 8 likely, 57 potential. Decisions: 0.

The table below shows the accessibility problems in your document sorted by HTML group (sort by accessibility guideline). Select any link in the table to view those problems or view all the problems.

All Problems By HTML Group

HTML Group	Known	Likely	Potential
headers	0	0	1
images	5	6	28
links	6	0	9
metadata	1	2	1
tables	0	0	9
text	0	0	9

Figure 8. Accessibility test summary report

Performance

Performance is institution’s ability to perform tasks within certain constraints in time and resources. Moreover, performance is also the criteria to be used for evaluation of e-learning (Kefalas, Retalis, Stamatis and Theodoros, 2003). Additionally, performance testing helps both identifying bottlenecks in a system and establishing a baseline for future testing. Weyuker and Vokolos (2000) stated that performance could be evaluated from a user’s perspective, which is typically assessed in terms of throughput, stimulus-response time, or some combination of the two. Alternatively, performance testing could be used to assess the level of system availability.

Performance, load, and stress tests are subcategories of performance testing. Performance testing evaluates compliance of a system or component with specified performance requirements. Performance testing evaluates of how the system can be expected to perform in the field. Performance testing determines or validates the speed, scalability, and/or stability characteristics of the system or application under test. Stress testing is focused on determining or validating performance characteristics of the system or application under test when subjected to conditions beyond the limits of its specification or requirement. Load testing is focused on determining or validating performance characteristics of the system or application under test when subjected to workloads and load volumes anticipated during production operations (http://msdn2.microsoft.com/en-us/library/bb924375.aspx).

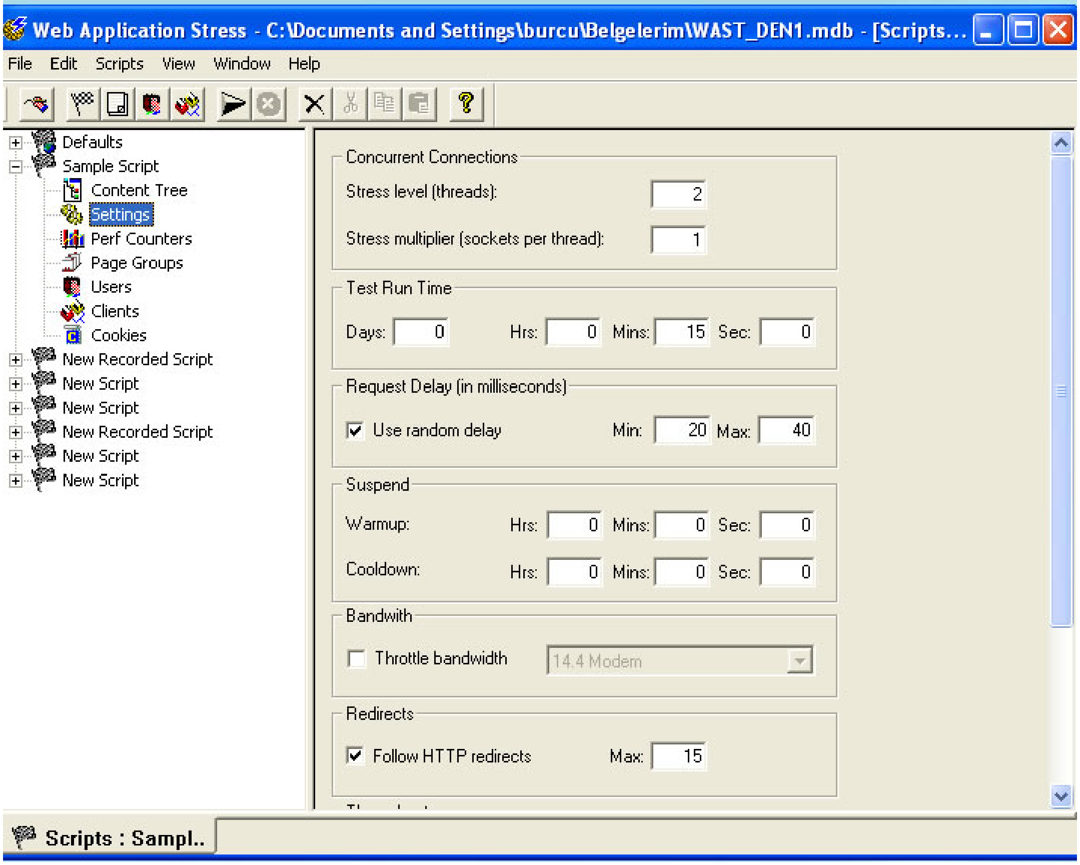
Avritzer and Weyuker (1996) introduced application-independent workload for doing performance evaluation by determining the given project as the quantity of software and system availability requirements made it impossible to port the system to the new platform in order to do the performance testing (as cited in Weyuker and Vokolos, 2000, p.1148).

ISO/IEC 24763 technical report is also includes information about participants in learning, education and training, provides a reference model which defines categories, objects, attributes and relationships of concepts such as competency, capability, performance and educational objective.

Tools

The Microsoft WAS web stress tool is designed to realistically simulate multiple browsers requesting pages from a web site. The tool can be used for gathering performance and stability information about web application. The tool simulates a large number of requests with a relatively small number of client machines. The goal is to create an environment that is as close to production as possible so that problems can be found and eliminated in the web application prior to deployment. (http://www.microsoft.com/technet/archive/itsolutions/intranet/downloads/webtutor.mspx?mfr=true)

The Settings dialog lets users to configure how the requests can be run against the server by adjusting the number of simulated clients, setting the number of threads and number of sockets on each thread. Settings of Web Application Stress tool can be seen at Figure 9.



Concurrent Connections

Stress level (threads): 2

Stress multiplier (sockets per thread): 1

Test Run Time

Days: 0 Hrs: 0 Mins: 15 Sec: 0

Request Delay (in milliseconds)

☑ Use random delay Min: 20 Max: 40

Suspend

Warmup: Hrs: 0 Mins: 0 Sec: 0

Cooldown: Hrs: 0 Mins: 0 Sec: 0

Bandwith

☐ Throttle bandwidth 14.4 Modem

Redirects

☑ Follow HTTP redirects Max: 15

Figure 9. Settings of Web Application Stress tool

The Web Application Stress tool has feature of capturing, configuring, running and displaying the result from running a stress test. Moreover, the captured Web links that can be accessed by the Web client when running the test are listed.

Browser Recorder is used to create a script. Browser Recorder is used to capture a browser session and WAS generates a test script from captured links. A manual mode is also available to allow user to manually add links and information about each link.

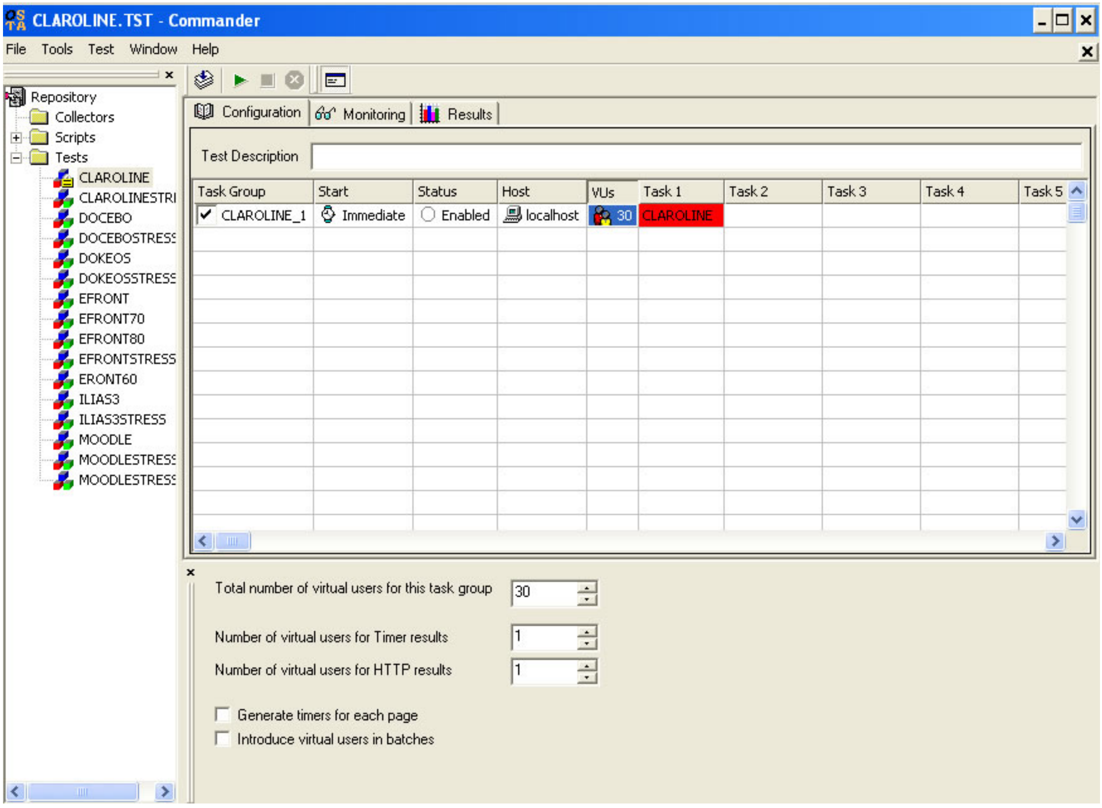
The user interface of WAS is straightforward with a list of scripts and sub-lists of the properties. There are a lot of options for most HTTP scenarios available, such as configuring users with usernames and passwords (optional) since large numbers of clients can be simulated or configuring specific pre-existing Cookie values for each user.

WAS test results of give useful information about the systems like throughput numbers that means the total number of hits and how many requests the Web server processed per second, the bandwidth information that signifies the average Kbytes received and sent per second, and more information about specific requests cn be seen at the page detail.

System requirements for WAS is required operating system: Windows 2000, Windows NT 4.0 Service Pack 4 with Internet Explorer 4.0 or newer. Microsoft Data Access Components 2.1 are required (http://www.microsoft.com/downloads/ details.aspx?FamilyID=e2c0585a062a439ea67d75a89aa36495&DisplayLang=en#QuickInfoContainer).

OpenSTA is open source software and its architecture is based on Common Object Request Broker Architecture (CORBA), which is Object Management Goup's (OMG) open, vendor-neutral architecture and infrastructure that computer applications use to work together over networks (http://corba-directory.omg.org/). A user can generate realistic heavy loads simulating the activity of hundreds to thousands of virtual users. OpenSTA graphs both virtual user response times and resource utilization information from all Web Servers, Application Servers, Database Servers and Operating Platforms under test; so that precise performance measurements can be gathered during load tests and analysis on these measurements can be performed (http://www.opensta.org/). Virtual user setting of OpenSTA can be seen at Figure 10.

At OpenSTA, the creation of Scripts, Collectors and Tests are separate processes that can be conducted independently. OpenSTA has HTTP/S recording and Script modeling functionality, using the Script Modeler Module, with Test creation and system data collection. It records browser requests issued during a Web session at the HTTP/S level, rather than recording the real time events of a browser, in order to create Scripts.



- Repository
  - Collectors
  - Scripts
  - Tests
    - CLAROLINE
    - CLAROLINESTRI
    - DOCEBO
    - DOCEBOSTRESS
    - DOKEOS
    - DOKEOSSTRESS
    - EFRONT
    - EFRONT70
    - EFRONT80
    - EFRONTSTRESS
    - ERONT60
    - ILIAS3
    - ILIAS3STRESS
    - MOODLE
    - MOODLESTRESS
    - MOODLESTRESS

Test Description

Task Group	Start	Status	Host	VUs	Task 1	Task 2	Task 3	Task 4	Task 5
☑ CLAROLINE_1	Immediate	○ Enabled	localhost	30	CLAROLINE				

Total number of virtual users for this task group 30

Number of virtual users for Timer results 1

Number of virtual users for HTTP results 1

□ Generate timers for each page

□ Introduce virtual users in batches

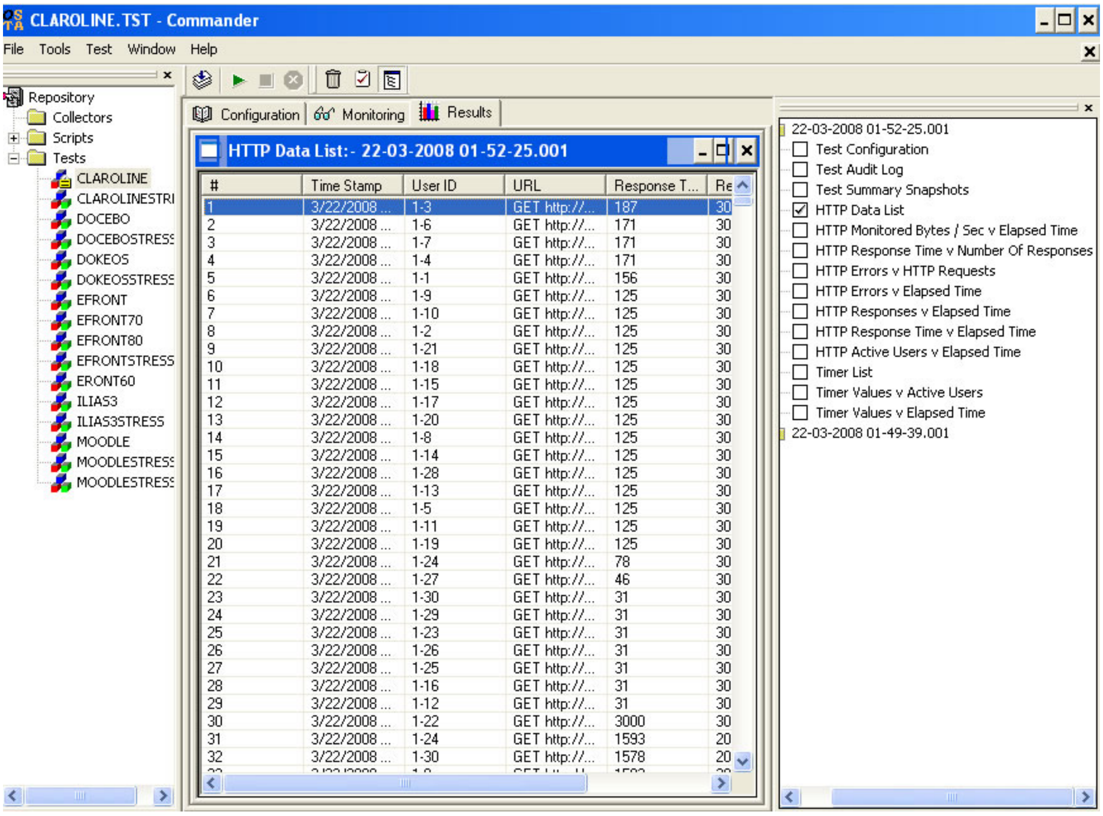
Figure 10. Virtual user setting of OpenSTA

The steps of OpenSTA performance test is:

(http://www.opensta.org/docs/ug/os-archi.htm#424072)

- Create Scripts (Script Modeler).
- Model Scripts if required (Script Modeler).
- Create data collection Collectors - optional (SNMP, NT Performance).
- Create Tests, by adding Task Groups containing the Scripts and Collectors required (Commander).
- Define Task Group settings (Commander), including:
- Schedule settings to control when Task Groups start and stop during a Test-run.
- Host computers used to run a Task Group: Script and Collector-based Task Groups.
- Number of Virtual Users used: Script-based Task Groups only.
- Task settings control the number of Script iterations and the delay between iterations during a Test-run: Script-based Task Groups only.
- Run a Test (Commander).
- Monitor a Test-run (Commander).
- Display Test results (Commander).

At the test result Repository Window, results are listed according to categories of test configuration, test audit log, test error log, test summary snapshots, HTTP data list, HTTP monitored bytes/sec v elapsed time, HTTP response time vs. Number of responses, HTTP errors vs. HTTP request, HTTP errors vs. HTTP elapsed time, HTTP responses vs. Elapsed time, HTTP response time vs. elapse time, HTTP active users vs. elapsed time, timer list, timer values v active users and timer values v elapsed time (Figure 11). According to the category of results chosen, data is displayed in graph or table format.



HTTP Data List:- 22-03-2008 01-52-25.001

#	Time Stamp	User ID	URL	Response T...	Re
1	3/22/2008 ...	1-3	GET http://...	187	30
2	3/22/2008 ...	1-6	GET http://...	171	30
3	3/22/2008 ...	1-7	GET http://...	171	30
4	3/22/2008 ...	1-4	GET http://...	171	30
5	3/22/2008 ...	1-1	GET http://...	156	30
6	3/22/2008 ...	1-9	GET http://...	125	30
7	3/22/2008 ...	1-10	GET http://...	125	30
8	3/22/2008 ...	1-2	GET http://...	125	30
9	3/22/2008 ...	1-21	GET http://...	125	30
10	3/22/2008 ...	1-18	GET http://...	125	30
11	3/22/2008 ...	1-15	GET http://...	125	30
12	3/22/2008 ...	1-17	GET http://...	125	30
13	3/22/2008 ...	1-20	GET http://...	125	30
14	3/22/2008 ...	1-8	GET http://...	125	30
15	3/22/2008 ...	1-14	GET http://...	125	30
16	3/22/2008 ...	1-28	GET http://...	125	30
17	3/22/2008 ...	1-13	GET http://...	125	30
18	3/22/2008 ...	1-5	GET http://...	125	30
19	3/22/2008 ...	1-11	GET http://...	125	30
20	3/22/2008 ...	1-19	GET http://...	125	30
21	3/22/2008 ...	1-24	GET http://...	78	30
22	3/22/2008 ...	1-27	GET http://...	46	30
23	3/22/2008 ...	1-30	GET http://...	31	30
24	3/22/2008 ...	1-29	GET http://...	31	30
25	3/22/2008 ...	1-23	GET http://...	31	30
26	3/22/2008 ...	1-26	GET http://...	31	30
27	3/22/2008 ...	1-25	GET http://...	31	30
28	3/22/2008 ...	1-16	GET http://...	31	30
29	3/22/2008 ...	1-12	GET http://...	31	30
30	3/22/2008 ...	1-22	GET http://...	3000	30
31	3/22/2008 ...	1-24	GET http://...	1593	20
32	3/22/2008 ...	1-30	GET http://...	1578	20

22-03-2008 01-52-25.001
☐ Test Configuration
☐ Test Audit Log
☐ Test Summary Snapshots
☑ HTTP Data List
☐ HTTP Monitored Bytes / Sec v Elapsed Time
☐ HTTP Response Time v Number Of Responses
☐ HTTP Errors v HTTP Requests
☐ HTTP Errors v Elapsed Time
☐ HTTP Responses v Elapsed Time
☐ HTTP Response Time v Elapsed Time
☐ HTTP Active Users v Elapsed Time
☐ Timer List
☐ Timer Values v Active Users
☐ Timer Values v Elapsed Time
22-03-2008 01-49-39.001

Figure 11 HTTP data list view of OpenSTA,

Systems requirements for OpenSTA are given at Table 6:

Table 6. Systems Requirements for OpenSTA

Hardware Specifications Pentium 200 processor 80MB RAM 20MB free hard disk space required for installation.
Web Browsers Supported for HTTP/S Recording in Script Modeler Internet Explorer 4 Internet Explorer 5 Netscape 4.7
Supported Protocols HTTP 1.0 and 1.1 HTTPS (SSL)

Software Prerequisites

Microsoft Windows 2000 or Microsoft Windows NT 4.0, with at least service pack 5

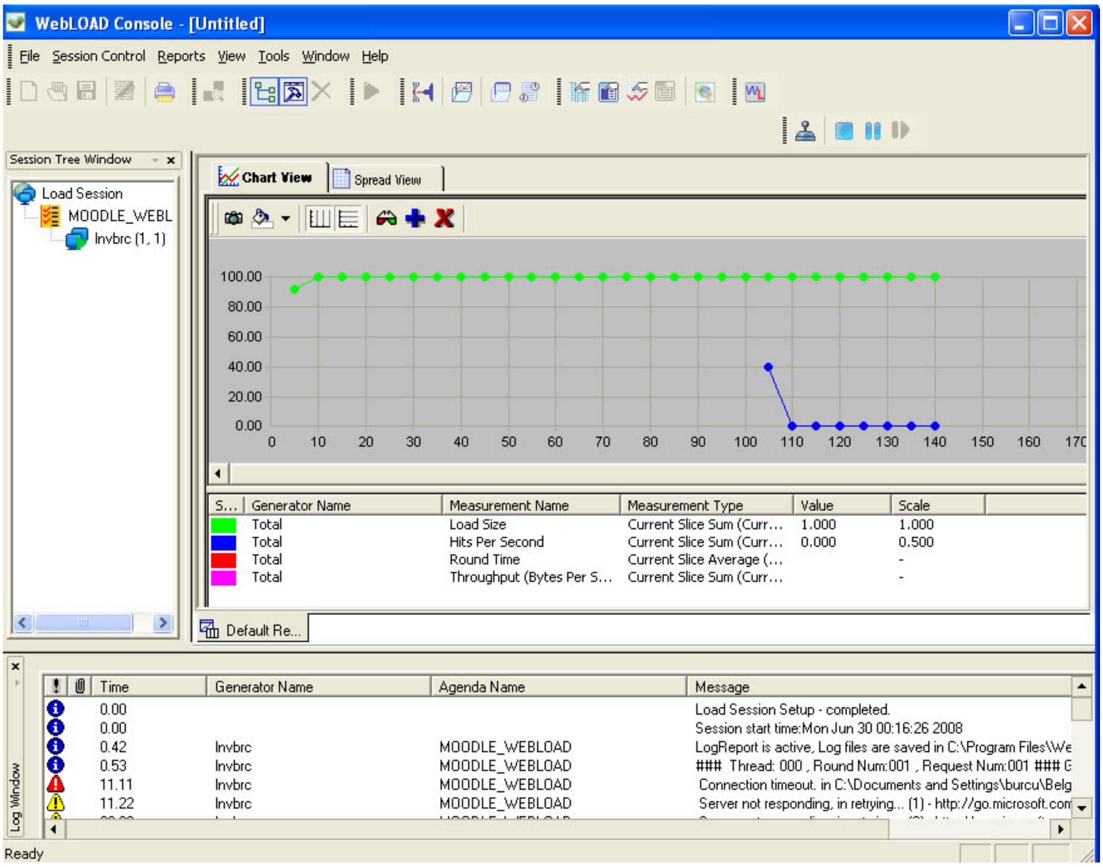
Windows Installer for Windows NT 1.1, instmsi.exe. This is not part of the basic installation of Windows NT 4.0.

An up-to-date HTML Help system.

OpenSTA also requires version 2.5, or later, of Microsoft Data Access Components MDAC_Typ.exe.

WebLOAD provides a comprehensive and robust environment for load testing. This includes a full authoring environment for recording, editing and debugging test scripts, a highly efficient execution environment for defining load parameters (virtual users), running and monitoring the tests as well as reporting tools for analyzing and presenting test results. The WebLOAD console also includes online reports for displaying the load session statistics. WebLOAD collects a complete set of statistics into its repository, enabling the user to define customized views that focus on any subset of the data. The user can toggle between a graphical view and a corresponding textual view of each user-defined report (http://www.webload.org/).

WebLOAD begins recording all of the actions that performed in the browser. When completing the performance testing scenario, user should stop the recording by saving the script to WebLOAD agenda. Moreover, the host and virtual clients can be adjusted. The number of Virtual Clients that can be generated depends on the power of the host machine.

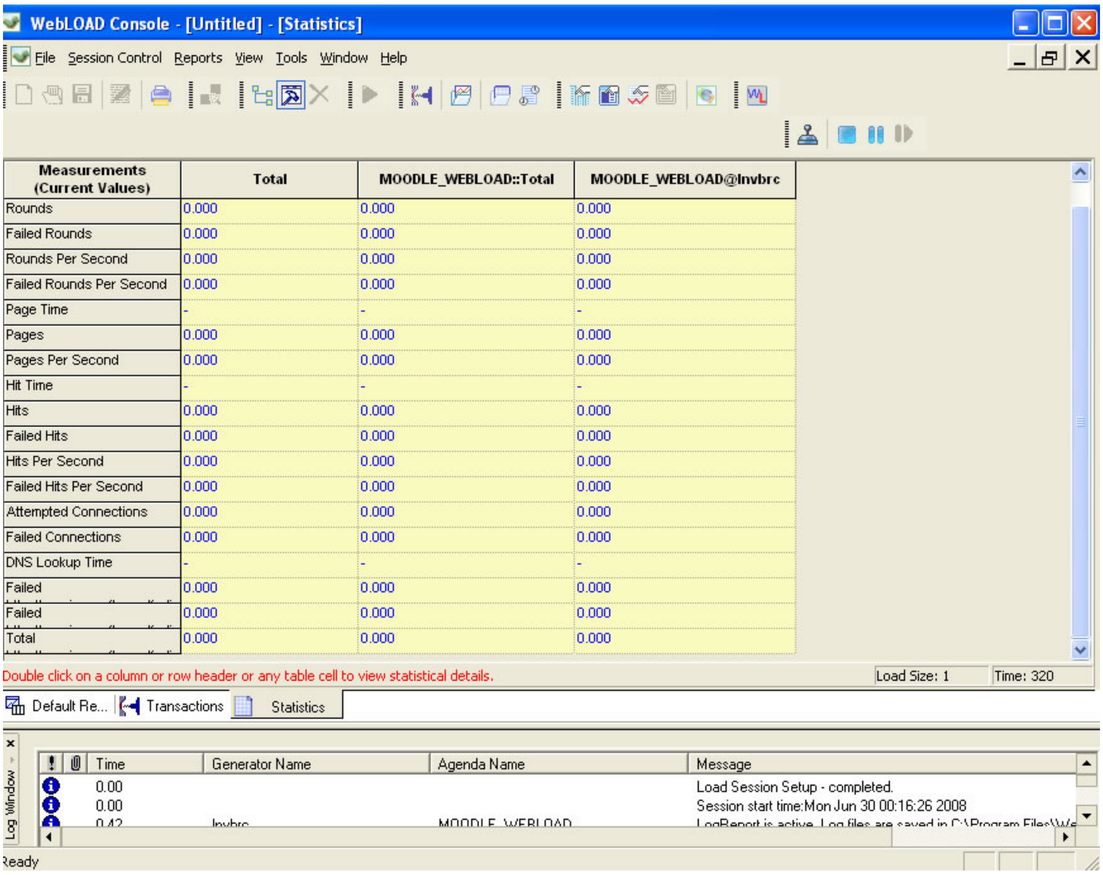


S...	Generator Name	Measurement Name	Measurement Type	Value	Scale
🟩	Total	Load Size	Current Slice Sum (Curr...	1.000	1.000
🟦	Total	Hits Per Second	Current Slice Sum (Curr...	0.000	0.500
🟥	Total	Round Time	Current Slice Average (...		-
🟪	Total	Throughput (Bytes Per S...	Current Slice Sum (Curr...		-

!		Time	Generator Name	Agenda Name	Message
ℹ		0.00			Load Session Setup - completed.
ℹ		0.00			Session start time:Mon Jun 30 00:16:26 2008
ℹ		0.42	lnvbrc	MOODLE_WEBLOAD	LogReport is active, Log files are saved in C:\Program Files\We
ℹ		0.53	lnvbrc	MOODLE_WEBLOAD	### Thread: 000 , Round Num:001 , Request Num:001 ### G
⚠		11.11	lnvbrc	MOODLE_WEBLOAD	Connection timeout. in C:\Documents and Settings\burcu\Belg
⚠		11.22	lnvbrc	MOODLE_WEBLOAD	Server not responding, in retrying... (1) - http://go.microsoft.com

Figure 12.Chart view of WebLOAD

The test results can be view in real time (Figure 12). Data on load size, hits per second, round time (one run through the Agenda) and throughput (bytes per second) appear in the Chart View window, with a color legend underneath the chart. Each data point represents a 20 second slice. Data is gathered every 20 seconds and is then either averaged (for hits, round time and throughout) or summed (for load size). The graphs are all normalized to scale so they can be viewed in a single Chart View window (http://www.webload.org/reporting.html). Result view of WebLOAD can be seen at Figure 13.



Measurements (Current Values)	Total	MOODLE_WEBLOAD::Total	MOODLE_WEBLOAD@lnvbrc
Rounds	0.000	0.000	0.000
Failed Rounds	0.000	0.000	0.000
Rounds Per Second	0.000	0.000	0.000
Failed Rounds Per Second	0.000	0.000	0.000
Page Time	-	-	-
Pages	0.000	0.000	0.000
Pages Per Second	0.000	0.000	0.000
Hit Time	-	-	-
Hits	0.000	0.000	0.000
Failed Hits	0.000	0.000	0.000
Hits Per Second	0.000	0.000	0.000
Failed Hits Per Second	0.000	0.000	0.000
Attempted Connections	0.000	0.000	0.000
Failed Connections	0.000	0.000	0.000
DNS Lookup Time	-	-	-
Failed	0.000	0.000	0.000
Failed	0.000	0.000	0.000
Total	0.000	0.000	0.000

Double click on a column or row header or any table cell to view statistical details.

Load Size: 1 Time: 320

Time	Generator Name	Agenda Name	Message
0.00			Load Session Setup - completed.
0.00			Session start time:Mon Jun 30 00:16:26 2008
0.42	lnvbrc	MOODLE_WEBLOAD	LogReport is active. Log files are saved in C:\Program Files\We

Ready

Figure 13. Result view of WebLOAD

Systems requirements for WebLOAD are givent at Table 7.

Table 7. Systems Requirements for WebLOAD

Hardware Specifications for WebLOAD Console:

Pentium III 800 MHz and above

512MB RAM minimum, 1GB recommended

Hardware Specifications for Load Machine(s):

Pentium III 800 MHz and above, Pentium 4 recommended;

For Linux- any supportedx86-32 based processor

Software Prerequisites

Microsoft Windows 2000 Professional/Server, XP Pro, Server 2003;

Red Hat Enterprise Linux 3,4,5;

Fedora Core 3, 6

Functionality

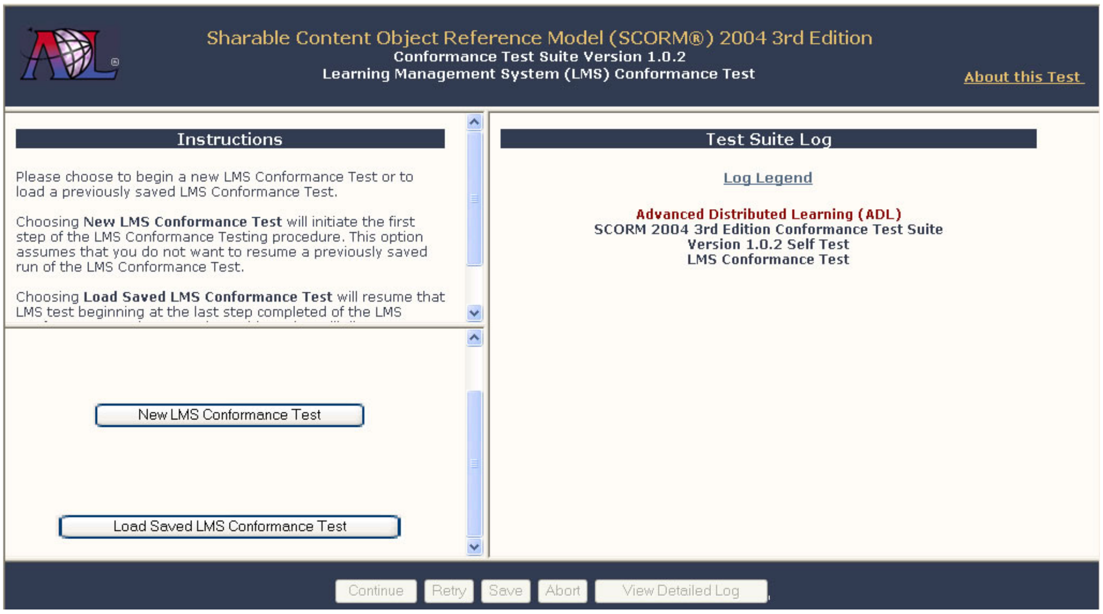
Functional testing involves ensuring that the functionality specified in the requirement specification works by determining whether or not a program does what it is supposed to do based on its functional requirements. Functional software testing verifies that the application supplies what the users need and also verifies that the systems work correctly from the perspective of user or business.

The function test must determine if each component or business event: performs in accordance to the specifications, responds correctly to all conditions that may be presented by incoming events / data, moves data correctly from one business event to the next (including data stores), and that business events are initiated in the order required to meet the business objectives of the system.

Standards Compliance Test

The results of evaluating the e-learning systems with SCORM Conformance test suite show up not only the strengths but also the inadequacies of existing products. SCORM Conformance test also contains the conformance testing software, procedures and supporting documents for organizations to perform self-testing on LMSs, SCOs and Content Packages (http://www.adlnet.gov/scorm).

ADL SCORM® 2004 3rd Edition Conformance Test Suite (Figure 14) can be used to test SCORM compliance of a LMS, a Sharable Content Object (SCO), or a Content Package. Beginning with SCORM 2004, LMSs must pass all of the tests to be considered conformant. The SCORM Conformance Test Suite used by the certification centers, which is also available for download free of charge from ADLNet.gov for self-testing. Only content packages and LMSs can be certified as SCORM conformant (not tools or organizations) (ADL, 2008).



Sharable Content Object Reference Model (SCORM®) 2004 3rd Edition
Conformance Test Suite Version 1.0.2
Learning Management System (LMS) Conformance Test

Instructions

Please choose to begin a new LMS Conformance Test or to load a previously saved LMS Conformance Test.

Choosing **New LMS Conformance Test** will initiate the first step of the LMS Conformance Testing procedure. This option assumes that you do not want to resume a previously saved run of the LMS Conformance Test.

Choosing **Load Saved LMS Conformance Test** will resume that LMS test beginning at the last step completed of the LMS

Test Suite Log

Advanced Distributed Learning (ADL)
SCORM 2004 3rd Edition Conformance Test Suite
Version 1.0.2 Self Test
LMS Conformance Test

Figure 14. Sample screenshot of SCORM conformance test suite

The LMS shall adhere to the conformance requirements defined for the following Conformance Categories; LMS Run-Time Environment Version 1.0 (LMS RTE 1.0), LMS Content Aggregation Model Version 1.0 (LMS CAM 1.0), LMS Sequencing and Navigation Version 1.0 (LMS SN 1.0). Test suits also can be used to decide whether a SCO is SCO Run-Time Environment Version 1.0 compliant. For a content package, conformance test suit can evaluate if content package implements the conformance requirements defined for Content Package Content Aggregation Model Version 1.0 (CP CAM 1.0), and Content Package Run-Time Environment Version 1.0 (CP RTE 1.0) (http://www.adlnet.gov/scorm).

CHAPTER III

METHODOLOGY

In this study, quantitative data is aimed to be formed in order to evaluate the systems. Accessibility, performance, security, standard compliance criteria and functionality criteria are chosen to evaluate the systems' success. However, accessibility, performance, security, standard compliance and functionality evaluation results cannot be definite indicators of the e-learning systems. Effects of human factor and other criteria like usability, compatibility, maintainability and modularity cannot be ignored while assessing whole e-learning systems' success, quality and efficiency.

LMS application helps to automate many of the e-learning processes, and LMS is more than just the administrative part of an e-learning deployment. Therefore, selection of an LMS is critical for both sides of learners and businesses. In the study, LMSs were chosen for assessing e-learning systems' success and efficiency by taking into consideration all aspects of organizational learning and benefits for all users. Moreover, open source and non-commercial LMSs were preferred in the study since these software are free to run, to study and to modify.

In the study, six LMSs were selected, which were Moodle, Ilias, Dokeos, Docebo, Claroline and Efront. The versions of LMSs are Moodle version 1.9, Ilias version 3.8.6, Dokeos version 3.5.0.4, Docebo version 1.8.4, Claroline version 1.8.9 and Efront version 3.1.3.These LMSs were tested according to previously defined criteria, and the functionality of these e-learning platforms was evaluated and compared. Furthermore, these e-learning systems were set up on a computer, which simulates as a web-server. The general feature of the computer was 1.6 GHz Intel Core 2 Duo processor, 1 GB RAM, and Microsoft Windows XP Professional operating system. All the LMSs in the study were also based on PHP scripting language and MySQL database. For web and database server, Apache Friends XAMPP (basic package) version 1.6.6a was installed on the server with Apache 2.2.8, MySQL 5.0.51a and PHP 5.2.5. Moreover, all tests were executed in LAN environment with 100Mbits infrastructure.

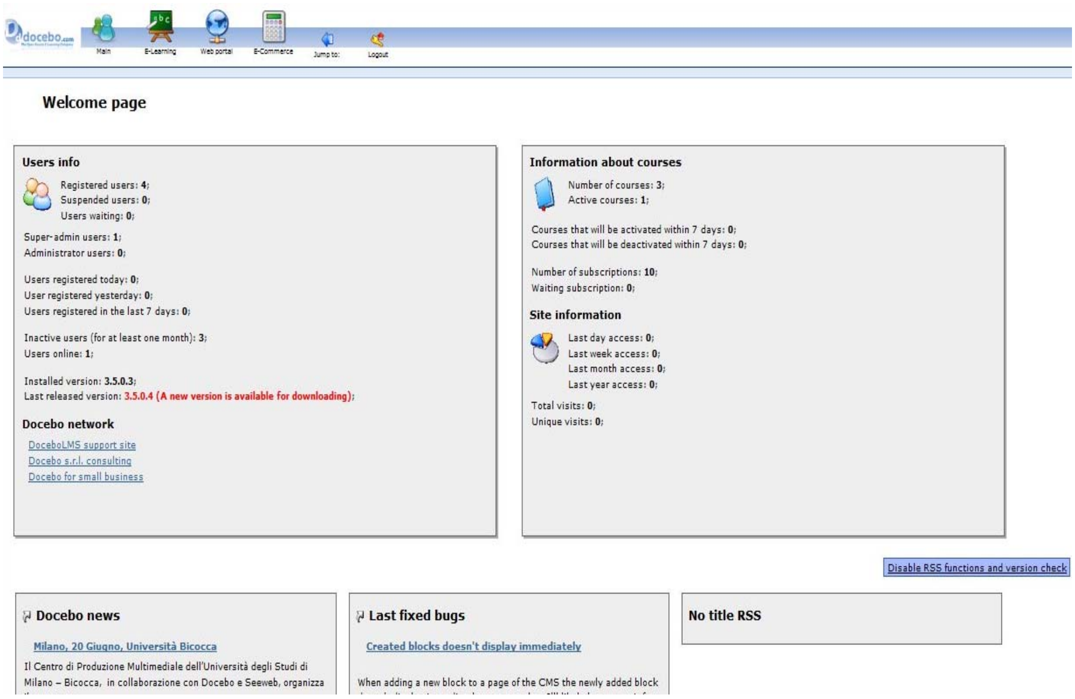
According to Bell (2006), software testing is a necessity to help attain any desired level of software quality. The whole testing process can be reduced with automated software testing since a test case that would have cost hours for manual testing to complete can be run in several minutes by automated testing. Therefore, accessibility, performance, security and standard compliance criteria were tested with automated software.

Learning Management Systems Used in the Study

Docebo

Docebo suite is Open Source and free software. Docebo suite is also completely free content management (CMS) and learning management systems (LMS) platform (Docebo, n.d.). Moreover, general information about user, site and courses can be viewed at Figure 15.

At the administration part of Docebo suite, there are main, e-learning, web portal and e-commerce menus, and choice of CMS and LMS options (Figure 16.). Users and group managements can be done at administrator part. Docebo suite includes wide variety of activities such as function choose to activate, reports, creation of groups and roles, user creation and sub-admins creation.



Welcome page

Users info

Registered users: 4;
Suspended users: 0;
Users waiting: 0;

Super-admin users: 1;
Administrator users: 0;

Users registered today: 0;
User registered yesterday: 0;
Users registered in the last 7 days: 0;

Inactive users (for at least one month): 3;
Users online: 1;

Installed version: 3.5.0.3;
Last released version: 3.5.0.4 (A new version is available for downloading);

Docebo network

DoceboLMS support site
Docebo s.r.l. consulting
Docebo for small business

Information about courses

Number of courses: 3;
Active courses: 1;

Courses that will be activated within 7 days: 0;
Courses that will be deactivated within 7 days: 0;

Number of subscriptions: 10;
Waiting subscription: 0;

Site information

Last day access: 0;
Last week access: 0;
Last month access: 0;
Last year access: 0;

Total visits: 0;
Unique visits: 0;

Disable RSS functions and version check

Docebo news

Milano, 20 Giugno, Università Bicocca

Il Centro di Produzione Multimediale dell'Università degli Studi di Milano – Bicocca, in collaborazione con Docebo e Seeweb, organizza

Last fixed bugs

Created blocks doesn't display immediately

When adding a new block to a page of the CMS the newly added block

No title RSS

Figure 15. Screenshot of welcome page of Docebo suite

Both e-learning lessons and users can be organized in a tree. Users can be created in a group. Groups can be distinguished or associated to applications of visible only to the administrator, visible also the user that can be auto-subscribe to it, visible to the user that can be auto subscribed but require admins approval and visible to the user but only the admin can subscribe. Not only e-learning user but also e-learning course subscription can be free or moderated. Different functions inside the learning management system page can be inserted by customizing languages via web interface. Notification via SMS is possible. There is also privacy and security management in order to offer better security. Report by user, groups or tree is also possible.

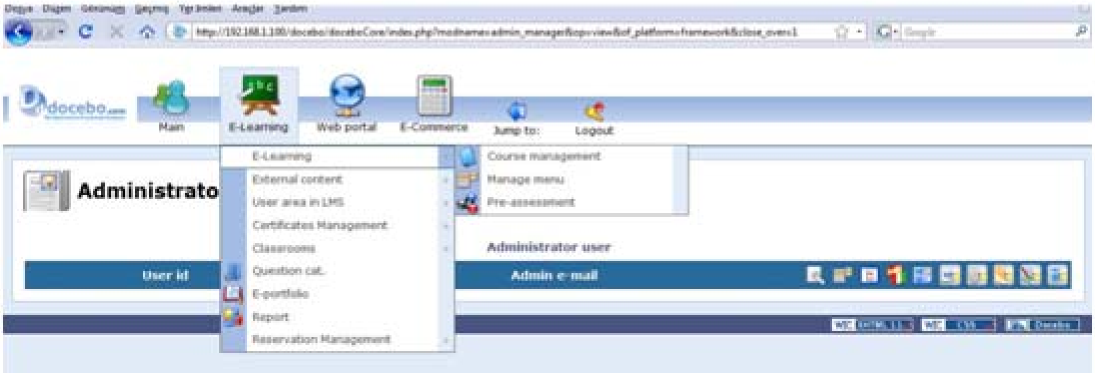
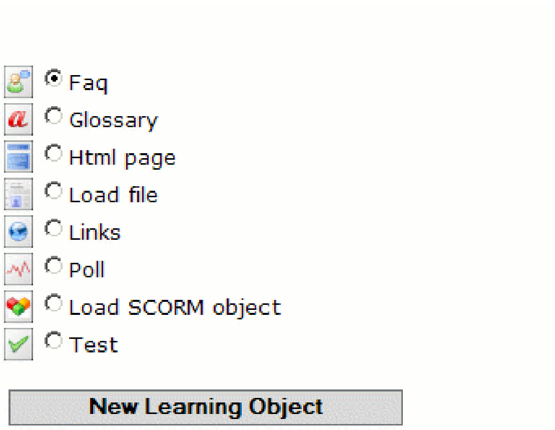


Figure 16. Screenshot of administration panel of Docebo suite

Docebo suite includes functionalities like chat, accessible chat version, videoconference, forum, course presentation page, Poll, FAQ, help upload files, messages, pre-requisites on learning object, report by learning object or user and test result table. Moreover, e-learning classroom can be divided in groups and learning objects can be stored and organized in a tree. Learning objects are all object that the students can use the Docebo LMS platform support. Docebo LMS platform supports learning object of FAQ, glossary, HTML page, file (upload), link list, poll/survey, SCORM objects 1.2 (upload) and test as it can be seen at Figure 17.



● Faq
○ Glossary
○ Html page
○ Load file
○ Links
○ Poll
○ Load SCORM object
○ Test

New Learning Object

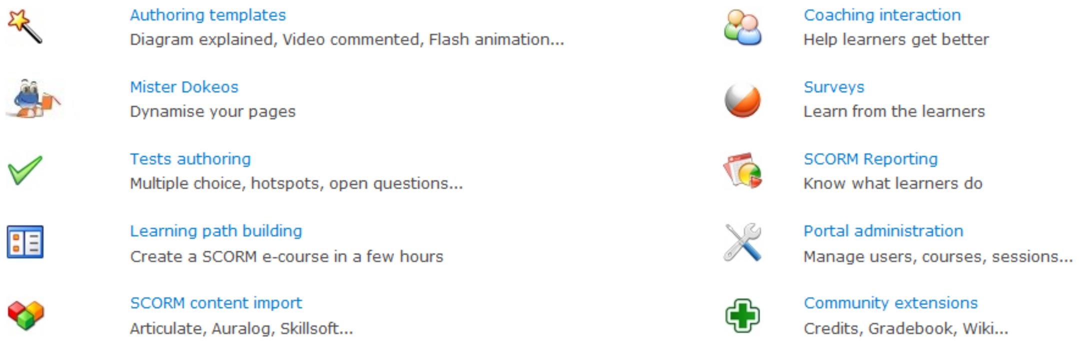
Figure 17. Screenshot of creating learning object at Docebo suite

At user profile options, user can edit his/her personal information, change his password and the platform template and edit some other information such as the mobile phone number, the birth date, etc...

Furthermore, test function at Docebo suite supports seven kinds of answers, which are single, multiple and open answers, choose right/wrong word, association, fill the blank and upload file. General statistic function is a summary that analyzes the whole statistics by the parameters of most used browser and operating system, access by country, most active crawler/robots, most visited pages and most used search engine keywords. In addition, category creation and permission assign to levels and groups are possible at forums. Forum functions of Docebo platform also supports upload a file into a thread, edit the replies, search discussion, notify via e-mail or SMS that a new discussion is started or reply to a discussion. Internal message system functions are used to share files and messages in every single course. Docebo suite is also supports project management system, which allows groups to work everybody on certain themes. Additionally, Docebo is interfaced with the videoconference system by supporting audio video conferencing, text chatroom and PowerPoint slideshow pre-loaded inside the chatroom.

Dokeos

Dokeos platform is Open Source professional learning suit. Dokeos provides learning management, Oogie rapid learning, accurate reporting and videoconferencing. Oogie Rapid Learning builds SCORM courses online from templates, from PowerPoint. Oogie Rapid Learning also inserts tests between slides and record audio. Moreover, accurate reporting provides user to export to excel or business objects. Furthermore, videoconferencing is virtual meeting and virtual classroom for live training sessions (Dokoes, n.d.).



Authoring templates
Diagram explained, Video commented, Flash animation...

Mister Dokeos
Dynamise your pages

Tests authoring
Multiple choice, hotspots, open questions...

Learning path building
Create a SCORM e-course in a few hours

SCORM content import
Articulate, Auralog, Skillsoft...

Coaching interaction
Help learners get better

Surveys
Learn from the learners

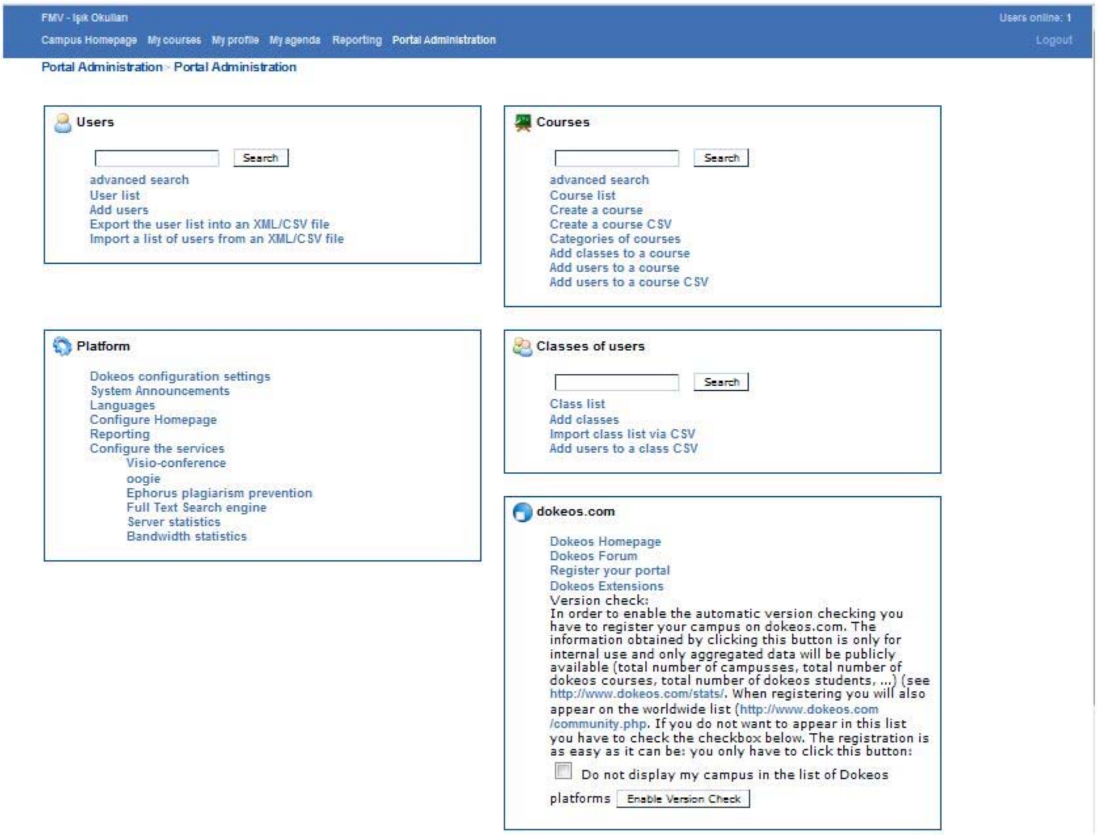
SCORM Reporting
Know what learners do

Portal administration
Manage users, courses, sessions...

Community extensions
Credits, Gradebook, Wiki...

Figure 18. Screenshot of learning management options at Dokeos suite

Learning management includes SCORM import, edit and export,social interaction, sophisticated test and surveys as it can be seen at Figure 18. Test function at Dokeos learning suite includes multiple choices, fill-in the blanks, matching, hotspots, open question and questions database. By the help of learning help builder creating a learning path, add content, tests, activities can be done through getting automatic SCORM sequencing, navigation and reporting. Moreover, coaching interaction menu includes interaction with learners through agenda, forums, chat, videoconference, open questions answers and assignment feedbacks. At survey menu, getting feedback from audience in both a qualitative and a quantitative way is possible. Display the results can be both in a graphical reporting and exporting into Excel format. Furthermore, at portal administration menu, managing users, courses and sessions can be done. Credits (pay-per-view), gradebook, wiki, reservations, stylesheets and icon sets functions exist at community extensions menu (Dokoes, n.d.).



Users

advanced search
User list
Add users
Export the user list into an XML/CSV file
Import a list of users from an XML/CSV file

Courses

advanced search
Course list
Create a course
Create a course CSV
Categories of courses
Add classes to a course
Add users to a course
Add users to a course CSV

Platform

Dokeos configuration settings
System Announcements
Languages
Configure Homepage
Reporting
Configure the services
Visio-conference
oogie
Ephorus plagiarism prevention
Full Text Search engine
Server statistics
Bandwidth statistics

Classes of users

Class list
Add classes
Import class list via CSV
Add users to a class CSV

dokeos.com

Dokeos Homepage
Dokeos Forum
Register your portal
Dokeos Extensions
Version check:
In order to enable the automatic version checking you have to register your campus on dokeos.com. The information obtained by clicking this button is only for internal use and only aggregated data will be publicly available (total number of campusses, total number of dokeos courses, total number of dokeos students, ...) (see http://www.dokeos.com/stats/. When registering you will also appear on the worldwide list (http://www.dokeos.com/community.php. If you do not want to appear in this list you have to check the checkbox below. The registration is as easy as it can be: you only have to click this button:

□ Do not display my campus in the list of Dokeos platforms

Figure 19. Screenshot of portal administration at Dokeos suite

By the help of user menu, user can be added easily by the options of creating user lists. User can be search through the LMS. Exporting and importing user list into XML/CSV file is also possible. Courses are also created easily by creating course list and categories. Users or classes can be added to the courses. Classes of users can be also import to CSV format, and users can also be added to class CSV. User, courses and classes can be search through the LMS. All these options are in administrator menu (Figure 19).

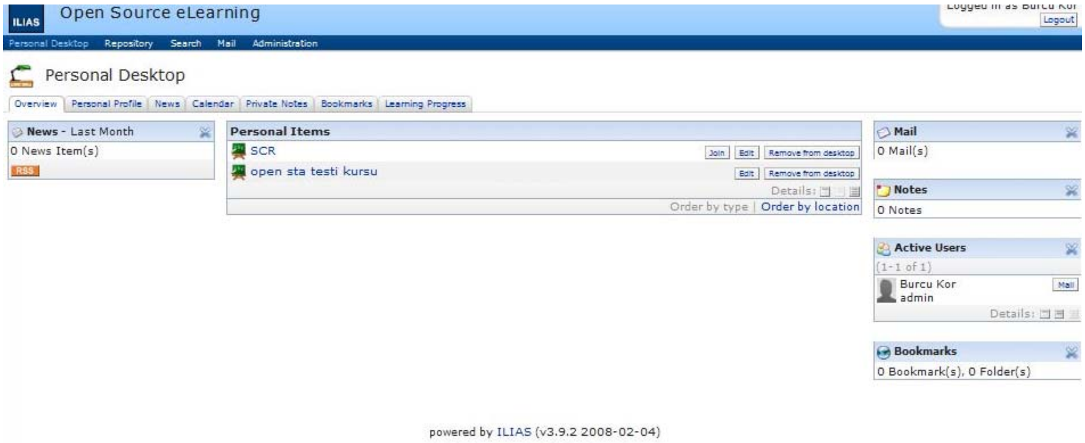
Apart from these functionalities, reporting, forum, calendar, announcements, multimedia, FAQ, links, e-mail notification, syllabus and file sharing are supported in Dokeos platform.

<u>Ilias</u>

Ilias 3.9.4 learning management system is Open Source with SCORM 2004 (3rd Edition) compliance (ILIAS Open Source LMS, n.d.). Ilias features are listed in the web site are

> "individual personal desktop, course management, group management, repository with role based access control, learning content (XML, SCORM, AICC), standards compliance (LOM, SCORM 1.2, SCORM 2004, IMS-QTI, AICC), SCORM 1.2 RTE Level 3 Certified, SCORM 2004 3rd Edition Certified, learning progress management, test & assessment, survey, chat, forums, exercises, RSS support, podcasting, google maps support, authentication (LDAP, Shibboleth, CAS, Radius, Soap), web service interface (Soap)."

Ilias provides users personal desktop (Figure 20). All resources that are needed by the learning to fulfill the daily learning tasks are collected. The personal desktop has features of news, personal messages, learning resources, personal notes, bookmarks, external web feeds and other information. The learner can re-arrange these blocks of information according to his needs.



Personal Desktop

News - Last Month
0 News Item(s)

Personal Items
SCR
open sta testi kursu

Mail
0 Mail(s)

Notes
0 Notes

Active Users
(1-1 of 1)
Burcu Kor
admin

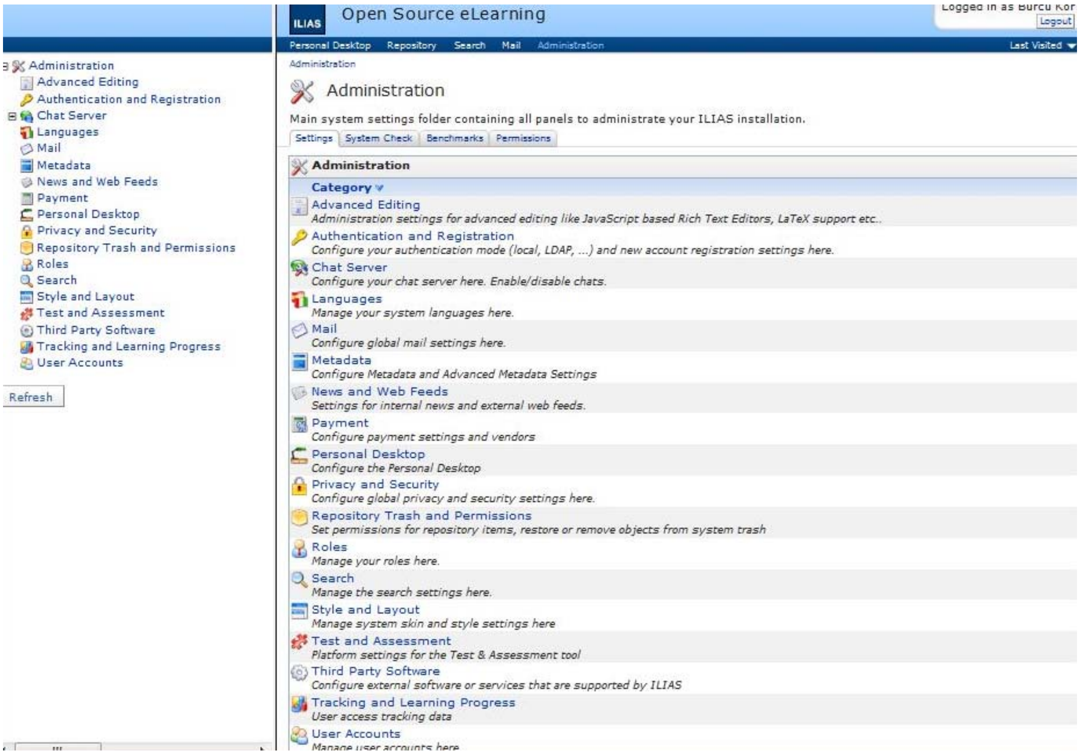
Bookmarks
0 Bookmark(s), 0 Folder(s)

powered by ILIAS (v3.9.2 2008-02-04)

Figure 20. Screenshot of personal desktop at Ilias suite

Ilias allows efficient creation of courses and course materials. Arbitrary learning modules of the repository can be arranged to courses with forums, chats, groups and other objects. Different entrance rules and current times are supported.

The test and assessment functions at Ilias support multiple choice, single choice, allocation questions, close questions (free text, select box), arrangements duties, hot spot (search images to click on) and unsettled questions types of questions. Moreover, administration of system includes system settings, language settings (over 20 languages available), enable modules, partial modules and tools, construction and configuration of clients, administration / import of users and courses, role based access control of repository objects, creation and modification of role templates (global roles, local roles), and administration of categories (structuring in form or content) (ILIAS Information Center, 2008). Administration of system can be seen at Figure 21.



Administration

Main system settings folder containing all panels to administrate your ILIAS installation.

Administration

Advanced Editing
Administration settings for advanced editing like JavaScript based Rich Text Editors, LaTeX support etc..

Authentication and Registration
Configure your authentication mode (local, LDAP, ...) and new account registration settings here.

Chat Server
Configure your chat server here. Enable/disable chats.

Languages
Manage your system languages here.

Mail
Configure global mail settings here.

Metadata
Configure Metadata and Advanced Metadata Settings

News and Web Feeds
Settings for internal news and external web feeds.

Payment
Configure payment settings and vendors

Personal Desktop
Configure the Personal Desktop

Privacy and Security
Configure global privacy and security settings here.

Repository Trash and Permissions
Set permissions for repository items, restore or remove objects from system trash

Roles
Manage your roles here.

Search
Manage the search settings here.

Style and Layout
Manage system skin and style settings here

Test and Assessment
Platform settings for the Test & Assessment tool

Third Party Software
Configure external software or services that are supported by ILIAS

Tracking and Learning Progress
User access tracking data

User Accounts
Manage user accounts here

Figure 21. Screenshot of administration module at Ilias suite

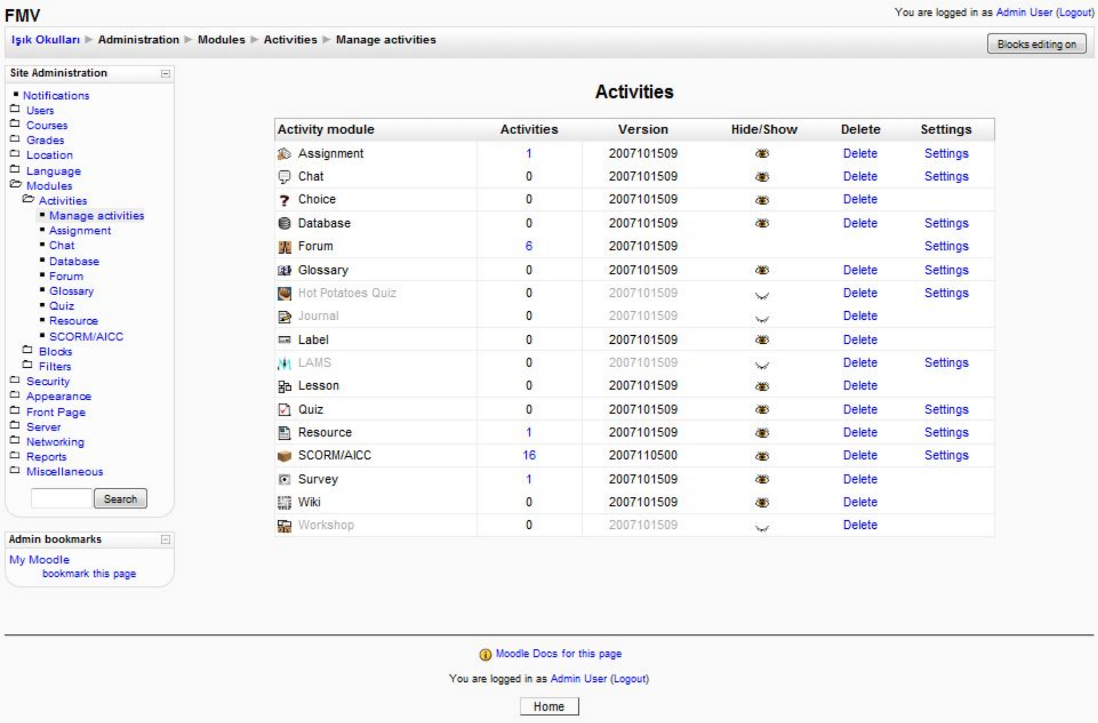
The simple search option includes some criteria of objects, which are one or more search terms, search area containing either any objects available to user and search type that can be restricted from any object types (default setting) to single ones.

Within the advanced search, titles, descriptions and keywords can be specified (ILIAS Information Center, 2008).

Moodle

Moodle (short for “Modular Object-Oriented Dynamic Learning Environment) is a free and Open Source software packages to produce internet-based courses and web sites by creating also effective online learning communities (Moodle - A Free, Open Source Course Management System for Online Learning, n.d.).

Moodle platform contains functions such as assignment-module for handing-in the assignments (including due-date and evaluation- and feedback functions for the teacher), a complete messenger-system to exchange messages as well as chats within a survey- or poll-module, the mandatory forums, glossaries, quizzes, news and announcements, workshop-tools, a learning-diary (comparable to a weblog), and relatively sophisticated test module for the integration of own interactive tests and learning content. Moodle is so flexible that the system can work with other e-learning and open-source tools like e.g. Hot Potatoes or ErfurtWiki (Moodle - A Free, Open Source Course Management System for Online Learning, n.d.).

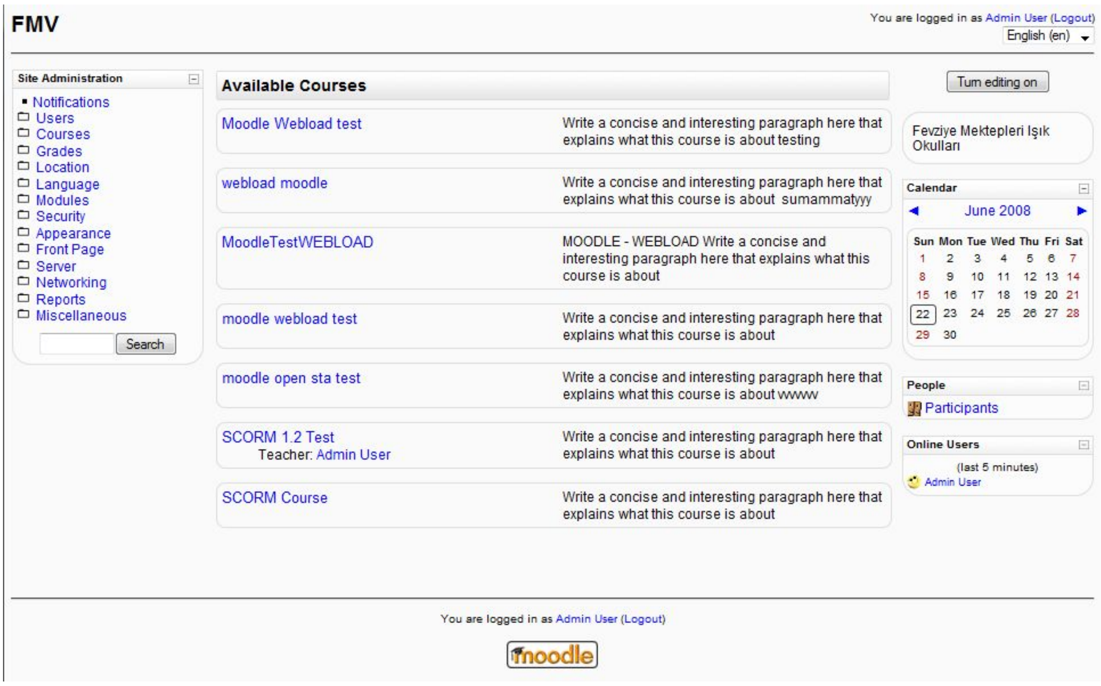


Activities

Activity module	Activities	Version	Hide/Show	Delete	Settings
Assignment	1	2007101509	👁	Delete	Settings
Chat	0	2007101509	👁	Delete	Settings
Choice	0	2007101509	👁	Delete	
Database	0	2007101509	👁	Delete	Settings
Forum	6	2007101509			Settings
Glossary	0	2007101509	👁	Delete	Settings
Hot Potatoes Quiz	0	2007101509	◡	Delete	Settings
Journal	0	2007101509	◡	Delete	
Label	0	2007101509	👁	Delete	
LAMS	0	2007101509	◡	Delete	Settings
Lesson	0	2007101509	👁	Delete	
Quiz	0	2007101509	👁	Delete	Settings
Resource	1	2007101509	👁	Delete	Settings
SCORM/AICC	16	2007110500	👁	Delete	Settings
Survey	1	2007101509	👁	Delete	
Wiki	0	2007101509	👁	Delete	
Workshop	0	2007101509	◡	Delete	

Figure 22. Screenshot of teacher activities at Moodle suite

Teachers' activities include functionalities of assignments, blogs, books, chats, forums, lessons, surveys, questionnaire, quizzes, wiki and workshops (Figure 22). Blogs are a form of online journal. Moreover, blogs in Moodle are user based - each user has their own Blog. Admins can create site level tags, teachers can create Course level tags, and students can create their own list of tags (Moodle - A Free, Open Source Course Management System for Online Learning, n.d.). The assignment module allows teachers to collect work from students, review it and provide feedback including grades. Additionally, both teacher and students can submit any digital content or files, including, for example, word-processed documents, spreadsheets, images, audio and video clips. A real-time synchronous discussion via the web is provided with chat activities. Lesson is a series of interactive pages that include questions, answers, responses, grading, teacher and student review and building lesson options. Furthermore, the quiz activity module allows the teacher to design and set quizzes consisting of a large variety of question types, among them multiple choice, true-false, and short answer questions, which may chosen form question banks (Moodle - A Free, Open Source Course Management System for Online Learning, n.d.). Screenshot of teacher activities is at Figure X and screenshot of site administration module is at Figure 23.



Available Courses

Moodle Webload test	Write a concise and interesting paragraph here that explains what this course is about testing
webload moodle	Write a concise and interesting paragraph here that explains what this course is about sumammatyyy
MoodleTestWEBLOAD	MOODLE - WEBLOAD Write a concise and interesting paragraph here that explains what this course is about
moodle webload test	Write a concise and interesting paragraph here that explains what this course is about
moodle open sta test	Write a concise and interesting paragraph here that explains what this course is about wwww
SCORM 1.2 Test Teacher: Admin User	Write a concise and interesting paragraph here that explains what this course is about
SCORM Course	Write a concise and interesting paragraph here that explains what this course is about

Fevziye Mektepleri Işık Okulları

◄ June 2008 ►

Sun	Mon	Tue	Wed	Thu	Fri	Sat
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30					

(last 5 minutes)
Admin User

You are logged in as Admin User (Logout)

Figure 23. Screenshot of site administration module at Moodle suite

Claroline

Claroline is Open Source learning platform. Claroline platform helps to build effective online courses and to manage learning and collaborative activities on the web by translating into 35 languages (Claroline .NET, 2008). The system primarily offers common functionalities for forums, administration of documents, online-test, question pools, assignment, calendar, announcement, multimedia, wiki, links, syllabus, file sharing, chat, learning-path (a tool, to show the users the planned procedure between documents, tests, HTML-pages, links, etc.), an upload area (in principle a so called dropbox; user can upload files to the system or hand in to the tutor), report and statistics (Figure 24). The statistics options are in both side of courses and systems. Administration of documents can be seen at Figure 25.

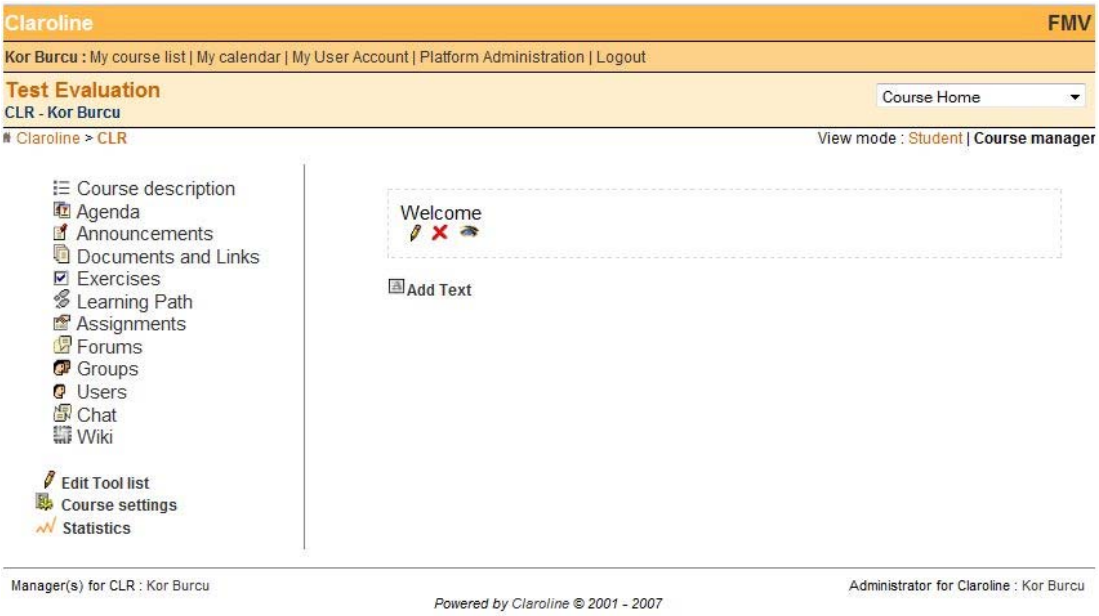
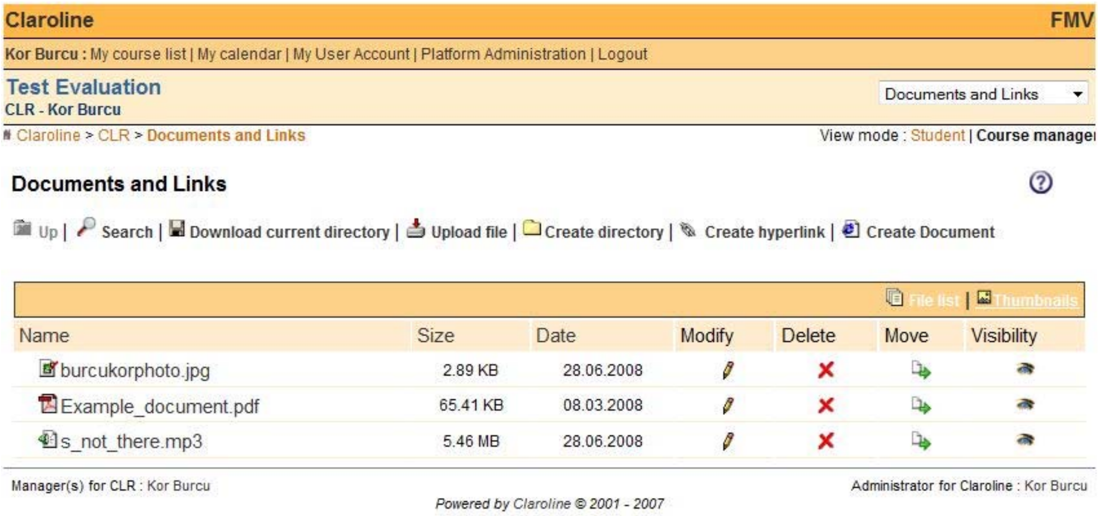


Figure 24. Screenshot of user entrance page module at Claroline suite


Claroline
FMV
Kor Burcu : My course list | My calendar | My User Account | Platform Administration | Logout
Test Evaluation
CLR - Kor Burcu
Documents and Links
Claroline > CLR > Documents and Links
View mode : Student | Course manager
Documents and Links
Up | Search | Download current directory | Upload file | Create directory | Create hyperlink | Create Document
File list | Thumbnails

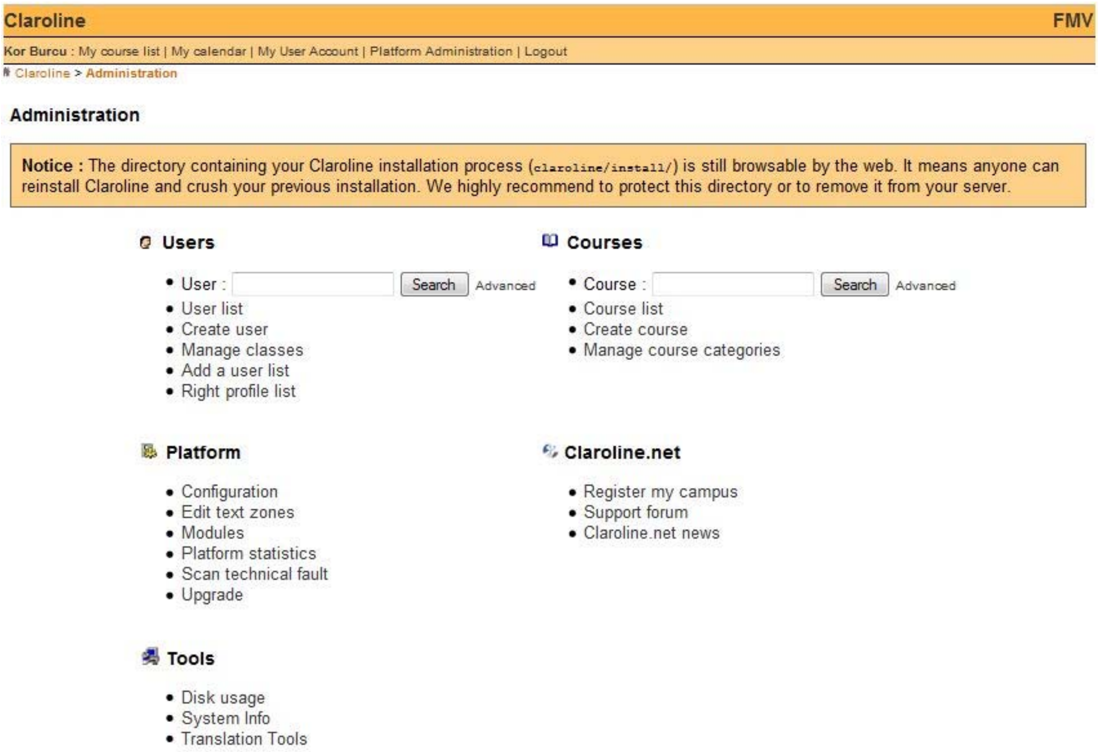
Name	Size	Date	Modify	Delete	Move	Visibility
burcukorphoto.jpg	2.89 KB	28.06.2008	✎	✗	➜	👁
Example_document.pdf	65.41 KB	08.03.2008	✎	✗	➜	👁
s_not_there.mp3	5.46 MB	28.06.2008	✎	✗	➜	👁

Manager(s) for CLR : Kor Burcu
Administrator for Claroline : Kor Burcu
Powered by Claroline © 2001 - 2007

Figure 25. Screenshot of documents and links at Claroline suite

Online-tests offer multiple choice. At the administration part, separate groups can be created. Moreover, the registration process is also simple. For the registration entering name, user-id and password is all that is needed.

Additionally, by the help of user menu, user can be added easily by the options of adding user lists. User and user list can be search through the LMS. Managing classes is also possible. Courses are also created easily by creating course list and categories. Courses are also searched through the LMS. By the help of the platform menu platform configuration and statistic can be seen easily. Information about the system and disk usage can be followed from tools menu. The screenshot of administrator menu is at Figure 26.



Claroline FMV

Kor Burcu : My course list | My calendar | My User Account | Platform Administration | Logout

Claroline > Administration

Administration

Notice : The directory containing your Claroline installation process (`claroline/install/`) is still browsable by the web. It means anyone can reinstall Claroline and crush your previous installation. We highly recommend to protect this directory or to remove it from your server.

Users

- User : Search Advanced
- User list
- Create user
- Manage classes
- Add a user list
- Right profile list

Courses

- Course : Search Advanced
- Course list
- Create course
- Manage course categories

Platform

- Configuration
- Edit text zones
- Modules
- Platform statistics
- Scan technical fault
- Upgrade

Claroline.net

- Register my campus
- Support forum
- Claroline.net news

Tools

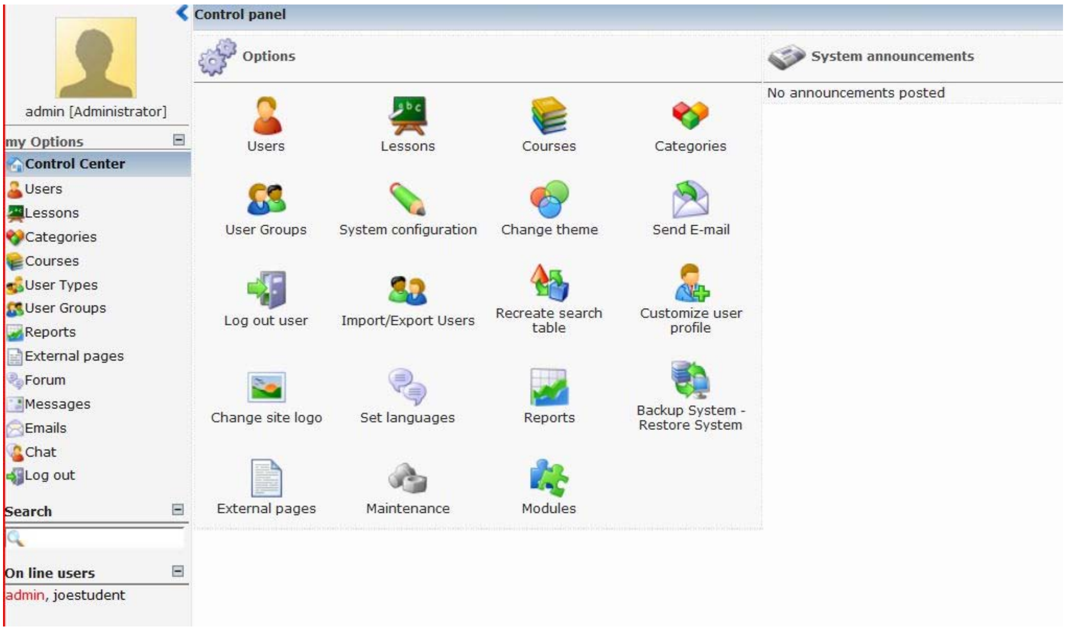
- Disk usage
- System Info
- Translation Tools

Figure 26. Screenshot of administrator menu at Claroline suite

Efront

Efront is an easy to use, visually attractive, SCORM compatible e-learning system, which is suitable for both company and educational usage (Efront, n.d.).

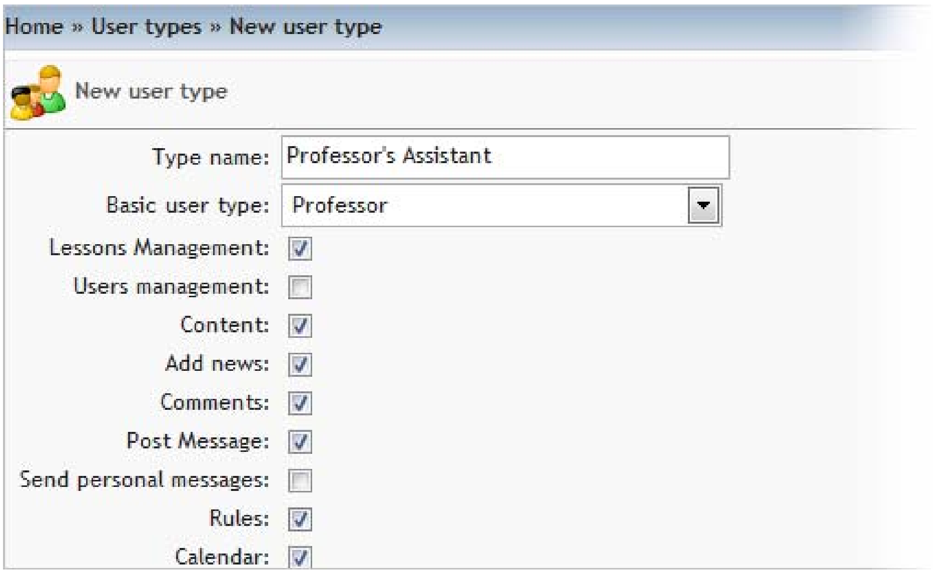
Efront includes a wide variety of functionalities such as create lesson structure and add content, build online-tests, communicate with others, track users' history and progress, conduct surveys, assign projects, glossary, search, e-mail, reports, external page creation, set language and create certifications (Figure 27).



No announcements posted

Figure 27. Screenshot of administrator menu at Efront suite

Both the user management and the course management can be done through the Administration's interface. User managements involve the management of users, the assignment of lessons/courses to users and the creation of new types of users by supporting three basic types of users, which are student, professor and administrator (Figure 28).


Home » User types » New user type

New user type

Type name: Professor's Assistant

Basic user type: Professor

Lessons Management: ☑

Users management: □

Content: ☑

Add news: ☑

Comments: ☑

Post Message: ☑

Send personal messages: □

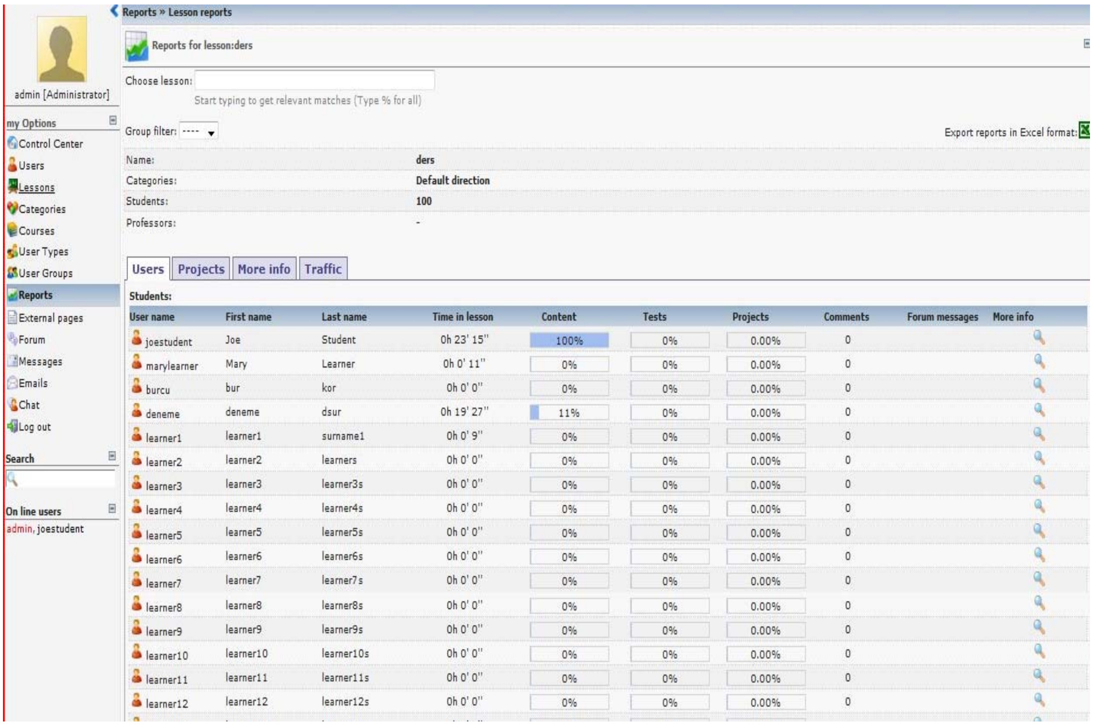
Rules: ☑

Calendar: ☑

Figure 28. Screenshot of user management at Efront suite

The course management involves the management of lessons, the logical organization of lessons to categories, the bundling of several lessons as a course, and the assignments of users to lessons and courses. Moreover, content can be created through adding images, videos, sounds, java applets and mathematical types in a visual way. By the help of the advanced file manager, user can upload, preview, share, zip, rename and delete files, and organize files to directories. Furthermore, Efront has an advanced internal test builder that supports several types of tests and a wide range of question types. The question types can be multiple choices, true/false, match and empty space. All questions are stored to a central pull and can be used in different tests.

Projects can be created with deadlines by assigning to users. The user must upload a file as answer to the project and the tutor can access it and give him a grade. Additionally, Efront has integrated support for surveys. A survey can be completed without having to enter Efront (through an external interface) (eFront Quick Guide, n.d). Efront also produces extensive reports on system access, generic system characteristics, users' progress and lesson information. Reports can be viewed in six types, which are user, tests, questions, more info and traffic. More info provides some general info on the lesson. Traffic defines a specific period of time to watch the lesson statistic information (Figure 29).



Reports for lesson:ders

Choose lesson:

Start typing to get relevant matches (Type % for all)

Group filter: ----

Export reports in Excel format:

Name:	ders
Categories:	Default direction
Students:	100
Professors:	-

Students:

User name	First name	Last name	Time in lesson	Content	Tests	Projects	Comments	Forum messages	More info
joestudent	Joe	Student	0h 23' 15''	100%	0%	0.00%	0		
marylearner	Mary	Learner	0h 0' 11''	0%	0%	0.00%	0		
burcu	bur	kor	0h 0' 0''	0%	0%	0.00%	0		
deneme	deneme	dsur	0h 19' 27''	11%	0%	0.00%	0		
learner1	learner1	surname1	0h 0' 9''	0%	0%	0.00%	0		
learner2	learner2	learners	0h 0' 0''	0%	0%	0.00%	0		
learner3	learner3	learner3s	0h 0' 0''	0%	0%	0.00%	0		
learner4	learner4	learner4s	0h 0' 0''	0%	0%	0.00%	0		
learner5	learner5	learner5s	0h 0' 0''	0%	0%	0.00%	0		
learner6	learner6	learner6s	0h 0' 0''	0%	0%	0.00%	0		
learner7	learner7	learner7s	0h 0' 0''	0%	0%	0.00%	0		
learner8	learner8	learner8s	0h 0' 0''	0%	0%	0.00%	0		
learner9	learner9	learner9s	0h 0' 0''	0%	0%	0.00%	0		
learner10	learner10	learner10s	0h 0' 0''	0%	0%	0.00%	0		
learner11	learner11	learner11s	0h 0' 0''	0%	0%	0.00%	0		
learner12	learner12	learner12s	0h 0' 0''	0%	0%	0.00%	0		

Figure 29. Screenshot of report menu at Efront suite

Software Tests

Performance Test

Research by Kefalas, Retalis, Stamatis and Theodoros (2003) showed that performance is one of the quality assurance systems. The performance criteria of LMSs were assessed with OpenSTA testing tool. Realistic heavy loads simulating the activity of hundreds to thousands of virtual users can be generated. OpenSTA graphs both virtual user response times and resource utilization information from all Web Servers, Application Servers, Database Servers and Operating Platforms under test; so that precise performance measurements can be gathered during load tests and analysis on these measurements can be performed (OpenSTA, 2007). In addition to these, user can generate and edit script with OpenSTA testing tool.

For creating test script with OpenSTA firstly the option of selected browser was chosen, which supports browsers Internet Explorer 4, 5, and 6 versions and

Netscape version. Then recording step for creating the scenario of test script was started. During the whole test, one task group was used with one task.

At the configuration part of the test menu, there were options for task group, start, host, virtual user and tasks. At task group, the task group description could be given. The test task groups could start immediately scheduled and delayed with the choices of manually, after fixed time and on completion. Additionally, total number of virtual users for the task group could be entered with the options of timer and HTTP results.

At the monitoring part of the test menu, test status could be seen, while executing the test. After the test process finished, the results of test could be seen on the results tab of the test pane. The results were grouped with test configuration, test audit log, test error log, test summary snapshots, HTTP data list, HTTP monitored bytes/sec v elapsed time, HTTP response time vs Number of responses, HTTP errors vs HTTP request, HTTP errors vs HTTP elapsed time, HTTP responses vs Elapsed time, HTTP response time vs elapse time, HTTP active users vs elapsed time, timer list, timer values v active users and timer values v elapsed time. Moreover, the test results can also be exported to spreadsheet software.

In the thesis, scripts were created in the same way as at all LMSs. The process and steps of generating script were almost the same. At each e-learning system, user logged in the system, entered a course forum and replied a topic, then checked the calendar of the course if there was a new event. After that, user went to his/her profile and updated it and finally logged out from the system. Tasks were chosen from the functionalities that all tested LMSs have in common. Performance-load and stress tests were executed after generating the script. Total number of virtual users for the task group was assumed that 30 for performance testing of each system in same testing condition. Besides that, total number of virtual users for stress testing was between 70 and 100. 70 virtual users were preferred for Efront because the system was overloaded and failed to finish the stress test with 100 virtual users. After each system test, the server and client machines were restarted in order torefresh the memory of the system. Additionally, antivirus and firewall of the systems were closed in order not to interfere with the test process.

In ideal systems, error percentage should be zero throughout the test run, and it is especially important in stress testing. If error percent is high, cause of the error should be analyzed. Types of error could be seen in the test result table. When the systems were compared on the base of error percentage, Dokeos, Docebo and Claroline were robust systems inasmuch as any errors were encountered during both performance and stress testing processes. Besides that, these systems not only had advantages on the number of errors but also had advantages on the number of failed request. Dokeos, Docebo and Claroline achieved successfully all of the requests. Additionally, in stress testing Dokeos had the maximum percentage of user, who finished all steps of script. Docebo had also a good score from finished user criteria in both performance and stress testing. Response time displayed in the HTTP data table is in seconds or milliseconds or minute while elapsed time describes how long the test has been running. Response time is how long, in seconds, a timed even took to complete. Another important criterion is average elapsed time for virtual user, as elapsed time signifies how long the test has been running in the specific script. When the performance and stress testing were examined, results in the aspect of number of errors, failed request, finished user and average elapsed time for virtual user, Dokeos, Docebo and Claroline show higher performance.

30 virtual users for load test and 100 virtual users for stress test were used to test the systems with OpenSTA performance testing tool. With given number of virtual users systems behaviors were given at following graphs. These graphs are HTTP Response Time / Elapsed Time which show the time to get a response from a HTTP request throughout the test run.

From Figure 30 to Figure 41, the graphs show HTTP Response Time / Elapsed Time for both load and stress tests of LMSs. These graphs signify the time to get a response from a HTTP request throughout the test run.

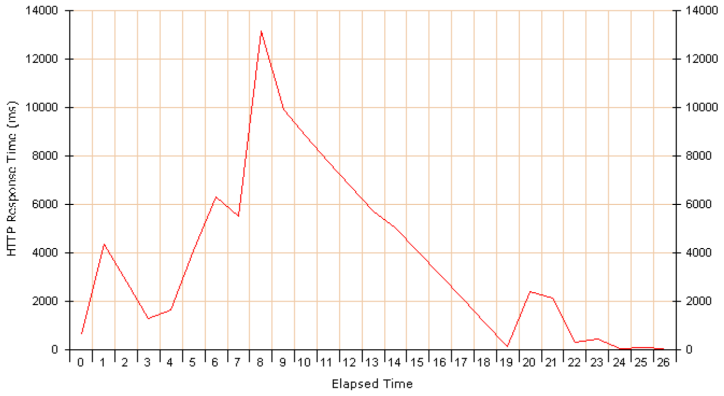


Figure 30. HTTP Response Time / Elapsed Time for Moodle for Load Test

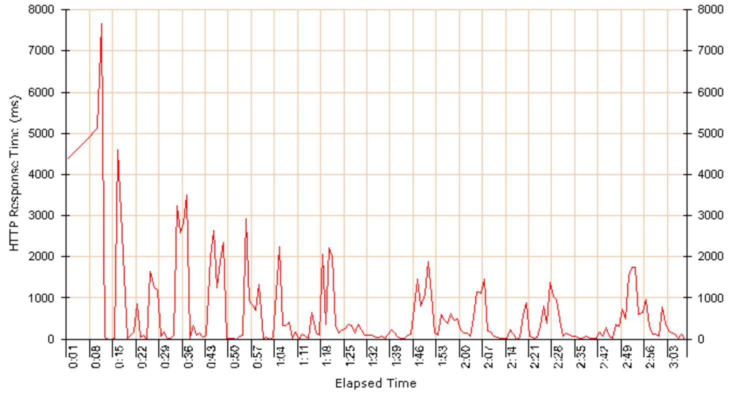


Figure 31. HTTP Response Time / Elapsed Time for Moodle for Stress Test

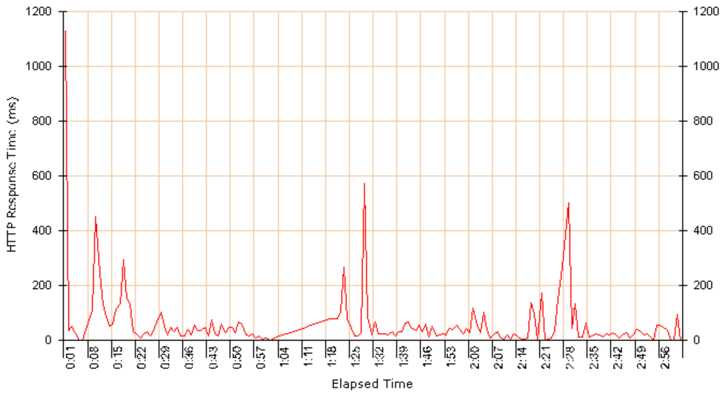


Figure 32. HTTP Response Time / Elapsed Time for Docebo for Load Test

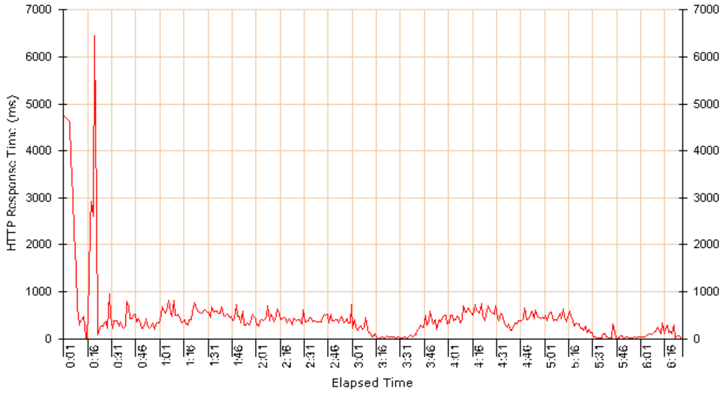


Figure 33. HTTP Response Time / Elapsed Time for Docebo for Stress Test

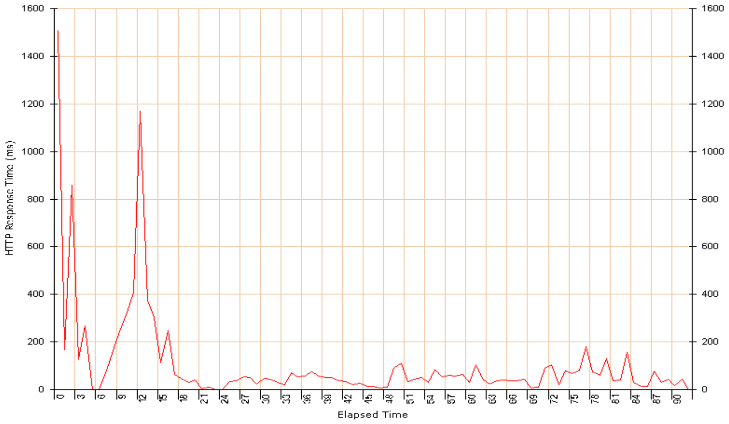


Figure 34. HTTP Response Time / Elapsed Time for Dokeos for Load Test

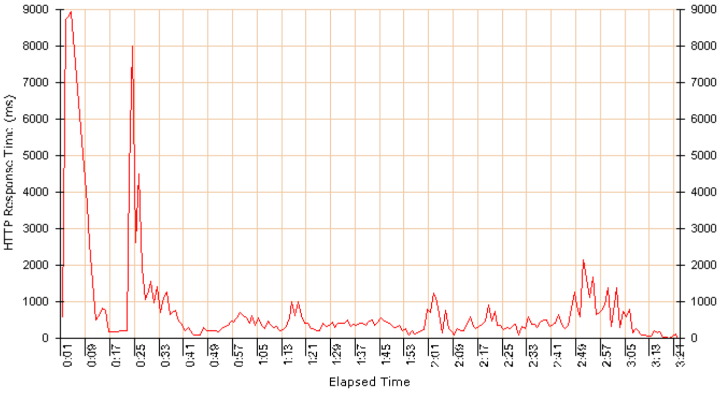


Figure 35. HTTP Response Time / Elapsed Time for Dokeos for Stress Test

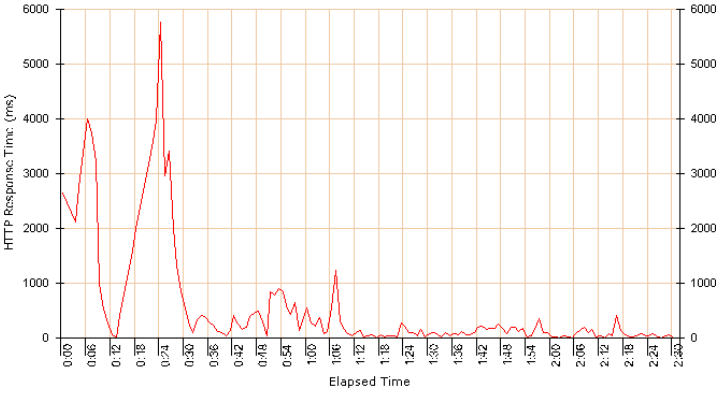


Figure 36. HTTP Response Time / Elapsed Time for Ilias for Load Test

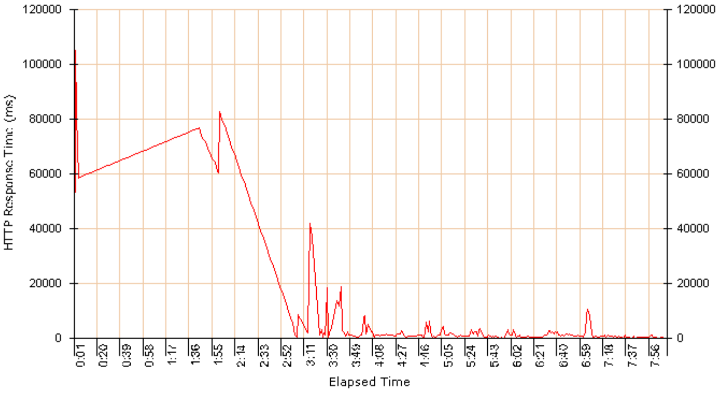


Figure 37. HTTP Response Time / Elapsed Time for Ilias for Stress Test

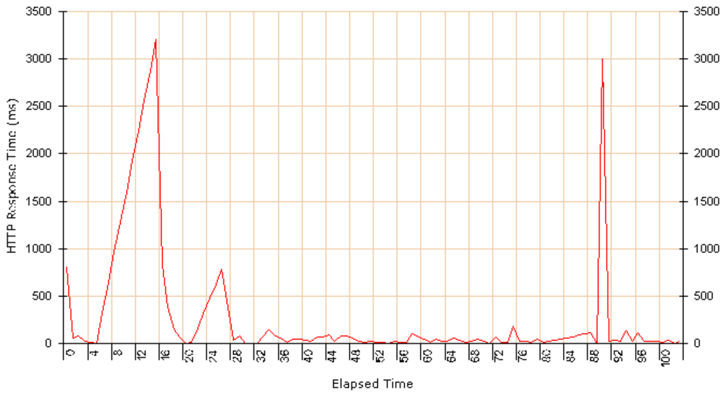


Figure 38. HTTP Response Time / Elapsed Time for Claroline for Load Test

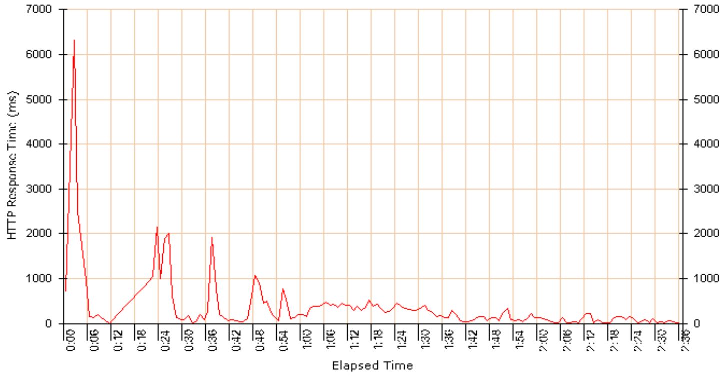


Figure 39. HTTP Response Time / Elapsed Time for Claroline for Stress Test

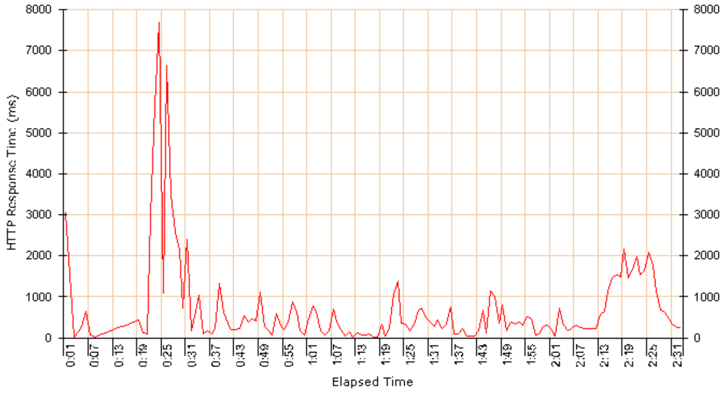


Figure 40. HTTP Response Time / Elapsed Time for Efront for Load Test

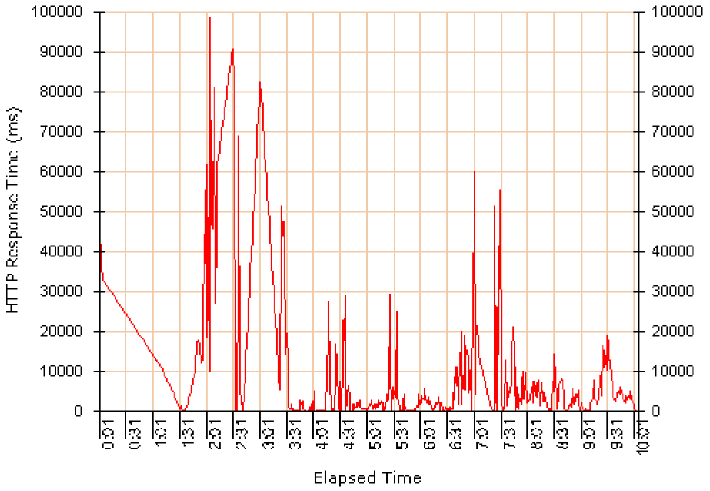


Figure 41. HTTP Response Time / Elapsed Time for Efront for Stress Test

The results of testing for each LMS are shown at Table 8 and Table 9. These tables included the results of average elapsed time for virtual users (sec), total script time, number of errors, average bytes responded from server (Kb), HTTP request, failed request, 3xx requests, 4xx request, 5xx request, finished user, timeout generated for socket, error 10038, error 10061, error 10060, error for virtual users: no data available for connection and IO failed for virtual users. Moreover, according to Hypertext Transfer Protocol -HTTP/1.1 source code definitions,

Redirection 3xx:

> This class of status code indicates that further action needs to be taken by the user agent in order to fulfill the request. The action required MAY be carried out by the user agent without interaction with the user if and only if the method used in the second request is GET or HEAD. A client SHOULD detect infinite redirection loops, since such loops generate network traffic for each redirection (Web Accessibility Initiative, 2008).

Client Error 4xx:

> The 4xx class of status code is intended for cases in which the client seems to have erred. Except when responding to a HEAD request, the server SHOULD include an entity containing an explanation of the error situation, and whether it is a temporary or permanent condition. These status codes are applicable to any request method. User agents

> SHOULD display any included entity to the user (Web Accessibility Initiative, 2008).

Server Error 5xx:

> Response status codes beginning with the digit ‘5’ indicate cases in which the server is aware that it has erred or is incapable of performing the request. Except when responding to a HEAD request, the server SHOULD include an entity containing an explanation of the error situation, and whether it is a temporary or permanent condition. User agents SHOULD display any included entity to the user. These response codes are applicable to any request method (Web Accessibility Initiative, 2008).

Additionally, according to OpenSTA portal,

10038 socket operation on nonsocket is:

> An operation was attempted on something that is not a socket. Either the socket handle parameter did not reference a valid socket, or for select, a member of an fd_set was not valid (OpenSTA Portal, n.d.).

10060 Connection timed out is:

> A connection attempt failed because the connected party did not properly respond after a period of time, or established connection failed because connected host has failed to respond(OpenSTA Portal, n.d.).

10061 Connection refused is:

> No connection could be made because the target machine actively refused it. This usually results from trying to connect to a service that is inactive o the foreign host - i.e. one with no server application running.” (OpenSTA Portal, n.d.).

Error for VU: No data available for connection is:

> The valid reason for no data available can be HTTP 304 return or the like; or an invalid one, like your Web server dropping the connection or having an error.
> cause: There is no returned data available to run the LOAD RESPONSE_INFO on.
> workaround: Check the HTTP CODE and STATUS to make sure you've actually got content before calling the LOAD RESPONSE_INFO.” (OpenSTA Portal, n.d.).

Table 8. Result of Performance/Load and Stress Testing-1

		Average elapsed Time for VU (sec)	Total Script Time	Number of errors	Average bytes responded from server (Kb)	HTTP request	Failed Request	3xx Request	4xx Request	5xx Request
ILIAS	30	69.18	02:31	0	290.91	9715	0	7337	0	0
	100	502.89	08:59	821	268.63	25758	241	19366	0	0
EFRONT	30	38.16	03:40	0	423.75	4263	0	3237	270	0
	70	503.45	10:39	370	56.04	6468	67	5309	46	0
DOCEBO	30	37.92	03:01	0	360.73	25839	0	24824	29	0
	100	337.86	06:26	0	261.08	73953	0	71008	83	0
DOKEOS	30	24.16	01:32	0	19.68	8112	0	7489	52	0
	100	141.64	03:21	0	260.29	30576	0	28220	195	0
CLAROLINE	30	71.90	01:43	0	14.60	6025	0	5700	0	0
	100	71.72	02:36	0	157.32	19521	0	18469	0	0
MOODLE	30	108.04	03:07	0	77.61	3497	0	3168	26	0
	100	2917.53	42:59	3617	68.88	21246	1267	19929	0	43

Table 9. Result of Performance/Load and Stress Testing-2

		Finished User	Timeout generated for socket	Error 10038	Error 10061	Error 10060	Error for VU: No data available for connection	IO failed for VU
ILIAS	30	29	0	0	0	0	0	0
	100	77	100	0	0	54	412	253
EFRONT	30	29	0	0	0	0	0	0
	70	44	155	0	0	0	57	156
DOCEBO	30	29	0	0	0	0	0	0
	100	83	0	0	0	0	0	0
DOKEOS	30	26	0	0	0	0	0	0
	100	98	0	0	0	0	0	0
CLAROLINE	30	25	0	0	0	0	0	0
	100	81	0	0	0	0	0	0
MOODLE	30	13	0	0	0	0	0	0
	100	79	1559	96	54	0	349	1559

The comparison of average elapsed time for virtual users' parameter for 30 virtual users is given at Figure 42. Average elapsed time for virtual users is important since elapsed time signifies how long the test has been running in the specific script. Average elapsed time for virtual user of Moodle platform was the highest score, whereas Dokeos had the least average elapsed time for 30 virtual users. Others e-learning platform average elapsed time for virtual users' scores order was Claroline, Ilias, Efront and Docebo.

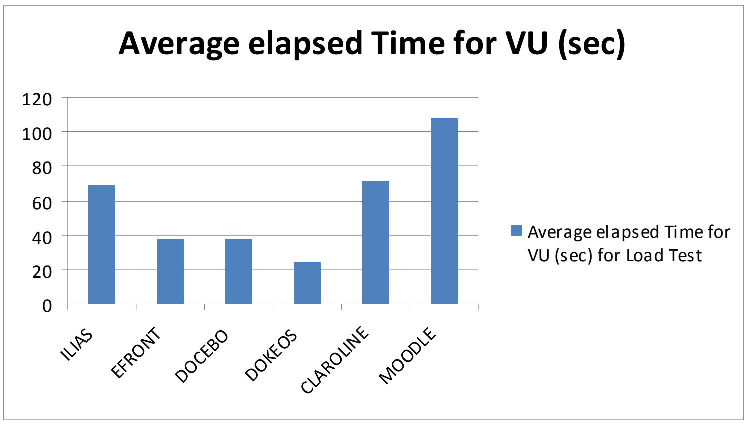


Figure 42. The comparison of Average Elapsed Time for Virtual User for Load Test

The comparison of average elapsed time for virtual users' parameter for 100 virtual users is given at Figure 43. Moodle had very high score for 100 virtual users when it was compared with the other systems. Claroline had the least elapsed time for 100 virtual users. Moodle average elapsed time for virtual users' score was approximately 40 times of Claroline score.

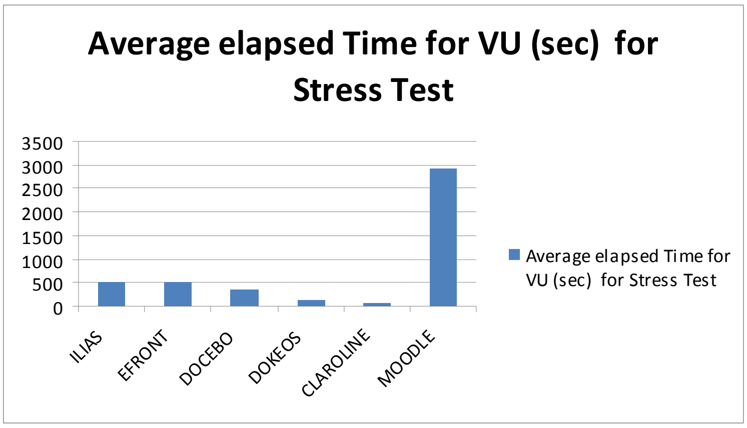


Figure 43. The comparison of Average Elapsed Time for Virtual Users for Stress Test

The comparison of total script time parameter for 30 virtual users is given at Figure 44. Efront had the highest total script time score, whereas Dokeos had the least score. The order of the total script time for load test from low to high was Dokeos, Claroline, Ilias, Docebo, Moodle and Efront.

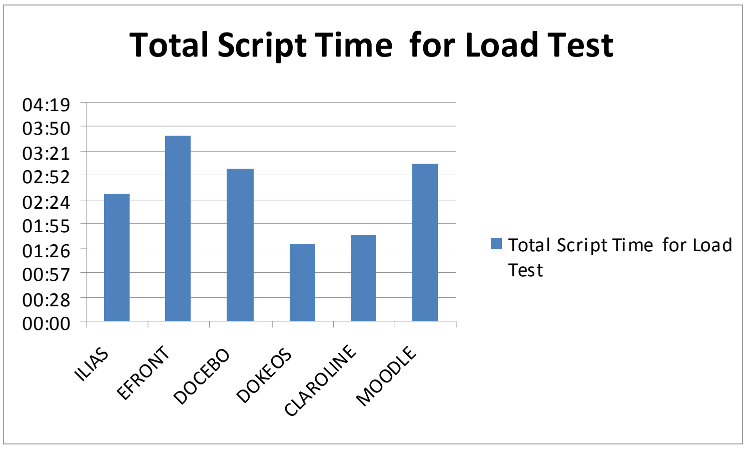


Figure 44. The comparison of total script time for Load Test

The comparison of total script time parameter for 100 virtual users is given at Figure 45. Moodle platform had highest total script time for 100 virtual users. Moreover, Moodle had big gap when it is compared with the other platforms. Claroline had the least total script time for stress test.

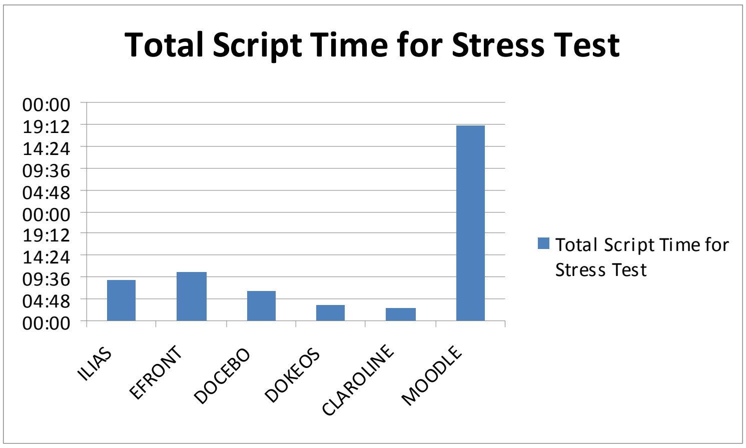


Figure 45. The comparison of total script time for Stress Test

The comparison of number of errors parameter for 100 virtual users is given at Figure 46. Docebo, Dokeos and Claroline finished the stress test without any error. However, Moodle had the largest number of errors and again there was a big gap when it was compared with the other systems.

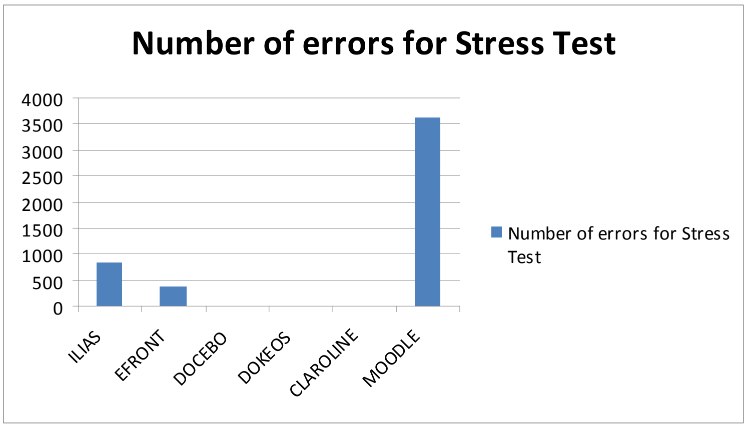


Figure 46. The comparison of number of errors parameter for Stress Test

The comparison of average bytes responded from server (Kb) parameter for 30 virtual users is given at Figure 47. Efront had the highest average bytes responded from server, whereas Claroline had the least score. Dokeos and Claroline average bytes responded from server score was almost the same. The order of average bytes responded from server (Kb) for load test from low to high was Claroline, Dokeos, Moodle, Ilias, Docebo and Efront.

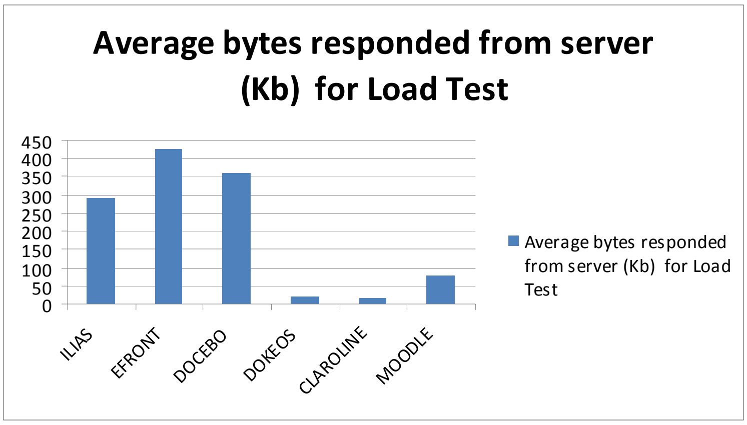


Figure 47. The comparison of Average Bytes Responded From Server (Kb) for Load Test

The comparison of average bytes responded from server (Kb) parameter for 100 virtual users is given at Figure 48. Ilias, Dokeos and Docebo had very similar scores of average bytes responded from server for stress test. Moreover, there was a gap between these systems' scores and others systems' scores. The order for that parameter from low to high was Efront, Moodle, Claroline, Dokeos, Docebo and Ilias.

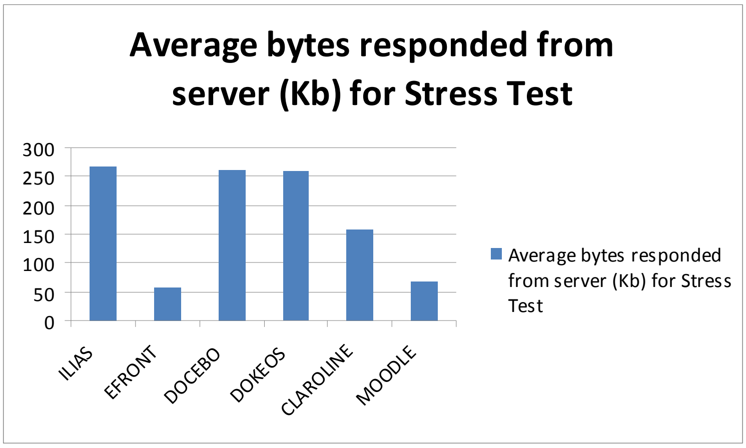


Figure 48. The comparison of Average Bytes Responded From Server (Kb) for Stress Test

The comparison of HTTP request parameter for 30 virtual users is given at Figure 49. The order for that parameter from high to low was Moodle, Efront, Claroline, Dokeos, Ilias and Docebo. Docebo platform HTTP request score was approximately 7 times higher than Moodle platform score.

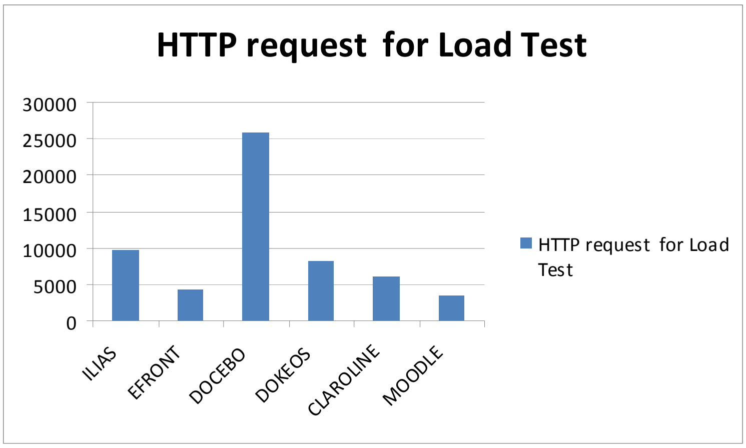


Figure 49. The comparison of HTTP request parameter for Load Test

The comparison of HTTP request parameter for 100 virtual users is given at Figure 50. The order for that parameter from high to low was Efront, Claroline, Moodle, Ilias, Dokeos and Docebo. Docebo had also highest score of HTTP request parameter.

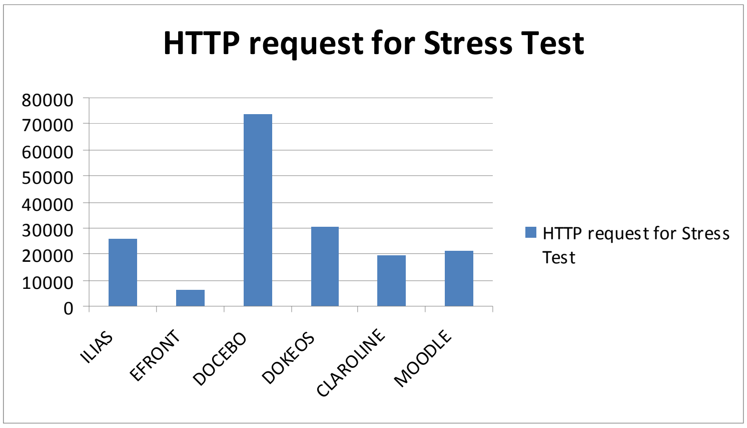


Figure 50. The comparison of HTTP request parameter for Stress Test

The comparison of finished user parameter for 30 virtual users is given at Figure 51. Almost all virtual users completed the test at Ilias, Efront and Docebo platforms.29 virtual users completed the test at Ilias, Efront and Docebo platforms. 26 virtual users at Dokeos, 25 virtual users at Claroline and 13 virtual users at Moodle completed the load test.

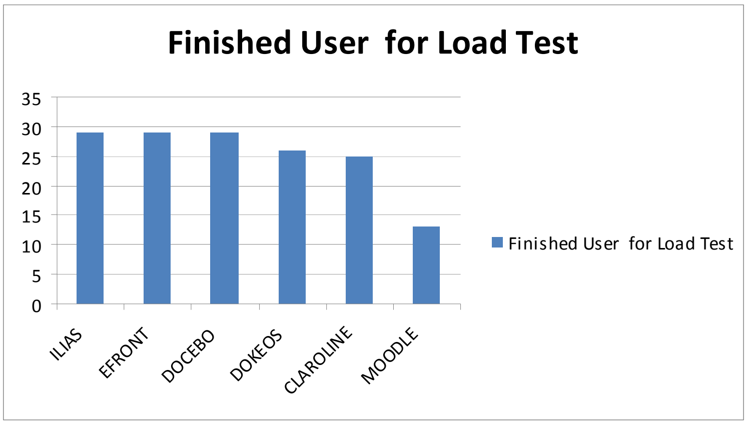


Figure 51. The comparison of finished user parameter for Load Test

The comparison of finished user parameter for 100 virtual users is given at Figure 52. Dokeos had the maximum numbers of finished user for stress test with the number of 98. The order for finished user was from minimum to maximum number was Efront, Ilias, Moodle, Claroline, Docebo and Dokeos.

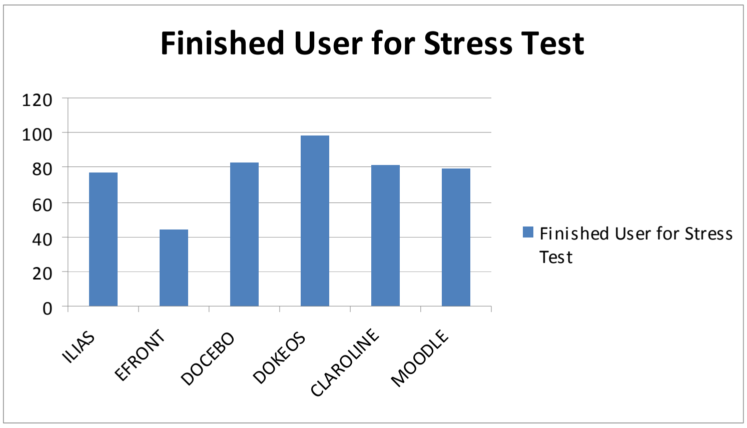


Figure 52. The comparison of finished user parameter for Stress Test

The comparison of error for virtual user: no data available for connection parameter for 100 virtual users is given at Figure 53. The reasons of that error could be while running a heavy load test and many of the HTTP response times in the log are large or connection that had been re-used for multiple HTTP requests previously. Docebo, Deokeos and Claroline had no error for virtual user: no data available for connection. Ilias had maximum number of error for virtual user:no data available for connection.

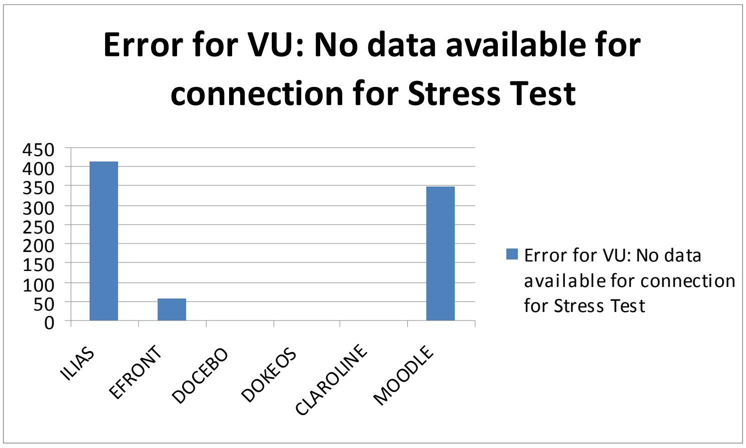


Figure 53. The comparison of Error For Virtual User:No Data Available for Connection for Stress Test

The comparison of IO failed for virtual user parameter for 100 virtual users is given at Figure 54. Docebo, Deokeos and Claroline had no IO failed for virtual user parameter. Moodle had the maximum number of IO failed for virtual user.

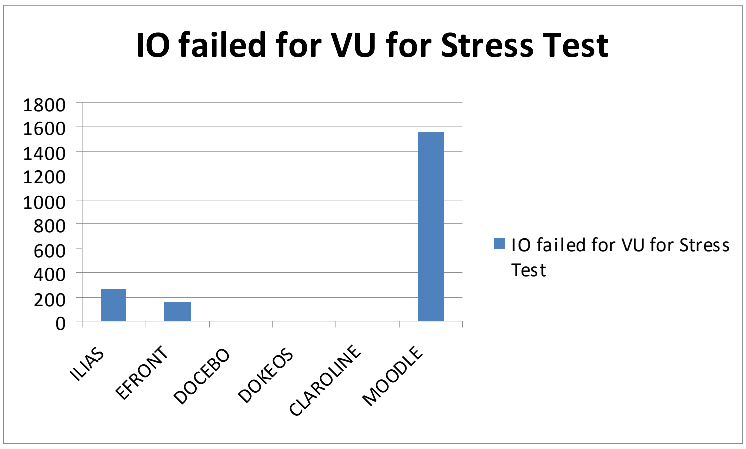


Figure 54. The comparison of IO Failed for Virtual User for Stress Test

Summary

In order to create a performance score for a LMS, stress and load test scores were considered. Firstly load and stress test scores were calculated and then average of these scores was used as performance score of an LMS. While calculating the stress and load scores, average elapsed time for virtual user, number of errors that occurred during test, failed HTTP request percentage and unfinished user of the test were taken into consideration. Weight factor for each criterion had taken 1 so the score was calculated as sum of these criteria. However, since each criterion had different scale and unit, raw score of the criteria should be normalized before summing. Standard deviation of the criteria was so high, thus t-score standardization was not used. Instead of t-score, min max normalization with range [0-100] was used. Formula of the min max normalization was

$$x^{'} = \frac{x - \min_A}{\max_A - \min_A}(\text{new\_max}_A - \text{new\_min}_A) + \text{new\_min}_A$$

When new min used as 0 and new max used as 100, formula becomes

$$x^{'} = \frac{x - \min_A}{\max_A - \min_A} * 100$$

Since in stress test Efront was not able to complete test with 100 virtual user, its scores in all criteria in stress test has assumed 100.Table 10 shows stress scores and

Table 11 shows load scores of the LMSs. For both scores, low score means better performance.

Table 10. Stress Test Scores of the Systems

	Average elapsed Time for VU		Number of errors		Failed HTTP request percentage		Unfinished User		Total Score ***
LMS	Time (sec)	Norm. Value**	Error Count	Norm. Value**	Failed Percent.	Norm. Value**	User Count	Norm. Value**	
CLAROLINE	71.72	0.00	0.00	0.00	0.00%	0.00	19.00	80.95	80.95
DOCEBO	337.86	9.35	0.00	0.00	0.00%	0.00	17.00	71.43	80.78
DOKEOS	141.64	2.46	0.00	0.00	0.00%	0.00	2.00	0.00	2.46
EFRONT*		100.00		100.00		100.00		100.00	400.00
ILIAS	502.89	15.15	821.00	22.70	0.94%	15.69	23.00	100.00	153.54
MOODLE	2917.53	100.00	3617.00	100.00	5.96%	100.00	21.00	90.48	390.48

* Since Efront does not complete test with 100 virtual used, its scores were assumed lowest for all criteria
* * Min max normalization formula with range 0-100 applied
*** Low score means better performance

Table 11. Load Test Scores of the Systems

	Average elapsed Time for VU		Number of errors		Failed HTTP request percentage		Unfinished User		Total Score ***
LMS	Time (sec)	Norm. Value**	Error Count	Norm. Value**	Failed Percent.	Norm. Value**	User Count	Norm. Value**	
CLAROLINE	71.90	56.92	0.00	0.00	0.00%	0.00	5.00	25.00	81.92
DOCEBO	37.92	16.40	0.00	0.00	0.00%	0.00	1.00	0.00	16.40
DOKEOS	24.16	0.00	0.00	0.00	0.00%	0.00	4.00	18.75	18.75
EFRONT	38.16	16.69	0.00	0.00	0.00%	0.00	1.00	0.00	16.69
ILIAS	69.18	53.67	0.00	0.00	0.00%	0.00	1.00	0.00	53.67
MOODLE	108.04	100.00	0.00	0.00	0.00%	0.00	17.00	100.00	200.00

* * Min max normalization formula with range 0-100 applied
*** Low score means better performance

After stress and load performance test scores were calculated for each LMS, performance test scores were calculated. Stress test and load test scores were firstly standardized using t-score (50+10((x-μ)/σ)) and then inversed ((1/x)*1000). After transformation again no weight factor was used for stress and load test, so simply by summing load and stress standardized score performance score was calculated for each system. Calculated performance scores are given in Table 12 and Figure 55 shows the comparison of the LMSs.

Table 12. Performance Scores of the LMSs.

	Stress Test		Load Test		Total	
LMS	Raw Score *	Standardized Score **	Raw Score ***	Standardized Score **	Raw Score ****	Standardize Value *****
CLAROLINE	80.95	22.78	81.92	19.07	41.85	51.05
DOCEBO	80.78	22.79	16.40	23.12	45.91	57.60
DOKEOS	2.46	25.46	18.75	22.95	48.41	61.64
EFRONT	400.00	15.96	16.69	23.10	39.06	46.53
ILIAS	153.54	20.76	53.67	20.63	41.39	50.31
MOODLE	390.48	16.10	200.00	14.50	30.60	32.87
AVERAGE	*184.70*	*20.64*	*64.57*	*20.56*	*41.20*	*50.00*
STDEV	*169.97*	*3.87*	*71.38*	*3.40*	*6.19*	*10.00*

* Value calculated at Table 10
** First t-score calculated with formula (50+ (10*(x-mean)/std)) then value inverted (1/value)*1000 in order to make high score better performance
*** Value calculated at Table 11
**** Standardized Score for Stress Test + Standardized Score for Load Test
***** t-score (50+(10*(x-mean)/std)) is calculated in order to use value for executive comparison table

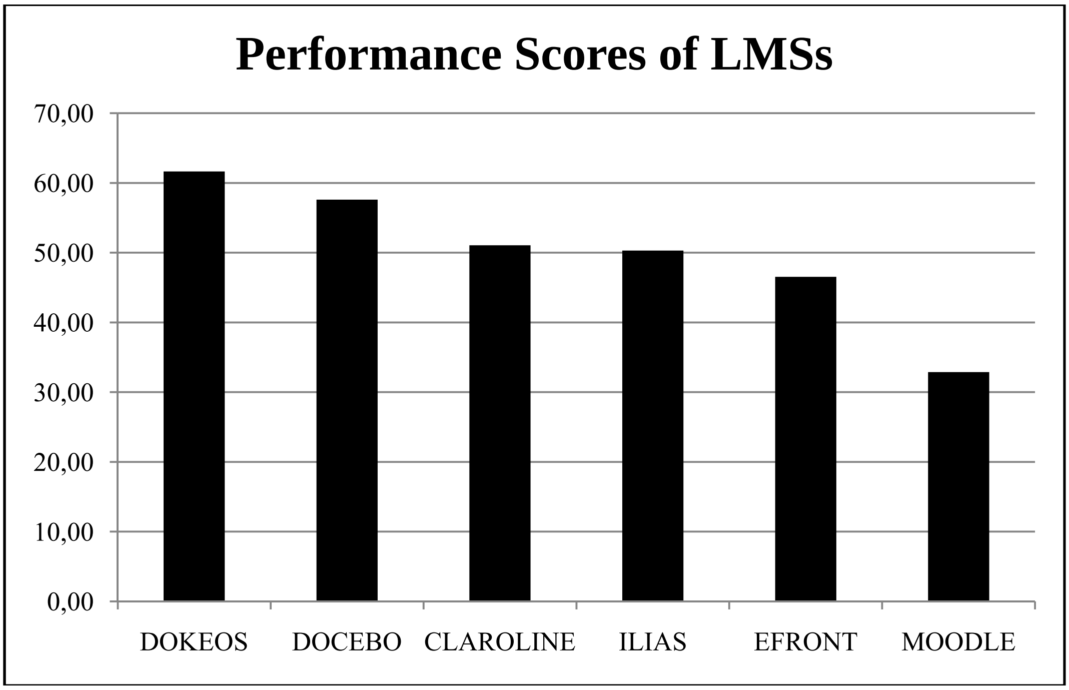


Figure 55. Performance comparisons of the systems

Security Test

Software security is about understanding software-induced security risks and how to manage them. Security is becoming more important. As majority of the systems connect to Internet, they become vulnerable to software-based attacks from distant sources. Security vulnerabilities in software systems can be scanned with automated tools (http://www.acunetix.com/vulnerability-scanner/wvs4manual.pdf).

Acunetix Web Vulnerability Scanner 4.0 was used for security testing. Acunetix Web Vulnerability Scanner offers its users real time reporting with four different scan types. One of the types offers a normal procedure where one web site gets all the attention. The others are options to select a file that contains the list of URls, to scan of a range of IP addresses with web servers running on ports specified by the user. The final scan type is if the software's built-in crawler module is already used, user can also act upon its results.

In Acunetix Web Vulnerability Scanner tool, High Risk Alert Level 3, Medium Risk Alert Level 2, Low Risk Alert Level 1 and Informational Alerts are descripted as below:

> High Risk Alert Level 3:
> Vulnerabilities categorised as the most dangerous, which put a site at maximum risk for hacking and data theft.
> Medium Risk Alert Level 2:
> Vulnerabilities caused by server misconfiguration and site-coding flaws, which facilitate server disruption and directory intrusion.
> Low Risk Alert Level 1:
> Vulnerabilities derived from lack of encription for data traffic, or directory path disclosures.
> Informational Alerts:
> Sites which are susceptible to revealing information through GHDB search strings, or email addresses disclosure (Acunetix Ltd, 2006).

In the thesis, while starting new scan, the target(s) to be scanned were specified.

The selected Web server technologies were PHP,mod_ssl and OpenSSL, and then at optimizing the technology options were Apache web server and Operation systems Windows were chosen. At scanning profile default options were chosen. At authentication level, login scripts were created by recording session. After that process, the tool started to scan the systems according to chosen technologies. After each system test, the server and client machines were restarted to refresh the memory of the system. Additionally, antivirus and firewall of the systems were closed in order not to interfere with the test process.

The results of security test of systems are shown at Table 13. The alert groups are grouped according to their severity level, which are high, medium and low levels. Some of the vulnerabilities are related with infrastructure or platform that the systems set up. These infrastructure vulnerabilities are shown at the vulnerabilities of infrastructure row. Additionally, these infrastructures weren't taken account of total vulnerabilities about LMS and total scored, as these vulnerabilities aren't related with the learning management systems software. The security vulnerabilities of the systems were also grouped by in both total vulnerabilities about LMS and total scored.

Table 13. LMS Security Results

		Claroline	Dokeos	Efront	Moodle	Ilias3	Docebo
Scan time (minutes)		87	404	265	48	364	194
Severity	Alert group	Number of alerts in this group					
High	Blind SQL/XPath injection for string inputs (double quotes)						1
High	Blind SQL/XPath injection for string inputs						1
High	Blind SQL/XPath injection for numeric inputs						1
High	Script source code disclosure		14	32	14	45	9

		Claroline	Dokeos	Efront	Moodle	Ilias3	Docebo
High	Cross Site Scripting	6	200	1		16	
High	File inclusion			2			
Medium	Source code disclosure		6	2	20	9	2
Medium	PHPSESSID session fixation	1					
Medium	Backup files					2	
Low	Possible sensitive files	11		25	3	34	6
Low	File inputs accepted		60	1		41	9
Low	Possible sensitive directories	2		1	3	11	14
Low	Broken links	1	33	15	32	200	24
Low	PHP script custom error message	1		4	1		30
Low	User credentials are sent in clear text	11	2		5	18	
Low	URL redirection	1	1				
Informational	Email address found	179		13			57
Informational	GHDB: FCKEditor script						200
Informational	GHDB: Script to display the source code for PHP scripts	3					
Informational	GHDB: Possible temporary file/directory	1					
Informational	GHDB: Files uploaded through FTP			18			
Informational	GHDB: Mp3 file			18			
	Vulnerabilities about Infrastucture	403	203	403	257	203	403
High		6	214	35	14	61	12
Medium		1	6	2	20	11	2
Low		27	96	46	44	304	83
Total Vulnerabilities About LMS		34	316	83	78	376	97
Informational		183	0	49	0	0	257

At Figure 56 Comparison of Total Vulnerabilities About LMS, total number of vulnerabilities except the vulnerabilities about infrastructure are taken into account. At that Figure 56, high, total number of medium and low vulnerabilities of each LMS were summed and then each system were compared with each other according to systems total vulnerabilities. When the results were compared, Ilias learning management system had maximum vulnerabilities in total and the least number of vulnerabilities are at Claroline learning management system. The order of the systems from maximum to minimum vulnerabilities was arranged like that Ilias, Dokeos, Docebo, Efront, Moodle and then Clarolline.

Moreover, each system's weighted total score is given at Table 14. High level vulnerabilities are more critical and low level vulnerabilities are less critical for the systems so that the impact of high to low level vulnerabilities isn't the same for the systems. Therefore, high level vulnerabilities were rated with 3, medium level vulnerabilities were rated with 2 and low level vulnerabilities were rated with 1. Total scores of the LMSs were found by multiplying the number of vulnerabilities with the rated scale of each severity. When these results were compared, Dokeos LMS has maximum vulnerabilities in total scored and the least number of total scored vulnerabilities is at Claroline LMS. The order of the systems from maximum to minimum total scored vulnerabilities was arranged like that Dokeos, Ilias, Efront, Moodle ,Docebo and then Clarolline.

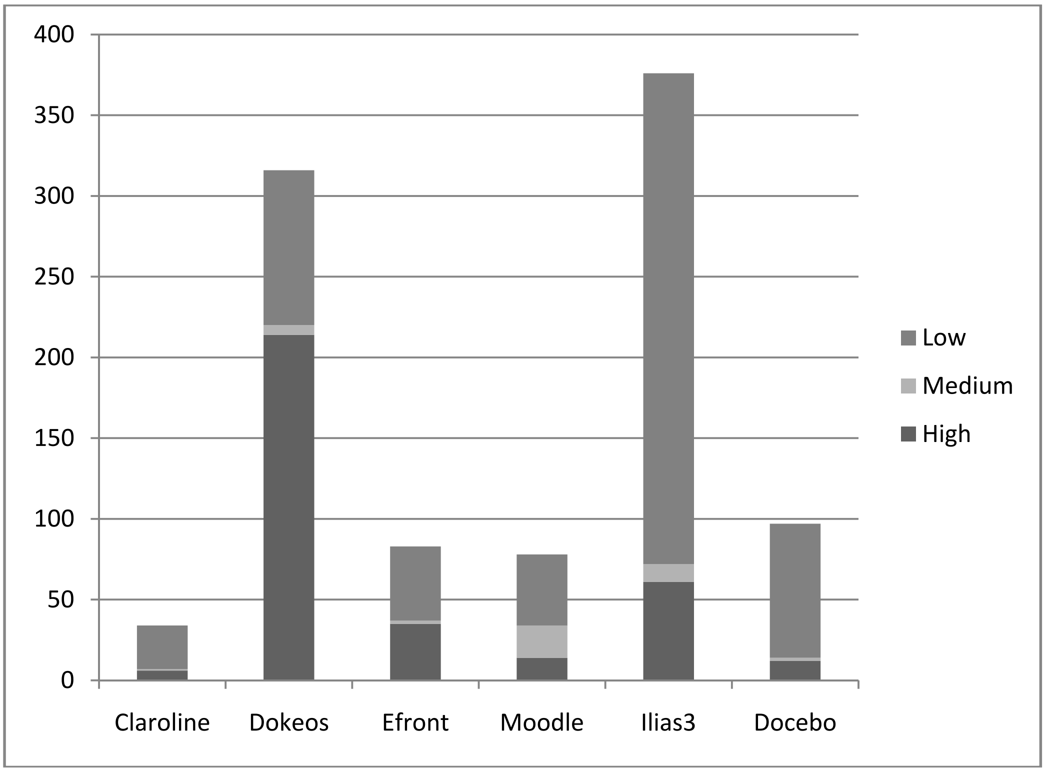


Figure 56. Comparison of total vulnerabilities about LMSs

Summary

While calculating the security score of the systems, high, medium and low vulnerability count were taken into account. Weight factors were used 3, 2, and 1 respectively. Thus, the formula of the security score was;

$$\text{Securtiy Score} = \frac{1}{3 * \text{N}_{\text{high}} + 2 * \text{N}_{\text{medium}} + \text{N}_{\text{low}}} * 1000$$

Standard score was calculated using t score formula. Table 14 shows the security scores of the systems and comparison of the scores is given in Figure 57.

Table 14. Security Scores of the LMSs.

	High	Medium	Low	Raw Score	Standart Score
Claroline	6.00	1.00	27.00	21.28	68.64
Docebo	12.00	2.00	83.00	8.13	50.39
Dokeos	214.00	6.00	96.00	1.33	40.95
Efront	35.00	2.00	46.00	6.45	48.06
Ilias3	61.00	11.00	304.00	1.96	41.83
Moodle	14.00	20.00	44.00	7.94	50.12
Average	57.00	7.00	100.00	7.85	50.00
Std. dev.	79.53	7.38	103.25	7.20	10.00

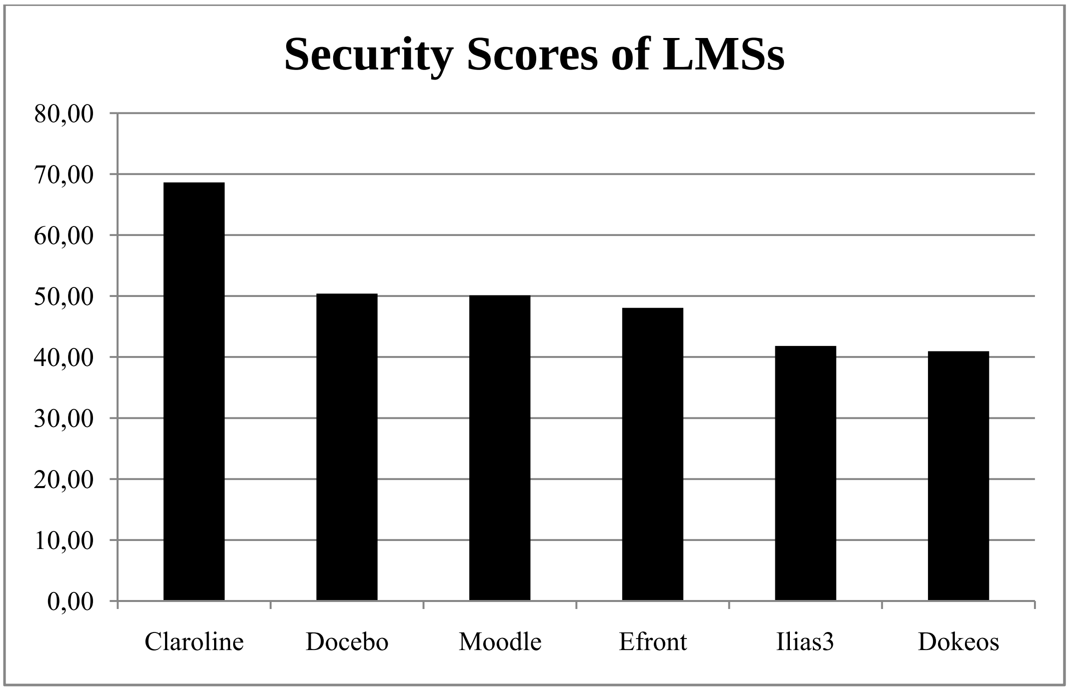


Figure 57. Security comparisons of the systems

Standards Compliance Test

Most e-learning projects require large investment involving a considerable amount of money, time, and human training. Thus, by taking into consideration these large investments, careful choosing of the appropriate e-learning systems is critical. Reusability and interoperability are most important factors for successful e-learning systems. Sharable Content Object Reference Model (SCORM), which is a collection of standards and specifications adapted from multiple sources to provide a comprehensive suite of e-learning capabilities, enable interoperability, accessibility and reusability of Web-based learning content (ADL, 2008), LMS and course-ware can be developed by different companies. They should apply some common rules in order to run them coherently. SCORM is a key factor since it provides common framework and rules for both LMSs and content packages. The Sharable Content Object Reference Model (SCORM) specifies how learning content should be coded, how others can later "discover" that content, how it fits into a sequence of learning activities, how its appearance through the delivery media can be customized for the individual learner, how it can communicate with LMS, and how the run-time (LMS) should be developed.

SCORM Conformance Test Suite (Self Test) contains the conformance testing software, procedures and supporting documents for organizations to perform self-testing on LMSs, SCOs and content packages. In this study e-learning systems were tested directly with “ADL SCORM® 2004 3rd Edition Conformance Test Suite” and “ADL SCORM® Version 1.2 Conformance Test Suite”. Conformance test suite results showed if the system was compliant or not. Ilias was the only open source LMS in the study, which passed “ADL SCORM® 2004 3rd Edition Conformance Test Suite” and reached SCORM 2004 (3rd Edition) compliance. Ilias was also both SCORM2004 and “SCORM 1.2 RTE Level 3” certified. Moodle completed each steps of “ADL SCORM® Version 1.2 Conformance Test”, which were “Import ADL SCORM Test Course I”, “Import ADL Test Course II”, “Enroll Student in Test Course I”, “Enroll Students in Test Course II”, “Take Test Course I and Take Test Course II”. At step 5, LMS was tested SCO 01 to SCO 09 of “course 1”; and at step 6, LMS was tested SCO 01 to SCO 03 of “course 2”. The result of “Moodle SCORM® Version 1.2 Conformance Test” was “SCORM Version 1.2 Run-Time Environment Conformant - Minimum with Some Optional Data Model Elements (LMS-RTE2)”. EFRONT didn’t pass the “ADL SCORM® Version 1.2 Conformance Test” because of some errors, such as “ERROR: The student name returned by the LMS is incorrect. The student name should be Learner, Mary", “ERROR: cmi.core.lesson_location value does not match previously set value”. Claroline, Docebo and Dokeos couldn’t finish the “ADL SCORM® Version 1.2 Conformance Test”. All LMSs’ SCORM standards compliance was also shown in

Table 15. Both SCORM 1.2 and SCORM 2004 Standards Compliance are examined by taking into consideration of certification and test results.

Table 15. SCORM Compliance Summary Table

	SCORM 1.2		SCORM 2004	
	Test	Certificate	Test	Certificate
Clarorine	No*	No	No	No
Docebo	No*	No	No	No
Dokeos	No*	No	No	No
Efront	Yes	Yes	No	No
Ilias	Yes	Yes	Yes	Yes
Moodle	Yes	No	No	No

* Although official sites of products claim they support SCORM standards, the systems could not pass the SCORM compliance test

While calculating total SCORM scores, certification and test results were taken into account with different weights. SCORM 1.2 and SCORM 2004 Standards Compliance were also evaluated with different weights. 40% weight is given to SCORM 1.2 Standard and 60% weight is given to SCORM 2004 Standard. For SCORM 1.2 Standard Compliance, weight of test results was 30% and weight of certification was 10%. Moreover, for SCORM 2004 Standard Compliance, weight of test results was 50% and weight of certification was 10%. Standard score is calculated using t score formula and comparison of LMSs for SCORM Compliance is given at Figure 58.

Table 16. SCORM Score Calculation

	SCORM 1.2		SCORM 2004		SCORE	
	Test (30%)	Certificate (10%)	Test (50%)	Certificate (10%)	Raw	Standard Score
Clarorine	0	0	0	0	0	42.77
Docebo	0	0	0	0	0	42.77
Dokeos	0	0	0	0	0	42.77
Efront	1	1	0	0	0.4	52.98
Ilias	1	1	1	1	1	68.28
Moodle	1	0	0	0	0.3	50.43
				Average	0.2833333	
				Std. Dev.	0.3920034	

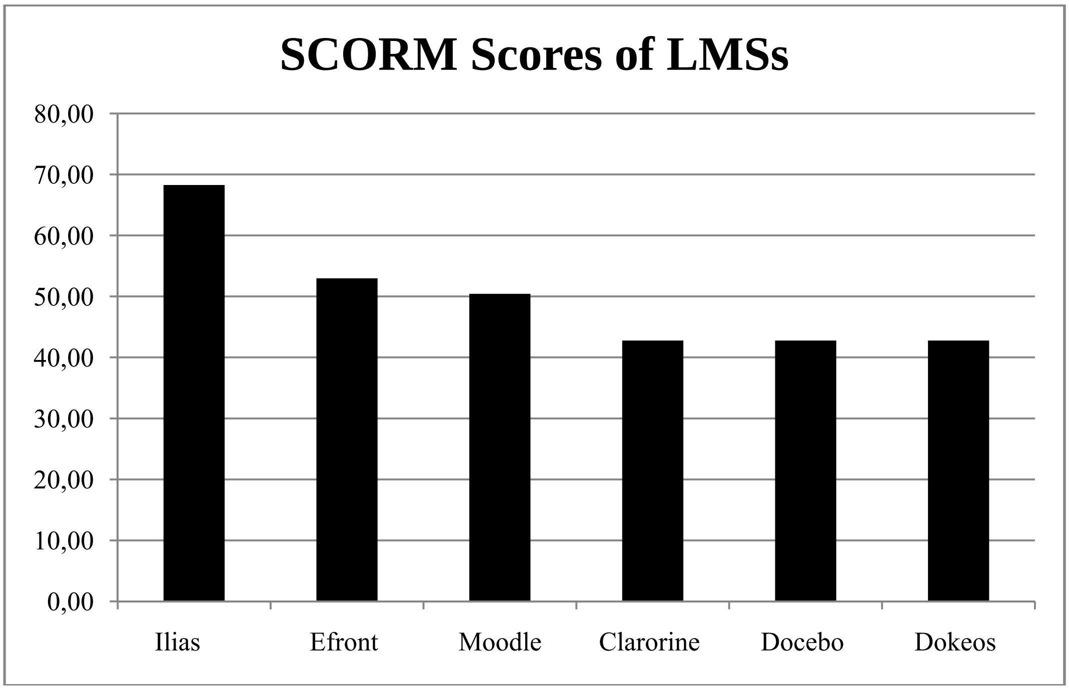


Figure 58. Comparison of standard compliance scores of LMSs

Functionality Comparison

Learning Management Systems (LMSs) typically provide a wide set of functionalities to support students' learning such as file storage, forums, calendar, news, bulk mail, submission management system, groups' surveys, organization, assessments, FAQs (Frequently Asked Questions) or scheduling. Functionality comparison was done for determining whether or not a program does the previously defined functional requirements. Moodle, Ilias, Dokeos, Docebo, Claroline and Efront systems were evaluated according to selected functionalities of the LMS Functionality Table, which was adapted from the research of Merino, Kloos, Seepold and García (2006). The existing functionalities of the systems were assumed as indicators of the success of the systems in terms of functionalities. When the existing functionalities of each system were compared, Dokeos e-learning platform had a substantial number of functionalities, whereas Claroline contained a minimum number of functionalities. Claroline and Efront had almost the same number of functionalist. Moodle, Ilias and Docebo had almost the same number of functionalities. In addition, all LMSs had forum, file download, calendar, news/announcement about the course, FAQ/glossary/wiki, lists of students (for all courses in which is enrolled), assessments, help about using the tool, application access log, knowledge of user own learning evolution and status at any time, knowledge of user`s own level for each topic and chat functionalities. Docebo was the only LMS, which had functionality of receiving relevant information about the course by SMS. Dokeos and Docebo had videoconference functionality.

Table 17. LMS Functionalities

	Moodle	Ilias	Dokeos	Docebo	Claroline	Efront
Forum	Yes	Yes	Yes	Yes	Yes	Yes
Forums can be viewed by title/topic/author/groups/date/threaded etc.	Yes	Yes	Yes	Yes	Yes	No
Instructors can create separate forums for small groups.	Yes	No	Yes	Yes	Yes	No
Calendar	Yes	Yes	Yes	Yes	Yes	Yes
Course Calendar	Yes	No	Yes	Yes	Yes	Yes
User calendar(student/teacher)	Yes	Yes	Yes	Yes	Yes	Yes
News/Announcement	Yes	Yes	Yes	Yes	Yes	Yes
News/Announcement about the course to student	Yes	Yes	Yes	Yes	Yes	Yes
News/Announcement about the course to teachers	Yes	Yes	Yes	Yes	Yes	Yes
Content	Yes	Yes	Yes	Yes	Yes	Yes
Multimedia	Yes	Yes	Yes	Yes	Yes	Yes
Wiki	Yes	No	No	Yes	Yes	No
Glossary	Yes	Yes	No	Yes	No	Yes
FAQ	No	No	Yes	Yes	No	No
Links	Yes	Yes	Yes	Yes	Yes	No
Games	No	No	No	No	No	No
On-line polls /Surveys	Yes	Yes	Yes	Yes	No	Yes
E-mail notifications	Yes	Yes	Yes	Yes	No	Yes
E-mail lists	Yes	Yes	Yes	Yes	No	Yes
Internal e-mail service	Yes	Yes	Yes	Yes	No	Yes
Receiving relevant information about the course by SMS	No	No	No	Yes	No	No
Test	Yes	Yes	Yes	Yes	Yes	Yes
Online Test	Yes	Yes	Yes	Yes	Yes	Yes
Question Bank	Yes	Yes	Yes	Yes	Yes	Yes
Create test (multiple choice questions, fill in the blanks, true/flase questions,...)	Yes	Yes	Yes	Yes	No	Yes
Assessments	Yes	Yes	Yes	Yes	Yes	Yes
Submission management	Yes	Yes	Yes	No	Yes	Yes

	Moodle	Ilias	Dokeos	Docebo	Claroline	Efront
system						
Grade Book	Yes	No	Yes	Yes	No	No
Syllabus	No	Yes	Yes	No	Yes	Yes
Create syllabus from template (via Wizard)	No	Yes	Yes	No	Yes	Yes
Import Syllabus	Yes	Yes	Yes	Yes	Yes	Yes
Additional information about the teaching staff (other courses, location, tutoring, research…)	Yes	Yes	Yes	Yes	No	No
File Sharing	Yes	Yes	Yes	Yes	Yes	Yes
Students can upload projects, images or any files to a shared library.	No	Yes	No	Yes	No	No
Instructors can upload projects, images or any files to a shared library.	Yes	Yes	Yes	Yes	Yes	Yes
Instructors can edit their text files in their folder using a browser.	Yes	Yes	Yes	No	Yes	No
Students can edit their text files in their folder using a browser.	Yes	Yes	No	No	No	No
Lists of students (for all courses in which is enrolled)	Yes	Yes	Yes	Yes	Yes	Yes
Users can create a home page for themselves	Yes	No	Yes	No	No	No
Virtual Classroom	Yes	No	Yes	Yes	Yes	Yes
Chat tool for messaging	Yes	No	Yes	Yes	Yes	Yes
The system creates archive logs for all chat rooms.	Yes	No	Yes	Yes	No	Yes
Real-time slides	No	No	Yes	No	No	No
E-blackboard	No	No	Yes	No	No	No
Videoconference	No	No	Yes	Yes	No	No
Student Presentation	No	No	Yes	No	No	No
Search	Yes	Yes	Yes	Yes	No	No
Incorporate web search engines (Google, Altavista, etc.)	No	No	Yes	No	No	No
Search within the portal	Yes	Yes	Yes	Yes	No	No
Help	Yes	Yes	Yes	Yes	Yes	Yes
Help about using the tool	Yes	Yes	Yes	Yes	Yes	Yes
Help desk	Yes	Yes	Yes	Yes	No	No
Reports	Yes	Yes	Yes	Yes	Yes	Yes
Student Reports	Yes	Yes	Yes	Yes	Yes	Yes
Course Reports	Yes	Yes	Yes	Yes	Yes	Yes
System Reports	Yes	Yes	Yes	Yes	Yes	Yes
Generic Queries	Yes	Yes	Yes	No	Yes	Yes

(Adapted from: Merino, Kloos, Seepold & García, 2006)

In the thesis, a survey methodology was implemented in order to collect data about the necessity of functionalities of e-learning systems and degree of importance of the criteria.

Subjects of the survey were selected among specialists in e-learning/distance learning or LMS developers who work in different sectors like universities, K12 schools and private. Survey information was sent to users by e-mail which described the aim of the survey and basic instructions about survey. 22 participants answered to survey, 10 of them were male and 12 of them were female (Table 18). Distribution of the sample according to the sector types is given in Table 19 and Table 20 shows the distribution according to job experience.

Table 18. Distribution of Sample According to Gender

	Frequency	Percent
Female	12	54,5
Male	10	45,5
Total	22	100,0

Table 19. Distribution of Sample According to Sector Type

	Frequency	Percent
Private	9	40,9
K12	7	31,8
Academic	6	27,3
Total	22	100,0

Table 20. Distribution of Sample According to Job Experience

	Frequency	Percent
1-3 Years	9	40,9
4-6 Years	9	40,9
7 and more	4	18,2
Total	22	100,0

The survey contained 3 parts. First part was about demographic information such as name surname, gender, current industry and company, department, job title and the length of time (approximately) in year that he/she has been working. Second part included list of functionalities. In that part, the data collection tool was used a Likert scale-rating scheme (five-point) with the range of 1 to 5; where 1 means least/not necessary and 5 means most necessary, while ranging the functionalities of the systems. At the survey, there were main functionality groups such as forum, calendar, test, report and content. Some of the groups had sub- functionalities, such as reports main functionality group had functionalities of teacher, system, lesson and generic queries. Both main functionality groups and sub-functionalities were evaluated by participants. There were 16 main-functionality groups and 41 sub-functionalities. Total numbers of functionalities were 57 in the second part of the survey. Additionally, the last part was about ranking learning management systems criteria, which were standards compliant, performance/load, security, accessibility and functionality criteria. The scale in the third part of the survey was used to rank the criteria from 1 to 5; where 1 means most important and 5 means least important, and each score could only be used once and for one criterion.

The scale was checked by two academicians and one specialist who are experts on e-learning. Cronbach's Alpha reliability coefficient of the questionnaire was calculated as 0.964, demonstrating that the survey is highly reliable.

The questionnaire was posted on a website, which is http://www.surveymonkey.com/. The participants used in the survey/study were specialists in e-learning/distance learning or LMS developers. The result of the importance of necessity of functionalities according to second part of the survey is shown at Table 21.

Table 21.Result of the Importance of Necessity of Functionalities

	1	2	3	4	5	Avg.	Std. Deviation
FORUM	0.0% (0)	0.0% (0)	22.7% (5)	27.3% (6)	50.0% (11)	4.27	.827
Forums can be viewed by title/topic/author/groups/date/etc	0.0% (0)	0.0% (0)	22.7% (5)	27.3% (6)	50.0% (11)	4.27	.827
Instructors can create separate forums for small groups	0.0% (0)	9.1% (2)	13.6% (3)	36.4% (8)	40.9% (9)	4.09	.971
CALENDER	0.0% (0)	13.6% (3)	4.5% (1)	40.9% (9)	40.9% (9)	4.09	1.019
Course Calendar	0.0% (0)	0.0% (0)	4.5% (1)	31.8% (7)	63.6% (14)	4.59	.590
User calendar(student/teacher)	0.0% (0)	0.0% (0)	22.7% (5)	31.8% (7)	45.5% (10)	4.23	.813
NEWS/ANNOUNCEMENT	0.0% (0)	4.5% (1)	9.1% (2)	31.8% (7)	54.5% (12)	4.36	.848
News/Announcement about the course to student	0.0% (0)	4.5% (1)	4.5% (1)	36.4% (8)	54.5% (12)	4.41	.796
News/Announcement about the course to teachers	0.0% (0)	4.5% (1)	4.5% (1)	45.5% (10)	45.5% (10)	4.32	.780
CONTENT	0.0% (0)	4.5% (1)	4.5% (1)	4.5% (1)	86.4% (19)	4.73	.767
Multimedia	0.0% (0)	4.5% (1)	9.1% (2)	13.6% (3)	72.7% (16)	4.55	.858
Wiki	4.5% (1)	0.0% (0)	22.7% (5)	50.0% (11)	22.7% (5)	3.86	.941
Glossary	4.5% (1)	0.0% (0)	22.7% (5)	45.5% (10)	27.3% (6)	3.91	.971
FAQ	0.0% (0)	0.0% (0)	36.4% (8)	31.8% (7)	31.8% (7)	3.95	.844
Links	0.0% (0)	9.1% (2)	18.2% (4)	50.0% (11)	22.7% (5)	3.86	.889
Games	4.5% (1)	9.1% (2)	36.4% (8)	45.5% (10)	4.5% (1)	3.36	.902
On-line polls /Surveys	0.0% (0)	4.5% (1)	18.2% (4)	40.9% (9)	36.4% (8)	4.09	.868
E-MAIL NOTIFICATIONS	0.0% (0)	0.0% (0)	9.1% (2)	40.9% (9)	50.0% (11)	4.41	.666
E-mail lists	0.0% (0)	0.0% (0)	27.3% (6)	31.8% (7)	40.9% (9)	4.14	.834
Internal e-mail service	9.1% (2)	18.2% (4)	18.2% (4)	27.3% (6)	27.3% (6)	3.45	1.335
RECEIVING RELEVANT INFORMATION ABOUT THE COURSE BY SMS	9.1% (2)	13.6% (3)	22.7% (5)	45.5% (10)	9.1% (2)	3.32	1.129
TEST	0.0% (0)	0.0% (0)	4.5% (1)	18.2% (4)	77.3% (17)	4.73	.550
Online Test	0.0% (0)	0.0% (0)	9.1% (2)	9.1% (2)	81.8% (18)	4.73	.631
Question Bank	0.0% (0)	4.5% (1)	9.1% (2)	22.7% (5)	63.6% (14)	4.45	.858
Create test (multiple choice questions, fill in the blanks, true/false questions,...)	0.0% (0)	0.0% (0)	9.1% (2)	18.2% (4)	72.7% (16)	4.64	.658
ASSESSMENTS	0.0% (0)	4.5% (1)	9.1% (2)	9.1% (2)	77.3% (17)	4.59	.854
Submission management system	0.0% (0)	4.5% (1)	4.5% (1)	18.2% (4)	72.7% (16)	4.59	.796
Grade Book	0.0% (0)	0.0% (0)	18.2% (4)	27.3% (6)	54.5% (12)	4.36	.790
SYLLABUS	0.0% (0)	0.0% (0)	4.5% (1)	50.0% (11)	45.5% (10)	4.41	.590

	1	2	3	4	5	Avg.	Std. Deviation
Create syllabus from template (via Wizard)	4.5% (1)	4.5% (1)	22.7% (5)	45.5% (10)	22.7% (5)	3.77	1.020
Import Syllabus	0.0% (0)	4.5% (1)	31.8% (7)	31.8% (7)	31.8% (7)	3.91	.921
Additional information about the teaching staff (other courses, location, tutoring, research…)	0.0% (0)	0.0% (0)	40.9% (9)	40.9% (9)	18.2% (4)	3.77	.752
FILE SHARING	0.0% (0)	4.5% (1)	18.2% (4)	22.7% (5)	54.5% (12)	4.27	.935
Students can upload projects, images or any files to a shared library.	0.0% (0)	9.1% (2)	9.1% (2)	31.8% (7)	50.0% (11)	4.23	.973
Instructors can upload projects, images or any files to a shared library.	0.0% (0)	9.1% (2)	0.0% (0)	36.4% (8)	54.5% (12)	4.36	.902
Instructors can edit their text files in their folder using a browser.	0.0% (0)	13.6% (3)	18.2% (4)	31.8% (7)	36.4% (8)	3.91	1.065
Students can edit their text files in their folder using a browser.	0.0% (0)	4.5% (1)	27.3% (6)	40.9% (9)	27.3% (6)	3.91	.868
LISTS OF STUDENTS (FOR ALL COURSES IN WHICH IS ENROLLED)	0.0% (0)	4.5% (1)	9.1% (2)	36.4% (8)	50.0% (11)	4.32	.839
USERS CAN CREATE A HOME PAGE FOR THEMSELVES.	9.1% (2)	4.5% (1)	54.5% (12)	13.6% (3)	18.2% (4)	3.27	1.120
VIRTUAL CLASSROOM	0.0% (0)	13.6% (3)	18.2% (4)	36.4% (8)	31.8% (7)	3.86	1.037
Chat tool for messaging	0.0% (0)	0.0% (0)	27.3% (6)	40.9% (9)	31.8% (7)	4.05	.785
The system creates archive logs for all chat rooms.	0.0% (0)	4.5% (1)	40.9% (9)	13.6% (3)	40.9% (9)	3.91	1.019
Real-time slides	4.5% (1)	4.5% (1)	31.8% (7)	27.3% (6)	31.8% (7)	3.77	1.110
E-blackboard	0.0% (0)	4.5% (1)	27.3% (6)	31.8% (7)	36.4% (8)	4.00	.926
Videoconference	0.0% (0)	0.0% (0)	27.3% (6)	22.7% (5)	50.0% (11)	4.23	.869
Student Presentation	0.0% (0)	4.5% (1)	22.7% (5)	27.3% (6)	45.5% (10)	4.14	.941
SEARCH	4.5% (1)	0.0% (0)	9.1% (2)	22.7% (5)	63.6% (14)	4.41	1.008
Incorporate web search engines (Google, Altavista, etc.)	0.0% (0)	18.2% (4)	27.3% (6)	27.3% (6)	27.3% (6)	3.64	1.093
Search within the portal	0.0% (0)	0.0% (0)	13.6% (3)	31.8% (7)	54.5% (12)	4.41	.734
HELP	0.0% (0)	4.5% (1)	4.5% (1)	9.1% (2)	81.8% (18)	4.68	.780
Help about using the tool	0.0% (0)	0.0% (0)	13.6% (3)	13.6% (3)	72.7% (16)	4.59	.734
Help desk	0.0% (0)	4.5% (1)	18.2% (4)	36.4% (8)	40.9% (9)	4.14	.889
REPORTS	4.5% (1)	0.0% (0)	0.0% (0)	22.7% (5)	72.7% (16)	4.59	.908
Student Reports	0.0% (0)	4.5% (1)	0.0% (0)	13.6% (3)	81.8% (18)	4.73	.703
Lesson Reports	0.0% (0)	0.0% (0)	0.0% (0)	18.2% (4)	81.8% (18)	4.82	.395
System Reports	0.0% (0)	0.0% (0)	4.5% (1)	22.7% (5)	72.7% (16)	4.68	.568
Generic Queries	0.0% (0)	4.5% (1)	9.1% (2)	31.8% (7)	54.5% (12)	4.36	.848

The result of ranking learning management systems criteria according to third part of the survey is show at Table 22.

Table 22. Summary of Criteria Ranking Score

	1	2	3	4	5	average
Standards Compliant	18.2% (4)	4.5% (1)	13.6% (3)	27.3% (6)	36.4% (8)	2.41
Performance/Load	22.7% (5)	13.6% (3)	22.7% (5)	31.8% (7)	9.1% (2)	3.09
Security	9.1% (2)	36.4% (8)	13.6% (3)	13.6% (3)	27.3% (6)	2.86
Accessibility	18.2% (4)	27.3% (6)	18.2% (4)	18.2% (4)	18.2% (4)	3.09
Functionality	31.8% (7)	18.2% (4)	31.8% (7)	9.1% (2)	9.1% (2)	3.55

Summary

In Table 22, functionally score calculated for each LMS. In calculation of the functionality score for each LMS

$$\mathrm{F_{LMS}} = \sum_{i=0}^{n} \mathrm{W_i S_i}$$

where $\mathrm{S_i}$ is 1 or -1 according to Table 23 (Yes means 1 and no means -1), $\mathrm{W_i}$ is the weight factor of a each criteria which is calculated standard t score

$$W_i = 50 + 10 * (\frac{x_i - \mu}{\sigma})$$

where x_i is average of criteria in the survey, μ is mean of the survey, and σ is the standard deviation of the survey.

Table 23. Functionality Score Calculation of LMSs

CRITERIA	Avg	t score	Moodle	Ilias	Dokeos	Docebo	Claroline	Efront
Forum	4.27	51.89	51.89	51.89	51.89	51.89	51.89	51.89
Forums can be viewed by title/topic/author/groups/date/threaded etc.	4.27	51.89	51.89	51.89	51.89	51.89	51.89	-51.89
Instructors can create separate forums for small groups.	4.09	47.13	47.13	-47.13	47.13	47.13	47.13	-47.13
Calendar	4.09	47.13	47.13	47.13	47.13	47.13	47.13	47.13
Course Calendar	4.59	60.23	60.23	-60.23	60.23	60.23	60.23	60.23
User calendar(student/teacher)	4.23	50.70	50.70	50.70	50.70	50.70	50.70	50.70
News/Announcement	4.36	54.27	54.27	54.27	54.27	54.27	54.27	54.27
News/Announcement about the course to student	4.41	55.46	55.46	55.46	55.46	55.46	55.46	55.46
News/Announcement about the course to teachers	4.32	53.08	53.08	53.08	53.08	53.08	53.08	53.08
Content	4.73	63.80	63.80	63.80	63.80	63.80	63.80	63.80

CRITERIA	Avg	t score	Moodle	Ilias	Dokeos	Docebo	Claroline	Efront
Multimedia	4.55	59.04	59.04	59.04	59.04	59.04	59.04	59.04
Wiki	3.86	41.18	41.18	-41.18	-41.18	41.18	41.18	-41.18
Glossary	3.91	42.37	42.37	42.37	-42.37	42.37	-42.37	42.37
FAQ	3.95	43.56	-43.56	-43.56	43.56	43.56	-43.56	-43.56
Links	3.86	41.18	41.18	41.18	41.18	41.18	41.18	-41.18
Games	3.36	28.08	-28.08	-28.08	-28.08	-28.08	-28.08	-28.08
On-line polls /Surveys	4.09	47.13	47.13	47.13	47.13	47.13	-47.13	47.13
E-mail notifications	4.41	55.46	55.46	55.46	55.46	55.46	-55.46	55.46
E-mail lists	4.14	48.32	48.32	48.32	48.32	48.32	-48.32	48.32
Internal e-mail service	3.45	30.46	30.46	30.46	30.46	30.46	-30.46	30.46
Receiving relevant information about the course by SMS	3.32	26.89	-26.89	-26.89	-26.89	26.89	-26.89	-26.89
Test	4.73	63.80	63.80	63.80	63.80	63.80	63.80	63.80
Online Test	4.73	63.80	63.80	63.80	63.80	63.80	63.80	63.80
Question Bank	4.45	56.66	56.66	56.66	56.66	56.66	56.66	56.66
Create test (multiple choice questions, fill in the blanks, true/flase questions,...)	4.64	61.42	61.42	61.42	61.42	61.42	-61.42	61.42
Assessments	4.59	60.23	60.23	60.23	60.23	60.23	60.23	60.23
Submission management system	4.59	60.23	60.23	60.23	60.23	-60.23	60.23	60.23
Grade Book	4.36	54.27	54.27	-54.27	54.27	54.27	-54.27	-54.27
Syllabus	4.41	55.46	-55.46	55.46	55.46	-55.46	55.46	55.46
Create syllabus from template (via Wizard)	3.77	38.79	-38.79	38.79	38.79	-38.79	38.79	38.79
Import Syllabus	3.91	42.37	42.37	42.37	42.37	42.37	42.37	42.37
Additional information about the teaching staff (other courses, location, tutoring, research…)	3.77	38.79	38.79	38.79	38.79	38.79	-38.79	-38.79
File Sharing	4.27	51.89	51.89	51.89	51.89	51.89	51.89	51.89
Students can upload projects, images or any files to a shared library.	4.23	50.70	-50.70	50.70	-50.70	50.70	-50.70	-50.70
Instructors can upload projects, images or any files to a shared library.	4.36	54.27	54.27	54.27	54.27	54.27	54.27	54.27
Instructors can edit their text files in their folder using a browser.	3.91	42.37	42.37	42.37	42.37	-42.37	42.37	-42.37
Students can edit their text files in their folder using a browser.	3.91	42.37	42.37	42.37	-42.37	-42.37	-42.37	-42.37
Lists of students (for all courses in which is enrolled)	4.32	53.08	53.08	53.08	53.08	53.08	53.08	53.08
Users can create a home page for themselves	3.27	25.70	25.70	-25.70	25.70	-25.70	-25.70	-25.70
Virtual Classroom	3.86	41.18	41.18	-41.18	41.18	41.18	41.18	41.18
Chat tool for messaging	4.05	45.94	45.94	-45.94	45.94	45.94	45.94	45.94
The system creates archive logs for all chat rooms.	3.91	42.37	42.37	-42.37	42.37	42.37	-42.37	42.37
Real-time slides	3.77	38.79	-38.79	-38.79	38.79	-38.79	-38.79	-38.79
E-blackboard	4.00	44.75	-44.75	-44.75	44.75	-44.75	-44.75	-44.75
Videoconference	4.23	50.70	-50.70	-50.70	50.70	50.70	-50.70	-50.70

CRITERIA	Avg	t score	Moodle	Ilias	Dokeos	Docebo	Claroline	Efront
Student Presentation	4.14	48.32	-48.32	-48.32	48.32	-48.32	-48.32	-48.32
Search	4.41	55.46	55.46	55.46	55.46	55.46	-55.46	-55.46
Incorporate web search engines (Google, AltaVista, etc.)	3.64	35.22	-35.22	-35.22	35.22	-35.22	-35.22	-35.22
Search within the portal	4.41	55.46	55.46	55.46	55.46	55.46	-55.46	-55.46
Help	4.68	62.61	62.61	62.61	62.61	62.61	62.61	62.61
Help about using the tool	4.59	60.23	60.23	60.23	60.23	60.23	60.23	60.23
Help desk	4.14	48.32	48.32	48.32	48.32	48.32	-48.32	-48.32
Reports	4.59	60.23	60.23	60.23	60.23	60.23	60.23	60.23
Student Reports	4.73	63.80	63.80	63.80	63.80	63.80	63.80	63.80
Course Reports	4.82	66.18	66.18	66.18	66.18	66.18	66.18	66.18
System Reports	4.68	62.61	62.61	62.61	62.61	62.61	62.61	62.61
Generic Queries	4.36	54.27	54.27	54.27	54.27	-54.27	54.27	54.27
Total Functionality Score			1929.35	1503.30	2388.73	1823.18	822.05	1029.62

Table 24 shows the total functionality scores as raw score and standard score. T-scores are used as standard score with formula ($t\ score = 50 + 10 * (\frac{x-\mu}{\sigma})$). According to standard scores, the comparison of functionality scores of LMSs is given at Figure 59.

Table 24. Summary of Functionality Score and Standard Functionality Score

	Raw Score	Standard Score
Claroline	822.05	37.03
Docebo	1823.18	54.10
Dokeos	2388.73	63.75
Efront	1029.62	40.57
Ilias	1503.30	48.65
Moodle	1929.35	55.91
Average	1582.70	
Std. Dev.	586.26	

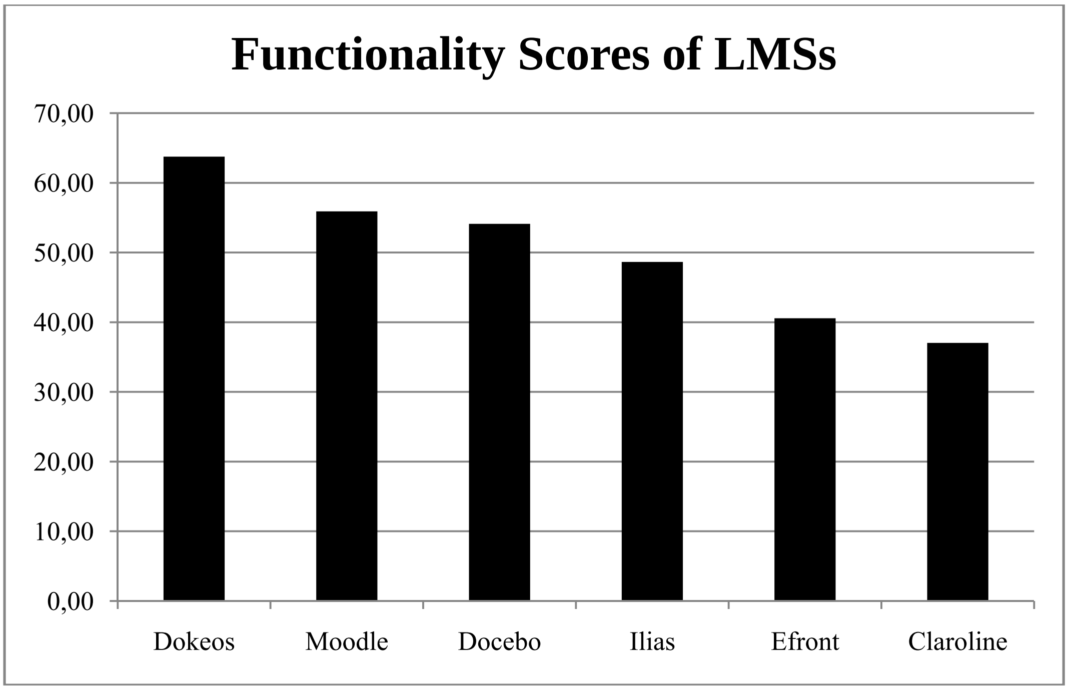


Figure 59. Comparison of functionality scores of LMSs

Accessibility

Additionally, ISO/IEC FCD 24751-1 Individualized Adaptability and Accessibility in E-learning, Education and Training Part 1 standard describes how learning systems can be more accessible. W3C/WAI Web Content Accessibility Guidelines describes how to create accessible content (Web Accessibility Initiative, 2008). Web Content Accessibility Guidelines 1.0 explains the ways of both making Web Content accessible to people with disabilities and promoting accessibility. Thus, people can find information on the Web more quickly by following these guidelines. That document also includes an appendix that organizes all of the checkpoints by topic and priority, which are identified in the appendix include images, multimedia, tables, frames, forms, and scripts, therefore checkpoints directly improve the performance of Web services while reducing the maintenance effort required (Web Accessibility Initiative, 2008).

For accessibility criteria measurement, Functional Accessibility Evaluator (FAE) free online service, which is developed by University of Illinois at Urbana-Champaign, was used. Moreover, FAE is useful for development of functionally accessible web resources and analyzes online systems for markup that is consistent with the use of CITES/DRES HTML Best Practices, which is statement of techniques for implementation of the W3C (Functional Accessibility Evaluator, 2007).

Running the Functional Accessibility Evaluator was on http://fae.cita.uiuc.edu/index.php web site by typing the URL of the systems to evaluate accessibility criteria. All systems were tested "Include all third-level pages" options of depth of evaluation. "Include all third-level pages" will cause all pages linked from the top- and second-level pages (with domain restrictions as explained below) to be included in the analysis. Test evaluation summaries of the learning management systems are at below.

From Table 25 to Table 30, each system accessibility test results summaries were listed according to navigation and orientation, test equivalents, scripting, styling and HTML Standard with status.

Table 25. Claroline Test Evaluation Summaries in HTML Best Practices Main Categories

	Status	% Pass	% Warn	% Fail
Navigation & Orientation	Partially Implemented	62	0	37
Text Equivalents	Almost Complete	72	27	0
Scripting	Not Applicable	0	0	0
Styling	Partially Implemented	88	0	11
HTML Standards	Partially Implemented	66	0	33

Table 26. Docebo Test Evaluation Summaries in HTML Best Practices Main Categories

	Status	% Pass	% Warn	% Fail
Navigation & Orientation	Partially Implemented	71	13	15
Text Equivalents	Complete	100	0	0
Scripting	Not Applicable	0	0	0
Styling	Partially Implemented	78	0	21
HTML Standards	Complete	100	0	0

Table 27. Dokeos Test Evaluation Summaries in HTML Best Practices Main Categories

	Status	% Pass	% Warn	% Fail
Navigation & Orientation	Partially Implemented	29	11	59
Text Equivalents	Complete	100	0	0
Scripting	Not Applicable	0	0	0
Styling	Partially Implemented	75	12	12
HTML Standards	Complete	100	0	0

Table 28. Efront Test Evaluation Summaries in HTML Best Practices Main Categories

	Status	% Pass	% Warn	% Fail
Navigation & Orientation	Not Implemented	12	0	87
Text Equivalents	Almost Complete	87	12	0
Scripting	Not Applicable	0	0	0
Styling	Partially Implemented	35	20	45
HTML Standards	Almost Complete	66	33	0

Table 29. Ilias Test Evaluation Summaries in HTML Best Practices Main Categories

	Status	% Pass	% Warn	% Fail
Navigation & Orientation	Partially Implemented	46	0	53
Text Equivalents	Not Implemented	0	0	100
Scripting	Not Implemented	0	0	100
Styling	Partially Implemented	83	0	16
HTML Standards	Complete	100	0	0

Table 30. Moodle Test Evaluation Summaries in HTML Best Practices Main Categories

	Status	% Pass	% Warn	% Fail
Navigation & Orientation	Partially Implemented	75	7	17
Text Equivalents	Complete	100	0	0
Scripting	Not Implemented	0	0	100
Styling	Almost Complete	83	14	2
HTML Standards	Partially Implemented	90	3	6

Summary

Accesibility scores simply calculated as sum of the pass percentage of navigation & orientation, text equivalents, styling and HTML standards of the systems (Table 31), after raw score is calculated standard score of is calculated with t-score formula (50+10*((x-µ)/σ)) (Table 32). Figure 60 shows the comparison of accessibility scores of LMSs.

Table 31. Accessibility Score of the Sytems

	Clarorine % Pass	Docebo % Pass	Dokeos % Pass	Efront % Pass	Ilias % Pass	Moodle % Pass
Navigation & Orientation	62	71	29	12	46	75
Text Equivalents	72	100	100	87	0	100
Scripting	Not Applicable					
Styling	88	78	75	35	83	83
HTML Standards	66	100	100	66	100	90
Total	288	349	304	200	229	348

Table 32.Summary of Accessibility Score and Standard Accessibility Score

	Raw Score	Standard Score
Clarorine	288.00	50.27
Docebo	349.00	60.22
Dokeos	304.00	52.88
Efront	200.00	35.92
Ilias	229.00	40.65
Moodle	348.00	60.06
Average	286.33	
Std. Dev	61.29	

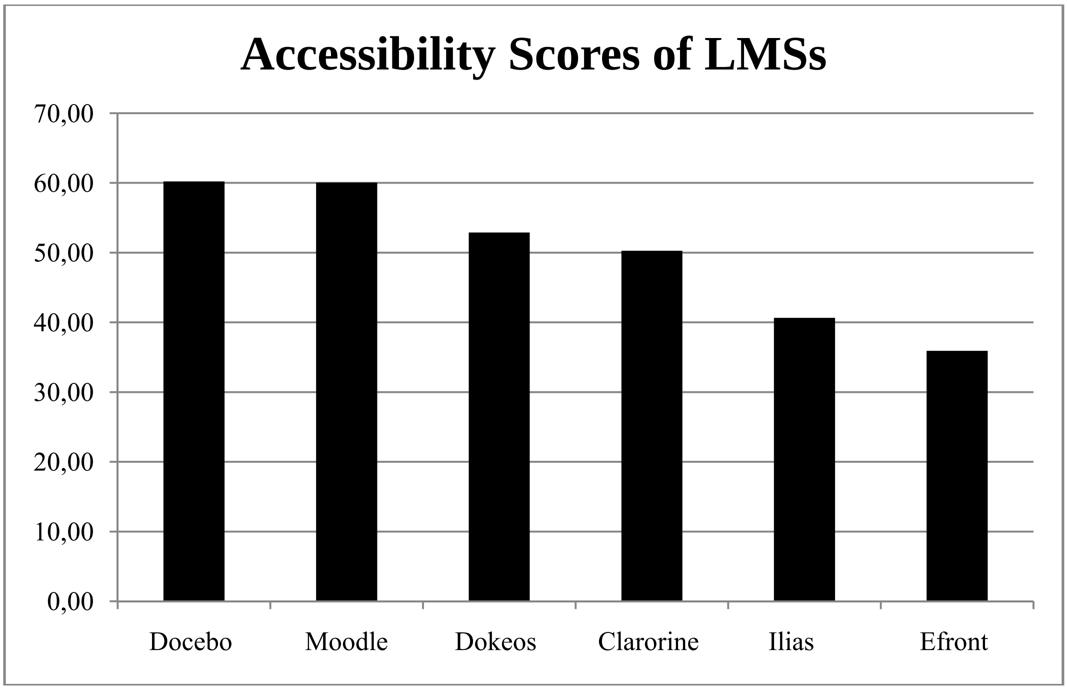


Figure 60. Comparison of accessibility scores of LMSs

General Comparison

In order to calculate general score for each LMS weighted sum formula is used.

$$\mathrm{Q_{LMS}} = \sum_{i=0}^{n} \mathrm{W_i S_i}$$

where $\mathrm{S_i}$ is the standard score of the LMS for each criterion (Table 33) , $\mathrm{W_i}$ is the weight factor of a each criteria which is calculated as standard t score

$$W_i = 50 + 10 * (\frac{x_i - \mu}{\sigma})$$

where x_i is average of criteria in the survey, μ is mean of the survey, and σ is the standard deviation(Table 34).

Table 33. Weight Factors of Criteria

Criteria	Average	Standard Score
Standards	2.41	35.69
Performance	3.09	52.20
Security	2.86	46.70
Accessibility	3.09	52.20
Functionality	3.55	63.21
Average	3.00	
Std. Dev.	0.41	

Table 34. General Comparison of the Systems.

	Standards (35.69)	Performance (52.20)	Security (46.70)	Accessibility (52.20)	Functionality (63.21)	Score	Standard Score	Rank
Docebo	42.77	57.60	50.39	60.22	54.10	2690.02	60.44	1
Dokeos	42.77	61.64	40.95	52.88	63.75	2689.36	60.40	2
Moodle	50.43	32.87	50.12	60.06	55.91	2505.09	50.28	3
Claroline	42.77	51.05	68.64	50.27	37.03	2472.32	48.48	4
Ilias	68.28	50.31	41.83	40.65	48.65	2442.60	46.85	5
Efront	52.98	46.53	48.06	35.92	40.57	2200.60	33.55	6
Average						2500.00	50	
Std. Dev.						182.04	10	

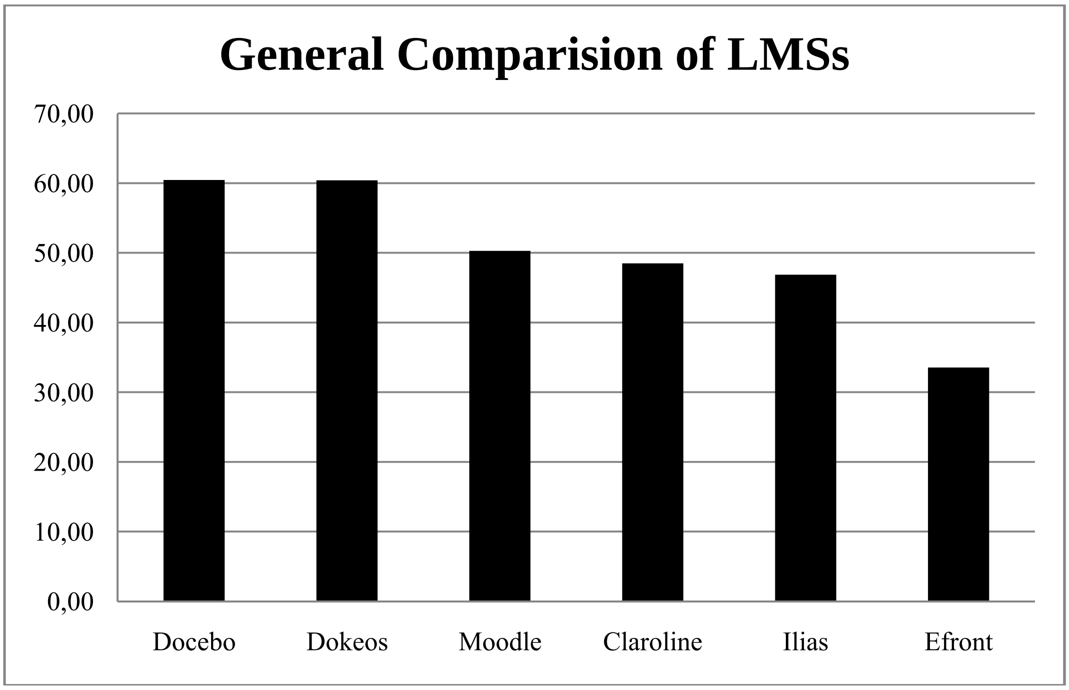


Figure 61. General comparison of LMSs

CHAPTER IV

CONCLUSION

The aim of this thesis is to provide a prototype to evaluate e-learning systems` success and effectiveness by addressing the success criteria and measuring these criteria with chosen tools. In this thesis study, some automated software tools are examined and then selected to execute to test the e-learning systems’ software. The variables that were used to evaluate e-learning system success are performance, stress, accessibility, security and standard compliance, and functionality.

E-learning systems have an impact on both individual development and other work systems in businesses. In addition, e-learning has new aspect to revolutionize the learning by making it individual rather than institution-based. Rokou and Rokos (2004) explained that on-line, web-based education represents one of the largest search and development areas with considerable financial interest. The goals of various sectors with different sizes are both working on producing more economical and productive systems, and concentrating on learning with technology to improve the effectiveness and quality of education. However, measuring the effectiveness and quality of e-learning is a sophisticated issue. For efficient e-learning applications, reliable ways is needed to measure the success, quality and effectiveness of the e-learning system, whereas there is no one exact way to measure the effectiveness and quality of e-learning. Besides, defining the effectiveness or quality of e-learning is a complicated issue on its own.

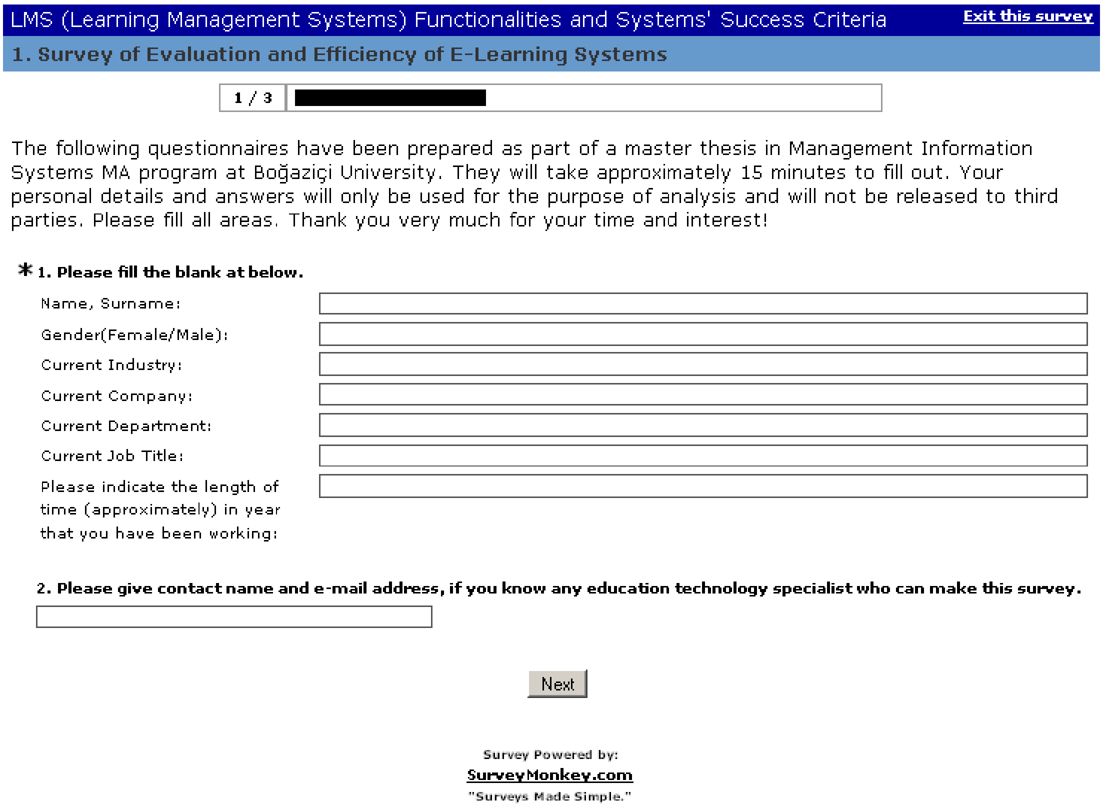
The technology of the systems should have several characteristics that make the learner's and the instructor's experience enjoyable in order for successful e-learning. The success of e-learning systems cannot be defined or evaluated by using single criterion, so the systems must take several criteria into account.

Various web-based learning systems have been developed to support e-learning. In this study, Learning Management Systems were chosen to assess e-learning systems' success and efficiency, inasmuch as LMSs automate many of the e-learning processes. Furthermore, LMSs provide e-learning platforms, which are delivery mechanisms through the Internet in order to allow learners from all over the world to access a number of learning tools such as student management and access control, assessment, forum, gradebook, a conferencing system, a chat area, electronic mail, survey, student self evaluation, on-line quizzes, searchable glossary, student progress tracking, course management, course content searching and more. With the help of these functionalities, e-learning platforms are answered to the needs and demands of the users. The intent of the study will be evaluating e-learning systems efficiency in the aspect of functionalities of the systems, performance, security, accessibility and standard compliance. Furthermore, it is assumed that functionality, accessibility, performance, security and standard compliance evaluation results of the system can be indicators of the whole e-learning system`s success. Some commercial and non-commercial tools are available for measuring or strengths of these success criteria. Software must be tested in order to achieve quality that meets expectations, and software testing is a necessity to help attain any desired level of software quality.

In the thesis, six open source learning management systems were selected, namely Moodle, Ilias, Dokeos, Docebo, Claroline and Efront to evaluate based on the previously defined criteria. Except functionality each criterion was evaluated with a tool. For each criterion a score was calculated, the scale of each criterion score was different due to calculation method. In order to compare different criterion scores of LMSs, the scores were standardized. T-score calculation was used for each criterion, so that it was possible to calculate a total score for each LMS using these scores and weights of criteria which was calculated using the results of the survey. The performance criterion of LMSs was assessed with OpenSTA testing tool. The process and steps of generating script for both load and stress tests were almost the same for all LMSs. After stress and load performance test scores were calculated for each LMS, performance test scores were calculated through these scores by standardization. According to overall calculation, the e-learning platform scores from minimum to maximum were in order of Moodle, Efront, Ilias, Claroline, Docebo and Dokeos. Moreover, the security criterion of LMSs was assessed with Acunetix Web Vulnerability Scanner 4.0. When security test results were compared by taking different levels of vulnerabilities into account, Ilias platform had maximum vulnerabilities in total. The order of the systems from maximum to minimum vulnerabilities was arranged like that Dokeos, Ilias, Efront, Moodle ,Docebo and then Clarolline. Furthermore, while calculating total SCORM scores with SCORM Conformance Test Suite, not only certification and test results but also SCORM 1.2 and SCORM 2004 Standards Compliance were taken into account with different weights. According to these test results, Claroline, Docebo and Dokeos had the same score, which was minimum, and the order of other systems' standards compliance from maximum to minimum was Moodle, Efront and then Ilias. Additionally, functionality comparison was done to determine whether or not the systems had the selected functional requirements. Different weights were calculated for each functional requirement using results of the survey which was answered by 22 specialists in e-learning/distance learning or LMS developers. Standardized average score of each item was used as weight for the functional requirements. The order of other systems' total score of functionalities from minimum to maximum was Claroline, Efront, Ilias, Docebo, Moodle and Dokeos. In addition, for accessibility criteria measurement, Functional Accessibility Evaluator (FAE) free online service was used. The order of accessibility scores, which was calculated as sum of the pass percentage of navigation & orientation, text equivalents, styling and HTML standards of the systems, was Efront, Ilias, Claroline, Dekeos, Moodle and Docebo from minimum to maximum scores. Finally, weighted sum of the standardized score of the automated software test results were used to evaluate systems overall success according to the defined criteria. Weights were calculated according to the third part of survey. The order of LMSs' general comparison was Docebo, Dokeos, Moodle, Claroline, Ilias and Efront, whereas the scores of Dokeos and Docebo were very close to each other. Assessment of e-learning systems success was focused on software testing, the whole e-learning process was not considered. While evaluation the criteria, software or technical properties of e-learning systems were taken into account by ignoring other criteria, such as compatibility, maintainability, modularity, users' perspective, social interaction, collaboration, teaching, learning and support of students' interactivity. Usability of the systems could be evaluated with wide range of time and users; therefore there was an opportunity to evaluate robustness of the e-learning systems in both sides of users' perspective and systems properties. Furthermore, just LMSs were assessed as an indicator of e-learning systems' success with related criteria. All LMSs in the study were based on PHP scripting language and MySQL database and Apache for web and database server. Different technologies depended e-learning systems could be used like Java, ASP.NET or PERL, while evaluating the e-learning systems' success. The other e-learning tools could also be used for evaluation, such as LCMS or Content Management System (CMS). In addition, other information systems can be also evaluated with the pre-defined criteria. The success of other domains, which can be ERP/MRP or CRM, and other web application systems can be evaluated with the similar evaluation methods of LMSs.

APPENDIX

Appendix A. Screenshots of the Survey



LMS (Learning Management Systems) Functionalities and Systems' Success Criteria

1. Survey of Evaluation and Efficiency of E-Learning Systems

The following questionnaires have been prepared as part of a master thesis in Management Information Systems MA program at Boğaziçi University. They will take approximately 15 minutes to fill out. Your personal details and answers will only be used for the purpose of analysis and will not be released to third parties. Please fill all areas. Thank you very much for your time and interest!

*** 1. Please fill the blank at below.**

Name, Surname: ____

Gender(Female/Male): ____

Current Industry: ____

Current Company: ____

Current Department: ____

Current Job Title: ____

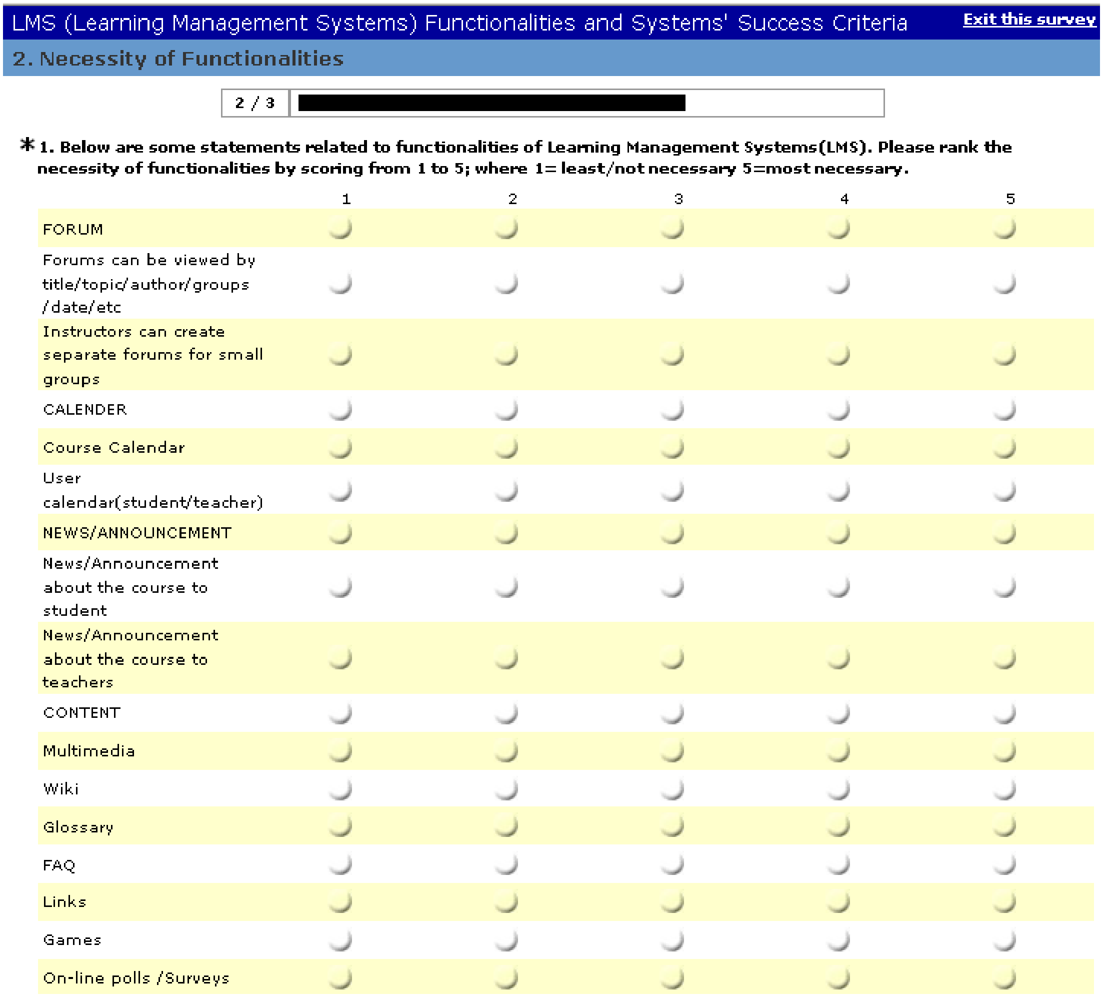
Please indicate the length of time (approximately) in year that you have been working: ____

2. Please give contact name and e-mail address, if you know any education technology specialist who can make this survey.

____

Survey Powered by:
SurveyMonkey.com
"Surveys Made Simple."

Figure 62. First part of the survey: demographic information



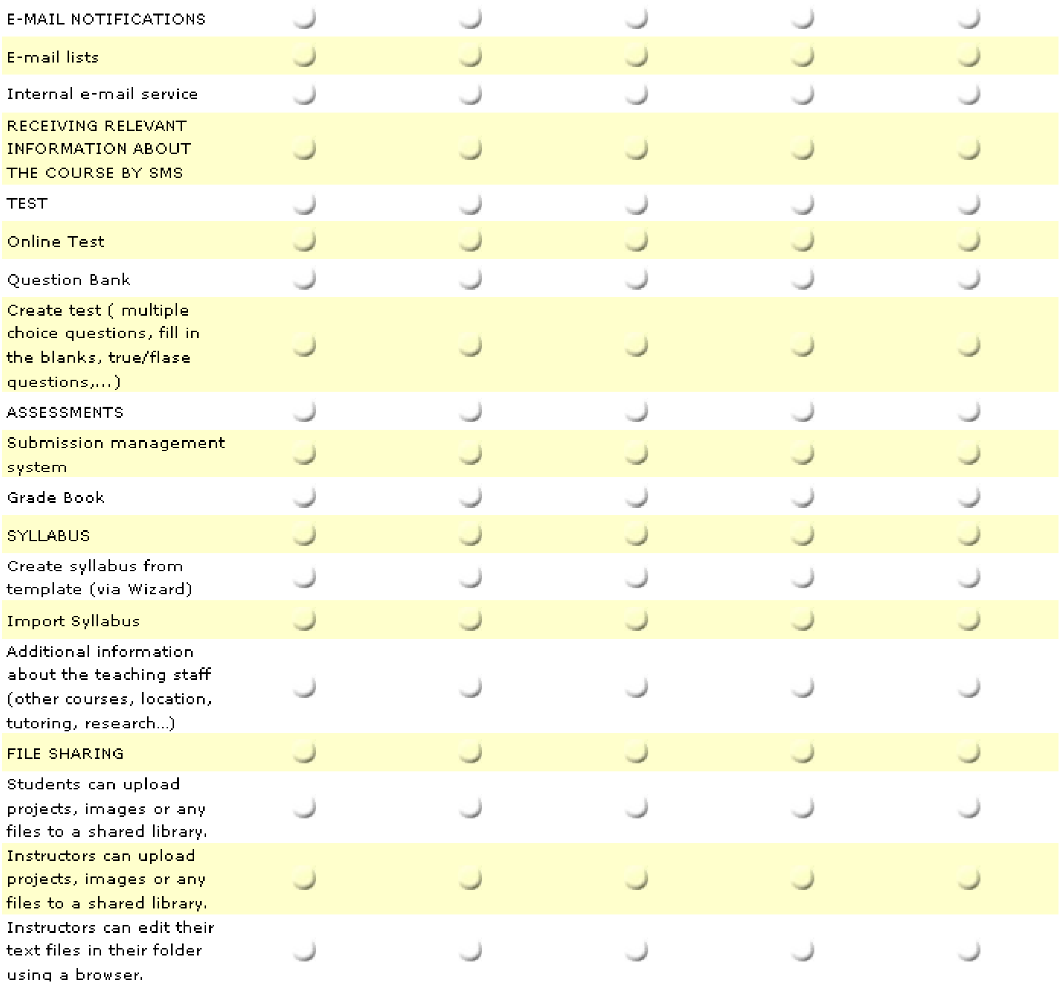
LMS (Learning Management Systems) Functionalities and Systems' Success Criteria

2. Necessity of Functionalities

* 1. Below are some statements related to functionalities of Learning Management Systems(LMS). Please rank the necessity of functionalities by scoring from 1 to 5; where 1= least/not necessary 5=most necessary.

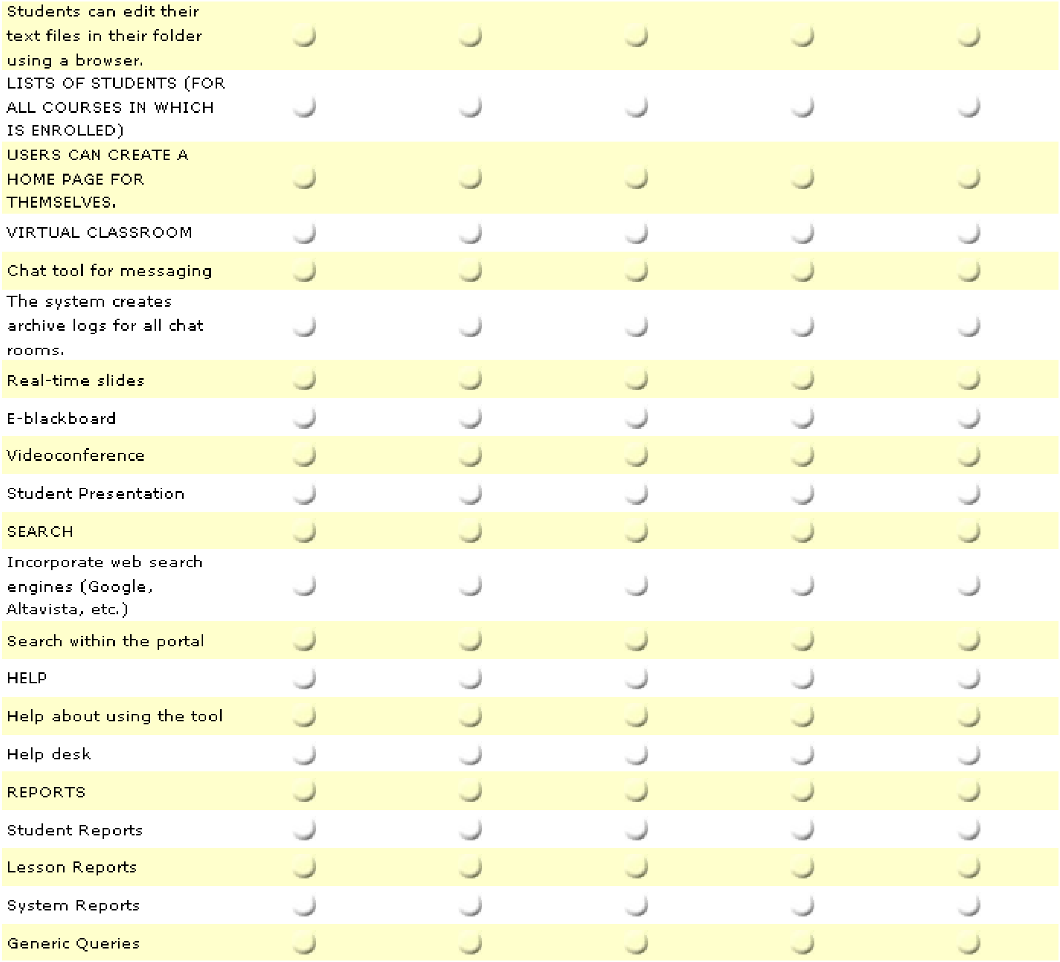
	1	2	3	4	5
FORUM	○	○	○	○	○
Forums can be viewed by title/topic/author/groups /date/etc	○	○	○	○	○
Instructors can create separate forums for small groups	○	○	○	○	○
CALENDER	○	○	○	○	○
Course Calendar	○	○	○	○	○
User calendar(student/teacher)	○	○	○	○	○
NEWS/ANNOUNCEMENT	○	○	○	○	○
News/Announcement about the course to student	○	○	○	○	○
News/Announcement about the course to teachers	○	○	○	○	○
CONTENT	○	○	○	○	○
Multimedia	○	○	○	○	○
Wiki	○	○	○	○	○
Glossary	○	○	○	○	○
FAQ	○	○	○	○	○
Links	○	○	○	○	○
Games	○	○	○	○	○
On-line polls /Surveys	○	○	○	○	○

Figure 63. Second part of the survey: necessity of functionalities (page 1)



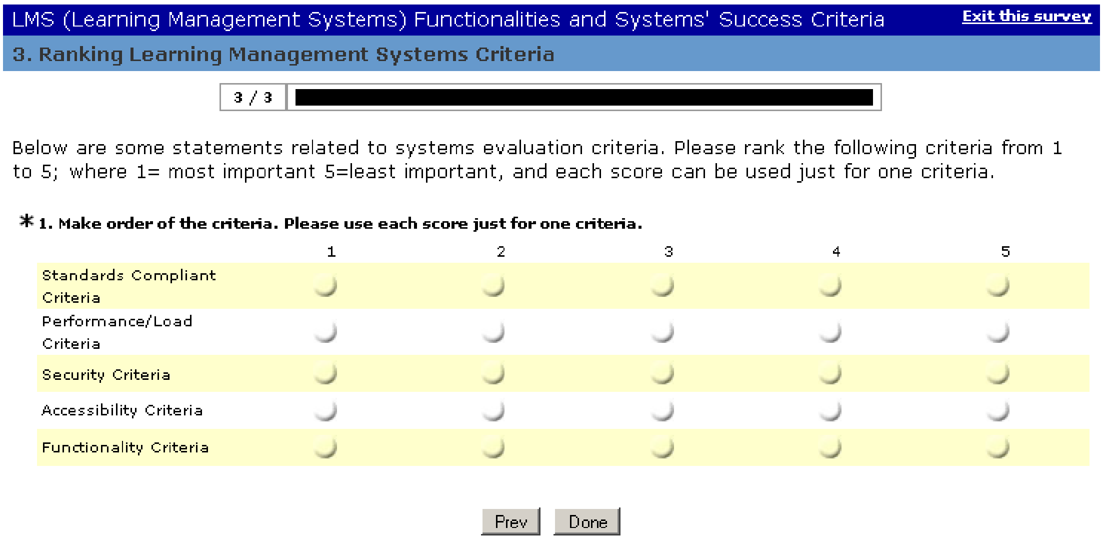
E-MAIL NOTIFICATIONS	○	○	○	○	○
E-mail lists	○	○	○	○	○
Internal e-mail service	○	○	○	○	○
RECEIVING RELEVANT INFORMATION ABOUT THE COURSE BY SMS	○	○	○	○	○
TEST	○	○	○	○	○
Online Test	○	○	○	○	○
Question Bank	○	○	○	○	○
Create test (multiple choice questions, fill in the blanks, true/flase questions,...)	○	○	○	○	○
ASSESSMENTS	○	○	○	○	○
Submission management system	○	○	○	○	○
Grade Book	○	○	○	○	○
SYLLABUS	○	○	○	○	○
Create syllabus from template (via Wizard)	○	○	○	○	○
Import Syllabus	○	○	○	○	○
Additional information about the teaching staff (other courses, location, tutoring, research...)	○	○	○	○	○
FILE SHARING	○	○	○	○	○
Students can upload projects, images or any files to a shared library.	○	○	○	○	○
Instructors can upload projects, images or any files to a shared library.	○	○	○	○	○
Instructors can edit their text files in their folder using a browser.	○	○	○	○	○

Figure 64. Second part of the survey: necessity of functionalities (page 2)



Students can edit their text files in their folder using a browser.	○	○	○	○	○
LISTS OF STUDENTS (FOR ALL COURSES IN WHICH IS ENROLLED)	○	○	○	○	○
USERS CAN CREATE A HOME PAGE FOR THEMSELVES.	○	○	○	○	○
VIRTUAL CLASSROOM	○	○	○	○	○
Chat tool for messaging	○	○	○	○	○
The system creates archive logs for all chat rooms.	○	○	○	○	○
Real-time slides	○	○	○	○	○
E-blackboard	○	○	○	○	○
Videoconference	○	○	○	○	○
Student Presentation	○	○	○	○	○
SEARCH	○	○	○	○	○
Incorporate web search engines (Google, Altavista, etc.)	○	○	○	○	○
Search within the portal	○	○	○	○	○
HELP	○	○	○	○	○
Help about using the tool	○	○	○	○	○
Help desk	○	○	○	○	○
REPORTS	○	○	○	○	○
Student Reports	○	○	○	○	○
Lesson Reports	○	○	○	○	○
System Reports	○	○	○	○	○
Generic Queries	○	○	○	○	○

Figure 65. Second part of the survey: necessity of functionalities (page 3)



LMS (Learning Management Systems) Functionalities and Systems' Success Criteria

3. Ranking Learning Management Systems Criteria

Below are some statements related to systems evaluation criteria. Please rank the following criteria from 1 to 5; where 1= most important 5=least important, and each score can be used just for one criteria.

*** 1. Make order of the criteria. Please use each score just for one criteria.**

	1	2	3	4	5
Standards Compliant Criteria	○	○	○	○	○
Performance/Load Criteria	○	○	○	○	○
Security Criteria	○	○	○	○	○
Accessibility Criteria	○	○	○	○	○
Functionality Criteria	○	○	○	○	○

Figure 66. Third part of the survey: ranking learning management systems criteria

REFERENCES

Abascal, J., Arrue, M., Fajardo, I., Garay, N., & Tomás. J. (2004). The use of guidelines to automatically verify Web accessibility. *Universal Access in the Information Society, 3(1)*, 71-79. Retrieved July 7, 2008, from ProQuest Computing database. (Document ID: 665325271).

About Moodle (n.d) Retrieved June 15, 2008, from MoodleDocs Wiki: http://docs.moodle.org/en/About_Moodle

Acunetix Ltd. (2006). Acunetix Web Vulnerability Scanner Manual v4.0. Retrieved June 15, 2008, from http://www.acunetix.com/ vulnerability-scanner/wvs4manual.pdf

Administrator documentation (n.d) Retrieved June 15, 2008, from MoodleDocs Wiki: http://docs.moodle.org/en/Administrator_documentation

Advanced Distributed Learning – SCORM (2008) Retrieved June 15, 2008, from http://www.adlnet.gov/scorm/

Almutairi, H. (2001). Evaluating Information System Success in Public Organizations: A Theoretical Model and Empirical Validation. Dissertation Abstracts International (UMI No. 3014588).

Alotaiby, F. T. (2005) A component-based functional model for e-learning systems. Dissertation Abstracts International (UMI No. AAT 3176555).

Andersson, T., Eriksson, I. (1996). Measuring the work quality of information systems, working paper. *Proceedings of the 28th annual Hawaii International Conference on Systems Sciences*, Hawaii

Barclay, C. (2008). Towards an integrated measurement of IS project performance: The project performance scorecard. *Information Systems Frontiers, 10*, 331-345, Retrieved June 15, 2008, from http://www.springerlink.com/content/x554r654724x6277/

Bates, R. (2004) A critical analysis of evaluation practice: the Kirkpatrick model and the principle of beneficence, *Evaluation and Program Planning, 27*,341–347

Bell, K. Z. (2006). Optimizing Effectiveness and Efficiency of Software Testing: A Hybrid Approach. Abstracts International (UMI No. 3247004).

Berg, K. (2005). Finding Open options: An Open Source software evaluation model with a case study on Course Management Systems. Unpublished master thesis. University of Tilburg, Tilburg.

Boldyreff, C. (2002). Determination and Evaluation of Web Accessibility. In *Proceedings of the 11th IEEE international Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises* (June 10 - 12, 2002), 35-42. Washington: IEEE Computer Society.

Bretthauer, D. (2002). Open source software: A history. *Information Technology and Libraries, 21 (1)*, 3-10

Cao, J. (2005). Learning with Virtual Mentors: How to Make E-learning Interactive and Effective? Dissertation Abstracts International (UMI No. 3176282).

Chapman, B. & Hall, B. (2004). *LCMS 2004-2005 Report: Comparative Analysis of Enterprise Learning Content Management Systems*, New York: Brandonhall.com Publications.

Chen Y. (2004). *Automated Software Testing in Resource Limited Platforms*. Unpublished master thesis. Dalhousie University, Halifax, Nova Scotia.

Chisholm, W., Vanderheiden, G. & Jacobs, I. (1999). *Web Content Accessibility Guidelines 1.0*. Retrieved June 15, 2008, from http://www.w3.org/TR/WCAG10/

Chu C., Chang, C., Yeh, C.& Yeh Y. (2004). A Web-service oriented framework for building SCORM compatible learning management systems, Information Technology: Coding and Computing. *Proceedings ITCC 2004. International Conference on Volume 1, 156 – 161.*

Claroline .NET (2008). Retrieved March 4, 2008, from http://www.claroline.net/

Coar, K. (2006, July7). *The Open Source Definition*. Retrieved June 15, 2008, from http://opensource.org/docs/osd

CORBA Directory (2008) Retrieved June 15, 2008, from http://corba-directory.omg.org/

Cowan, C. (2003). Software security for open-source systems. *IEEE Security & Privacy Magazine, 1(1)*, 38 – 45.

DeLone, W. & McLean, E. (1992). Information Systems Success: TheQuest for the DependentVariable. *Information Systems Research, 3(1),* 60-95.

Docebo Open Source scorm e-learning, LMS (n.d.). Retrieved March 5, 2008, from http://www.docebo.org/

Dokeos Open Source e-Learning.(n.d.) Retrieved March 1, 2008, from http://www.dokeos.com/ .

Edward R. J. (2002) Implications of SCORM and Emerging E-learning Standards On Engineering Education, *Proceedings of the 2002 ASEE Gulf-Southwest Annual Conference (March 20 – 22, 2002),* American Society for Engineering Education

eFront - A Free, Open Source eLearning and Human Capital Development System (n.d.), Retrieved March 4, from http://www.efrontlearning.com/index.php

eFront Quick Guide (n.d), Retrieved June 15, 2008, from http://www.efrontlearning.com/eFront_quick_guide.pdf

Eriksson, A. T. (1991). A model for IS quality. *Software Engineering Journal*, 152-158

Fallon, C. & Brown, S. (2003). *E-learning Standards: A guide to Purchasing Developing, and Deploying standards-conformant e-learning.* New York: St. Lucie Press.

Feller, J. & Fitzgerald, B. (2000). A framework analysis of the open source software development paradigm. In *Proceedings of the Twenty First international Conference on information Systems* (Brisbane, Queensland, Australia). International Conference on Information Systems. Association for Information Systems, Atlanta, GA, 58-69.

Fielding, R., Gettys, J., Mogul, J., Frystyk, H., Masinter, L., Leach, P., & Berners-Lee T. (1999), Hypertext Transfer Protocol -- HTTP/1.1, RFC 2616,

Free Software Foundation. (2008, March 20). *The Free Software Definition*. Retrieved June 15, 2008, from http://www.gnu.org/philosophy/free-sw.html

Friesen, N.(2005). Interoperability and Learning Objects: An Overview of E-Learning Standardization, *Interdisciplinary Journal of Knowledge and Learning Objects, 1.*

Functional Accessibility Evaluator. (2007). Retrieved June 15, 2008, from http://fae.cita.uiuc.edu/about.php

Gallagher, P. S. (2007). Assessing SCORM 2004 for its Affordances in Facilitating a Simulation as a Pedagogical. Unpublished doctoral dissertation. George Mason University, Fairfax, VA.

Ganchev, I., O'Dromal, M. & Andreev, R. (2007). Functionality and SCORM-compliancy Evaluation of e-Learning Tools, *Seventh IEEE International Conference on Advanced Learning Technologies.*

Hancı, R, (2007). The Effect of Individual Differences on Acceptance of Web-Based Learning Management System: A Case of the Central Bank of the Republic of Turkey, Unpublished master thesis. Middle East Technical University. Ankara.

Harrington, H. J. (1991). *Business Process Improvement: The Breakthrough Strategy for Total Quality, Productivity, and Competitiveness,* McGraw-Hill.

Henley M., Kemp R. (2008). Open Source Software: An introduction. *Computer law & Security report, 24,* 77-85. London: Kemp Little LLP

Home - load testing tool - stress testing tool – WebLOAD (2008). Retrieved June 15, 2008, from http://www.webload.org/

Hwang, S. (2007) Generalized imperfect delay-debugging software reliability models. Abstracts International (UMI No. AAT 3252994).

IBM Corporation Software Group. (2007). *The IBM Rational AppScan lifecycle solution: building Web application security into software and systems delivery*. Retrieved June 15, 2008, from ftp://ftp.software.ibm.com/software /rational/web/brochures/r_appscan_lifecycle.pdf

IBM Corporation Software Group. (2007). *Watchfire AppScan, Version 7.6.* Retrieved June 15, 2008, from ftp://ftp.software.ibm.com/software/ rational/web/datasheets/watchfire_appscan_ds.pdf

ILIAS Information Center - General Information about ILIAS. (2008) Retrieved June 15, 2008, from http://www.ilias.de/docu/goto_docu_cat_580.html

ILIAS Open Source LMS - SCORM 2004, Assessment, Course Management and more.(n.d.) Retrieved March 6, 2008, from http://www.ilias.de/

International Organization for Standardization. (1991). *Software Product Evaluation – Quality Characteristics for Their Use* (ISO-9126). Genéve, International Organization for Standardization.

Islas, E., Pérez, M., Rodriguez, G., Paredes, I., Ávila, I., & Mendoza, M. (2007). E-learning tools evaluation and roadmap development for an electrical utility. *Journal of Theoretical and Applied Electronic Commerce Research, 2(1),* 63-75.

Ismail, J. (2001). The Design of an E-Learning System: Beyond the Hype. *Internet and Higher Education, 4,* 329-36.

Jones, E.R. (2002). Implications of SCORM and Emerging e-learning Standards on Engineering Education*, Proceedings of the 2002 ASEE Gulf-Southwest Annual Conference, Univ. of Louisiana at Lafayette, March 20-22, 2002.*

Jun S. (2002). E-learning: An Evaluation of Knowledge Acquisition in Training. Dissertation Abstracts International (UMI No. 3058298).

Kals, S., Kirda, E., Kruegel, C.,& Jovanovic, N. (2006). SecuBat: a web vulnerability scanner. In *Proceedings of the 15th International Conference on World Wide Web* (Edinburgh, Scotland, May 23 - 26, 2006), 247-256. New York: ACM Press

Kolat S. (2006). Web Tabanlı Uygulamalarda Otomatik Güvenlik Denetim Yazılımlarının İyileştirilmesi [Improving Automated Web Application Vulnerability Scanners]. Unpublished master thesis. Istanbul University, Istanbul.

Kramer, H. (2007). Measuring the Effect of E-Learning on Job Performance. Unpublished doctoral dissertation. Nova Southeastern University.

Krogh, G. & Spaeth, S. (2007). The open source software phenomenon:Characteristics that promote research. *Journal of Strategic Information Systems, 16,* 236–253

Kwun, O. (2004). Information Systems Development and User Behavior: Effects of Organizational Justice. Unpublished doctoral dissertation. The University of Mississippi. Mississippi.

Leiserson E. (n.d). Learning Circuits Glossary. Retrieved June 15, 2008, from http://www.learningcircuits.org/glossary

Madison, M. J. (2005). The Legitimacy of Open Source and Other Software Licenses. *Journal of Internet Law, 8,* 12-25.

McGraw, G. (2003). From the ground up: the DIMACS software security workshop. *IEEE Security & Privacy Magazine, 1(2)*, 59 – 66.

Misra, S. (2000). A Software Test Plan Generation Approach for Pedagogical Purposes. Dissertation Abstracts International (UMI No. AAT MQ65510).

Moodle - A Free, Open Source Course Management System for Online Learning. (n.d.) Retrieved March 1, 2008, from http://moodle.org/

MS Web Application Stress Tutorial (2008) Retrieved June 15, 2008, from http://www.microsoft.com/technet/archive/itsolutions/intranet/downloads/webtutor.mspx?mfr=true

Nagappan, N. (2005). A *Software Testing and Reliability Early Warning (STREW) metric Suite* . Dissertation Abstracts International (UMI No. AAI3162465).

Nathanael, D. & Vassilakopoulou P. (2008). Integrating quality-in-use in the IS acquisition process: an activity perspective. *Cognition, Technology & Work*, Retrieved June 15, 2008, from http://www.springerlink.com/content/yw37gv14666466m5

Oakes, K. (2002). LCMS, LMS--They're not just acronyms but powerful systems for learning. *American Society for Training &Development, Inc., 56(3),* 73-74.

OpenSTA Portal :: HTTP Performance Testing (n.d.) Retrieved June 15, 2008, from http://portal.opensta.org/

OpenSTA User Home Page (2007) Retrieved June 15, 2008, from http://www.opensta.org/

Pantazis C. (2001). *Executive Summary: A Vision of E-Learning for America's Workforce*. Retrieved June 15, 2008, from http://www.learningcircuits.org/2001/aug2001/pantazis.html

Parmar, K., Anane, R. & , Hendley, R. J. Architecture of a SCORM-Compliant Assessment Authoring Tool, *Fifth International Conference on Computational Science and Applications,* pp. 110-117, IEEE

Pedro, J. M. M., Carlos, D. K., Ralf S., & Raquel M, C.G. (2006). Rating the Importance of Different LMS Functionalities. *Frontiers in Education Conference, 36th Annual, 13 – 18*

Poels, G., and Cherfi, S. S. (2006) Information Quality, System Quality and Information System Effectiveness: Introduction to QoIS'06. In *Advances in Conceptual Modeling - Theory and Practice* (pp. 325-328). Retrieved June 15, 2008, from http://www.springerlink.com/content/c4q58314255nh76v

Potter, B. & McGraw, G. (2004). Software security testing. *IEEE Security & Privacy Magazine, 2(5),* 81 – 85.

Rokou, F.P.; & Rokos, Y. (2004). E-learning system for multiple devices presentation with XML technology, *Advanced Learning Technologies, 2004. Proceedings. IEEE International Conference on 30 Aug.-1 Sept. 2004*, 874 – 875

Sabherwal, R., Jeyaraj, A. and Chowa, C. (2006). Information System Success: Individual and Organizational Determinants. *Management Science, 52 (12),* 1849- 1864

Salmela, H. (1997). From information systems quality to sustainable business quality. *Information and Software Technology, 39*, 819-825

Shelly, G.B., Cashman, T. J. & Vermaat, M. E. (1999). *Discovering Computers 2000 – Concepts for a Connected World*, Course Technology Ptr.

Tai L. (2005). Corporate E-learning How E-learning is Created in Three Large Corporations. Dissertation Abstracts International (UMI No. 3168048).

Teacher documentation (n.d) Retrieved June 15, 2008, from MoodleDocs Wiki: http://docs.moodle.org/en/Teacher_documentation

Von Hellens L.A. (1997). Information systems quality versus software quality a discussion from a managerial, an organisational and an engineering viewpoint. *Information and Software Technology, 39 (12)*, 801-808. Retrieved June 15, 2008, from http://dx.doi.org/10.1016/S0950-5849(97)00038-4

Wang, H. & Wang, C. (2003). Taxonomy of security considerations and software quality. *Commun. ACM, 46(6),* 75-78. DOI= http://doi.acm.org/10.1145/777313.777315

Wang, J., Niu, Z., Song, H. & Liu, L. (2007). The Design and Realization of Distributed Learning Management System Based on Internet. *Information Technologies and Applications in Education, 2007. ISITAE '07. First IEEE International Symposium on 23-25 Nov. 2007, 162 – 166*

Watchfire® AppScan® 6.0 Automating Web Application Security Management. (2006) Retrieved June 15, 2008, from http://www.spectrum-systems.com/vendors/watchfire/appscansix-overview.pdf

Web Accessibility Initiative (WAI). (2008). Retrieved June 15, 2008, from http://www.w3.org/WAI/

Web Application Stress Tool (Software) (2002) Retrieved June 15, 2008, from http://www.microsoft.com/downloads/details.aspx?FamilyID=e2c0585a-062a-439e-a67d-75a89aa36495&DisplayLang=en

Weyuker, E. J., & Vokolos, F. I. (2000). Experience with Performance Testing of Software Systems: Issues, an Approach, and Case Study, *IEEE Transactions on Software Engineering, 26 (12).*

Whittaker, J.A.(2000).What is software testing? And why is it so hard? *IEEE Software, 17(1),* 70 – 79.

Wiley, D. A. (2000). Connecting learning objects to instructional design theory: A definition, a metaphor, and a taxonomy. In D. A. Wiley (Ed.), *The Instructional Use of Learning Objects: Online Version*. Retrieved June 15, 2008, from http://reusability.org/read/chapters/wiley.doc

Womble, J. C. (2007). E-Learning: The Relationship among Learner Satisfaction, Self-Efficacy, and Usefulness. Dissertation Abstracts International (UMI No. 3298903).

Xie, T. (2005) Improving effectiveness of automated software testing in the absence of specifications. Dissertation Abstracts International (UMI No. AAT 3183442).

Yaw, D. C. (2005). An Evaluation of E-Learning in Industrial Level Three Based Upon the Kirkpatrick Model. Unpublished doctoral dissertation. College of Technology, Indiana State University, Terre Haute, Indiana