Collaborative Working e-Learning Environments Supported by Rule-Based e-Tutor

Salaheddin Odeh and Eiman Ketaneh
Electronic and Computer Engineering Master Program, Faculty of Engineering, Al-Quds University, Abu Dies, Jerusalem, Palestine

Abstract—Collaborative working environments for distance education sets a goal of convenience and an adaptation into our technologically advanced societies. To achieve this revolutionary new way of learning, environments must allow the different participants to communicate and coordinate with each other in a productive manner [1]. Productivity and efficiency is obtained through synchronized communication between the different coordinating partners [2], which means that multiple users can execute an experiment simultaneously. Within this process, coordination can be accomplished by voice communication and chat tools. In recent times, multi-user environments have been successfully applied in many applications such as air traffic control systems [3], team-oriented military systems [4], chat text tools [5], [6] and multi-player games [7]. Thus, understanding the ideas and the techniques behind these systems can be of great significance regarding the contribution of newer ideas to collaborative working e-learning environments. However, many problems still exist in distance learning and tele-education [8], such as not finding the proper assistance while performing the remote experiment. Therefore, the students become overwhelmed [8] and the experiment will fail. In this paper, we are going to discuss a solution that enables students to obtain an automated help by either a human tutor or a rule-based e-tutor (embedded rule-based system) for the purpose of student support in complex remote experimentative environments. The technical implementation of the system can be realized by using the powerful Microsoft .NET, which offers a complete integrated developmental environment (IDE) with a wide collection of products and technologies. Once the system is developed, groups of students are independently able to coordinate and to execute the experiment at any time and from any place, organizing the work between them positively.

Index Terms—Collaborative working environment, human tutor, rule-based e-tutor, multi-user environments, socio-technical system, human-computer interaction, .NET.

I. INTRODUCTION

Recently, the university can go to students and not vice versa; this idea is what is designated as distance learning. We are going to argue how two or more students can execute an experiment in real time, at the same time and on different places. People collaborate because doing so is satisfying or productive. Human connections have prompted millions of users to join mailing lists, visit chat rooms peppered with humor. Without cooperation, many computer users have felt introversive and isolated, but as in any human community, there is also controversy and slander. In a collaborative environment, goals of collaboration are to allow two or more students in conjunction with a tutor to communicate with each other at the same time, where they are distributed in space.

Before we begin to discuss how we can improve and develop the field of collaborative working e-learning, it is very necessary to study and analyze other socio-technical systems, where collaboration is successfully realized such as:

- **Air traffic control systems [3]**: The activity of the airport is a highly distributed activity. The function of the control tower at an airport is to provide information and instructions. This data is in the form of clearances to aircrafts in certain areas of the airport and the airspace immediately around it. The objective of this is to ensure the safe and efficient movement of air traffic. The collaboration between the different partners is achieved by partitioning and coordinating the tasks between them. For example, the tower controller is responsible for controlling the aircrafts on the runway, and the ground controller is responsible for certain other areas of the airport. Whereas, the planner or coordinator typically maintains contact by telephone with other air traffic control facilities that manage the airspace around the airport.
- **Team-oriented military systems [4]**: Advanced technologies are used to facilitate military team work collaboration through communication und information exchange for enabling remote operations. Collaborative technologies currently used in remote military operations, such as email, instant messaging, and desktop conferencing, assist explicit communications between distributed team members around the world as part of physically distributed teams.
- **Chat tools [5], [6]**: Chat tools are an integral part of many collaborative environments. They mostly sample instant messenger and allow chatting with all participants.
- **Share desktop and share applications [6]**: Share desktop displays what you see on your desktop to all other participants and can both give control to others and take back control. Whereas share applications in addition to their ability to allow participants to see your application on their screen, they make it possible for a participant to control someone's application such as PowerPoint and Word. That is, other participants can make their own edits and updates. Principles and techniques suggested in share desktop and share applications have been

successfully applied in web conferencing and net meeting tools.

A well-known architectural model that which fits into the designing of collaborative working e-learning environments is the client-server architecture [9], [10] that facilitates what is called multi-user e-learning environment; more than one student and a tutor have access to an experiment at the same time and from different places.

II. LEARNING WITH INSTRUCTIONAL SUPPORT

Complex problem solving tasks without instructional support will often demand too much from the students and will lead to ineffective learning [2]. As long as students need instructions and help in solving scientific problems, the learning environment shall provide knowledge for solving these problems. Instructional support is an important element especially in problem-based learning settings; therefore, remote laboratories should provide support for students. In remote laboratories, a tele-tutor communicates via synchronized or asynchronized communication tools with his students, resulting as a central role regarding instructional support. Table 1 classifies collaborative environments using a time-space matrix. In the following, we are going to discuss some advantages and disadvantages of the different approaches. One form of communication between students and tutors is asynchronized; this kind of communication has several disadvantages [2]:

- Students will not get any support from the tutor during the laboratory session, therefore, students might not able to find answers and solutions for their questions and problems in the experiment.
- Students have to accept time delays for getting answers to their questions, and therefore, if time delays are too long, they could lose their motivations.

By contrast, the synchronized communication bypasses the above mentioned problems:

- The synchronized communication restricts the time flexibility of the students.
- Upcoming problems can then be solved immediately by using chat and application sharing.

From the dimension "place" perspective, the "same place-same time" approach, the local lab, enables the students to discuss problems so that they achieve solutions easily and the lab purpose is reached in that each student complements the other. The "different place-same time" approach, the distance lab, has several contemporary advantages [8]:

- Solving the experiment problems with less time is possible.
- Allowing remotely located students to complete lab experiments unconstrained by time or geographical considerations.
- Expanding the student’s skills on using real-time systems and its instrumentations.
- Providing sufficient time for the students to complete labs.

The disadvantages of the distributed laboratories are [8]:

- Inequality in task division between students.
- Network congestion.
- The fact that all the students do not have the same level of skills in using system features.
- Students feel isolated because of lack of feed-back from other students.
- Groups without supervision will face a lot of difficult problems left unsolved.

Our focus in this research will be about the development of education through the use of distributed e-laboratories for collaborative working; so after our previous discussion, the main problem caused by the asynchronized approach is that if the student groups are not under enough control by their teachers, they will have a time delay for getting the answers to their questions. Once the delay becomes too long, the students could lose their motivation, and face many problems left unsolved. One way the discussed problems can be solved is by making a remote tutor to support students available over a distance via synchronized communication tools, such as chat and voice sharing that allows quicker assistance. However, a remote human tutor can not support students at any time and will not be available throughout the day. Therefore, it is intended to implement an automated helping system in the form of a rule-based e-tutor for user support in complex remote experimentative environments. A rule-based system embraces a rule-base including stored knowledge about the correct experiment configuration, an inference machine, and a dialog component (see next section). The rules determine what should be done in different situations and are initially designed by a human experiment, the experiment designer. Each rule has two parts: conditions and action.

TABLE I.
TIME-SPACE MATRIX AS A TRADITIONAL WAY TO DECOMPOSE COLLABORATIVE SYSTEMS

Time / Place	Same Time	Different Times
Same Place	Face to face	Asynchronous interaction
Different Places	Synchronous distributed	Asynchronous distributed

Every student should commit to handing work on time and required hours to complete the experiment, thus scheduling sessions to work is of great significance. Because of inequality in task division between students, color coding can be used to distinguish between them. To achieve a powerful user-interface, we can benefit from some human-computer aspects such as perceptive and cognitive ergonomics ([11], [12] and [13], see section "User-Interface Design for Remotely Collaborating Partners"). One widely used technique to establish the synchronized communication in our collaborative working environment is chat tools; in addition to advanced visualization techniques, it is direct to an effective interaction between students and tutor.

III. DISTRIBUTED SYSTEM ARCHITECTURE FOR COLLABORATIVE E-LEARNING

Fig. 1 illustrates a suggested system architecture tailored to the special needs of this research for establishing a collaborative working e-learning environment, which is not only supported by a human tutor, but by a rule-based e-tutor as well. The architecture applied to this system follows the simplest client-server architecture, the two-tier client-server architecture, where an application is organized as a server and a set of clients [14]. The two-client architecture can take two forms: the thin-client and the fat-client model. Since all of the application processing and data management is carried out on our application server for collaborative working e-learning environment and the application is only responsible for running the representation software, our system architecture follows the thin-client model. In the fat-client model, the server undertakes only the data management and the client software implements the application logic and the interaction with the users.

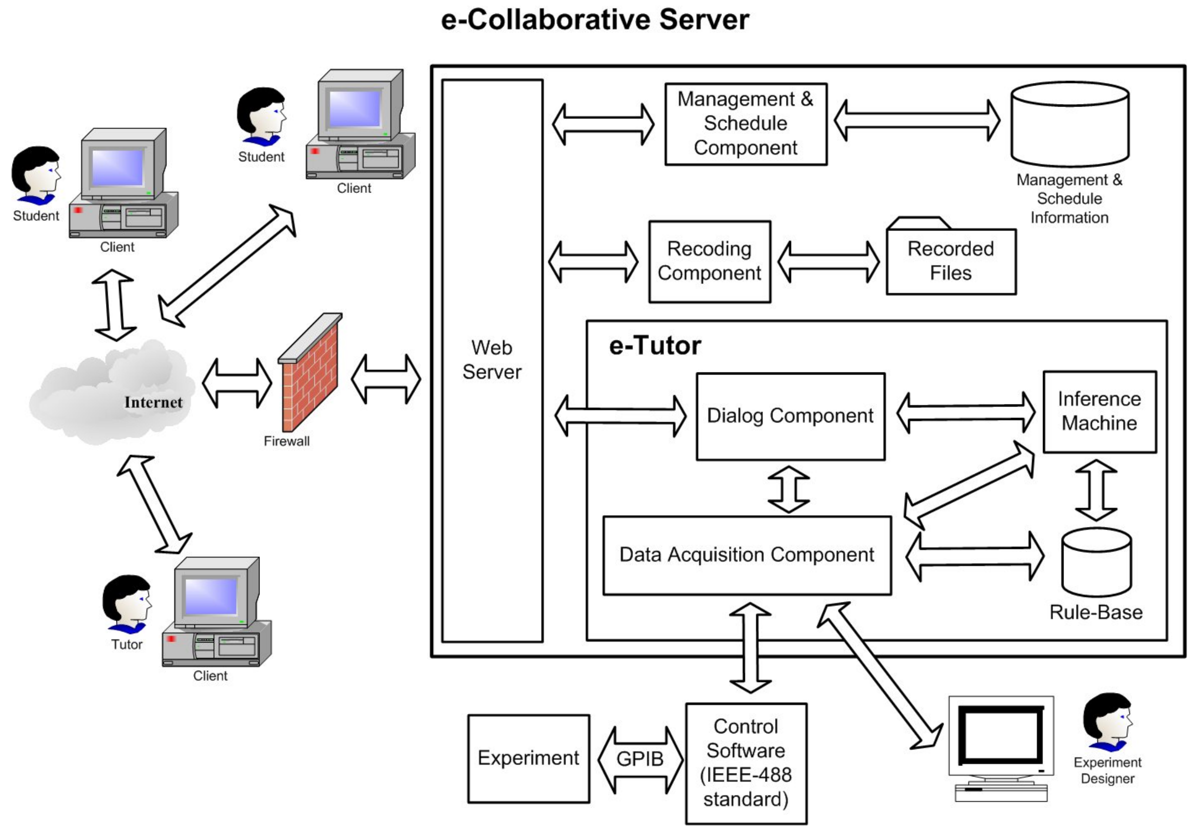

Figure 1. System architecture of the suggested e-collaborative environment for remote experimentation supplemented by an embedded rule-based e-tutor

A system realized after this architecture enables the different users such as students and tutors to perform on real (physical) experiments remotely whenever they want and anywhere they are. Our application is classified as an event-processing system due to the fact that the actions of the system depend on interpreting events occurring in the collaborative working environment. These events could be either the input commands of the different users or variable changes of the experiment itself. The architecture proposes that our distributed application should be made of these components:

A. Web server

The web server provides several services for the clients through a web browser. In addition to its role as a middleware to communicate with the clients and the other system components via the internet, it represents the central unit of the collaborative e-learning server and functions as a coordinator between the various server components.

B. Web-based user-interface

All clients of the distributed e-collaborative system, students and tutors, use the same web-interface. In our case, the clients are only responsible to run the presentation software, which will be mediated by a conventional web browser such as the Microsoft Internet Explorer or Netscape. When the web-based user interfaces have been designed, several human-computer interaction rules for user-interface design have to be taken into account such as consistency of data display (labeling and graphic conventions), efficient information assimilation by the user, minimal memory load on user, compatibility of data display with data entry, flexibility for user control of data display, presentation of information graphically where appropriate, standardized abbreviations, presentation of digital values only where knowledge of numerical value is necessary and useful [15].

C. Management and schedule component

The schedule data of accessible time that is stored in a schedule data base. When a student group tries to access to the experiment, the management and the schedule component examines whether they are allowed to do this according to a schedule time-table. Moreover, this component manages the registration procedures of the students for enabling them to execute the experiment. The entered data will be temporarily stored and after its verification by the experiment administrator, it will be stored in the database permanently. The system informs the students about the failure or the success of their registration attempts by email confirmations.

D. Rule-based e-Tutor (Embedded rule-based system)

Main goal of this research is helping students to execute experiments remotely at anytime and from anywhere. This goal can be fulfilled when the synchronized human tutor is continually available. Realistically, a remote human tutor will not be online all day. Therefore, we want to implement an automated help system (e-tutor) for user support in complex remote experimentative environments. Hayes-Roth [16] notes that rule-based systems automate problem-solving know-how, providing a means for capturing human expertise, and are proving to be commercially viable. Rule-based systems share certain key properties: incorporating practical human knowledge in conditional if-then rules, increasing their skill at a rate proportional to the enlargement of their knowledge bases, being able to solve a wide range of possibly complex problems by selecting relevant rules and then combining the results in appropriate ways, determining the best sequence of rules to execute adaptively, and explaining their conclusions by retracing their actual lines of reasoning and translating the logic of each rule employed into natural language.

As our focus is mainly on collaborative systems and not on knowledge-based systems, we will realize a simplified rule-based system whose rules consist of two parts: conditions and action. The e-tutor embraces a rule-base including stored knowledge about the correct experiment configuration, an inference machine, and a dialog component. Evaluating the rules periodically causes the system to react in accordance to how students deal with the experiment. The experiment designer in cooperation with the experiment tutor maintains the e-tutor rule-base with plausible rules so that they verify and validate the fired rules after every experiment session. While carrying out an experiment, the recording component archives the students' interactions with the experiment along with the e-tutor reactions. Johannsen [11] expanded the rule-based system architecture presented by Raulefs [17] with additional components that are what makes a rule-based system to be embeddable in a real-time system. Various interfaces for signal/symbol and symbol/signal transformations are necessary to embed the e-tutor in the collaborative system. From Fig. 1, the rule-based e-tutor includes a dialog (presentation) component, an inference machine, an explaining component, a data acquisition component, and an experiment-specific rule-based.

E. Recording component

As the experiment designer, along with the human tutor, verify and validate the e-tutor reactions against incorrect students' operations on the experiment, it is important to have a component that records the students' interactions with the user interface. Furthermore, this data is useful for the objective evaluation [18].

F. The experiment

The remote experiment can be any one of an engineering lab covering topic related to electric circuits or electronics and so on. However, remote experiment designers must reform and expand conventional experiment kits so that they can be easily integrated in contemporary remote e-laboratories. Fortunately, most of the current instrumentations such as oscilloscopes and multimeters are provided with control through PCI GPIB (General Purpose Interface Bus) card and GPIB cable, allowing PCs to communicate with over 2000 instruments made by over 200 manufacturers. The main purpose of the general purpose-interface bus (GPIB) is to send information between two or more devices. Before any data is sent, the devices must be configured to send the data in the proper order and according to the proper protocol [19]. The electrical specifications as well as the cables, connectors, control protocol, and messages required to allow information transfer between devices are defined by the IEEE-488 standard [20]. For instance, by chaining IEEE-488 cables from one device to the next, it is possible to connect up to 14 devices together. IEEE-488 supports data transfer at up to 1 Mbytes/sec. In addition to simple data transfers, the IEEE-488 standard defines a number of specialized commands for interface programming in the form of subroutines available as programming libraries for different programming languages such as C, Pascal, C# etc.

IV. SYSTEM DEVELOPMENT AND IMPLEMENTATION

The development of the system follows the prototyping approach because of its appropriateness in helping resolve requirements and design uncertainties as it is the situation in most of research projects [14], [21]. In the iterative process, the stages: specification, design, development, and testing are not chained, but rather interleaved and concurrent. Software is developed in a series of increments with each increment including new system functionality. For developing the user interfaces, it is recommendable to use an interactive development system like Microsoft Visual Web Developer 2005 Express Edition in addition to these tools: Microsoft Visual Basic 2005 Express Edition [22], Microsoft SQL Server 2005, Web Server: IIS (Internet Information Service), as well as Web Development. Additional tools include: Dream weaver MX, and Macromedia Flash 9. The first two tools are based on the .NET technology, allowing quick creation through drawing and placing of graphical objects on the user display. .Net technology includes graphical user interface tools with rich libraries for user interface components, enabling data to be displayed in many forms.

V. USER-INTERFACE DESIGN FOR REMOTELY COLLABORATING PARTNERS

For designing the user-interface, it is to suggest the usage of simple screen sketches and key-screen prototyping. The latter encourages that the user-interface concept can be illustrated by simple screen sketches (paper or on-screen) aimed at conveying the system concept to non-technical users; whereas the latter focuses on key-screen prototyping. That is, key-screen prototypes show users the design of the proposed system and allow them to evaluate and refine it; they can be used for usability testing and heuristic review. They usually evoke strong reactions, generate early participation, and create momentum for the project.

Before discussing the key-screen prototype of the web-based user-interface for the clients shown in Fig. 2, it will be of great significance if we clarify some term definitions related to human factors that must be taken into account when we have designed the user-interface. This will not only simplify distinguishing between the terms used in this contribution, but it will mediate the concept as well. The key-screen prototype is the basis for a web-based user-interface for the clients: the students and the tutor. Ergonomically, we have to distinguish between aspects of perceptive and cognitive ergonomics [11]. The cognitive ergonomics relates to reasoning, memory and knowledge [12]. Here, we are more concerned with perceptive ergonomics focusing on designing issues such as color, shape form, dimension and allocation, highlighting and so on.

From Fig. 2, the desktop of the web-based user-interface is divided into several windows for representing different functionalities. The windows fixed at particular loci on the desktop for strengthening the ergonomic aspect "allocation and dimension" represent different information. The windows are:

A. Experiment window

The experiment window represents the remote tool kit and the instrumentation necessary for that experiment, which will be virtually visualized. By using the tools of the interaction configuration area, the student can select the suitable wiring tool to connect the electronic elements together. For a better visualization, she/he can adjust the wire color according to the function of the wire, i.e. black for ground, blue for positive VCC etc. The instrumentation area includes the equipments such as oscilloscopes to pursue the signals after building the circuit; as well as various function generators to feed the circuit with input signals. The students are able to connect the inputs and outputs of a virtual oscilloscope with other electronic elements. After every building step the students can select the accept button, the user-interface sends the circuit configuration to the web server, where the e-tutor initially carries out a consistence check to the sent circuit data. If these data passes the check, this data are used to control the real experiment.

B. Chat tool window

As previously discussed, chat tools play a central role in e-collaborative environments. Through the chat text window, a user can send a text message to a particular user or to all users at once. To select a destination user, she/he can select the item "chat" within the pop-up menu associated with each active user icon.

C. Active user window

The user windows show the users who are logged into the experiment by means of an icon titled with the student's name. These icons are selectable as pop-up menus leading to usable user-interface. The menu item "highlighting" helps the users know what configurations on the experiment are done. The highlighting tool frames all electronic elements such as resistors, wires, ICs etc., with a dashed line whose color corresponds to the color code assigned to the student who has done the changes. The menu item "information" causes the system to display information about the student, how long the student is being logged-in etc.

D. Session control window

This window consists of buttons with meaningful icons for controlling session concerns such as logging-in and –out, session recording or playing back etc.

When the web-based user-interface has been developed, several human-computer interaction issues such as visual thinking and icons [15] must be taken into consideration. An icon is an image, picture, or symbol representing a concept.

VI. USABILITY TESTING OF THE COLLABORATIVE SYSTEM FOR E-LEARNING

When we have implemented the whole system, a comprehensive usability testing for revealing the advantages and disadvantages of introducing collaborative working e-learning environments in higher education will be carried out. Subjects (the students and tutors) interact with the system via the different web-interfaces and have to solve several scenarios [23]. The student groups consisting of two students carry out the experiment in different sessions:

- Session 1: collaborative working e-learning environments supported by a human tutor
- Session 2: collaborative working e-learning environments supported by an e-tutor
- Session 3: conventional experiment (face-to face)

One reason three different collaborative environments are used is because of the fact that the obtained result should be comparative. In this evaluation, the different environments serve as independent variables, whereas the evaluation criteria serve as dependent variables. It will be possible to measure the differences between these environments according to what extent the main features of collaborative environments communication between partners, coordination such as roles and task distribution, and production are fulfilled. These features are discussed at the beginning of the paper. The raw data of the experiments were handled statistically by using the Student's test (t-test) [24] and then analyzed by SPSS [25].

The statistical outcome should be analyzed and reviewed by the usability engineers and the system designers, so that the final results helped in revising and optimizing the design of the interactive software system on the one hand; and the system designers could have defined new or corrected existing design guidelines for future e-learning tools on the other hand. As a useful means for comparing mean values of two sets of numbers, usability engineers have the opportunity to select between either the Student's test (t-test) [26] or one-way ANOVA, through which a comparison can be carried out, providing us with a statistic for evaluation exposing the statistical significance of the difference between two means. During a usability testing session, the usability engineer explains to the subjects (the students and the tutor in session 1 and 3; and only the students in session 2) all operations related to the experiment and the web-interfaces. Several perception-ergonomic aspects were taken into account whilst developing and designing the collaborative e-learning tool. However, the perception of ergonomics is concerned with designing aspects such as color, shape form, dimension and allocation [13].

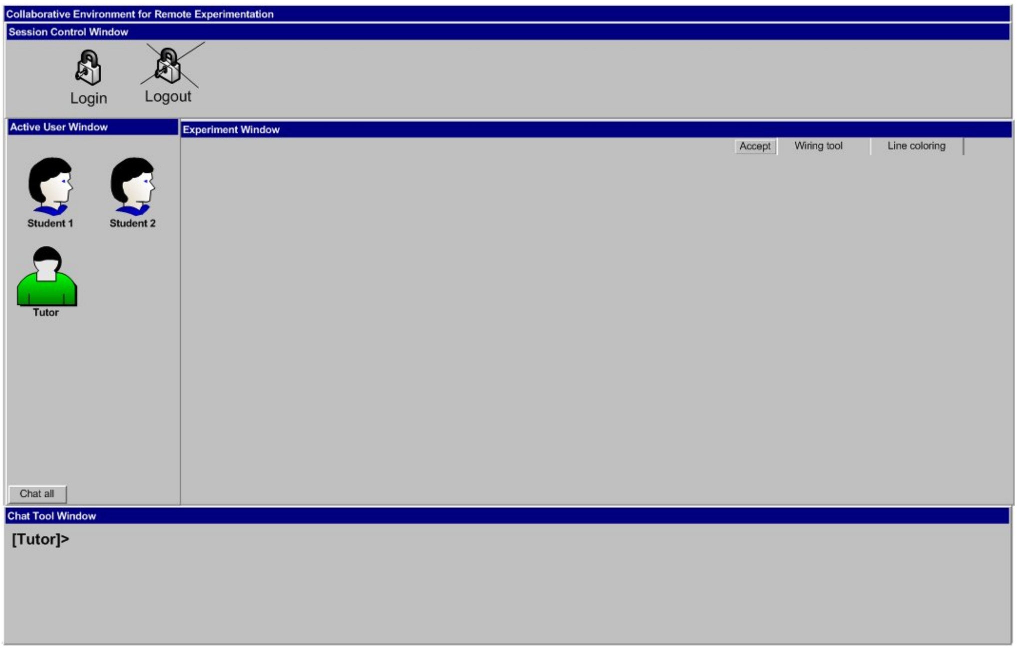

Figure 2. A key-screen prototype of the web-based user-interface for the clients

VII. CONCLUSION

In order to contribute to the collaborative working environments, it was necessary to consider other collaborative systems implemented in other fields such as air traffic and military. It is obvious that an e-collaborative system is characterized as effective if it enables the collaborating partners to be productive through a coordinated communication. A comparison between the various collaborative approaches shows the advantages and disadvantages of each one, concluding that for collaborative distributed e-learning systems the synchronized form is superior to the asynchronous regarding instructional support.

A remote human tutor can't be online every time to support students. The implementation of an automated helping system (e-tutor) to support students complements the human tutor. Moreover, the e-tutor or the embedded rule-based system functions as an observer for students' actions taken on the experiment. Once the connections of the electronic elements on the virtual experiment are incorrect, the e-tutor displays a warring message and prevents the real contacting on the remote experiment board.

From software engineering perspective, the system obeys the thin-client architectural model because the e-collaborative server handles the application processing and the data management, whereas a web-based user-interface is only responsible for running the representation software (Fig. 1). Our distributed e-collaborative system could be made of the following components: a web-server, web-based user-interfaces, a management and a schedule component, an embedded rule-based system (e-tutor), a recording component, and the experiment itself.

The web-based user-interface plays a central role within the distributed e-collaborative system as it represents the window to the experiment; therefore, we have to pay particular attention to its development, aimed at taking various human-computer interaction rules for user-interface into account. The web-based interface, which is realized as an integrated desktop, includes windows for displaying the experiment, the chat tool, the active users and session control.

The expected results from this research are: providing the students with an automated help at any time, putting students in a real environment lab, targeting the system at a real interaction between students and tutor and accessing the lab from any place and at any time. These are possible through designing of and innovations in web-based collaborative working for online engineering education, designing of innovative web-based interfaces, architectures and environments to support collaborative working, and applying of contemporary technologies to facilitate collaborative remote experimentation.

REFERENCES

[1] M. Fang, T. Li, J. Rao, X. Su, "A collaborative educational IS based on WWW," Information Systems in the WWW Environment, pp. 277-297, Beijing, China, 1998.

[2] A. Böhne, N. Faltin, B. Wagner, "Synchronous tele-tutorial support in a remote laboratory for process control," INNOVATIONS 2004: World Innovations in Engineering Education and Research, edited by W. Aung, R. Altenkirch, T. Cermak, R. W. King, L. M. Sanchez Ruiz, iNEER, USA, 2004.

[3] B. Fields, P. Amaldi, A. Tassi, Representing collaborative work: The Airport as Common Information Space, Technical Report: IDC-TR-2003-003, Interaction Design Centre, School of Computing Science, Middlesex University, UK, 2003.

[4] S.D. Scott, M.L. Cummings, D.A. Graeber, W.T. Nelson, R.S. Bolia, "Collaboration Technology in Military Team Operations: Lessons Learned from the Corporate Domain," Proceedings of CCRTS 2006: the Command and Control Research and Technology Symposium, San Diego, CA, USA, 2006.

[5] J. Lonchamp, "A Structured Chat Framework for Distributed Educational Settings," 6th International Conference on Computer Supported Collaborative Learning - CSCL'05, Taipeh, Taiwan, 2005.

[6] "Collaborative Tool Demonstration," Peace and stability education, Workshop report, the United States Army Peacekeeping and Stability Operations Institute at Carlisle Barracks, Pennsylvania, USA, pp. 189-205, 2006.

[7] Y. Cao, G. Sharifi, Y. Upadrashta, J. Vassileva, "A Case Study on Social Network in a Computer Game," Proceedings of the second international joint conference on Autonomous agents and multiagent systems, Melbourne, Australia, pp. 954 – 955, 2003.

[8] O. C. Santos, A. Rodríguez, E. Gaudioso. J. G. Boticario, "Helping the tutor to manage a collaborative task in a web-based learning environment," 11th international conference on artificial intelligence in education, Sydney, Australia, 2003.

[9] MJ. Callaghan, J. Harkin, TM. McGinnity, LP. Maguire, "Client-Server Architecture for Collaborative Lecture-Led Remote Experimentation," 7th Distance Learning and the Internet Conference APRU DLI 2006, Japan, 2006.

[10] M. Cefalo, L. Lanari, G. Oriolo, M. Vendittelli, "The REAL Lab: Remote experiments for active learning," XLI AICA Annual Congress, Trento, IT, 2003.

[11] G. Johannsen. Mensch-Maschine-Systeme. Berlin: Springer, 1993.

[12] J. R. Anderson, Cognitive psychology and its implications (5th ed.), New York: Worth, 2000.

[13] N. A. Streitz, "Cognitive compatibility as a central issue in human-computer interaction: Theoretical framework and empirical findings," Cognitive engineering in the design of human-computer interaction and expert systems, in G. Salvendy (Ed.), Amsterdam: Elsevier, 1987, pp. 75-82.

[14] I. Sommerville, Software Engineering (8th Edition). Addison Wesley, 2007.

[15] B. Shneiderman, and C. Plaisant, Designing the User Interface: Strategies for Effective Human-Computer Interaction (4th Edition). Addison Wesley Longman, 2004.

[16] F. Hayes-Roth, "Rule-based systems," Communications of the ACM, vol. 28, Nr. 9, pp. 921-932, 1985.

[17] P. Raulefs: "Expertensysteme," Kuenstliche Intelligenz, Informatik-Fachbereiche, W. Bibel, J. H. Siekmann (Ed.), Proc. 59, Berlin: Springer, 1982, pp. 61-98.

[18] S. Odeh, O. Qaraeen, "Evaluation methods and techniques for e-learning software for school students in primary stages,"International conference on interactive computer aided learning (ICL 2006), Villach, Austria, 2006.

[19] CEC, CEC 488 programming and reference, Part number 370966A-01, 2003.

[20] A. J. Caristi. Ieee-488: General purpose instrumentation bus manual (professional and technical series), Academic Press, 1989.

[21] T. Faison, Component-Based Development with Visual C#, John Wiley & Sons, 2002.

[22] Deitel Associates, Visual Basic 2005 How to Program, Prentice Hall, 2006.

[23] M. B. Rosson, J. M. Carroll, Usability Engineering: Scenario-Based Development of Human-Computer Interaction, Morgan Kaufmann Publishers, 2002.

[24] R. A. Johnson and G. K. Bhattacharyya: Statistics: Principles and Methods, Wiley, 2000.

[25] Pallant, J. (2004). SPSS Survival Manual, Open University Press.

[26] P. L. Gardner: "Discusses assumptions of the t-test," Scales and Statistics: Review of Educational Research, 45: 43-57, 1975.

AUTHORS

Salaheddin Odeh received his masters degree in electrical engineering, specialization area computer and control engineering, from the University of Stuttgart and the PhD degree from the University of Kassel; both universities are in Germany. He is currently an assistant professor in the Department of Computer Engineering and coordinator of the masters program of the Faculty of Engineering at Al-Quds University in Jerusalem. His interests include software engineering, control engineering, robotics, advanced programming, operating systems, human-computer interaction, and multimedia. In 1999, for his doctoral thesis, he was the recipient of the first prize of the Association of German Engineers (VDI) in the state Hessen in Germany for the best technical-scientific research.

Salaheddin Odeh, Department of Computer Engineering, Faculty of Engineering, Al-Quds University. P.O. Box 20002, Abu Dies, Jerusalem, Email: sodeh@eng.alquds.edu

Eiman Ketaneh received her B.A. degree in electrical engineering department of communication, from the University of Omer-Almoktar/Libya. She is currently a Master student at the Faculty of Engineering, Al-Quds University, Palestine.

Eiman Ketaneh, Electronic and Computer Engineering Master Program, Faculty of Engineering, Al-Quds University, P.O. Box 20002, Abu Dies, Jerusalem, Email: eng.eiman@yahoo.com

Manuscript received 24 July, 2007. Published as submitted by the author(s).