E-learning Indicators Approach to Developing E-learning Software Solutions

Bekim Fetaji[1], Majlinda Fetaji[2]
[1,2] South East European University/Computer Sciences, Tetovo, Macedonia, e-mail: b.fetaji@seeu.edu.mk, m.fetaji@seeu.edu.mk

***Abstract*—The research is proposing a new way of tackling the process of creation of e-learning solutions as interactive environments by integrating and undertaking the software engineering approach based on e-learning indicators. The study proposes and recommends as starting point in any e-learning initiative assessment, measurement and evaluation of concepts and factors that influence e-learning further represented as e-learning indicators. The main research was focused towards software engineering 3 (three) different e-learning interactive environments based on e-learning indicators approach taking into consideration theories of learning and pedagogical approach in their development. The main focus was set on assessing and evaluating the e-learning indicators and based on them designing and building the e-learning software solutions. In order to further analyze issues and propose solutions several e-learning software solutions are designed and build and further analyzed as case study experiments. Also a methodology for evaluation of the e-learning effectiveness is also proposed and assessed. This research makes several contributions: First it proposes a new development methodology called e-learning indicators methodology approach to creation of e-learning software solutions. Secondly it proposes an assessment and evaluation novel methodology called ELUAT (E-learning Usability Attributes Testing) and as measuring instrument the PET (predefined evaluation tasks) inspection technique and gives insights and evaluation of their usage. Thirdly it investigates different learning approaches: 1) the content-oriented, 2) the tool-oriented, and the 3) task-oriented approach and analyses of these models in supporting: project, problem and task based learning.**

***Keywords*—e-learning, indicators, effectiveness, software, solutions, engineering.**

I. Introduction

One of the strongest arguments for promoting e-learning lies in its potential to improve and even revolutionize teaching as well as in aspects of learning to minimize the dimensional constrains of time and location. However different studies review that e-learning at present is still of the static style in a form of "online newspaper form and information transmission" and fails to provide higher level of learning that it would differentiate it and made it better than the classical classroom. Also the same studies show a lot of skepticism in thoughts about e-learning solutions successfulness in general. In order to asses the overall obstacles and barriers to enhanced e-learning this research has been undertaken to asses and measure factors influencing e-learning that are represented as e-learning indicators. Using this approach the objective was to determine the general issues, deficiencies and barriers in e-learning and propose a solution for them by undertaking different Case Studies and experiments. The software engineering issues in designing learning environments are analyzed. Based on that proposed is a model of understanding of e-learning processes in developing e-learning software solutions that are provided as interactive virtual environments. The designed and developed solutions were: 1) XHTML and XML e-learning Interactive tool, 2) e-learning interactive mathematical tool, 3) Information Retrieval Courseware system. Each software solution was designed and build upon e-learning indicators assessment, measurement and evaluation. At the same time we had in mind supporting: project, problem and task based learning for each particular solution separately with motivation to realize a comparative analyses of each one in order to find what is the best approach.

II. Method

The research method was exploratory research to determine the best research design and then constructive research to build the software solution followed by empirical research to evaluate and describe accurately the interaction between the learners and the system being observed. We have defined the e-learning indicators as the important concepts and factors that are used to communicate information about the level of e-learning and their impact on learning that could be measured and described then in simpler terms. We have defined e-learning indicators based on our previous study [7] as: (1) learner education background; (2) computing skills level (3) type of learners they are, (4) their learning style and intelligence, (5) obstacles they face in e-learning (e-learning barriers), (6) attention, (7) e-content (suitability, format preferences), (8) instructional design, (9) organizational specifics, (10) preferences of e-learning logistics; (11) preferences of e-learning design; (12) technical capabilities available to respondents; (13) collaboration; (14) accessibility available to respondents; (15) motivation, (16) attitudes and interest; and (17) performance-self-efficacy (the learner sense their effectiveness in e-learning environment). We have used focus group and a web based survey of academic staff and students for the research of e-learning indicators following guidelines from Fetaji et al. (2007). We have considered the next learning modeling approaches: 1) the content-oriented, 2) the tool-oriented, and the 3) task-oriented approach [5]. We have decided to use different approaches for each software solution in order to support: project, problem and task based learning. This was done in order to have different views and to be in order to analyze, asses and evaluate the impact of each approach in e-learning effectiveness. The data were collected throw ELUAT methodology, focus groups and interviews with prospective users. The purpose of the research realized is in order: (1) to gather information and asses e-learning interactions between human actors and the developed medium of instruction-the software solution, (intervention strategies and content), (2) determine the distance between learner activities and preconceived scenarios. The observed route of a learner was used to give feedback information on the effective learning.

We have used the general principles and guidelines for HCI regarding the software design from [8], and general principles and guidelines for document design and guidelines for online documentation [3]. All this guidelines were closely advised and reviewed when designing each software solution. In order for the software solutions to be successful they have been developed in close consultation, contact and feedback with users. In the case of technology to support learning that means that we consulted with both instructors/teachers and learners.

III. Research Instrument Development

Major challenge for e-learning researchers is to assess e-learning effectiveness.

In order to do that we have proposed a methodology, called ELUAT (E-learning Usability Attributes Testing), which combines an inspection technique with e-learning effectiveness evaluation based on 4 (four) usability attributes we have set. The usability attributes we have set are: 1) Time to learn, 2) Performance speed; 3) Rate of errors; 4) Subjective satisfaction. The e-learning-methodology is necessary for presenting and evaluating e-learning in an efficient aspect. Measuring the given variables an empirical measurement of e-learning effectiveness expressed through usability attributes could be reached. The theoretical basis are pedagogical conceptions defined from [5]:

- Learning according to the constructivist perspective,
- usability of the e-learning environment and
- research about user opinions.

We have based the measuring instrument on the use of predefined evaluation tasks (PET), which precisely describe the activities to be performed during inspection in the form of a predefined tasks, measuring previously assessed usability attributes. We have named it as PET inspection technique and using this technique we evaluated usability attributes using evaluation tasks for a particular scenario. Evaluation tasks in this technique are determined throw designing several user scenarios and choosing the scenarios that include the most of the options of the software. This kind of approach using this technique has shown very effective, straightforward and useful in determining the distance between learner activities and preconceived scenarios in several research project we conducted. Using the ELUAT methodology and PET inspection technique we have gathered information on interactions between human actors (intervention strategies and content). Scenario contains at least a collection of components and a method. The components are roles, activities or activity-structures, which role does what (which activity) and at which moment is determined by the method which is made up of one or many plays formed by a series of acts. In an e-learning environment, information obtained from learner activity contain a certain pedagogical semantic. After deciding to implement the ELUAT evaluation method, we have decided on using prototyping and evaluation during the design process. After each prototyping stage we have decided to conduct ELUAT testing using the Co-discovery technique [3, 4] where more participants worked together to achieve the given tasks. We could not use summative approach because one of its drawbacks is that the design could be fail test on the last minute. However, this approach could be adopted on paper designs. We decided on using formative approach, that is, at specified intervals a prototype is built, based on the current state of the design. Afterwards, the prototype is tested on users, and the results are fed into next stage of enhancement of the design. In addition, we conducted performance measurement test to quantify usability requirements such as time to complete a task, time to learn, rate of errors and subjective satisfaction. We also made observations by watching the users with different backgrounds in computing and took notes by watching them using the prototype in every stage of development. For the final prototype we have conducted ELUAT testing on full product and made testing on several users, different then the ones we used in the prototyping stage. We also prepared ELUAT testing materials like the observation form, pretest questionnaire in order to choose the users and posttest questionnaire in order to get the proper feedback, written explanations for users together with the tasks they have to achieve, logging sheet and observation task outlines checklist.

The observed route of a learner has been used to give feedback information on the level of learning and its effectiveness. We have considered the next learning modeling approaches: the content-oriented, the tool-oriented, and the task-oriented approach, for which we developed the methodology to suite to our specifics.

TABLE I.
PET Inspection Technique Task Based Form

Task n#	Time for:			M	S	E	R	O	H	F	*
	Task completion	Help search	Recover from errors								
		Time to Learn:		Total:							

The PET inspection technique uses the next measurements: M – Menu Error; R – Repeat task; F-Frustrations; S – Selection error; O – Uses online Help, E – Other errors, H - Help calls, *-Subjective Satisfaction (5-very high, 4-high, 3-average, 2-low, 1-very low).

This methodology and the inspection techniques has been used in several different research software solutions as well as for the 3 (three) discussed in this study.

IV. EXPERIMENT 1 - XHTML AND XML E-LEARNING INTERACTIVE TOOL

The e-learning software solution developed has evolved as an idea to be a valuable tool for students and others who want to learn XHTML and XML and Web design.

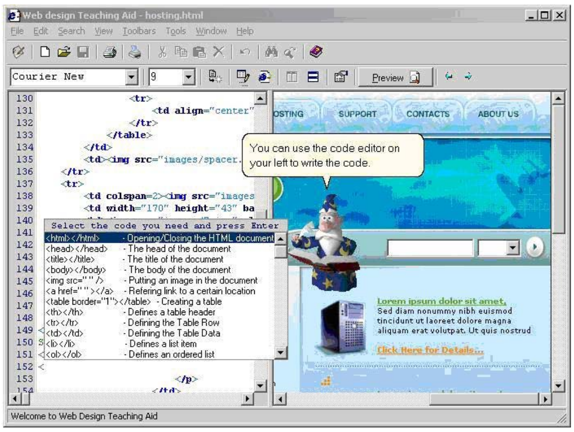

Fig. 1. XHTML and XML E-learning Software Solution Interface.

It offers possibility to learn, design and create web pages only by coding and instantly preview the results.

This benefits both instructors/teachers and students and novice users who want to learn. The creation of web pages only by coding enables to concentrate on learning the principles of the XHTML and XML and provides a solid bases as an introduction for the structural programming principles and create the habit of proper coding styles. The integration of virtual learning environment (VLE) in the form of an e-learning interactive tool and integrated developing environment (IDE) for XHTML and XML contributes in improving the efficiency and quality in learning because of the "hands on approach" and learners having everything they need in one place without the need to leave the application framework results in faster progress and efficiency in learning and grasping concepts. The e-learning interactive software tool as software solution was created in Borland Delphi 6 using Visual Component Library (VCL) classes. The component structure design is a major design issue that has to be realized during the development. In our approach for the software solution we have decided to be modeled and used for project based learning. In Project-Based Learning, students have a great deal of control of the project they will work on and what they will do in the project. [2] suggest that a "deeper understanding of research will come from consideration of the process by which it is conducted". In order to asses the research activities we have used the network analyses planning method.

Activity			Estimation duration (weeks)				
Code	Name	Total	Research	Requirement Specification	Software Design	Help Design	Report
A1	Research	3	0.70	0.40	0.20	0.20	0.50
A2	Software Design	2	0.50	-	0.50	0.50	0.50
A3	Multimedia help Design	2	3	0.50	1	2.50	2
A4	Implementation	9	0.20	0.20	1	2.50	2
A5	Developing Test Plans	1	0.20	0.20	0.20	0.20	0.20
A6	Test	1	0.20	0.20	0.20	0.20	0.20
A7	Evaluation	1	0.20	0.20	0.20	0.20	0.20
A8	Documentation	3	0.70	0.40	0.20	0.20	0.50
	TOTAL	22					

Activity Code	Activity Name	Duration (Weeks)	Dependency
A1	Research	3	
A2	Design	2	
A3	Multimedia Help Design	2	A2
A4	Implementation	9	A2
A5	Develop test plans	1	A2
A6	Test	1	A3, A4, A5
A7	Evaluation	1	A6
A8	Documentation	3	A1, A7

Fig. 2. Calculating Activities, Duration and Dependencies.

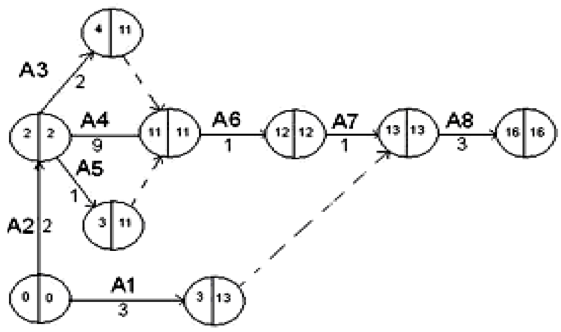

Fig. 3. Calculated Project Activities Network.

The project was estimated to be finished in 16 weeks according to the project activities network calculated. This diagram did not take account of such things as holidays and other such events that may extend the timescales when the project is mapped onto a calendar. There is one critical path in this project, it is shown by a thick line (A2-A4-A6-A7-A8), covering the next activities: Design, Implementation, Test, Evaluation, and Documentation. There is a risk that this project could be late through either the documentation is not being done on time or the software development slipping the planed time schedule. The estimation is calculated barring in mind that some activities can be done in parallel. Each activity in the network is given a time. It is estimated as being the time required for the work to complete the task. In PERT (program evaluation and review techniques), three such estimates are required for every activity these are:

Optimistic time (OT) - the best time possible for completing the activity (16 weeks)

Pessimistic time (PT) - that is, worst possible time (22 weeks for this project)

The most likely **time** (MLT) - (19 weeks)

These three times are used to give a weighted mean form the formula:

$$\text{Time} = \frac{OT + 4*MLT + PT}{6}$$

This calculation is repeated on all activities in the network and is used to predict the probability of completing the project by the required deadline.

$$\text{Time} = \frac{16 + 4 * 19 + 22}{6} = \frac{16 + 76 + 22}{6} = \frac{114}{6} = 19$$

After applying the activity timings approximation approach in PERT, the new estimations are that the project can be completed in 19 weeks.

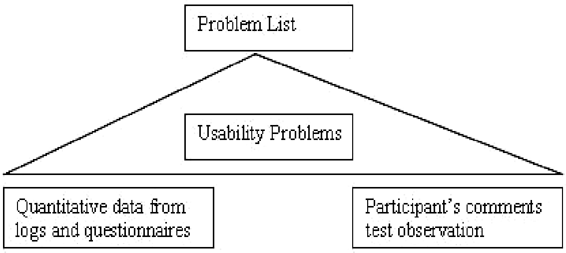

Fig. 4. Triangulation Technique[3].

We have chosen Empirical evaluation method and contextual inquiry, interviewing and observing users in context, in which a prototype is constructed and tested by users in real-world environment iteratively because available analytical tools are limited in scope and strongly theory-based. As the nature of the software solution that we named as Web Design Teaching Aid requires that probable users of the system will be students or occasional people who are novice in Web Design and XHTML/XML and probably also are novice as computer users. The analytical methods such as GOMS or cognitive walkthrough would leave us making a decision that we cannot immediately check for validity, as we do not know how the novice people will react on the system. Given the limited time in disposal to complete the project, we could not afford to leave such a decision unchecked for so long. We have decided that testing must be carried out with participants that will play real user roles. We have recruited students and other people belonging to the two defined user classes, experts and novices, as a representatives of the real users to carry out tasks typical of those they would do in the real world.

According to research studies of [6] for usability testing 5 users are enough, however although we are following guidelines from [6] we have used 10 users just to add weight to the results. After the usability test we had collected the data from the 10 participants we had, were 5 of them were experts while the other 5 novices. In order to handle those data we have used the triangulation technique from [3], were we look at all data at the same time to see how the different data supports each other. We also tabulated the data for the performance measurements using the next usability attributes: time to learn, speed of performance, rate of errors, Subjective satisfaction, and Frustration for the both classes of users Experts and novices.

Here is the tabulated data sheet for time to learn, and speed of performance as well as the general usability requirements measures.

TABLE II.
USABILITY REQUIREMENTS FOR CLASS-2 (USER/ADMINISTRATOR)

Usability Attribute	Measuring instrument	Value to be measured	Current Level average	Worst acceptable	Planned target level	Best possible
Time to learn	Task Scenario	Time to complete task	82.8 s	100 s	80 s	70 s
Speed of performance	Task Scenario	Time to complete task	93 s	110 s	90 s	80 s
Rate of errors	Task Scenario	Number of errors	0.43	4	1	0
Subjective satisfaction	Task Scenario	Satisfaction degree of users	4.03	3	4	5
* number. Subject satisfaction scale: very high 5, high 4, average 3, low 2, very low 1						

TABLE III.
USABILITY REQUIREMENTS FOR CLASS-3 (USER/READER)

Usability Attribute	Measuring instrument	Value to be measured	Current Level average	Worst acceptable	Planned target level	Best possible
Time to learn	Task Scenario	Time to complete task	331 s	400 s	300 s	200 s
Speed of performance	Task Scenario	Time to complete task	346 s	420 s	350 s	250 s
Rate of errors	Task Scenario	Number of errors	5.53	6	4	0
Subjective satisfaction	Task Scenario	Satisfaction degree of users	3.06	1	3	5
* number. Subject satisfaction scale: very high 5, high 4, average 3, low 2, very low 1						

We have organized and analyzed the problems appeared from the testing in two dimensions:

- Scope (how widespread is the problem) and
- Severity (how critical is the problem)

Global problems by scope: Since the polled computer rooms that were used as the test laboratory had no sound card and therefore no sound capabilities the users specially the novice ones were disadvantaged since the main help was the multimedia help were the user can see and hear a live presentation of the software capabilities and usage.

Problems by level of severity: The novice users had a high learning curve, because they did not know XHTML or XML and during the short time of the test they could not learn it, and therefore see and use the full capabilities of the software.

The non-existence of written help topics combined with the disadvantage of not having sound capabilities for the multimedia help has been on of the main severe problems for the users. The final conclusions are summarized for all experimental software solutions and also addresses the specifics of each in the conclusions section of this paper.

V. Experiment 2 - E-learning Interactive Mathematical Tool

The software solution developed has evolved as an idea to be a valuable tool for students and others who want to learn mathematical operation of different types such as system of linear equations, matrices, determinants, functions etc.

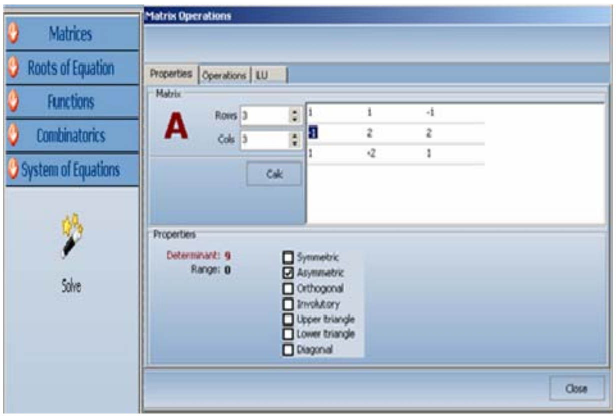

Fig. 5. The Interactive E-learning Mathematical Interface.

During the course “Discrete Mathematics” held in 5th semester in the University, which mainly deals with solving linear systems, finding the roots of the function, drawing and evaluating the graphic of the function and other mathematical operations, the students complained about the course learning content. Their opinion was that the learning content was not enough, and they spent to much time solving and calculating mathematical operations such as matrix and system operations with higher range. It means that they spent a lot of time in operations that are second hand and are simple calculations. We have therefore designed and build a software that will serve as learning tool. The motivations for the research were several. First in order to test our hypothesis which is the best approach in developing e-learning software solutions in our approach for the software solution we have decided to be modeled and used for Problem Based Learning. In Problem-Based Learning, students think, retrieve information for themselves, search for new ideas and apply them using the software solution. Secondly it is a standalone software that has no need to be installed and can be executed from any account especially from student accounts that do not have sufficient privileges in university network. Thirdly compared to Mathematica, Maple, and mathlab it is much simple to use and the learning curve is lower than in those software packages. All this combined with the easiness of execution and usage without exposing learners to hassle with installation procedures and since it does not employ restrictions of any kind it proved it self as very usable software solution. The matrix form contains three tabs: Properties, Operations and LU. The user can calculate the property of the matrix by following actions: (Note: The matrix should be a square matrix to perform these operations.) In order to do that they need to enter number of rows and columns in textboxes (the numbers should be positive) 1) On the grid they enter the matrix values; 2) click on Calc button; 3) As the result users will get the determinant of the matrix, range and the properties in form of checkboxes.

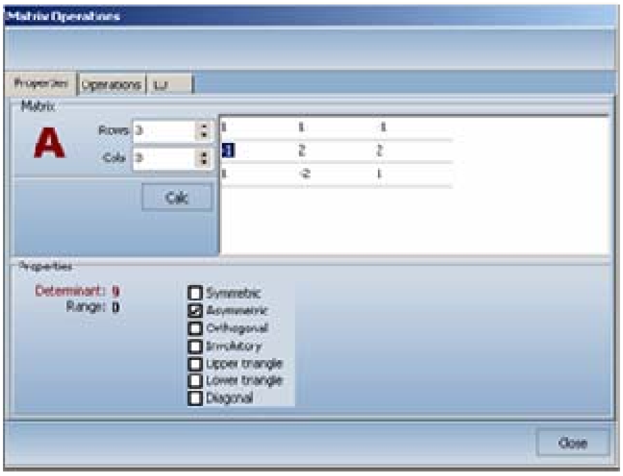

Fig. 6. Properties Tab of the Matrix Operations.

In the operations tab users can calculate few matrix operations, as following: − Enter rows and columns of the first matrix

− On the grid below input values

− On the drop-down list select the action to perform

− If operation addition, subtraction or multiplication is selected the enter number of rows and columns of the second matrix the input the values.

− Click on the Operate button

The result users will se in the result grid, depends on the action selected. In the screenshot there is an example of inverse of matrix.

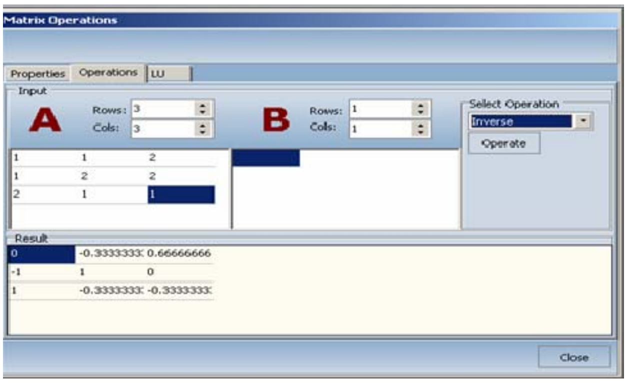

Fig. 7. Operations Tab of the Matrix Operations.

In the following tab can be calculated the LU factorization as following: 1) As an input here users have one matrix, and output two matrices. 2) Input number of rows and columns of the matrix. 3) Input the values on the grid. 4) Click Factorize button

As a result users have two matrices Lower and Upper matrix. The example cab be seen in Figure below.

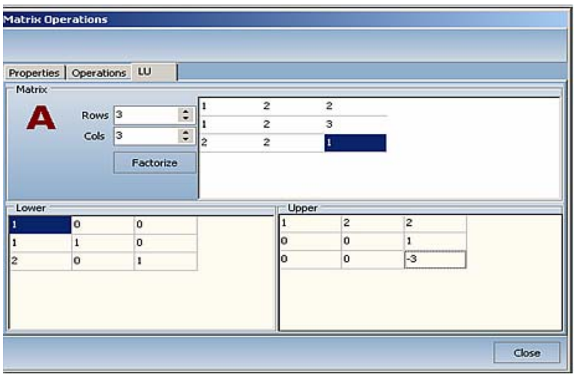

Fig. 8. Lower Upper (LU) Tab of the Matrix Operations

Roots of Equation – Bisection. To find the root of equation with method of Bisection users follows these steps:

- In the f(x) textbox input the function
- Enter Tolerance, Endpoint A, Endpoint B and Steps
- Click Solve button

The result is displayed in the grid. To see the graph of this function user should click on the graph button. An example of this method is given in the Figure below.

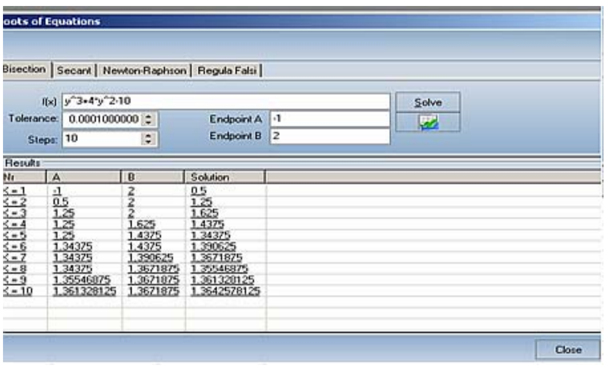

Fig. 9. Roots of Equation - Bisection

Newton-Raphson. This method differs from two upper because there is an additional input box. That is the derived function of main function.To find the root of equation with method of Bisection follows these steps

- In the f(x) textbox input the function
- In the f'(x) textbox input the first derived function
- Enter Tolerance, Endpoint A, Endpoint B and Steps
- Click Solve button

The result is displayed in the grid. To see the graph of this function users should click on the graph button. An example of this method is given in the Figure below.

Regula Falsi. To find the root of equation with method of Regula Falsi users should follow these steps:

- In the f(x) textbox input the function
- Enter Tolerance, Endpoint A, Endpoint B and Steps
- Click Solve button

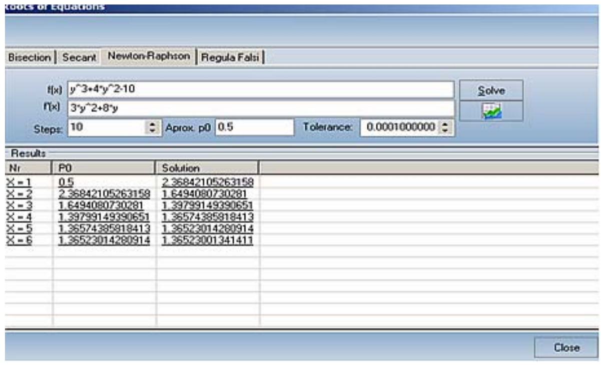

Fig. 10. Newton-Raphson Method

The result is displayed in the grid. To see the graph of this function users should click on the graph button. An example of this method is given in the Figure 9. The most interesting part of this project is evaluation and graphic of the functions. This section uses inline parser which evaluates the expressions entered in text format. In order to evaluate the function follow these steps:

- Input function in Expression textbox
- Input Start, End and Scale by textboxes – start is the first value of the iteration to begin, End is the number that iteration will end and Scale By is the increment value of the function
- If you want to use formula like Sin or Cos you can write them of just click on the list of formulas
- Click Eval button to evaluate this function

The result is shown on the grid, the value X and f(x). Here there are some statistic for the function in statistic section f(min) is the minimum evaluated value of this series while min is the index of the minimum value. f(max) is maximum value of the series while max is the index.

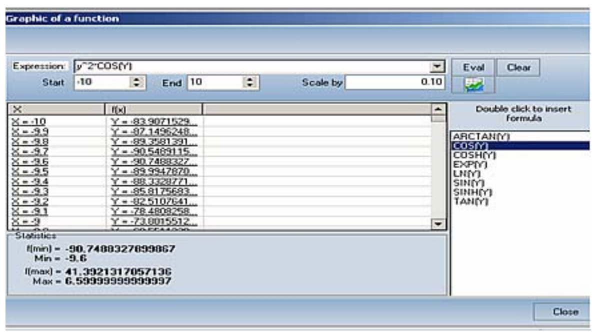

Fig. 11. Function Evaluation.

In the screenshot can be seen that the f(min) = -90.74 for x = -9.6 and f(max) = 41.39 for x = 6.59

According to [6] for usability test 5 users are enough, however we have used 10 users in order to be on the safe side. After the ELUAT and PET test we had collected data from the 10 participants we had all students users. In order to handle those data we have used the triangulation technique from [3], were we look at all data at the same time to see how the different data supports each other.

We also tabulated the data for the performance measurements using the next usability attributes: time to learn, speed of performance, rate of errors, Subjective satisfaction, and Frustration for the both classes of users Experts and novices. Please look at the appendix E for the tabulated data sheets and results. Here is the tabulated data sheet for time to learn, and speed of performance as well as the general usability requirements measures.

TABLE IV.
USABILITY REQUIREMENTS FOR STUDENTS

Usability Attribute	Measuring instrument	Value to be measured	Current Level average	Worst acceptable	Planned target level	Best possible
Time to learn	Task Scenario	Time to complete task	134.7 s	200 s	120 s	90 s
Speed of performance	Task Scenario	Time to complete task	112 s	200 s	110 s	80 s
Rate of errors	Task Scenario	Number of errors	0.63	4	1	0
Subjective satisfaction	Task Scenario	Satisfaction degree of users	3.74	1	3	5
* number. Subject satisfaction scale: very high 5, high 4, average 3, low 2, very low 1						

VI. EXPERIMENT 3 - INFORMATION RETRIEVAL COURSEWARE SYSTEM

In our approach for the Information Retrieval courseware system as software solution we have decided to be modeled and used for Task Based Learning. In Task-Based Learning, students are instructed and leaded by the instructor giving them exact tasks that they will need to go throw during each week in order to apply their knowledge and learning using the software solution.

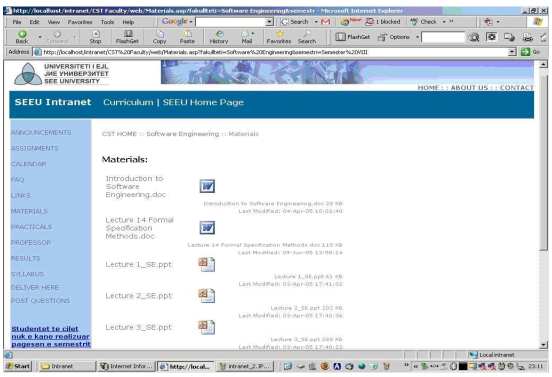

Fig. 12. Intranet Gateway - Lecture View.

In developing the Information Retrieval courseware system we have approached using the spiral life cycle model and followed the usability principles recommended by [1], and [3]. As a platform we have used Microsoft Active Server Pages- ASP. The spiral methodology reflects the relationship of tasks with rapid prototyping, increased parallelism, and concurrency in design and builds activities.

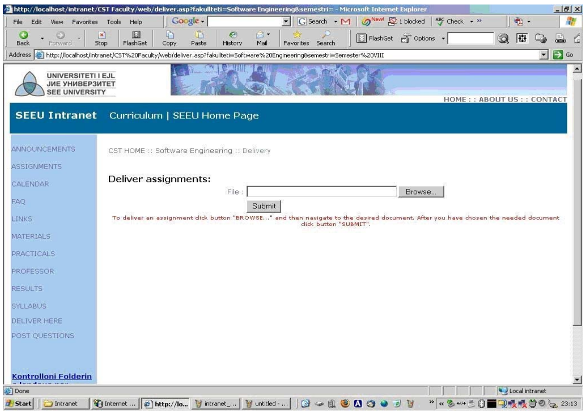

Fig.. 13. Intranet Gateway Interactivity - Uploading Content.

The motivation for designing and developing this e-learning software solution was primarily to test our previously set hypothesis discussed above. Secondly it targets the user skills level and computing knowledge and builds upon that. This is done in order to be simple and easy to use and with this to increase substantially the level of usage. It has been based and build upon previous lowest level assessed user skills and computing knowledge. Compared to commercial courseware management systems this one has no need for training staff since it has a flat learning curve (it targets and does not expect any particular computing knowledge except the fundamental Operating system operations). It also has a minimized maintenance and support since each instructor them self's take care of the content and changes of content happens in real time. Based on all of the findings it proved it business perspective and proved as very cost effective.

TABLE V.
WORK BREAKDOWN ACTIVITIES WITH DEPENDENCIES, DURATION, PESSIMISTIC (PT) AND OPTIMISTIC TIME (OT)

Activity Number	ACTIVITY Name	DURATION (Weeks)	DEPENDS ON (Dependencies)	PT	OT
10	Feasibility study	2		+ 2	
15	Requirement analyses	6		+ 1	- 1
20	Modeling-Design using UML	9	15	+ 1	- 1
25	Planning activities	2	20		
30	Write data entry software	3	10		
40	Create Data files	5	30		
50	Create multilingual content	3	40		
60	Write Program Specifications	3	10	+ 1	-1
70	Construct Program	12	60	+ 3	- 2
80	Test System	5	25,40,70		
90	Familiarize users	2	10		
100	Train Staff	6	90		
110	Deployment	2	50,80,100	+ 1	-1

In defining the initial requirements using this model before building our first prototype we first anticipated all the requirements and created a project management time table that would involve all anticipated activities. We introduced 13 work breakdown activities for the retrieval courseware system. We have defined our critical activities and time of delivery (time when the project can be delivered) calculated to be in 25 weeks. We also realized a risk analyses of the system, analyzing all the possible risks. The project had two critical paths (CP) lying between the activities: CP1: 10-60-70-80-110 and CP2: 15-20-25-80-110. The system was constantly critically evaluated during its all stages first from the development team during meeting sessions and then from the users. Based on the interviews with students and lecturers it was concluded as very easy to use and at the same time the courses are very effectively managed using the software solution. The menu content and links are based on research study of [6] and also included several comparative analyses of course management systems like Blackboard, and WebCT as well on our experiences and surveys on the users needs analyses. Following the guidelines from [6] the interface is clear and navigational structure is clearly marked using breadcrumbs allowing the user to orient him self at each point about his position in the content structure hierarchy. Also a clear exit or shortcut to the other main content groups is provided so the user can easily navigate depending on his preferences and needs observing the navigational guidelines and not so much the aesthetic perspective of the interface. The aesthetics can also be addressed later and at this point had to be sacrificed for the accessibility, content availableness and overall functionality since it is not a priority for the intended role of the created courseware system. The entire content is retrieved in the browser at the moment when students access the required subject content throw the web based interface and the course administrator-lecturer and assistants engaged in the subject have no limitations and can offer content in different formats. The course administrator using copy past functionalities puts the content in the remote folder were the students retrieve it through the web interface.

VII. Conclusion

We have proposed a new way of tackling the process of creation of e-learning solutions as interactive environments by integrating and undertaking the software engineering approach based on e-learning indicators. The software solutions as a new systems were functioning practically and correctly as defined in their specifications. The experience introduced suggests the positive effects of using the interactive tools: 1) XHTML and XML e-learning Interactive tool, 2) e-learning interactive mathematical tool, 3) Information Retrieval Courseware system. Randomly assigned testing groups experienced and worked with the software solution.

Our conclusion regarding the first goal of the research study (1) to gather information and asses e-learning level and interactions between human actors and the developed medium of instruction - the software solution. Our analyses have shown that the e-learning interaction based on PET technique is quite high and the learning curve is quite high for the I (first) XHTML and XML e-learning Interactive tool software solution, as well as for the second software solution (e-learning mathematical tool). It is obvious that the student learners are faced with a lot of decisions and they need previous knowledge in order to use the software. The high learning curve of the system however is based on student interaction without any previous instructions. If the system is taught and instructed how to be used in classes then the learning curve might drop significantly and therefore the benefit of the usage of the system cold be much higher. The analyses show that for the third (3) software solution the Information Retrieval Courseware system the learning curve is quite flat and users are already familiar with all of the concepts because the software it self was build and targeted the knowledge level of the users.

Our conclusion regarding the second goal (2) to determine the distance between learner activities and preconceived scenarios, our e-learning research analyses based on EULAT methodology and PET technique as well on focus group, we have seen, evidenced and concluded the next: 1) The learners interpret their experiences according to their own perceptions and doing that they construct their own knowledge. 2) Active construction demands a high level of independence and self organization. 3) Construction of knowledge of the learners and the refinement of the ability to do so do not happen passively and autonomously. 4) Learning is situated. The social, motivational and emotional contextual factors of the learning situation decisively control the ways and means of the learning- and retention-process as well as the use of the knowledge and abilities.

Generally each software solution is appreciated and the presented methodology approach should be reused in designing and developing e-learning software solutions since then they are tailored to users needs and encapture the important factors that influence learning, with what they have proven effective and usable. Students achieve better results and learn more when they can reflect what they learn. This is especially achieved using our developed software solution where they can reflect what they have learned previously, relate it to past experiences, and apply it practically using the software solution.

References

[1] Pressman, R. (2005) "Software Engineering: A practitioners approach 6 Ed" McGraw-HILL, inc, pp.81-117

[2] Easterby-Smith, Mark and Thorpe, R, (2002) Management research: An introduction, 2nd Edition, London : Sage Publications

[3] Joseph S Dumas and Janice C. Redish (1999) "A practical guide to Usability Testing" revised edition, Pearson Education Limited

[4] Elfriede, D. (2004) "Effective Software Testing: 50 Specific Ways to Improve Your Testing" , Addison-Wesley Pub Co

[5] Klauser, F.; Schoop, E.; Gersdorf, R.; Jungmann, B. & Wirth, K. (2004): The Construction of Complex Internet-Based Learning Environments in the field of Tension of Pedagogical and technical Rationality, Research Report ImpulsEC 10, Osnabrück, 2004.

[6] Nielsen, J. (2000). Designing Web Usability: The Practice of Simplicity. New Riders Publishing, Indianapolis, ISBN 1-56205-810-X

[7] Fetaji, B., & Fetaji,. M. (2007). Assessing, measuring and evaluating e-learning indicators. In P. Kommers & G. Richards (Eds.), Proceedings of World Conference on Educational Multimedia, Hypermedia and Telecommunications 2007. Chesapeake, VA: AACE.