RACOL

Rural Advanced Community of Learners

EVALUATION OF THE RACOL USER INTERFACE: By Robyn Taylor

1.1 Abstract

This paper provides an overview of two well-established user interface evaluation techniques: heuristic analysis and cognitive walkthrough. This background information is used to evaluate the user interface of the Rural Advanced Community of Learners (RACOL) project.

1.2 Introduction

The Rural Advanced Community of Learners (RACOL) project is a distance learning endeavor which intends to use advanced communications technologies in order to provide real-time classroom instruction to students in remote regions of Alberta [1]. One of the major goals of the RACOL project is that the integrated classroom be as intuitive and accommodating as possible, in order for students and teachers to feel at ease when interacting within the classroom. For this reason it is critical that the technological components of the classroom are unrestrictive of user behavior and provide a high degree of usability for the classroom participants. Careful evaluation of the classroom interface is required in order to identify and rectify sources of concern in order to provide the most positive classroom experience.

Currently, the RACOL system is in the pre-deployment phase, so evaluation at this point focuses on evaluation of the computerized control panel user interface and how its presence and usage requirements affect the greater context of the integrated classroom.

This paper will discuss the issues and challenges surrounding the evaluation of a distance learning environment, then focus on the pre-deployment evaluation of the computerized control panel user interface. Using the techniques of heuristic analysis and cognitive walkthrough, an assessment will be made regarding the usability of RACOL's computerized control panel user interface.

1.2.1 Evaluating a technologically integrated distance learning environment

When evaluating a technologically integrated system, there is more to evaluate than simply a computer screen interface. It is important to

remember that every aspect of the integrated environment contributes to the user experience.

One such integrated environment is currently being developed at the University of Alberta. The Rural Advanced Community of Learners (RACOL) project is a distance learning endeavor that intends to allow students in rural Alberta to communicate via a two-way videoconferencing interface with teachers and fellow students across the province [1]. The intent of the RACOL classroom is to provide an environment where teachers and students can interact visually and audibly with one another, and present data using various audio-visual media (computer presentations, VCR/DVD, visualizer). A major goal of this system is that teachers and students be able to interact naturally with the environment, without feeling constrained by the technology in their midst. For this reason, evaluation of the environment before the system is deployed is currently being undertaken in order to provide the users with an effective environment. Evaluation of the RACOL classroom will continue after actual use of the system is initiated, in order to determine how the system functions under real-world operation.

1.2.2 Evaluation before system deployment

The pre-deployment evaluation process of the RACOL project is already underway. There are several aspects of the integrated classroom project which are subject to evaluation before actual teachers and students begin using the system. The needs of teachers and students must be taken into account when any decisions are made, either regarding the integrated computer applications, or the classroom layout itself. Neither the technological nor physical aspect of the integrated system should be allowed to dominate the environment to the detriment of the other. In order to address all the components, which affect usability within an integrated classroom, the evaluation must address the user interface of any computer applications, the physical layout of the classroom itself, as well as any technical aspects of the system, which are fundamental to the classroom's operation.

The RACOL classroom layout has been the subject of careful consideration. The classroom must be laid out in such a way that the students and teacher are allowed to be physically comfortable in the environment while still being able to easily interact with the

technology. In order to enable students to function with ease inside the classroom, any technical interfaces (computers, microphones, screens, etc…) must be readily physically accessible to the students, while not being physically dominant to the point where 'normal' classroom behavior (speaking to an instructor, communicating with other students) is obstructed or overly regimented by the constraints of the technology. Care should also be taken to ensure that undue noise, heat and visual distraction caused by any large pieces of equipment is minimized in order not to cause discomfort to the people spending time in the environment [2].

The integrated classroom contains computer applications, which require evaluation. The screen interfaces can be evaluated using the methodologies one would use for stand-alone applications, but as well it is important to study the implications of the screen interface's usage requirements (Does the operator have to sit rooted at a computer screen, or may s/he move around the room and access the computer system via a wireless device?) in terms of how they may restrict the user's behaviors within the integrated environment [3]. With regards to the RACOL distance learning environment [1], early inspection of the interface has revealed that the originally planned touch-screen panel control system is overly restrictive of the instructor's movements, and that in order to achieve the goal of seamless technological integration, additional control mechanisms need to be installed in order to allow the instructor freedom to move more naturally throughout the classroom while still maintaining control of all audio-visual system components via strategically placed control buttons. This is the type of concern which an evaluation of an integrated environment must address.

To address all possible issues that could arise and impede system use, along with the classroom layout and computer system interface evaluation, any and all technical aspects of the system should be inspected to ensure that they function effectively. In the case of the RACOL project [1], the integrated classroom allows students and instructors to communicate remotely via videoconferencing channels. To properly evaluate this system, the technical effectiveness of the videoconferencing method (speed, reliability of the network, etc…) must be assessed in order to ensure that the RACOL classroom will provide adequate functionality for its users.

Principal Investigator: Tel: (780) 492-3667 Ext. 227 Project Communications: Tel: (780) 492-5016 Ext 216 Fax: (780) 492-3179
3-105 Education North T6G 2G5 University of Alberta Edmonton, Alberta Canada

1.2.3 Evaluation once the system is in use

Once an integrated classroom system is deployed, evaluation of the system in use can begin. The ODYSSEUS distance learning project based at the University of Cyprus [7] has outlined its evaluation goals, including evaluation of the instructors' ability to teach comfortably within the environment, evaluation of the performance of the environment's technical components, evaluation of the students' learning performance, evaluation of the integrated teaching methodology, and evaluation of the social relationships between students as well as between students and instructors [7]. Gathering data on these topics can be done by self-evaluation (questionnaires, surveys), evaluation of one another, as well as by observing the students' and instructors' technical proficiency in the classroom [7]. Additionally, students' academic performance may provide an indication as to the environment's ability to facilitate learning [2].

1.2.4 Technological determinism

When attempting to integrate technology into a learning environment, designers and educators must be aware of the potential disruption the integrated technology may cause to traditional notions of how a classroom is to operate. The term "technological determinism" defines the concept of technology's ability to encompass not only physical artifacts and systems, but rather in itself exist as a driving force which can shape and direct human "progress" [6, p.87].

In terms of a collaborative and interactive learning environment, evaluators should attempt to discern whether "the technology [changes] the nature of the collaboration" [6, p.109]. Care must be taken to determine what degree of technological determinism is acceptable in the classroom environment.

An example situation, evident in the RACOL project may illustrate the concept [1]. To signal to the instructor that they wish to comment or pose a question when the instructor is giving a lecture, remote students must press a button to signal that they wish to take a place in the question queue. This queue is then processed in a first-in-first-out manner at a time of the instructor's choosing. The design of the technology restricts the way in which the students are free to behave in the classroom (in traditional classrooms, students might spontaneously comment without waiting for their contribution to be

formally acknowledged or requested) leading to a modification in the way students and instructors view student classroom participation (student-driven interruption into an instructor's lecture is, for good or ill, not possible when the RACOL system is in lecture mode).

While this seems a relatively innocuous restriction to place on classroom behavior (in most cases, it is likely that instructors do not wish to be randomly interrupted when lecturing), designers must be aware that the system design is determining available classroom discussion mechanisms, and may in some circumstances be prohibiting perfectly desirable classroom interactions (spontaneous spirited debate is curtailed by the question queue restriction). If these technical restrictions continue to inhibit student behavior over a lengthy period of time, students may begin to internalize the technically dictated manner of classroom interaction, and in fact modify their personally held views on how classroom interaction should be carried out. This is the notion of technological determinism – that the influence of technology may effect change in the way humans perceive and interact with societal and cultural constructs [6].

Whether technologically dictated system design decisions are ultimately ruled to be acceptable or unacceptable, the ramifications of such decisions should be examined and weighed, in order to determine the role technology is playing in shaping the methods of learning within a technologically integrated classroom environment.

1.3 Analysis of the RACOL screen interface

1.3.1 Heuristic analysis

1.3.1.1 Heuristic Analysis Methodology

A heuristic analysis is an evaluation which is based upon a set of guidelines. These guidelines are termed 'heuristics'. In a context relevant to user interfaces, these heuristics are a list of interface 'rules', adherence to which has been pre-determined to be indicative of user interface effectiveness.

When performing a heuristic analysis of a user interface, an analyst first selects an appropriate set of heuristics upon which to base the analysis. User interface experts Nielsen and Molich have created a nine point heuristic [8] which has been tested and proven effective for use in user interface analysis [9].

Once the heuristic has been chosen, the analyst studies an application, taking careful note of any aspects of the interface which contradict particular guidelines, or of missing interface components which could make the application satisfy the guidelines more readily.

Attention to detail is imperative, as the application must be studied thoroughly if subtle yet important usability considerations are to be noticed. For this reason, Nielsen and Molich (who have formalized a process of effective heuristic analysis [9]) suggest it is desirable for more than one individual to analyze an application. After several independent analyses have been completed, they advise that an interface expert (or alternatively the entire group of independent analysts) combine the results of multiple analyses in order to address the most complete set of interface concerns.

When done with diligent care and attention, heuristic analysis of a user interface can provide detailed and concrete information as to where an application could improve its usability.

1.3.1.2 The contributions of Nielsen and Molich

Nielsen and Molich developed their generalized nine-point heuristic in 1990 [8]. Previous to their research, Nielsen and Molich noted that existing heuristics often contained hundreds or even thousands of guidelines, making heuristic analysis cumbersome and complicated.

Their simplified nine-point heuristic addresses what their years of user interface research identified as critical user interface requirements. Evaluation of an application's user interface with respect to these nine guidelines can capture many frequently occurring interface errors.

To test the effectiveness of a heuristic analysis based upon their nine point heuristic, Nielsen and Molich set up experiments [9] in which evaluators would use their nine points to conduct a heuristic analysis of a pre-designed system with built in usability concerns. Each test evaluator was able to identify a subset of the interface concerns, but no one analyst was able to identify the complete list. As a result, Nielsen and Molich recommend that a small number of analysts each assess the application independently. These separate assessments can then be combined to address a greater number of concerns. Nielsen and Molich's tests have shown that their nine point heuristic can be used to effectively identify substantial concerns in most types of screen-based interfaces when a small group (three to five individuals [12]) collates their individual analyses into a comprehensive report.

1.3.1.3 Merits and shortcomings of heuristic analysis

One of the most attractive aspects of heuristic analysis is that it is relatively easy and inexpensive to carry out. Heuristic analysis can be performed in the early stages of product development so as to greatly reduce the number of interface issues early on in the development process. This of course eliminates the need to go back and change interface elements after large sections of code have already been written. An analysis of interface components can be done even before development has begun, using mockups or interface diagrams, although certainly some usage issues (unreasonable run-time delays un-ameliorated by warnings or progress indicators, to use Nielsen and Molich's 'MANTEL' example [9]) may not become apparent until the application is in a runnable form.

The process of heuristic analysis is quite convenient in comparison to methods of interface evaluation (observation, interviews, questionnaires) which require costly and time-intensive user participation[5]. Since a team of experts evaluates the interface based on a list of interface rules, adherence to which has been verified to be of benefit to users and enhance ease-of-use in user interfaces, the experts can apply heuristic analysis before requiring user participation, in order to identify fundamental interface flaws. Of

course, the merits of user participatory studies should not be ignored. Even the most careful expert review may overlook certain interface concerns which only a real user who is an actual member of the target user community can identify [4]. However, the fewer heuristically identifiable flaws which remain in an interface if and when user participatory studies are undertaken, the more meaningful and useful eventual actual user feedback will be.

Results obtained from heuristic analysis differ from results obtained from task-oriented evaluation techniques (thinking aloud, cognitive walkthrough, etc…)[4]. Heuristic analysis is not a task-oriented evaluation technique, as it evaluates the interface's adherence to a list of rules rather than the interface's tendencies to help or hinder the performance of required tasks. Task-oriented evaluation techniques yield feedback about the ability to perform specific tasks, which is of great interest, but the holistic nature of heuristic analysis can produce additional interface feedback which is not tied to the performance of specific tasks (feedback on the overall consistency of an application's 'look and feel', for instance can be easily addressed by heuristic analysis, but is not the target focus of a task-oriented evaluation [4]). For this reason, heuristic analysis should be performed in addition to task-oriented evaluation in order to achieve the maximum insight into an application's usability.

A drawback of heuristic analysis is that the results of heuristic analysis provide only a list of mistakes and evaluators' complaints. While this is true, this list of very specific concerns can be used to formulate a definite plan of action [10]. By focusing the analyst's mind on identifying violations of 'rules' [12], heuristic analysis simplifies the discovery process required to identify why exactly portions of an interface may appear confusing. Although admittedly heuristic analysis does not directly propose solutions to identified problems, clarifying the exact nature of each problem is highly beneficial.

Another limitation of heuristic analysis is that the best evaluation is only as good as the set of heuristics used in its generation. For this reason it is crucial to use the most comprehensive and well-tested heuristics available for analysis purposes. This set of heuristics should be as minimal as possible, so as to reduce the evaluator's workload [8], but must still be detailed enough so as to enable the evaluator to identify critical usability concerns. Nielsen and Molich's heuristics are well respected as valid indicators of usability [13], but the limitation

remains that only interface issues which are addressed by the chosen heuristic can be reported by heuristic analysis.

1.3.1.4 Heuristic analysis of RACOL screen interface

To perform this study, the example study described in [9] (a group of individuals was required to use Nielsen and Molich's nine point heuristic to perform a heuristic analysis of a sample system) was used as a model.

I functioned as the 'interface expert', collating the results of two independent analyses One was performed by myself, the other performed by Christopher Thompson (view his complete report in Appendix C) an experienced software developer with a Computing Science degree from the University of Alberta. I obtained Christopher Thompson's analysis, then integrated the results into a comprehensive heuristic analysis representing both our findings.

Observations marked with (CT) are the observations contributed by Christopher Thompson which were not also present in my independent analysis. Observations marked with (RT) were present in my analysis, but not Christopher Thompson's analysis. Unmarked observations were found in both analyses. This is an important characteristic of heuristic analysis – no one observer will catch all of the heuristic infractions contained in an interface, so multiple analyses are valuable in obtaining more complete feedback.

When attempting to identify areas of the RACOL screen interface which conflicted with Nielsen and Molich's usability guidelines, we considered the clarity of the user interface as it might appear to a computer novice, since the target user community for the RACOL application – teachers – may contain some individuals who have less computer experience and technical confidence than their colleagues.

While all observed issues have been noted, including some which may appear minor, problems marked 'SERIOUS' are ones which severely impede system usage. Rectification of these serious issues should be considered a high priority.

Bold text indicates the heuristic definitions, with italics signifying Nielsen and Molich's own elaboration further describing each guideline.

Please see Appendix A for the screen diagrams corresponding with each referenced figure.

Use simple and natural dialogue.

Dialogues should not contain irrelevant or rarely needed information. Every extraneous unit of information in a dialogue competes with the relevant units of information and diminishes their relative visibility. All information should appear in a natural and logical order.

1. Using all capital letters makes text hard to read. (RT)
2. Whenever possible abbreviations and acronyms should be avoided.
3. The user's typed password should not be visibly echoed (fig 2). (RT)
4. Echoing the currently selected site in the display bar loses semantic value when multiple sites are selected (fig 6). (RT)
5. It should be decided whether or not the time and date is really required to be present on every screen. Some screens like the

Password screen (fig 2) have nothing to do with time or date. However, consistency should also be considered (see problem 20).

6. Many displays contain superfluous information. Displaying which network mode is selected (fig 3) or which schools are selected (fig 4,5,6,7,8) is redundant, since the selected schools/modes change colors. Using the display as additional feedback is not required. (CT)
7. It is unclear what units (hours:minutes or minutes:seconds?) the timer represents (fig 9,12) (CT)

Speak the user's language.

The dialogue should be expressed clearly in words, phrases, and concepts familiar to the user rather than in system-oriented terms.

8. The compression information does not belong on the user's screen interface. Instead of having the user choose between "MPEG-2" and "H.320/H.323" perhaps a label explaining the scenario in which one would choose a particular CODEC would be more intuitive (fig 3).
9. In the list of possible sites and their status (idle, connecting, etc…) it is not clear what "H.320 IDSN / H.323 IP CODEC" signifies. If it is a special connection to a certain site a more intuitive label should be found (fig 4-8).
10. The use of the term 'ENTER' instead of 'OK' may confuse novice users familiar with Windows dialogue terms (fig 2,3,9). (RT)
11. When 'MULTICAST' is selected, the status bar (see problem 6) displays 'FVSD MPEG HUB SELECTED', which is unintuitive (fig 5).
12. When 'BROADCAST' is selected, the status bar (see problem 6) displays 'MPEG-2 MULTICAST SELECTED' which is unintuitive (fig 7).
13. On the system timer menu 'ENTER TIME REQUIRED' is less explanatory than 'ENTER THE LENGTH OF TIME REQUIRED FOR YOUR SESSION'. The overall brevity of this application's text contributes to this application's highly technical feel (fig 9). (RT)
14. There is an unintelligible button appearing on the Teleconference screen with a strange symbol '<<<' on it. This button should be labeled in a more intuitive way (fig 19).
15. It is not explicit what 'RESET QUEUE' will do – a better term might be 'EMPTY QUEUE' or 'PURGE QUEUE' in order to

express the idea that the question queue will be erased (fig 14) (CT)

Minimize the user's memory load.

The user's short-term memory is limited. The user should not have to remember information from one part of the dialogue to another. Instructions for use of the system should be visible or easily retrievable whenever appropriate. Complicated instructions should be simplified.

16. The Student Questions screen and any other screen which may be displayed during an ongoing session should definitely display the 'TIME REMAINING' countdown. The instructor should be kept aware at all times how much time is remaining in the session (fig 13, 14,15, 16, 17, 18, 19, 20). (RT)

Be consistent.

Users should not have to wonder whether different words, situations, or actions mean the same thing. A particular system action – when appropriate – should always be achievable by one particular user action. Consistency also means coordination between subsystems and between major independent systems with common user populations.

17. The application has a 'look and feel' which is consistent with existing Smart Classroom interfaces, but different from a standard Windows/Mac interface. This may confuse novice users who are used to seeing buttons and display panels which match familiar operating systems. (See problems 10, 18, 19).
18. **SERIOUS:** There is no difference between a pressed button and an information display. Both have thickened edges, maroon backgrounds and yellow text. This causes confusion when a user is confronted with an unfamiliar screen in which it may not be readily apparent which interface items are buttons and which are displays. This is particularly obvious in the Student Question and Camera Preset screens (fig 13,14, 17). (RT)
19. Some buttons are colored, with rounded edges and "glassy" rims, and some have sharp bevels and are gray in color. This inconsistency is visually unpleasant, plus serves to further confuse the user as to what is a button and what is a display (see problem 18) since not all buttons are recognizably similar. This is noticeable in screens where both types of buttons are present (fig 2, 4, 5, 6, 7, 8, 9, 10, 11, 12, 15, 17, 18, 19, 20, 22, 23). (RT)
20. The placement of the time/date displays are consistent everywhere but on the session selection screens (fig 4, 5, 6, 7, 8) where they appear on the left hand side, and on the

Multipoint Camera, Videoconference and Camera Preset screens where they are unaccountably absent. If they are considered important enough to be featured in every other screen layout (see problem 5) there is nothing unusual about the semantics of these screens. For consistency reasons they should exist there as well (fig 18, 20, 23).

21. The placement of the 'ENTER' and 'CLEAR' buttons in the numeric keypad is unusual and particularly in the case of the 'ENTER' button may be easily overlooked (fig 2, 23). (RT)

Provide feedback to the user.

The system should always keep the user informed about what is going on by providing him or her with appropriate feedback within reasonable time.

22. **SERIOUS:** When a student indicates to the instructor that s/he has a question, the system automatically jumps to the Student Question screen. While the instructor may choose to ignore the question and return to his/her previous screen this presents a feedback problem, as the instructor sees the system behaving in a way (the switching to the Student Question screen) inconsistent with the instructor's actions (the instructor has done nothing to warrant this feedback). A more appropriate solution might be to flash an indicator on the Menu Screen when a student question is queued rather than overriding the instructor's actions to switch the screens automatically (fig 13, 14). (RT)

Provide clearly marked exits.

A system should never capture users in situations that have no visible escape. Users often choose system functions by mistake and will need a clearly marked "emergency exit" to leave the unwanted state without having to go through an extended dialogue.

23. Is the 'EXIT' button a signal to quit the application? If so, 'QUIT' might be a better choice of word, as it is more consistent with other applications (fig 3, 4, 5, 6, 7, 8, 10, 11).
24. 'RETURN' is in the same stylized font as 'EXIT'. The use of a traditional font and color associated with the 'real world' exit sign on a door does not intuitively indicate 'RETURN TO THE PREVIOUS SCREEN' (fig 13, 14, 15, 16, 17, 18, 19, 20, 23, 24).
25. **SERIOUS:** The screens which contain a 'RETURN' sign appear to save changes when the user returns to the previous screen. There is no way for the user to decide not to apply changes made on these screens which may have been made in

error if s/he does not remember exactly the previous state of the screen (fig 13, 14, 15, 16, 17, 18, 19, 20, 23, 24). (RT)

26. It is not apparent how the user can exit from the Password, Service or System Timer screen (fig 2, 9, 22).
27. **SERIOUS**: There is no indication to the user as to how to progress from the 'splash screen'. The user has no way of knowing that s/he must click this screen to enter the application (fig 1). (CT)

Provide shortcuts.

The features that make a system easy to learn – such as verbose dialogues and few entry fields on each display – are often cumbersome to the experienced user. Clever shortcuts – unseen by the novice user – may often be included in a system such that the system caters to both inexperienced and experienced users.

(there are no lengthy dialogues or informational messages to avoid)

28. There does not appear to be a way to adjust hours/minutes separately on the timer, which would imply that the arrow keys raise and lower the time by the smallest units. This could make setting longer sessions very inconvenient (fig 9,12). (CT)

Provide good error messages.

Good error messages are defensive, precise, and constructive. Defensive error messages blame the problem on system deficiencies and never criticize the user. Precise error messages provide the user with exact information about the cause of the problem. Constructive error messages provide meaningful suggestions to the user about what to do next.

29. An error message should be provided if a connection to a requested site is unable to be made (fig 4, 5, 6, 7, 8). (RT)
30. Users should be easily able to restore default camera and audio settings if they adjust them in the Audio or Camera screens and are not satisfied with their results (fig 15, 17, 18). (RT)

Prevent errors.

Even better than good error messages is a careful design that prevents a problem from occurring in the first place.

31. Users should be asked whether they want to apply changes in sub-screens (see problem 25). (RT)

1.3.2 Cognitive walkthrough

1.3.2.1 Cognitive walkthrough methodology

A cognitive walkthrough is an expert simulation of the user experience [4]. An expert (or group of experts) 'walks through' the steps required to carry out a selected task using an application, attempting to place him/herself in the mind of the target user. The expert must consider whether each action required to carry out the task is intuitive to the target user, based on the target user's knowledge base as well as the interface's guidance and feedback.

Before beginning the evaluation the expert must clearly define the knowledge base of the application's target user group, the task to be 'walked through', and a list of the actions the user would have to complete in order to complete the task [11].

The expert then needs to describe the task completion process from the target user's perspective, outlining each required step. The description should be in the form of a believable narrative [4]. The expert needs to then assess this description, and determine if the user would have a reasonable motivation for completing each step of the task completion process.

To do this, the expert should make four considerations about each step of the task completion process. The expert should "consider the user's goal", "consider the accessibility of the correct control", "consider the quality of the match between the control's label and the goal", and "consider the feedback provided after the control is acted upon"[11]. Each step of the task completion process should be consistent with the goal the user would reasonably have in mind at this stage of the progress, should be accomplished by a mechanism which is visible and accessible to the user at the time it is required to be performed, should be the obvious course of action for the user, and should provide adequate feedback to let the user know s/he has performed the appropriate action.

If the expert cannot explain how these four considerations provide motivation to each required action, then s/he has discovered a problem with the user interface. Having to formally identify reasonable motivations for user actions helps identify any unrealistic assumptions which designers and developers may have made when

determining whether a user will be able to figure out how to use an interface [4].

Cognitive walkthroughs should be performed to address major and often-performed tasks ("the day in the life of the user" [2, p.126]) in order to assess the application's major functionality, as well as unusual, atypical tasks, in order to ensure that less commonly accessed areas of the application receive adequate inspection, particularly since these unfamiliar events occur with less frequency during actual use, requiring them to be especially intuitive to the user.

1.3.2.2 Merits and shortcomings of the cognitive walkthrough technique

The cognitive walkthrough technique is ideal for exploring how novice users might immediately interpret and interact with an unfamiliar interface, but it is also applicable to situations where users are expected to become accustomed to an application over a long period of use [4]. Regardless of the fact that repeated use will allow users to familiarize themselves with even the most unintuitive interfaces, a *good* interface which is intended for long-term use should nonetheless require users to perform tasks via a series of actions which can be attributed to reasonable motivations [4].

A cognitive walkthrough should be performed during the development process [4] in order to obtain insight at an early stage as to where the interface could be improved. It is a valuable and cost-effective method of obtaining insight into where an interface could potentially confuse a user, without having to go to the expense of actually having users involved in the exploratory process. Of course, the expert simulation of a user walkthrough has the obvious drawback of being only a simulation – the expert's feedback should not be considered an exact substitute for genuine user feedback [14].

1.4 Cognitive walkthrough of 'initiating a lecture' via the RACOL screen interface

Description of users and the users' relevant knowledge

The target users of this system who will perform this task are teachers who are attempting to initiate a videoconference with remote students. These teachers may come from a variety of backgrounds, with varying degrees of technical proficiency. It should be assumed that the teachers will have at least some familiarity with Windows or Macintosh computer systems, since their occupational requirements should have brought them into some contact with personal computing, but it would be unwise to assume that they are all highly confident computer users, comfortable with exploring unfamiliar systems. For this reason, the less skilled members of the target user group may become more easily confused by an unintuitive interface.

1.4.1 Description of the task to be analyzed

The task to be analyzed by this cognitive walkthrough is the process of initiating a H.320/H.323 videoconference with the Calgary Cyberport. This task is representative of the type of task an instructor who conducts videoconferenced lectures would be required to perform on a regular basis.

1.4.2 Correct actions required in order to complete the task using the RACOL interface

Each action is analyzed by answering the following four questions. These questions are quoted from text authored by Clayton Lewis and John Rieman, creators of the cognitive walkthrough method [4].

A. Will users be trying to produce whatever effect the action has?

B. Will users see the control (button, menu, switch, etc.) for the action?

C. Once users find the control, will they recognize that it produces the effect they want?

D. After the action is taken, will users understand the feedback they get, so they can go on to the next action with confidence?

Click upon the logo to enter the system (fig 1)

The user looks at the screen and attempts to figure out how to get the system up and running. All s/he sees is the logo splash screen. The user must determine that the way to enter the system is to click on the splash screen.

A. Yes, the users will be attempting to begin using the system.
B. Yes, the users will see the logo.
C. No, there is no reason to believe that the user would intuitively know that clicking the logo would produce the desired effect of allowing entry to the system.
D. Yes, the provided feedback is that the screen will change to a password screen (fig 2). They should then feel confident that they have entered the system.

Summary: Currently, the esthetic merit of an attractive splash screen does not compensate for the fact that the user has no intuitive way to progress past it. While the logo splash screen does provide an attractive introduction to the system, the screen would still look

attractive and professional if text was posted below it instructing user needed to click the screen in order to enter the system.

Enter a password (fig 2)

The user sees a screen prompting him/her to enter a password. The user must enter the password by pressing buttons on the numeric keypad.

A. Yes, the prompt above the password entry mechanism makes it clear that the user is to enter a password.
B. Yes, the users will see the numeric keypad.
C. Yes, the users should be able to understand that they are to use the on-screen numeric keypad to enter the password.
D. Yes, the feedback is that the password will appear in the display screen as they type it. They should be confident that their numeric entries are being accepted by the system.

Summary: The password entry mechanism is satisfactory, however, it would be more acceptable to echo a symbol instead of visibly echoing the numeric password, for security reasons.

Click the button labeled 'ENTER' (fig 2)

After entering the password, the user must click 'ENTER' to submit the password for authentication.

A. Yes, even a moderately experienced user should be familiar with the necessity to press a key to signify that the entire password string has been entered, but to remove any confusion, it could be explicitly stated in the prompt box where only 'ENTER PASSWORD' is currently listed.
B. Yes, the 'ENTER' button is visible.
C. Yes, the user should infer that 'ENTER' is the key to press after entering the entire password, however, Windows/Macintosh interfaces, with which the user is likely familiar would use the label 'OK' for this purpose.
D. Yes, the user receives feedback after entering the password – the system moves on to the next screen (fig 3), signifying that the password has in fact allowed entry into the system.

Summary: There are no major problems with the way this task is accomplished.

If the system was to be made more explicitly instructional, the fact that the user needs to press 'ENTER' after inputting his/her password could be added to the prompt box. As well, the designers might want to re-think the use of the term 'ENTER' over the more familiar term 'OK'.

These are relatively minor problems given that most users should be familiar with the numeric keypad password entry process based on their everyday interactions with bank machines or password key-lock systems.

Click the button which is labeled H.320/H.323 (fig 3)

The user must select the button labeled H.320/H.323. The user knows that s/he wants to initiate a videoconference with the Calgary Cyberport, but s/he must also know that H.320/H.323 is the CODEC required to connect to the Calgary Cyberport, since not all sites are accessible via both an H.320/H.323 and MPEG-2 connection.

> A. No, at this point the user is concentrating on creating a connection to the Calgary Cyberport, not choosing what type of CODEC (H.320/H.323 or MPEG-2) to use. In the case where the desired remote site can be reached via either an H.320/H.323 or MPEG-2 connection, it is still unclear to the target user what advantage the use of each CODEC serves.
> B. Yes, the user is able to see the mechanism for selecting the CODEC.
> C. Yes, it is clear which type of CODEC each button will select, since they are clearly labeled, however it is not clear that the user will understand what each labeled term means.
> D. Yes, feedback is given in two places as each button is selected: the button changes color, and the display screen echoes the selected mode.

Summary: This screen is problematic. Firstly, the user is being required to make a decision based on inadequate information since the screen does not offer any suggestion as to which remote sites require the MPEG-2 CODEC and which require the H.320/H.323 CODEC. There is no reason the user should have to memorize which type of CODEC

will provide access to the desired remote site. In the situation where a site is accessible either way, and the CODEC choice reflects bandwidth/quality compromises, the user should be able to make his/her decision based on less arcane terminology. The user has no need to be concerned with the acronyms relating to the networking software.

A better approach would be to have the user select the site s/he wishes to connect to out of a list of available sites, letting the computer decide (transparently to the user) whether this site requires the MPEG-2 or H.320/H.323 CODEC, configuring itself accordingly, or if necessary, allowing the user to make a choice about CODEC related issues (bandwidth and quality) using more explanatory terminology.

Click the button labeled 'ENTER' (fig 3)

Once the user has highlighted the CODEC s/he wishes to use, s/he must press 'ENTER' in order to select the CODEC and progress to the next screen.

A. Yes, the user is wanting to process the highlighted data.
B. Yes, the 'ENTER' button is visible.
C. Yes, the user should infer that 'ENTER' is the key to press after highlighting one of the above three buttons, as requiring a signal to process a selection choice is a commonplace GUI feature.
D. Yes, the user receives feedback after pressing 'ENTER' – the system moves on to the timer screen (fig 9), signifying that the selection has been accepted.

Summary: There are no major concerns with this step, however, as stated above, the need to press enter after selecting a button could be explicitly stated, and the more familiar term 'OK' could be considered as a replacement for 'ENTER'.

Adjust the length of the videoconference by clicking the arrow keys (fig 9)

To set the length of time the videoconference is to run for, the user must adjust the default time (1 hour) by incrementing or decrementing using the arrow keys.

A. Yes, it is clear that the user must set a time length for the upcoming videoconference.
B. Yes, the mechanisms for shortening and lengthening the time frame are visible.
C. No, the users may be confused as to what time increments are represented by the arrow keys, as well as what units the time is displayed in.
D. Yes, the user receives feedback when the displayed presentation time changes as they press the arrow keys.

Summary: This screen would be more intuitive if time units were stated, and the time increments were specified on the arrow keys.

Click the button labeled 'ENTER' (fig 9)

The user must click 'ENTER' to start the clock after s/he is satisfied with the time length displayed.

A. Yes, the user is now satisfied with the adjusted time and wanting to process it.
B. Yes, the 'ENTER' button is visible.
C. Yes, the user should infer that 'ENTER' is the key to press after adjusting the time because requiring a signal to process adjusted data is a commonplace GUI feature.
D. Yes, the user receives feedback after pressing 'ENTER' – the system moves on to the main screen (fig 11), signifying that the videoconference length has been accepted.

Summary: There are no major concerns with this step, however, as stated above, the need to press enter after adjusting the time could be explicitly stated, and the more familiar term 'OK' could be considered as a replacement for 'ENTER'.

Select 'VIDEOCONF' (fig 11)

The user must click 'VIDEOCONF' in order to continue initiating the videoconference.

A. Yes, the user is still wanting to initiate a videoconference with a remote site. (It is unfortunate, however, that the

selection of the 'VIDEOCONF' button will still only get the user to another sub-screen that must be clicked through before achieving this goal)
B. Yes, the 'VIDEOCONF' button is visible, although it is not in a prominent position of importance on the screen.
C. No, it is not reasonable to assume that the user will recognize the 'VIDEOCONF' button as the button which will allow him/her to connect to a remote site.
D. No, the feedback provided to the user at this point is not sufficient to inform him/her that s/he is progressing towards making a connection to a remote site. The screen which appears after the 'VIDEOCONF' button is pressed (fig 20) does not have any reference to videoconferencing on it. The feedback of seeing this screen may make the user wonder if s/he has done something wrong.

Summary: This step and the next five steps following it require serious reconsideration if an intuitive method of initiating a videoconference is to be developed. This screen and the screens which follow do not offer the user any guidance as to how to initiate a videoconference.

Select 'MENU' (fig 20)

The user must click 'MENU' in order to reach the menu choices which will allow him/her to choose a remote site and make a connection. These menu choices will appear on a different screen located across the room from the touch panel s/he is working on, so s/he must be prepared to observe all further interaction with the menu choices by watching this screen across the room, yet still navigating this screen via the touch panel at his/her fingertips.

A. Yes, the user is still wanting to initiate a videoconference with a remote site. Clicking the 'MENU' button will help them progress towards this goal, however as mentioned before, achieving this goal should be possible via fewer stages.
B. Yes, the 'MENU' button is visible, although it is not in a prominent position of importance on the screen.
C. No, it is not reasonable to assume that the user will recognize the 'MENU' button as the button which will allow him/her to connect to a remote site. The label 'MENU has no semantic value which would indicate this.

D. No, the feedback provided to the user after s/he clicks 'MENU' is that an entirely unrelated and inconsistent screen interface (the POLYCOM interface appears to have been spliced in at this point to avoid duplicating functionality) appears on an entirely different screen from the panel the user is currently interacting with (fig 25). The user may not even realize that a separate window is now visible on an entirely different physical screen.

Summary: This step is extremely unintuitive to the user. The user should not expect that pressing one button on a previously consistent user interface will cause a completely separate interface to appear on a different screen positioned at a different point in the room. (See fig 27 for a view of the 2-screens and their physical positioning).

Additionally, there is nothing about the screen displayed in fig 20 that encourages the user to click 'MENU' in order to continue initiating a videoconference – there is nothing about fig 20 that even indicates to the user that s/he is still on the correct path towards initiating a videoconference.

Use the arrow keys on the panel to highlight 'ADDRESS BOOK' (fig 20, 25)

The user must select the remote site s/he wishes to connect to. In order to do this s/he must first highlight the 'ADDRESS BOOK' icon on the screen located across the room, via the arrow keys on the touch panel at his/her fingertips.

A. Yes, the user is wanting to initiate a videoconference with a remote site which is contained in the Address Book, so selecting the book would further that goal.
B. No, the arrow keys are not visible to the user if s/he is looking at the screen which contains the target icon (the 'ADDRESS BOOK'). If the user is looking at the arrow keys on the screen in front of him/her s/he cannot see the 'ADDRESS BOOK' target.
C. No, it is unreasonable to expect the user to understand that the arrow keys on one display panel affect what is selected on a different display panel. Furthermore, it is not clear that selecting the 'ADDRESS BOOK' icon will bring the user any closer to his/her goal of connecting to a remote site.

D. No, the feedback provided to the user is not sufficient. The user must continually move his/her eyes back and forth from the control panel screen to the target menu screen in order to see the highlighting of the menu which serves as feedback. This is confusing and frustrating, and is made even more difficult by the control panel's touch screen interface – there is no way to adjust the arrow keys by touch, because the user has to visually observe the control screen in order to position his/her fingers over the touch target.

Summary: This method of interface control (a touch panel on one screen affecting the highlighting of information on a separate screen) is unwieldy, unintuitive and inefficient. Additionally, it is not obvious that selecting the 'ADDRESS BOOK' will bring the user any closer to his/her goal of starting a videoconference.

Click the center button on the panel's keypad to select the address book (fig 20, 25)

Once the user has highlighted the 'ADDRESS BOOK' icon on the screen located across the room, s/he must use the button marked with a circle at the center of the arrow keyset on the touch panel to select this icon.

A. Yes, the user wants to initiate a videoconference with a remote site which is contained in the Address Book, and since the Address Book is now highlighted, s/he will want to select it.
B. No, the selection button is not visible to the user if s/he is looking at the screen which contains the target icon (the 'ADDRESS BOOK'). If the user is looking at the selection button on the screen in front of him/her s/he cannot see the 'ADDRESS BOOK' target.
C. No, it is unreasonable to expect the user to understand that the selection button on one display pane affect what is selected on a different display panel. Furthermore, it is not clear that selecting the 'ADDRESS BOOK' icon will bring the user any closer to his/her goal of connecting to a remote site. Most importantly there is no semantic meaning to a round circle in the center of the arrow key layout which would

suggest to the user that this key should be used to make a selection.
D. Yes, when the key is pressed the Address Book (fig 26) opens, providing feedback.

Summary: As stated above, having the controls on one screen and the feedback on another is confusing. Additionally the circle used to represent the selection mechanism has no semantic meaning to the user.

Use the arrow keys on the panel to highlight Calgary Cyberport (fig 20, 26)

The user must select the remote site s/he wishes to connect to out of the list in the Address Book. In order to do this s/he must first highlight the appropriate remote site entry on the screen located across the room, via the arrow keys on the touch panel at his/her fingertips.

A. Yes, the user wants to highlight the 'Calgary Cyberport' entry in the Address Book.
B. No, the arrow keys are not visible to the user if s/he is looking at the screen which contains the contents of the Address Book. If the user is looking at the arrow keys on the screen in front of him/her s/he cannot see the contents of the Address Book.
C. No, it is unreasonable to expect the user to understand that the arrow keys on one display panel affect what is selected on a different display panel.
D. No, the feedback provided to the user is not sufficient. The user must continually move his/her eyes back and forth from the control panel screen to the target menu screen in order to see the highlighting of the menu which serves as feedback. This is confusing and frustrating, and is made even more difficult by the control panel's touch screen interface – there is no way to adjust the arrow keys by touch, because the user has to visually observe the control screen in order to position his/her fingers over the touch target.

Summary: This method of interface control (a touch panel on one screen affecting the highlighting of information on a separate screen) is unwieldy, unintuitive and inefficient.

Click the center button on the panel's arrow pad to select Calgary Cyberport (fig 20, 26)

Once the user has highlighted the appropriate remote site entry on the screen located across the room, s/he must use the button marked with a circle at the center of the arrow keyset on the touch panel to select this entry and initiate the videoconference.

A. Yes, the user wants to initiate a videoconference with the remote site 'Calgary Cyberport' which is highlighted in the Address Book, and since the remote site is now highlighted, s/he will want to select it.
B. No, the selection button is not visible to the user if s/he is looking at the screen which contains the site name. If the user is looking at the selection button on the screen in front of him/her s/he cannot see the highlighted site name.
C. No, it is unreasonable to expect the user to understand that the selection button on one display pane affect what is selected on a different display panel. Most importantly there is no semantic meaning to a round circle in the center of the arrow key layout which would suggest to the user that this key should be used to make a selection.
D. Yes, when the key is pressed the videoconference is initiated and the screen displays the remote site, providing feedback.

Summary: As stated above, having the controls on one screen and the feedback on another is confusing. Additionally the circle used to represent the selection mechanism has no semantic meaning to the user.

Press 'RETURN' to return to the panel from which the lecture can be controlled (fig 20)

The user must then return to the main control panel in order to continue controlling the devices used in the videoconference. To do so, s/he presses the 'RETURN' button in the top left corner of the screen, remembering that the screen seen on this display immediately before the current screen was the main menu screen. The 'RETURN' button will return to this previous screen.

A. Yes, at this point the user would wish to return to the main screen (fig 11) from which s/he could control the data which is presented to the connected remote site.
B. Yes, the 'RETURN' button is visible to the user, although not in a prominent position.
C. No, it is not adequately clear to the user what the 'RETURN' button would do if pressed. It cannot be assumed that the user would infer that this would return back to the control screen, as a variety of tasks have been accomplished since the user last saw the main control screen, and the fact that it was immediately previous to this screen may not be easily remembered by the user.
D. Yes, when the button is pressed the main screen (fig 11) reappears.

Summary: This functionality would be more intuitive if 'RETURN' was replaced with text instructing the user what screen s/he would actually be returning to.

1.5 Conclusions

There are many methods of evaluating the usability of a user interface. Two evaluatory techniques – heuristic analysis and cognitive walkthrough – have been applied to the RACOL distance learning project in order to identify some specific usability concerns within the application.

The RACOL system contains a large and impressive amount of functionality, but attempts to evaluate its usability reveal that the complete functionality is not intuitively accessible. While it would easily be possible for a user to memorize the series of steps required in order to perform needed tasks using the system, some adjustments are still required in order for the goal of creating a readily intuitive and user-friendly distance learning classroom to be met.

Systematic performance of well-defined evaluation techniques is valuable in identifying very specific usability concerns within an application. It is hoped that the identification of these specific concerns will guide the direction of further revisions to the RACOL application in order to enhance the overall usability.

1.6 References

1. University of Alberta. "Rural Advanced Community Of Learners." Accessed April 21, 2003. <http://www.racol.ualberta.ca>.

2. Shneiderman, Ben. *Designing the user interface: Strategies for Effective Human-Computer Interaction*. 3rd ed. Reading, MA: Addison-Wesley, 1998.

3. Winer, Laura R. and Cooperstock, Jeremy. "The "Intelligent Classroom": changing teaching and learning with an evolving technological environment." *Computers and Education* 38 (2002): 253-266.

4. Lewis, Carl and Rieman, John. "Task-Centered User Interface Design – Evaluating the Design Without Users." 1993-1994. Accessed January 22, 2003. <http://www.hcibib.org/tcuid/chap-4.html>.

5. Lewis, Carl and Rieman, John. "Task-Centered User Interface Design – Evaluating the Design With Users." 1993-1994. Accessed January 22, 2003. <http://www.hcibib.org/tcuid/chap-5.html>.

6. Johnson, Robert R. *User-centered Technology*. Albany, NY: State University of New York Press, 1998.

7. Anastasiades and Panagiotes, S. "Distance learning in elementary schools in Cyprus: the evaluation methodology and results." *Computers and Education* 40 (2003):17-40.

8. Nielsen, J. and Molich, R. "Heuristic evaluation of user interfaces." ACM (1990): 249-256.

9. Nielsen, J. and Molich, R. "Improving a Human-Computer Dialogue." ACM 33 (March 1990): 338-347.

10. Cox, Donald Allan. "Supporting Results Synthesis in Heuristic Evaluation."

M.Sc. thesis, Department of Computer Science, Calgary: University of Calgary, 1998.

11. Rieman, J. and Franzke, M. and Redmiles, D. "Usability Evaluation With The
Cognitive Walkthrough." CHI 95 Conference Companion 1995: 387-388.

12. Nielsen, J. "Heuristic Evaluation How-to." Accessed April 21, 2003.

<http://www.useit.com/papers/heuristic/heuristic_evaluation.html>.

13. Iannella, Renato. "Heuristic User Interface Evaluation: Three Case Studies of Dialog
Design." Proceedings of OZCHI94 - Australian CHISIG Annual Conference. Eds. Howard & Leung. Melbourne (Nov/Dec 1994):127-132.

14. Axup, Jeff. "Usability Evaluation Methods." Accessed January 22, 2003.
<http://www.userdesign.com/usability_uem.html>.

1.7 RACOL Screen Shots

Fig 1. Logo Screen

Fig 2. Password Screen

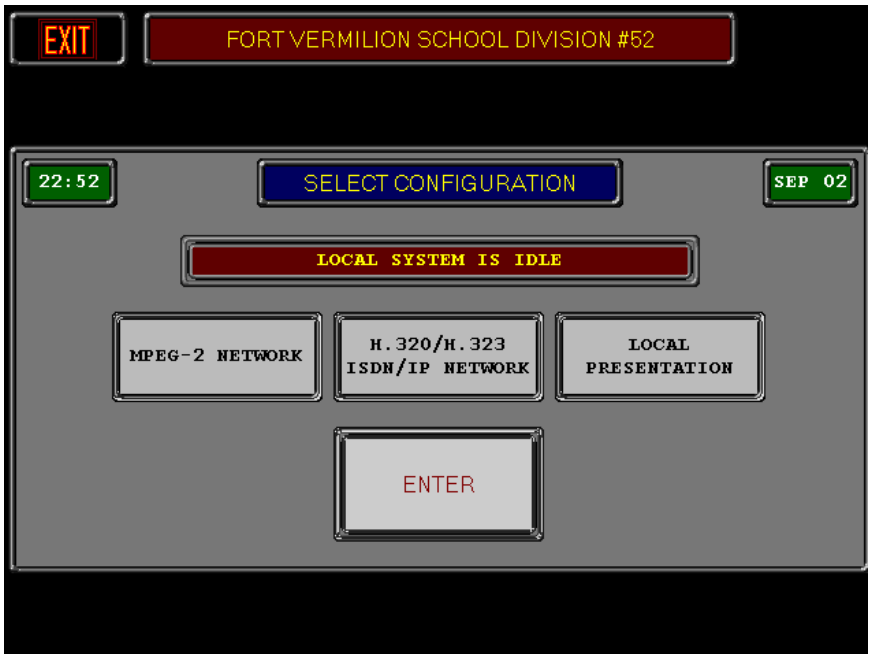

Fig 3. System Configuration Screen

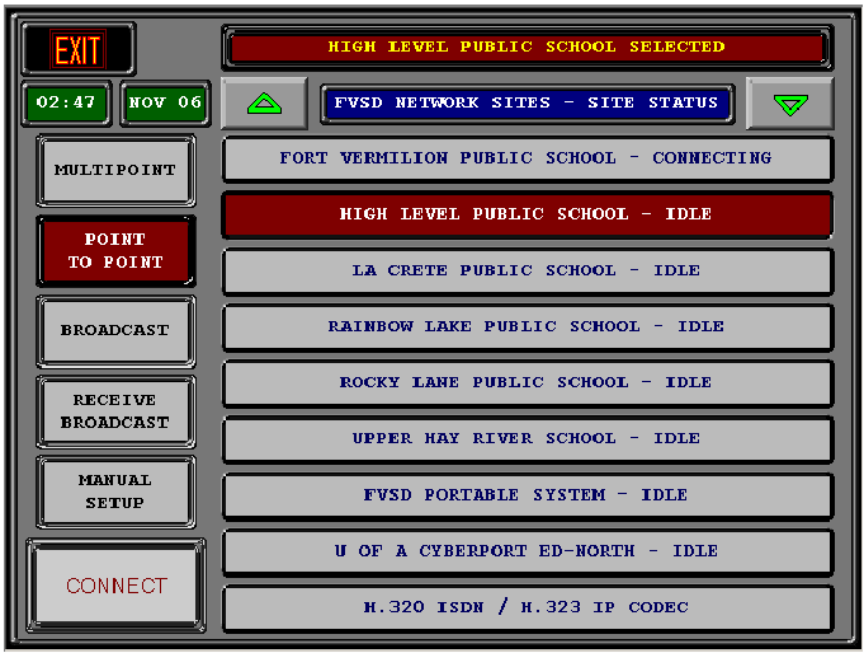

Fig 4. MPEG Screen Point-to-Point

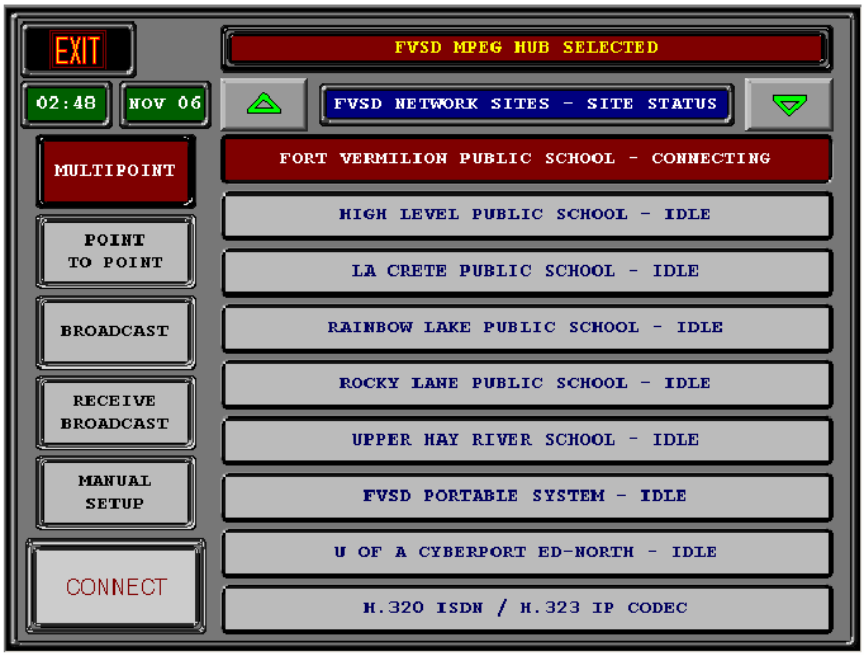

Fig 5. MPEG Screen Multipoint Selected

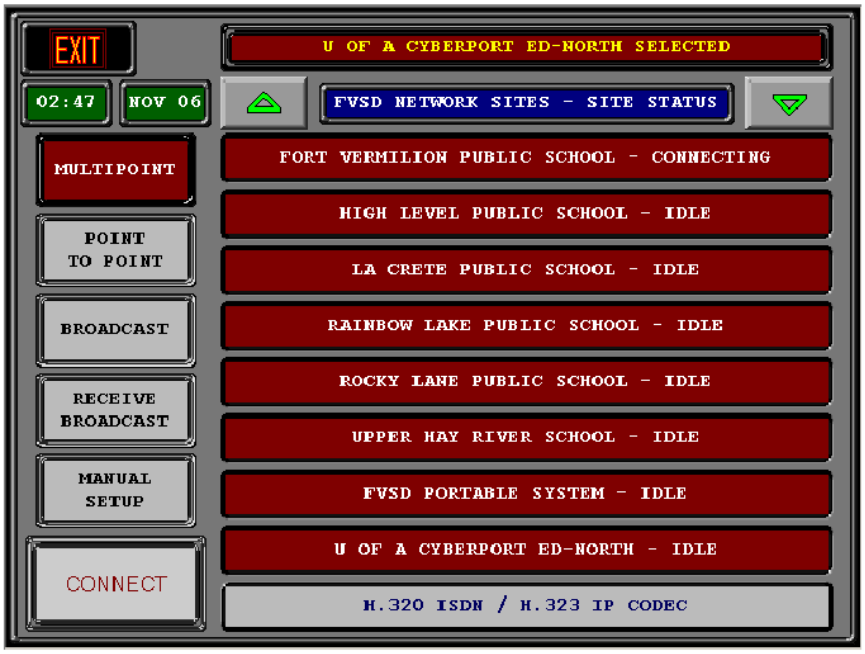

Fig 6. MPEG Screen Multipoint (8 Sites Selected)

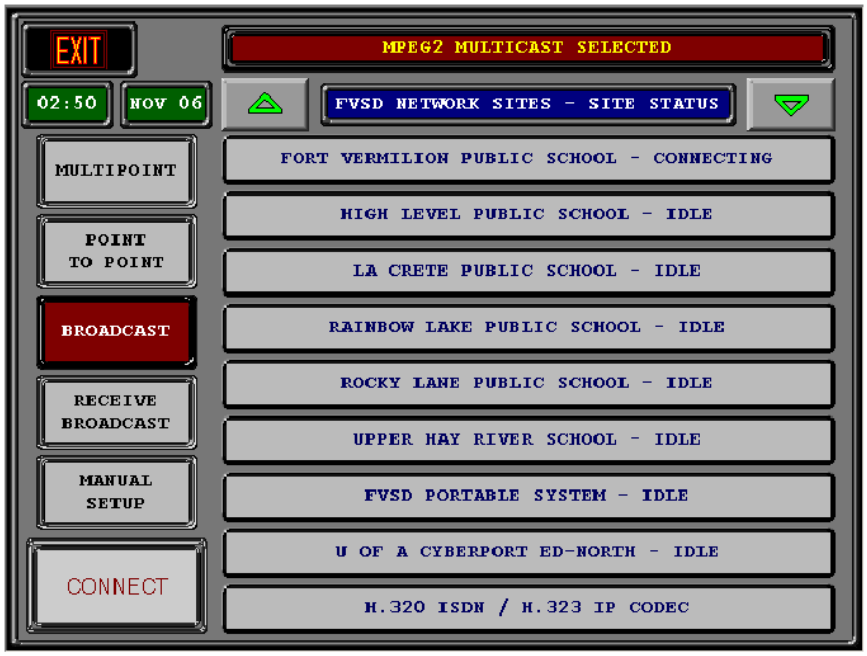

Fig 7. MPEG Screen Broadcast Selected

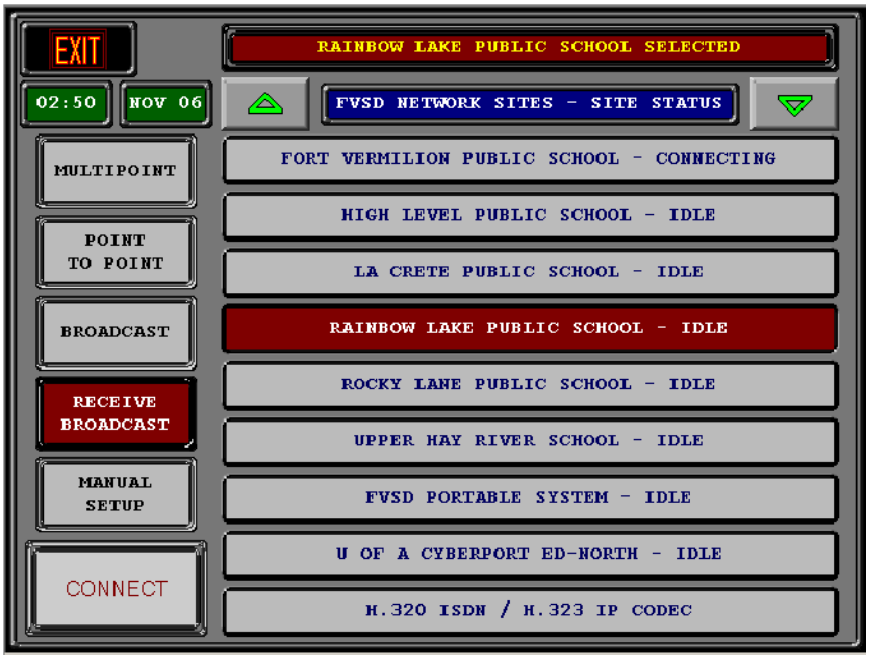

Fig 8. MPEG Screen Receiving Broadcast

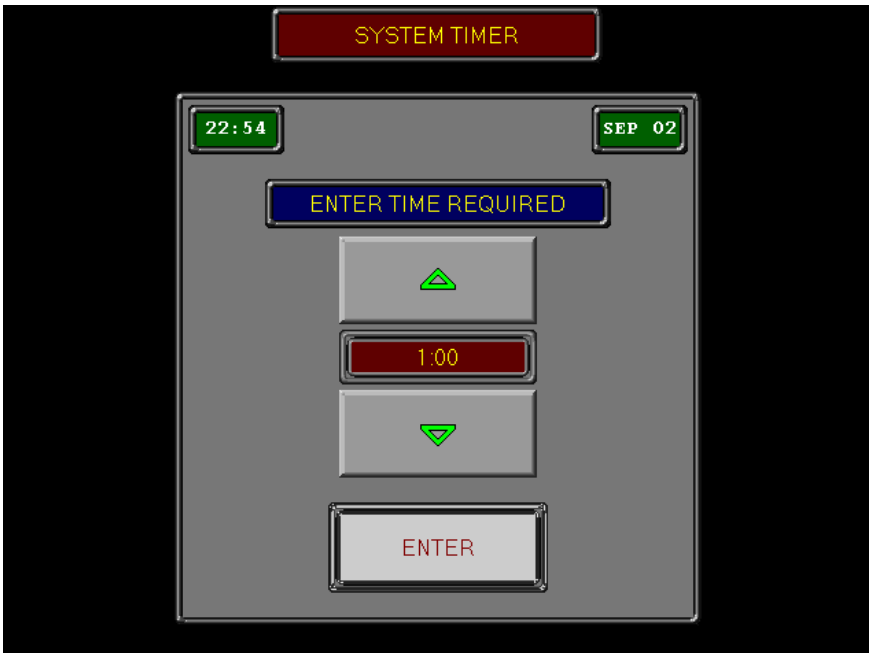

Fig 9. Time Entry Screen

Fig 10. Local Presentation Menu Screen

Fig 11. Main Menu

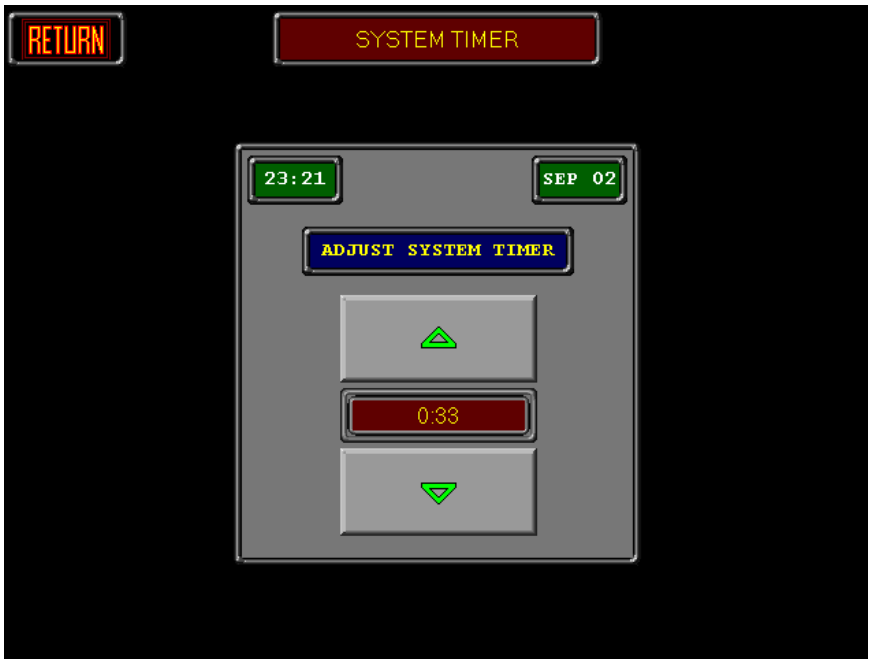

Fig 12. Adjusting Timer

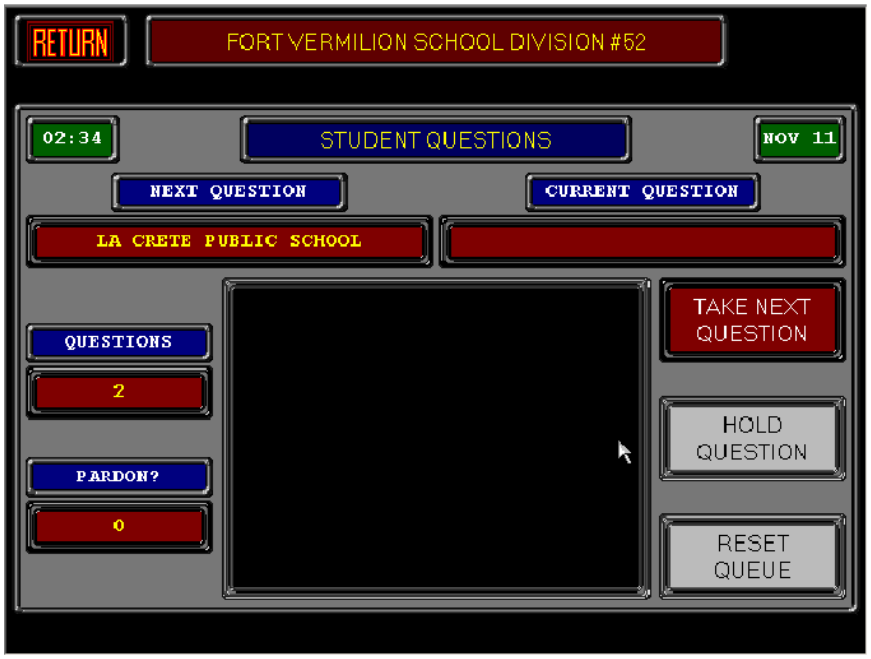

Fig 13. Question Screen (2 Questions In Queue)

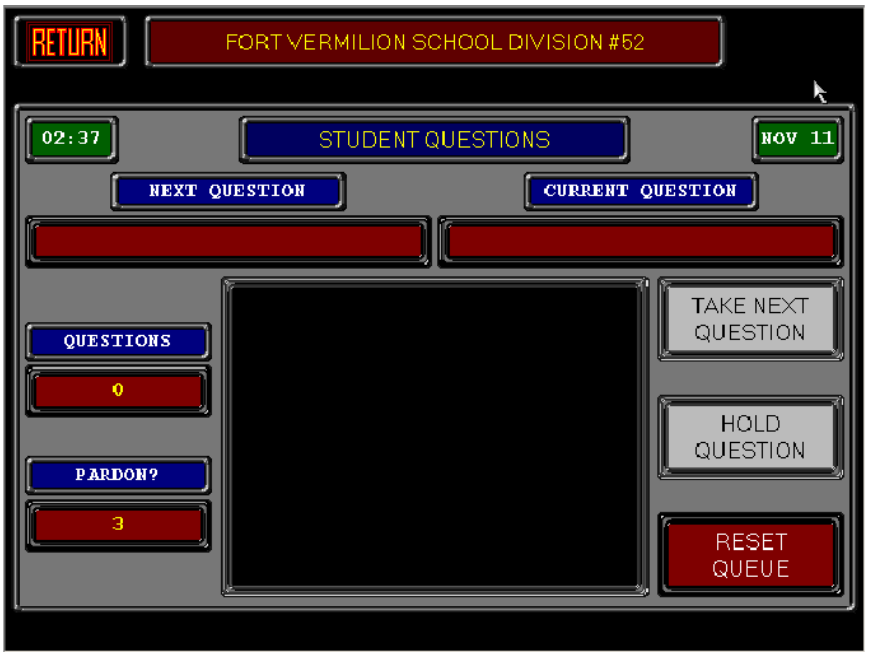

Fig 14. Question Screen (3 'confused' inputs)

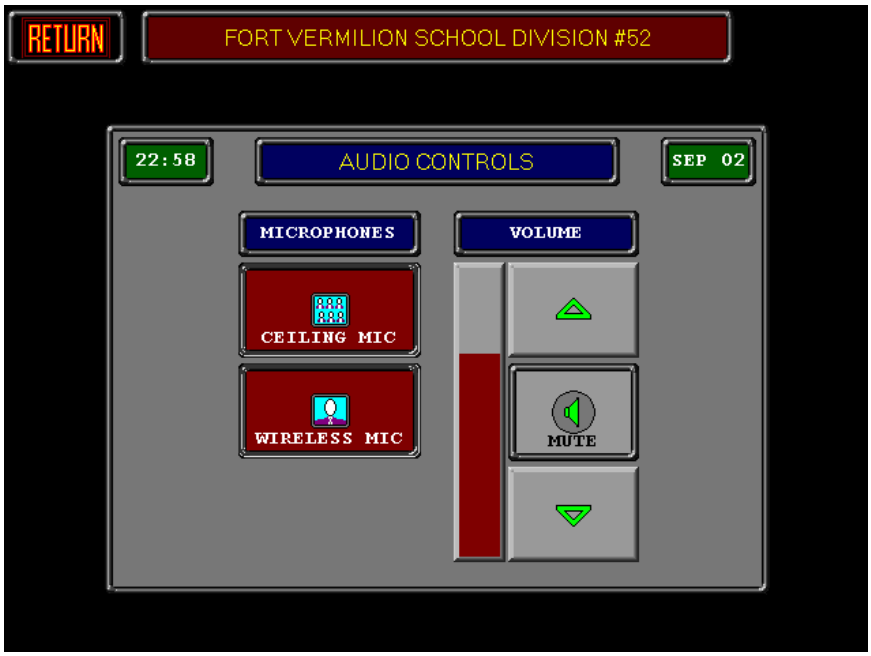

Fig 15. Audio Screen

Fig 16. Student Presentation Site Select

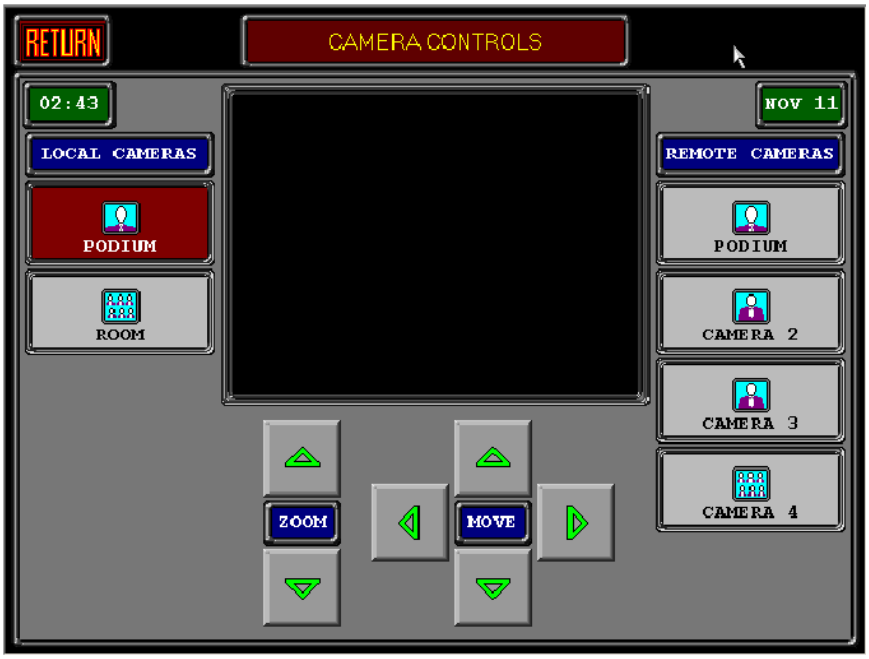

Fig 17. Camera (Point-to-Point)

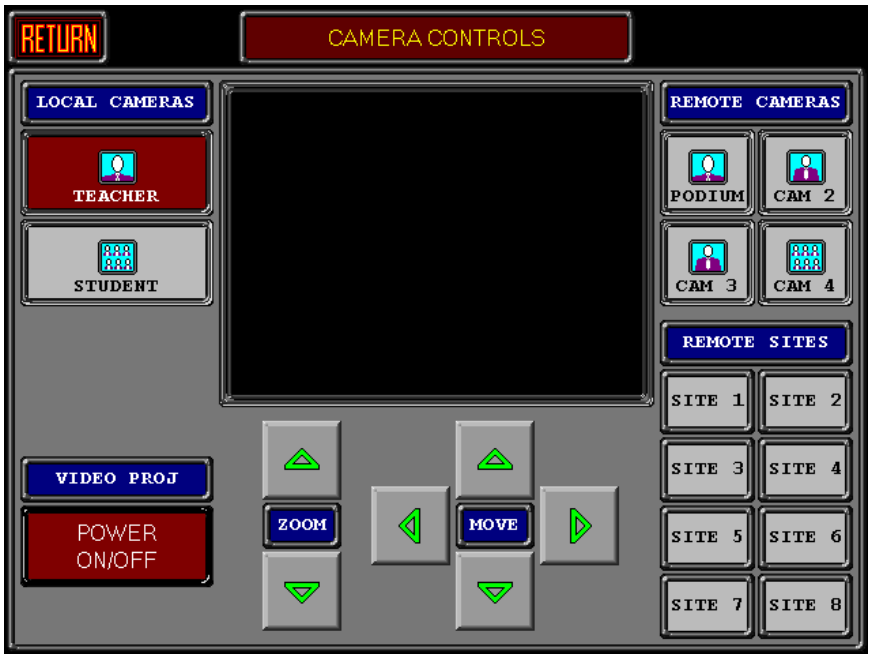

Fig 18. Camera (Multipoint)

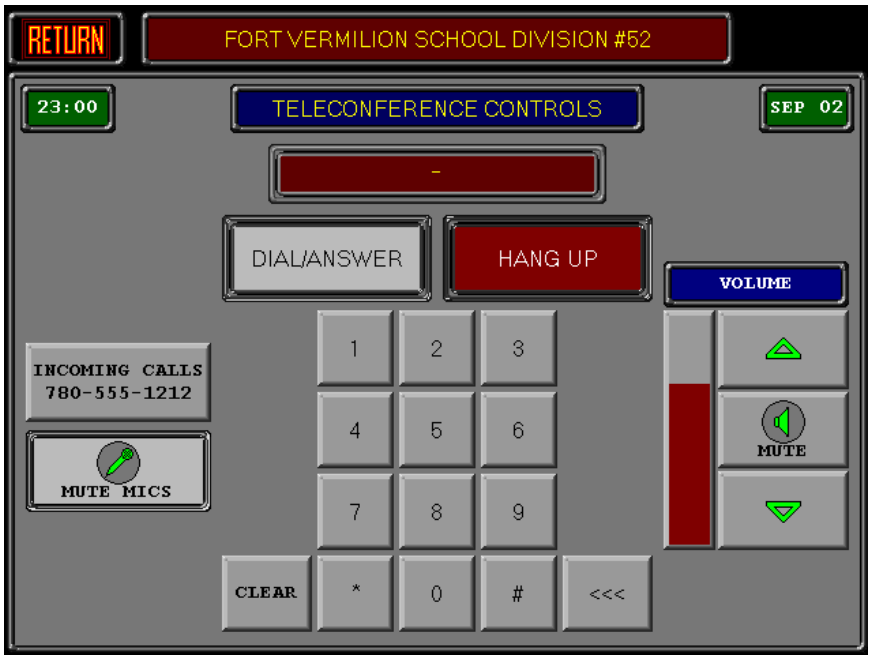

Fig 19. Teleconference

Fig 20. Videoconference

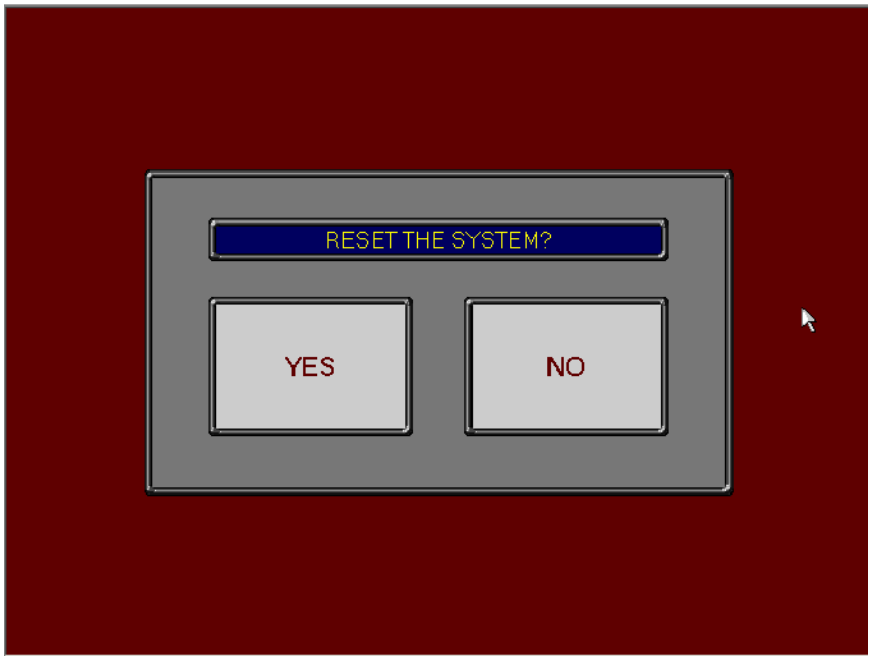

Fig 21. Reset

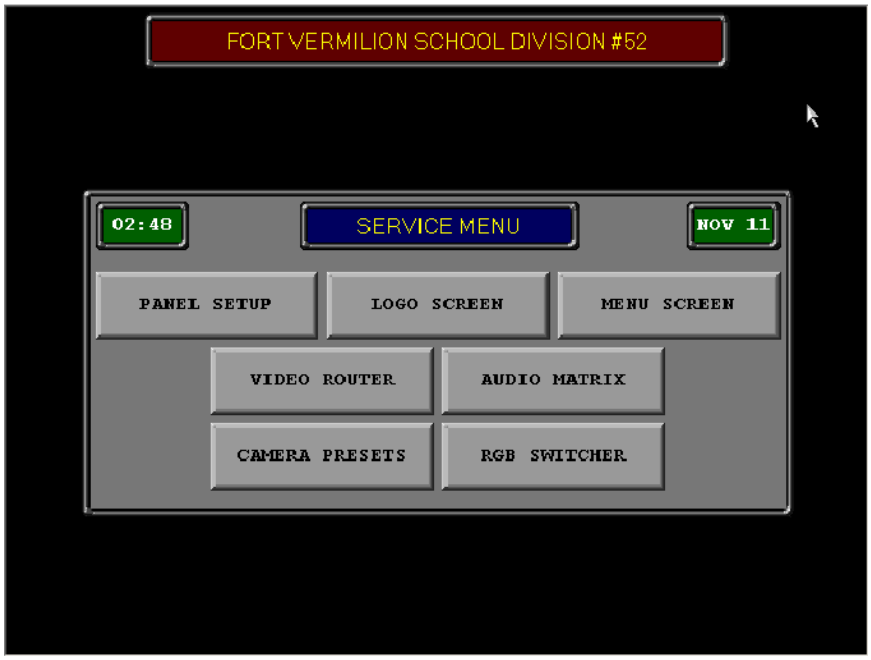

Fig 22. Service (techs only)

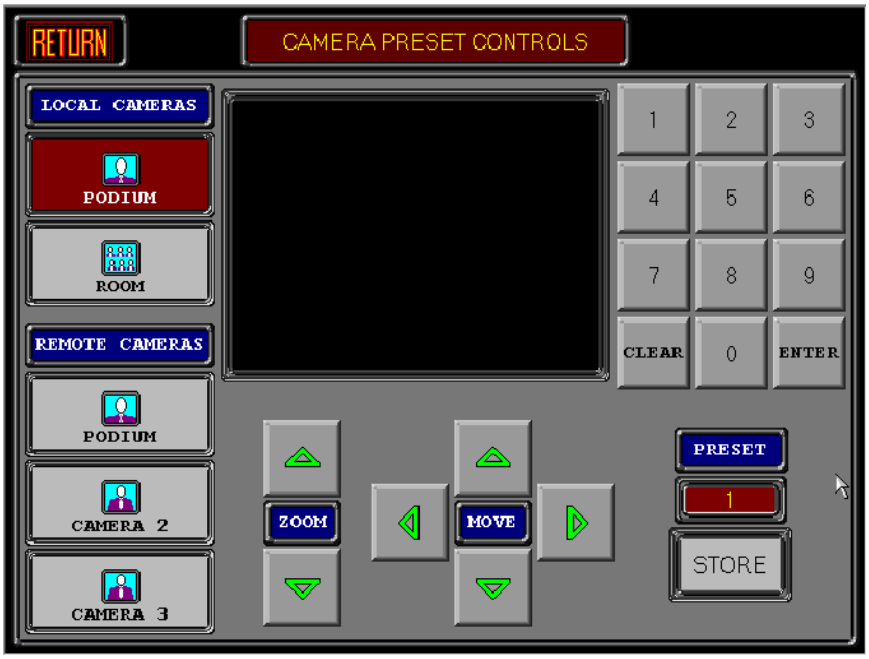

Fig 23. Camera Preset (techs only)

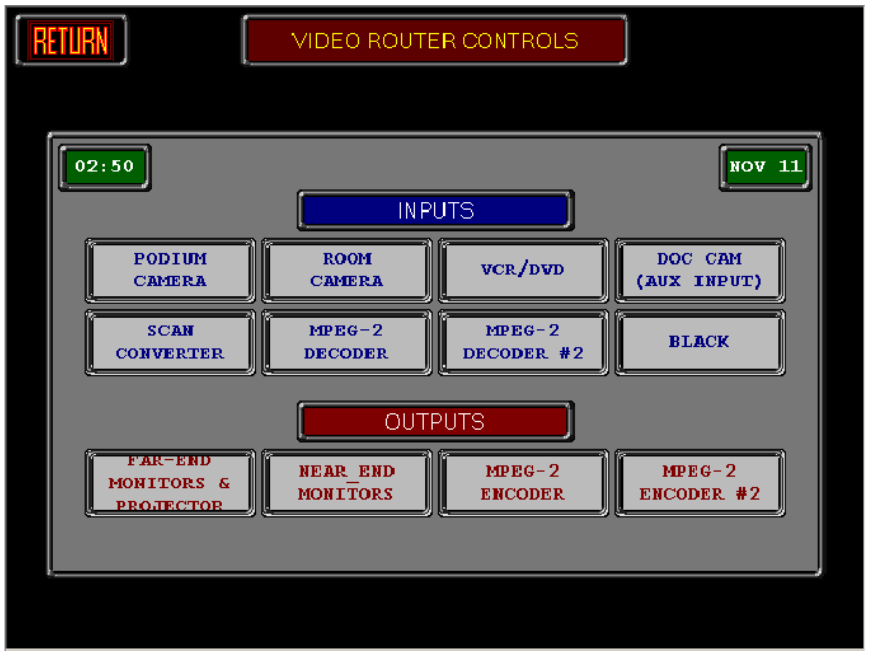

Fig 24. Video Router (techs only)

1.8 Appendix B -- Images from the RACOL classroom

Fig 25 – the POLYCOM main screen

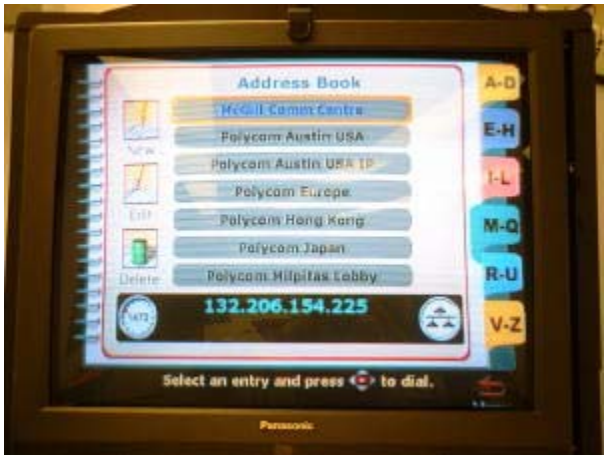

Fig 26 – the POLYCOM address book

Fig 27 – the layout of the screens within the classroom

1.9 Christopher Thompson's heuristic analysis

Throughout:

In every dialog, non-windows widgets are used. This fundamentally violates Microsoft's user-interface guidelines. More specifically, it is difficult to tell when something is a button or is a label. This violates rule #4 at least.

Most screenshots include the date and time. These are extraneous (violating rule 1) and inconsistent (not displayed in same location, not displayed at all in some screens) violating rule 4.

It is also unclear to me whether the 'Exit' button on most (but not all) screens exits back to the main menu or exits the application.

Screen 1:

No method of advancing from this screen is shown (or at least, clearly available). This screen may be displayed temporarily while the rest of the app loads. If so, this screen violates rule 5, provide feedback. A 'loading' progress bar would be good.

Screen 2:

This is not asking for a password, it is asking for a passcode. This violates rule 2. It does not provide a clearly marked exit, violating rule 6.

Screen 3:

A huge violation of rule 2. I don't even know what this screen means.Furthermore, the three select buttons seem to function like radio buttons. You select one and it becomes red, showing it is selected. If that is true, it is superfluous to display which mode is currently selected in the label immediately above the radio buttons. This therefore violates rule 1.

Screen 4:

Not even mentioning the lack of Windows-standard scroll widgets on this page.

Display is inconsistent (violates rule 4) as it displays a seeming list of schools along with a single codec. These seem to be two different categories.

Display includes superfluous information, violating rule 1. I don't need to know that 'High Level public school [has been] selected', for example, as it is highlighted below.

Screen 5:

All comments for screen 4 apply here.

Screen 6:

All comments for screen 4 apply here.

Screen 7:

It is unclear to me what the difference between 'multipoint' and 'broadcast' is. Presumably, therefore, this violates rule 2.

Screen 8:

It is unclear to me when the user should select 'receive broadcast'. It is likely this should be selected when a remote site is broadcasting at this terminal. If that is the case, it would perhaps be more efficient to prompt the user once a broadcast comes in.

Screen 9:

This screen does not say what units the time display is in, violating rule 1. There also seems to be no easy way to modify the hours column independently of the minutes column (if this is in fact what the columns are), violating rule 7.

Furthermore, it is unclear from this screen what the system timer actually does. Perhaps this times how long until your tea is brewed.

Furthermore, there's no easy way to return to the previous dialog, violating rule 6.

Screen 10:

As this is asking the user to select the source, it is unclear why two of the options are 'podium camera' and 'room camera' and yet at the bottom of the screen, there's another option entitled 'Cameras'.

Screen 11:

It is unclear the difference between 'presentation mode' and 'discussion mode'. The meaning of 'Questions' is unclear. The meaning of 'videoconf' is unclear. I am confused.

Screen 12:

This screen seems to be a duplicate of screen 9 but with a different label. This is obviously not consistent, violating rule 4. Additionally, my previous comments for screen 9 apply equally to this screen.

Screen 13:

'Pardon?' is a nonsense title for a widget. This should be reworded. It violates rule 1. It is unclear what 'reset queue' will do, violating rule 5.

Screen 14:

I am uncertain as to what has happened to bring this screen to this state.

Screen 15:

This screen seems to present a method to select the volume of multiple microphones simultaneously or individually. If this is really how this screen works, I have no significant complaints. If this is not the intention of this screen, it is violating rule 5.

Screen 16:

It is unclear why this screen is needed. If it is to allow a student to give a presentation, it should perhaps be displayed in response to a student attempting to give a presentation. At the moment, it seems possible to select a site not trying to give a presentation, violating rule 5.

It is also unclear to me why a school is selected when it would seem to make more sense to select a specific student. This may violate rule 2.

Screen 17:

I have no significant complaints about this screen.

Screen 18:

The selection of a 'remote site' seems to change the mode with respect to 'remote camera'. This could be considered a violation of rule 3 but is likely insignificant.

Screen 19:

It is unclear what the '<<<' button does. It is unclear what the difference between 'mute mics' and 'mute' is. These are violations of rule 2 and rule 1.

Screen 20:

This screen should not be titled 'H.323 Codec Controls'. It should instead be something meaningful to end-users. This violates rule 2. The rest of the screen is too confusing to comment on. Fundamentally, I have no idea what the result of my changes will be, violating rule 5 pretty badly.

Screen 21:

If this screen really causes a hard-reset of the operating system and all attached peripherals, I have no comments on this page. If not, it should be reworded so as to comply to rule 2.

Screen 22:

This screen is apparently only available to techs. If so, they will presumably know what the buttons all mean. No method of exiting from this screen seems to be available, violating rule 6.

Screen 23:

This seems to be a reasonable screen but it may make more sense to store presets using words rather than numbers, thus minimizing the user's memory load (rule 3).

Screen 24:

This screen is apparently only available to techs who would presumably understand the buttons.