

Virtual Classroom Tool

INITIAL DESIGN REPORT

CENG 491 Fall 2004

December 05, 2004

Mehmet Akif TAŞKIRMAZ
Serhat MEŞE
Abdullah ÖZTÜRK
Fırat KIZILER

Computer Engineering Department
Middle East Technical University

TABLE OF CONTENTS

1. INTRODUCTION	4
<i>Goals and Objectives</i>	4
<i>Project Scope</i>	4
<i>Major Constraints</i>	5
Performance/Behaviour Issues	5
Management and Technical Constraints	6
2. DATA DESIGN	7
<i>Internal Software and Data Structures</i>	7
<i>Global Data Structures</i>	9
<i>Temporary Data Structures</i>	9
<i>Database Description</i>	9
3. ARCHITECTURAL AND COMPONENT-LEVEL DESIGN	10
<i>Program Structure</i>	10
Architecture Diagram	10
Instructor State Transition Diagram	11
Student State Transition Diagram	12
Server Data Flow Diagram	13
Instructor Data Flow Diagram	14
Student Data Flow Diagram	15
Entity - Relation Diagram	16
<i>Component Descriptions</i>	17

<i>Software Interface Description</i>	32
4. USER INTERFACE DESIGN	33
<i>Description of the User Interface</i>	33
<i>Components Available</i>	36
5. RESTRICTIONS, LIMITATIONS AND CONSTRAINTS	37
<i>Performance/Behaviour Issues</i>	37
<i>Program Limitations</i>	37
6. TESTING ISSUES	38
<i>Classes of Tests</i>	38
Unit Testing	38
Integration Testing	38
High-Order Testing	38
<i>Expected Software Response</i>	38
General Hardware Requirements	38
<i>Identification Of Critical Components</i>	38
AudioVisual Streaming	38
Synchronization of Components	38
7. GLOSSARY	40
8. REFERENCES	45

1. INTRODUCTION

Goals and Objectives

The purpose of the project is to develop a virtual classroom tool (VCT) for synchronous distance learning with multimedia facilities. This distance learning takes place when a teacher and student(s) are separated by physical distance, and technology (i.e., voice, video, data, and print), often in concert with face-to-face communication. These types of programs can provide adults with a second chance at a college education, reach those disadvantaged by limited time, distance or physical disability, and update the knowledge base of workers at their places of employment [1; 2].

VCT will provide network broadcasting of a multimedia lecture given by a trainer to multiple clients, interaction with the students, preparation, online editing and offline replay of the courses.

Project Scope

In a physical classroom there is a standard set of audio-visual equipment and tools available to the instructor. These might include a chalkboard, overhead projector, video cassette player, possibly a sound system, and even the textbook. The virtual classroom has equivalent equipment and tools in the form of network-based software applications [3]. This equipment and the equivalent software applications in VCT are as follows:

- chalkboard: whiteboard
- pencil: multimedia pointer
- video cassette recorder: multimediaboard
- textbook: e-book
- sound and video conferencing: audiovisual system

The software will consist of a number of inputs and input files including the following:

- Video Files
- Sound Files
- MS Powerpoint Files
- Animation Files
- Bitmaps
- Camera
- Microphone

The software will consist of a number of outputs including the following:

- Audiovisual Streams
- Video Files
- Sound Files
- Text Files

- Database Files
- Snapshots

Major Constraints

Performance/Behaviour Issues

VCT requires the following software:

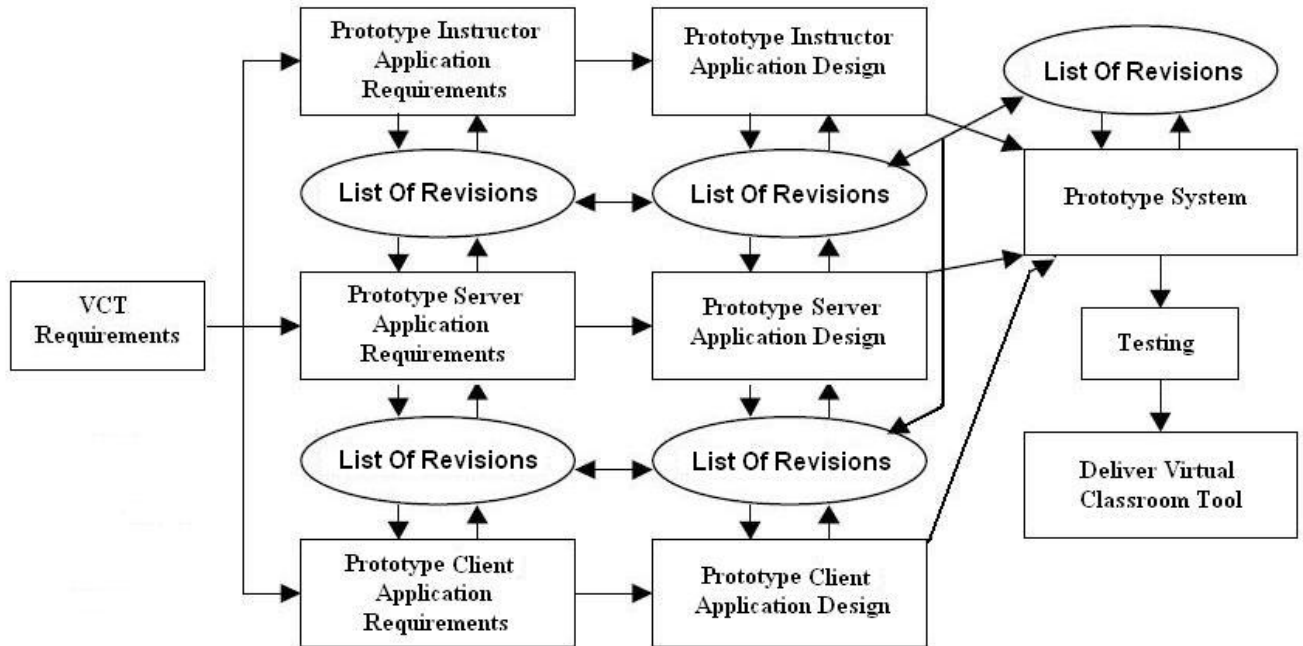
- Microsoft Windows Media SDK
- Microsoft Windows Media Server
- Microsoft Windows Media Encoder
- Microsoft Windows Media Player 9.0 or higher
- Microsoft SQL Server 7.0 or higher
- DirectX 8.1 or higher
- Microsoft Powerpoint 2000 or higher
- Microsoft Word 2000 or higher
- Microsoft Excel 2000 or higher
- Macromedia Flash Player

The server side of the VCT can only work on Microsoft Windows 2000/2003 server operating systems because Windows Media Server is distributed only with these operating systems.

The client side of the VCT is designed to be compatible with Microsoft Windows 2000/XP/2003 operating systems.

The reason for choosing DirectX 8.1 or higher is that the GDI+ Libraries (which is used for creating interfaces) requires at least DirectX 8.1.

Management and Technical Constraints



2. DATA DESIGN

Internal Software and Data Structures

multimediaboard

multimediaboard consists of the following attributes:

- **TopleftX:** The topleft x coordinate of the multimediaboard window
- **TopleftY:** The topleft y coordinate of the multimediaboard window
- **BottomRightX:** The bottomright x coordinate of the multimediaboard window
- **BottomRightY:** The bottomright y coordinate of the multimediaboard window
- **State:** Determine whether any file is shown or not

whiteboard

whiteboard consists of the following attributes:

- **TopleftX:** The topleft x coordinate of the whiteboard window
- **TopleftY:** The topleft y coordinate of the whiteboard window
- **BottomRightX:** The bottomright x coordinate of the whiteboard window
- **BottomRightY:** The bottomright y coordinate of the whiteboard window
- **State:** Determine whether anything is drawn or not

player

player consists of the following attributes:

- **TopleftX:** The topleft x coordinate of the player window
- **TopleftY:** The topleft y coordinate of the player window
- **VolumeChange:** The status of the volume_change button
- **State:** Determines the play state (playing, not playing)

audiovisual_data

audiovisual_data consists of the following attributes:

- **Alias:** The filename of the audiovisual_data
- **Path:** The directory that the audiovisual_data is found in
- **Buffer:** The buffer to load the audiovisual_data into
- **VolumeLevel:** Determines the volume level

help

help consists of the following attributes:

- **TopleftX:** The topleft x coordinate of the chat window

- **TopleftY:** The topleft y coordinate of the chat window
- **BottomRightX:** The bottomright x coordinate of the chat window
- **BottomRightY:** The bottomright y coordinate of the chat window
- **Topic:** The help topic that will be displayed
- **Explanation:** The explanation of the help topic
- **SearchText:** The string that is to be searched

chat

chat consists of the following attributes:

- **TopleftX:** The topleft x coordinate of the chat window
- **TopleftY:** The topleft y coordinate of the chat window
- **BottomRightX:** The bottomright x coordinate of the chat window
- **BottomRightY:** The bottomright y coordinate of the chat window
- **MessageType:** Determines the type of the message (question or chat message)
- **MessageText:** The message string
- **MessageLength:** The length of the message
- **MessageState:** Determines the state (send or get message)

online_users

online_users consists of the following attributes:

- **TopleftX:** The topleft x coordinate of the online_users window
- **TopleftY:** The topleft y coordinate of the online_users window
- **BottomRightX:** The bottomright x coordinate of the online_users window
- **BottomRightY:** The bottomright y coordinate of the online_users window
- **OnlineUsersArray:** The array that contains the usernames of the online users.

course_content

course_content consists of the following attributes:

- **TopleftX:** The topleft x coordinate of the course_content window
- **TopleftY:** The topleft y coordinate of the course_content window
- **BottomRightX:** The bottomright x coordinate of the course_content window
- **BottomRightY:** The bottomright y coordinate of the course_content window
- **Content:** The string that contains the contents.

Global Data Structures

object_handler

object_handler consists of the following attributes:

- **BroadcastControl:** State of the broadcast component
- **DatabaseControl:** State of the database component
- **QuestionControl:** State of the question component
- **HelpControl:** State of the help component
- **LoginManagerControl:** State of the login_manager component
- **ErrorControl:** State of the error component
- **ChatServerControl:** State of the chat_server component
- **CourseContentControl:** State of the course_content component
- **PlayerControl:** State of the player component

Temporary Data Structures

No temporary data structures are created.

Database Description

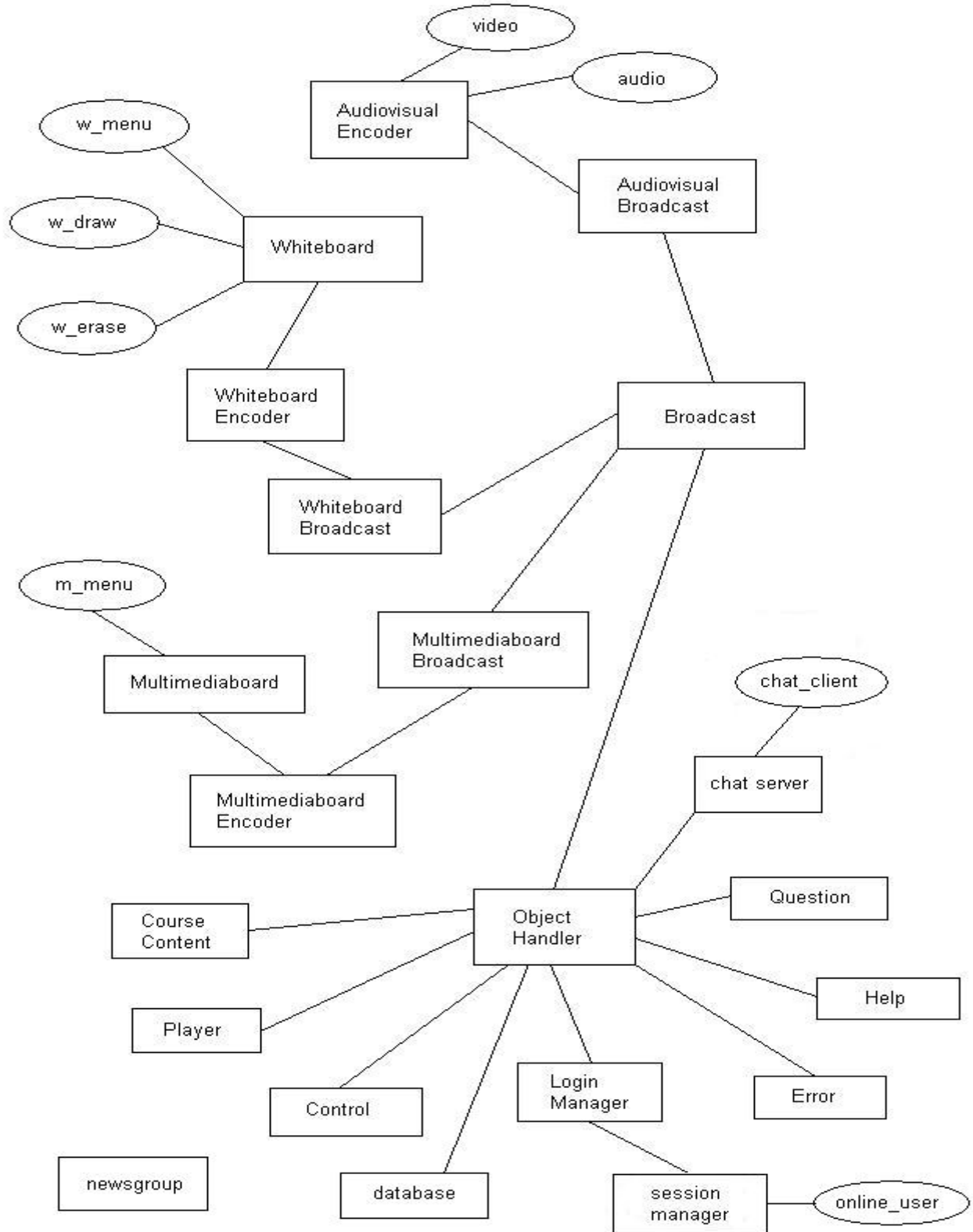
The database we are using is Microsoft SQL Server 7.0. It will hold the following information:

- Users account information
- Course information
- Help content
- Audiovisual content
- Multimedia content
- Chat messages

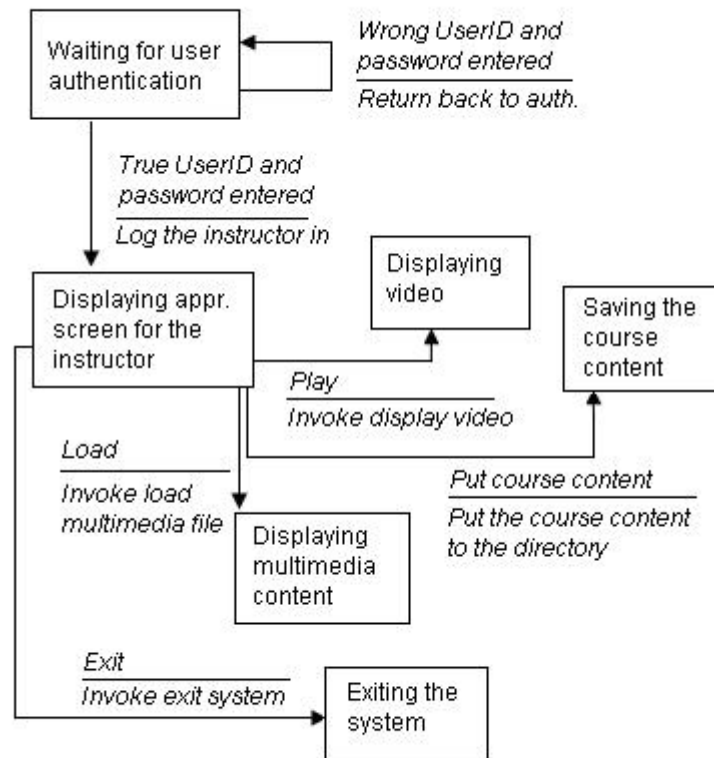
3. ARCHITECTURAL AND COMPONENT-LEVEL DESIGN

Program Structure

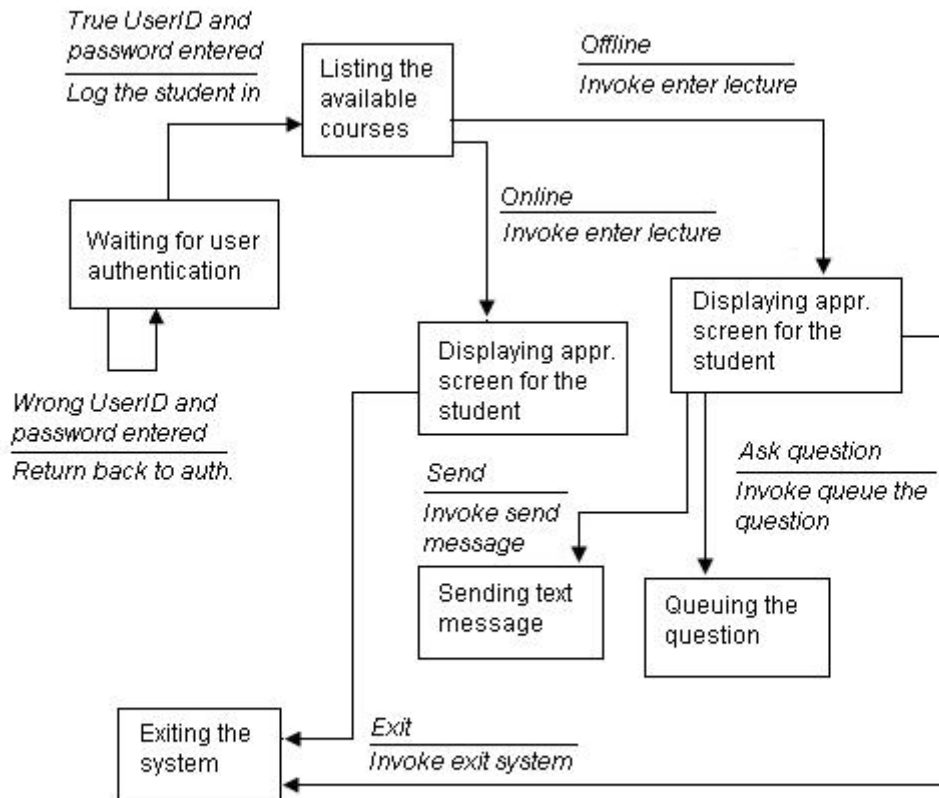
Architecture Diagram



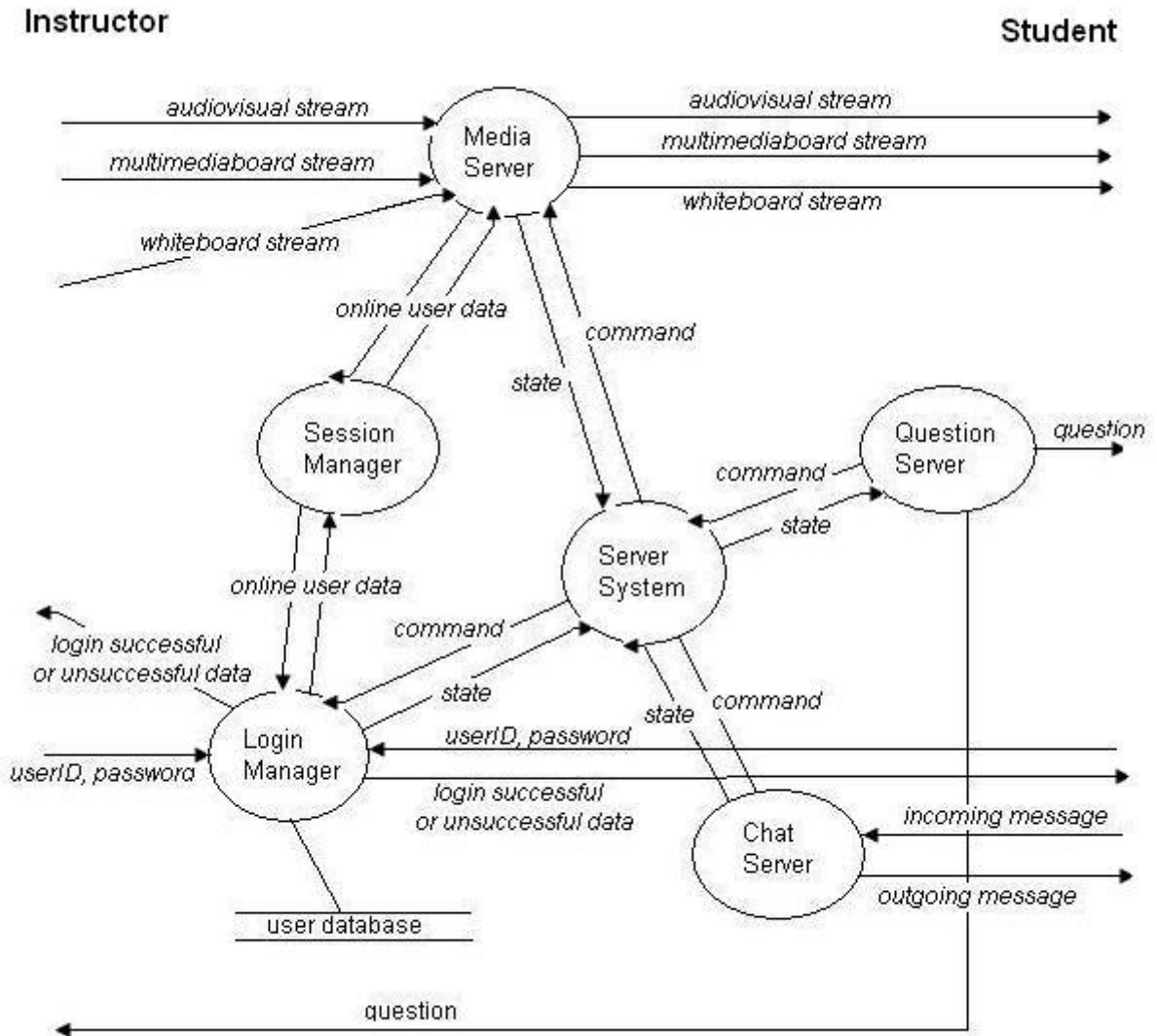
Instructor State Transition Diagram



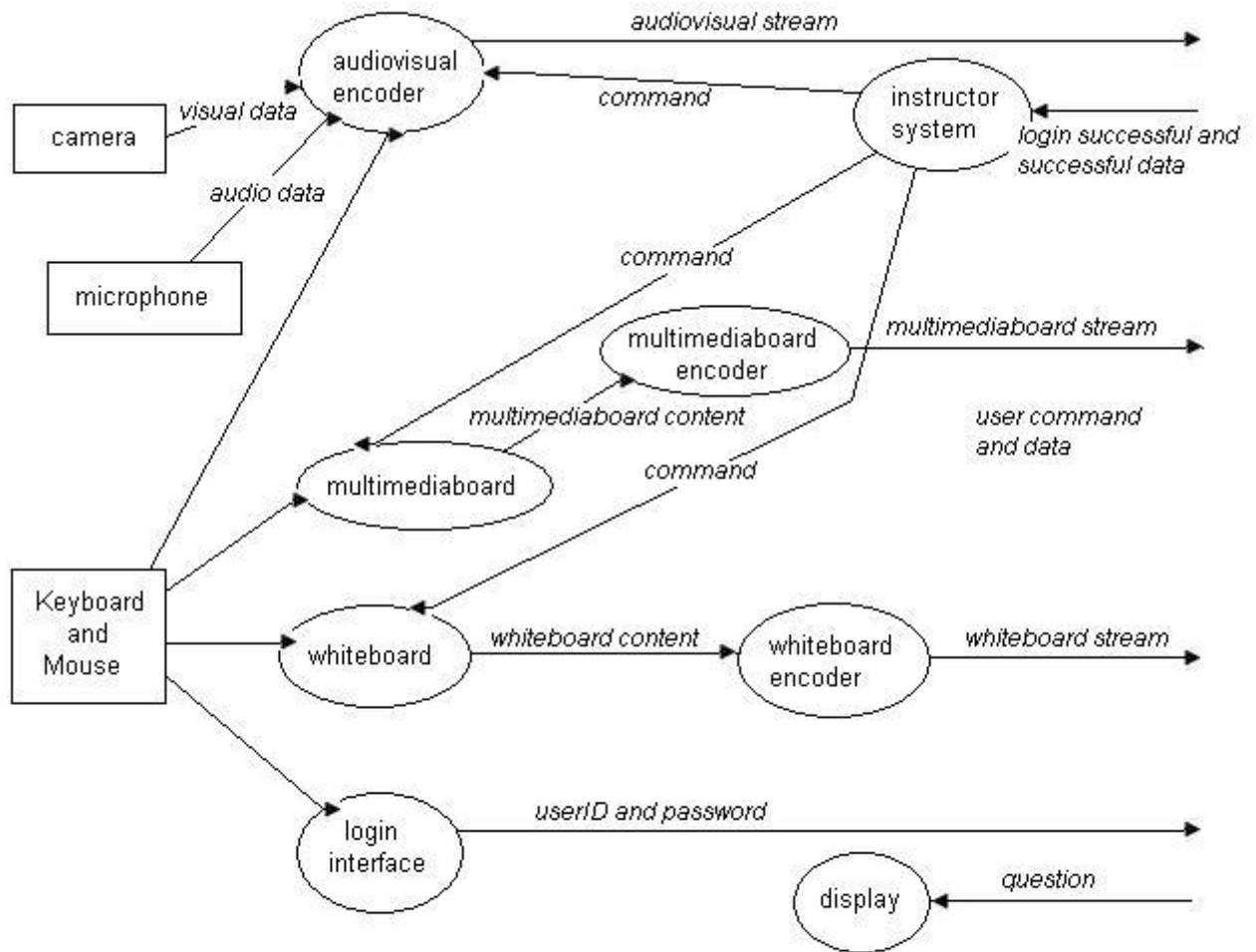
Student State Transition Diagram



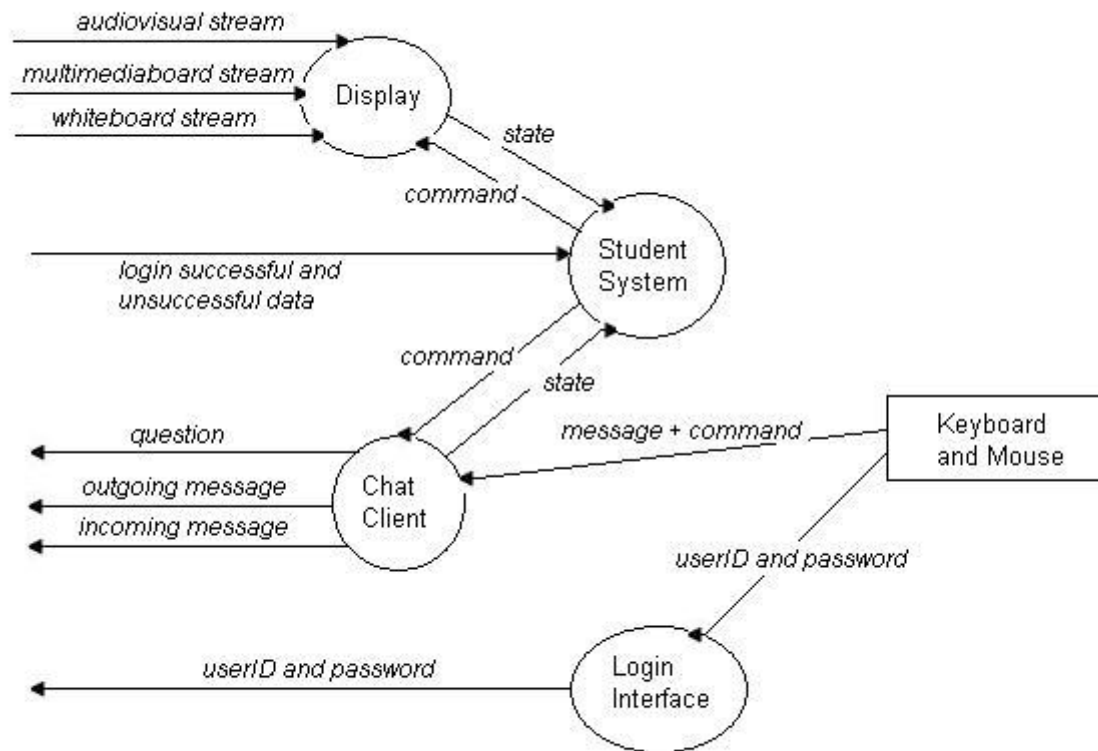
Server Data Flow Diagram



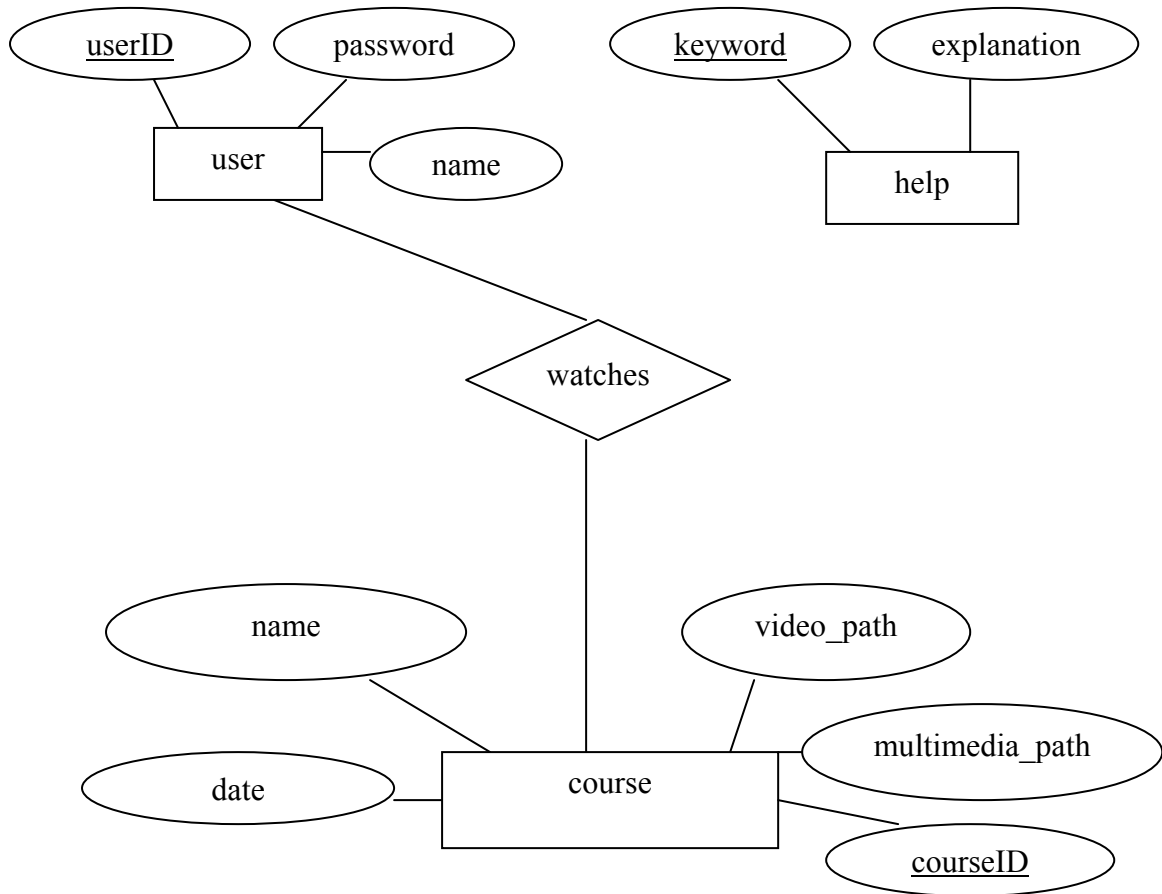
Instructor Data Flow Diagram



Student Data Flow Diagram



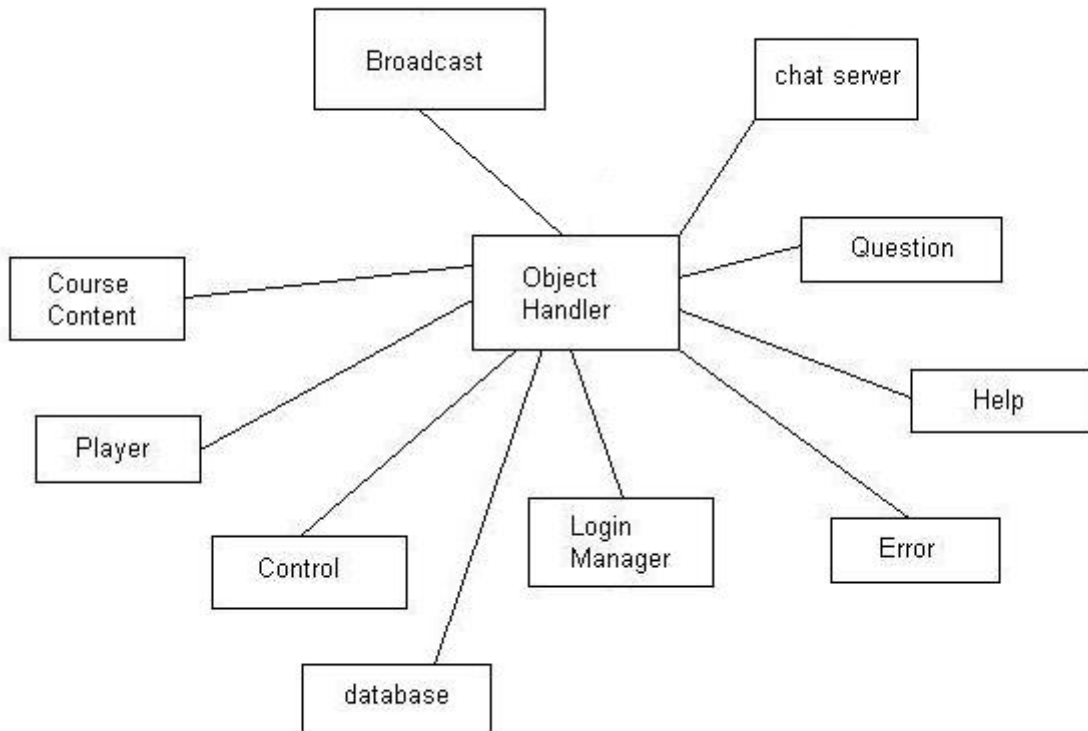
Entity-Relation Diagram



Component Descriptions

object_handler

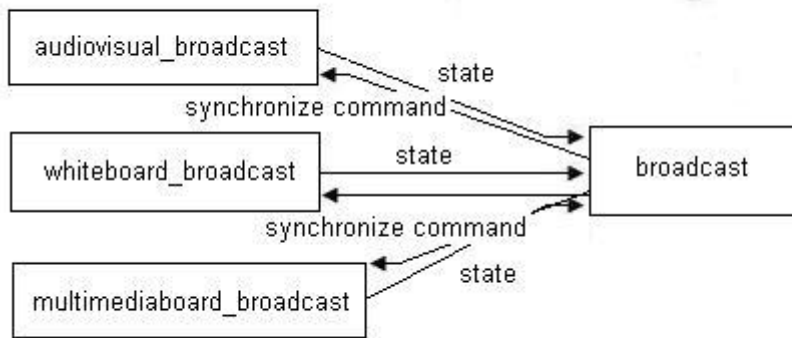
- **Narrative:** Receives the states of broadcast, chat_server, question, help, error, login_manager, control, course_content objects as inputs. It controls these components.
- **Diagram:**



- **Interface:** This component interfaces with the broadcast, chat_server, question, help, error, login_manager, control, course_content objects.
- **Functions:** GetState(), SendCommand()
- **Issues:** This is the most critical component. It controls all the components to work together.
- **Constraints:** None.

broadcast

- **Narrative:** Receives the states of the audiovisual_broadcast, multimediboard_broadcast and whiteboard_broadcast components. It controls (synchronizes) these objects.
- **Diagram:**



- **Interface:** This component interfaces with the audiovisual_broadcast, multimediaboard_broadcast and whiteboard_broadcast objects.
- **Functions:** GetState(), SendSynchronizeCommand()
- **Issues:** None.
- **Constraints:** None.

audiovisual_broadcast

- **Narrative:** Receives the audiovisual stream as the input from the audiovisual_encoder. It broadcasts this stream.
- **Diagram:**

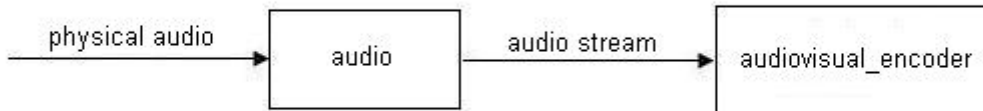


- **Interface:** This component interfaces with the player and audiovisual_encoder objects.
- **Functions:** BroadcastAVContStream(), GetAVContStream().
- **Issues:** None.
- **Constraints:** None.

audio

- **Narrative:** Receives physical audio. It converts the physical audio to digital audio and sends this digital audio data to the audiovisual encoder.

- **Diagram:**

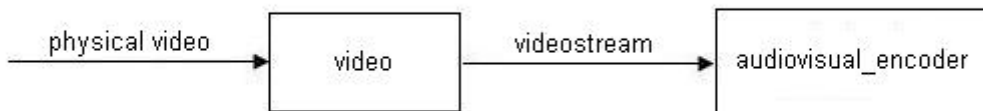


- **Interface:** This component interfaces with the audiovisual encoder.
- **Functions:** GetAudio(), SendAudio()
- **Issues:** None.
- **Constraints:** None.

video

- **Narrative:** Receives physical video. It converts the physical video to digital video and sends this digital video data to the audiovisual encoder.

- **Diagram:**

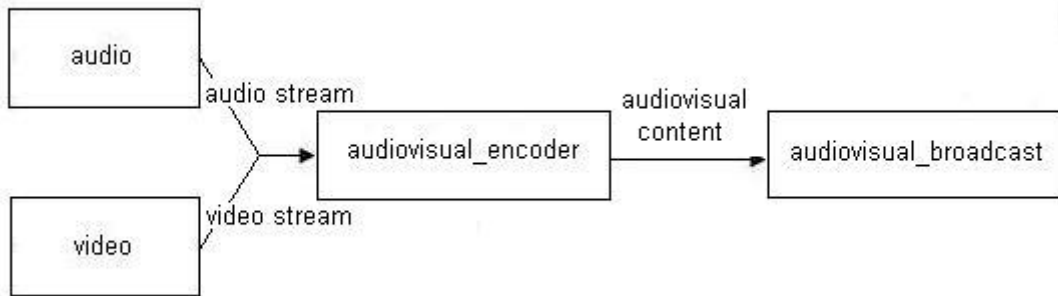


- **Interface:** This component interfaces with the audiovisual encoder.
- **Functions:** GetVideo(), SendVideo()
- **Issues:** None.

- **Constraints:** None.

audiovisual_encoder

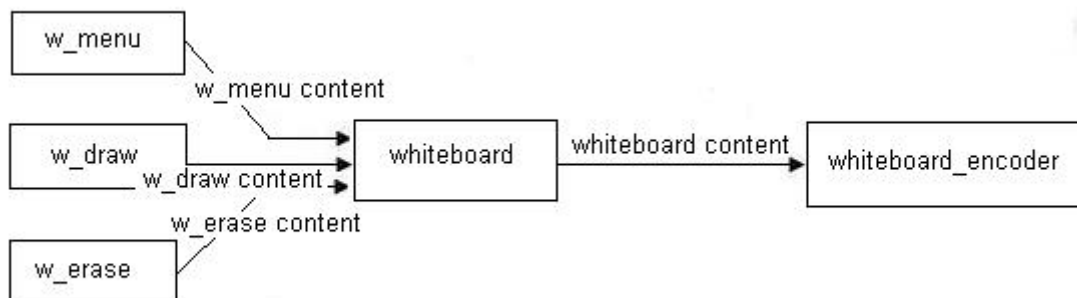
- **Narrative:** Receives digital audio and digital video from the audio and video objects. It combines these and sends the audiovisual stream to the audiovisual_broadcast object.
- **Diagram:**



- **Interface:** This component interfaces with the audio, video and audiovisual_broadcast objects.
- **Functions:** GetDigitalVideo(), GetDigitalAudio(), SendAudioVideo()
- **Issues:** The digital video files should be in the appropriate format (asf, avi, mpg and wmv). The digital audio files should be in the appropriate format (mp3, wav, wma).
- **Constraints:** None.

whiteboard

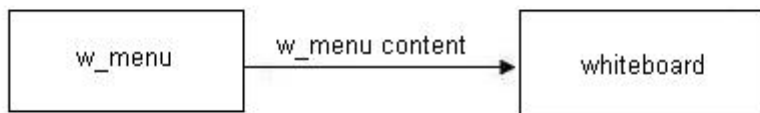
- **Narrative:** Receives w_menu, w_draw and w_erase contents as inputs. Its job is sending the whole whiteboard content to the whiteboard_encoder.
- **Diagram:**



- **Interface:** This component interfaces with the w_menu, w_draw, w_erase and whiteboard_encoder objects.
- **Functions:** GetWMenuContent(), GetWDrawContent(), GetWEraseContent(), SendWContent().
- **Issues:** None.
- **Constraints:** None.

w_menu

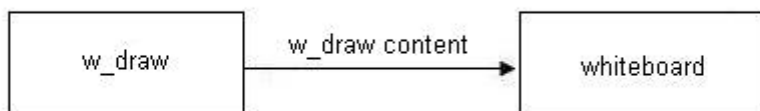
- **Narrative:** Receives no input. It performs the specified menu function.
- **Diagram:**



- **Interface:** This component interfaces with the whiteboard object.
- **Functions:** New(), Save(), Load(), SendWMenuContent().
- **Issues:** None.
- **Constraints:** None.

w_draw

- **Narrative:** Receives no input. It performs the drawing jobs.
- **Diagram:**

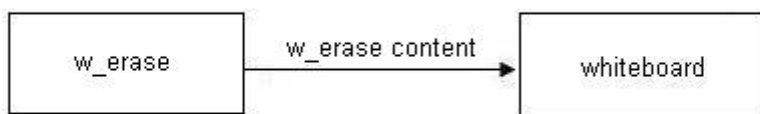


- **Interface:** This component interfaces with the whiteboard object.
- **Functions:** DrawLine(), DrawDot(), DrawCircle(), WriteText(), DrawPolygon(), SendWDrawContent().

- **Issues:** None.
- **Constraints:** None.

w_erase

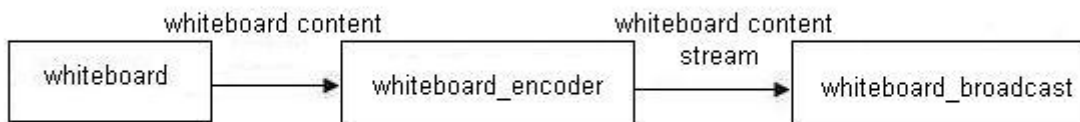
- **Narrative:** Receives no input. It erases the specified part(s) on the whiteboard.
- **Diagram:**



- **Interface:** This component interfaces with the whiteboard object.
- **Functions:** SendEraseContent()
- **Issues:** None.
- **Constraints:** None.

whiteboard_encoder

- **Narrative:** Receives the whiteboard content as the input. It converts the whiteboard content into stream and sends this stream to the whiteboard_broadcast object.
- **Diagram:**



- **Interface:** This component interfaces with the whiteboard and whiteboard_broadcast objects.

- **Functions:** GetWContent(), SendWContentStream().
- **Issues:** None.
- **Constraints:** None.

whiteboard_broadcast

- **Narrative:** Receives the whiteboard content stream as the input. It broadcasts this stream.
- **Diagram:**



- **Interface:** This component interfaces with the player and whiteboard_encoder objects.
- **Functions:** BroadcastWContentStream(), GetWContentStream().
- **Issues:** None.
- **Constraints:** None.

multimediaboard

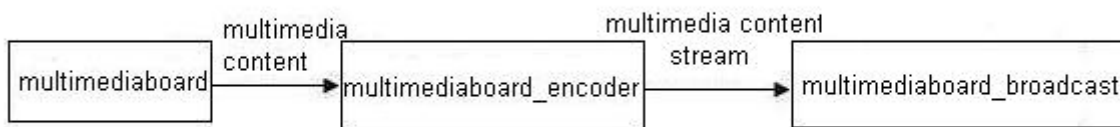
- **Narrative:** Receives multimediaboard content and m_menu content as inputs. Its job is sending the whole multimediaboard content to the multimediaboard_encoder.
- **Diagram:**



- **Interface:** This component interfaces with the m_menu and multimediboard_encoder objects.
- **Functions:** GetMContent(), GetMMenuContent(), SendMContent().
- **Issues:** None.
- **Constraints:** None.

multimediboard_encoder

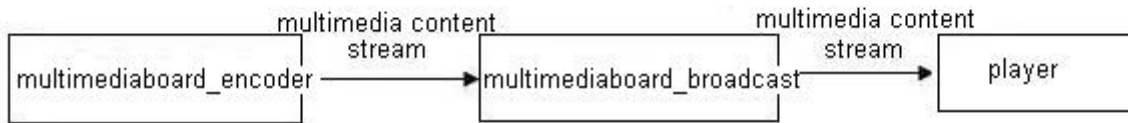
- **Narrative:** Receives the multimediboard content as the input. It converts the multimediboard content into stream and sends this stream to the multimediboard_broadcast object.
- **Diagram:**



- **Interface:** This component interfaces with the multimediboard and multimediboard_broadcast objects.
 - **Functions:** GetMContent(), SendMContentStream().
 - **Issues:** None.
- Constraints:** None.

multimediboard_broadcast

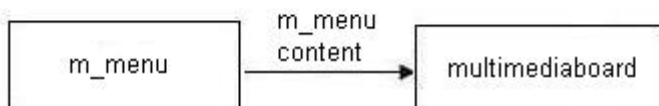
- **Narrative:** Receives the multimediboard content stream as the input. It broadcasts this stream.
- **Diagram:**



- **Interface:** This component interfaces with the player and multimediaboard_encoder objects.
- **Functions:** BroadcastMContentStream(), GetMContentStream().
- **Issues:** None.
- **Constraints:** None.

m_menu

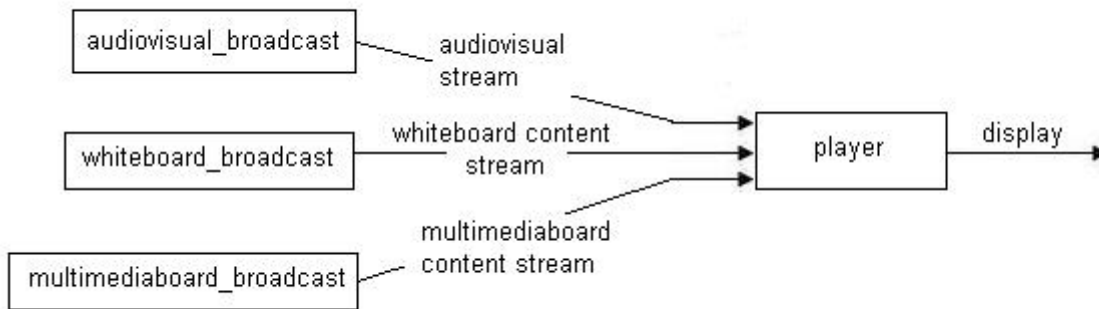
- **Narrative:** Receives no input. It performs the specified menu function.
- **Diagram:**



- **Interface:** This component interfaces with the multimediaboard object.
- **Functions:** Open(), SendMMenuContent().
- **Issues:** None.
- **Constraints:** None.

player

- **Narrative:** Receives the audiovisual stream, whiteboard content stream and multimedia content stream. It displays these streams on the screen.
- **Diagram:**



- **Interface:** This component interfaces with the audiovisual_broadcast, whiteboard_broadcast and multimedia_broadcast.
- **Functions:** Display(), Play(), Stop(), SetVolumeLevel()
- **Issues:** The stream format will be in wmv format.
- **Constraints:** If the stream is corrupted, the player can not display it correctly.

chat_server

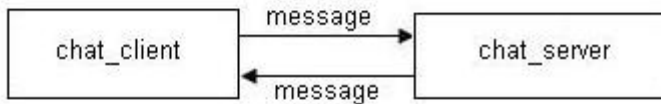
- **Narrative:** Receives messages as inputs. It transmits the messages between the chat clients.
- **Diagram:**



- **Interface:** This component interfaces with the chat_client objects.
- **Functions:** ServerGetMessage(), ServerSendMessage()
- **Issues:** None.
- **Constraints:** None.

chat_client

- **Narrative:** Receives messages as inputs. It receives and send messages to the chat server.
- **Diagram:**



- **Interface:** This component interfaces with the chat_server object.
- **Functions:** ClientGetMessage(), ClientSendMessage()
- **Issues:** None.
- **Constraints:** None.

online_users

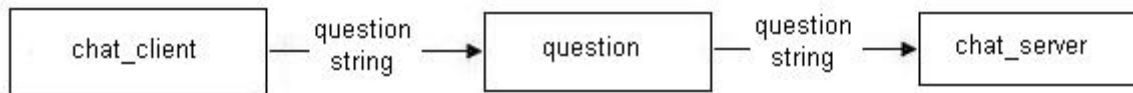
- **Narrative:** Receives the online users data from the session_manger object as the input. It displays the clients who are online.
- **Diagram:**



- **Interface:** This component interfaces with the session_manager object.
- **Functions:** DisplayOnlineUsers()
- **Issues:** None.
- **Constraints:** None.

question

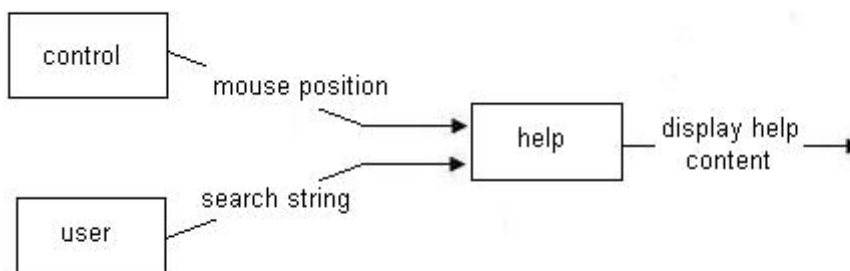
- **Narrative:** Receives the question string from the chat_client object as the input. It sends the question string to the chat_server object.
- **Diagram:**



- **Interface:** This component interfaces with the chat_server and chat_client objects.
- **Functions:** GetQuestion(), SendQuestion()
- **Issues:** None.
- **Constraints:** None.

help

- **Narrative:** Receives the mouse position from the control component or the search string from the user as the inputs. It searches the database and displays the help content dynamically.
- **Diagram:**

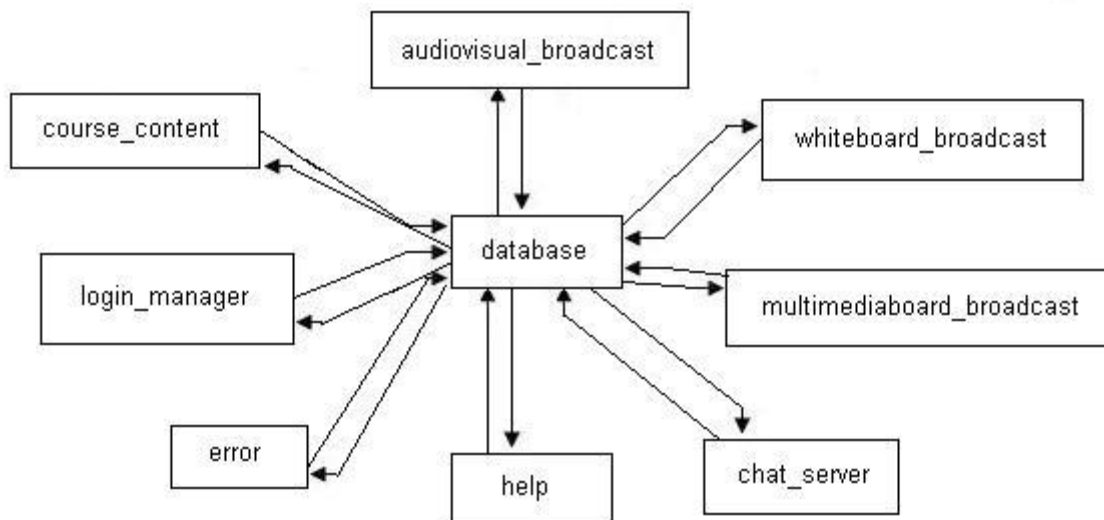


- **Interface:** This component interfaces with the database and control components.
- **Functions:** ConnectDatabaseServer(), Search(), DisplayHelpContent()
- **Issues:** The help content for all Mouse positions must be defined correctly, otherwise, the help will not work properly.

- **Constraints:** If the search string is not found in the database, nothing will be displayed.

database

- **Narrative:** Receives the requests and sends the required contents.
- **Diagram:**



- **Interface:** This component interfaces with the login_manager, error, help, chat_server, multimediaboard_broadcast, whiteboard_broadcast, audiovisual_broadcast and course_content.
- **Functions:** GetRequst(), SendContent()
- **Issues:** If the connection fails, the required content can not be obtained.
- **Constraints:** None.

error

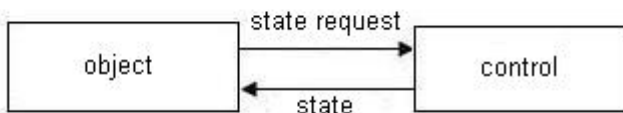
- **Narrative:** Receives exceptions as the input. It handles these.
- **Diagram:**



- **Interface:** This component interfaces with all of the components.
- **Functions:** GetException(), HandleException()
- **Issues:** If the errors are not handled, the program may crash.
- **Constraints:** All the error types should be handled correctly.

control

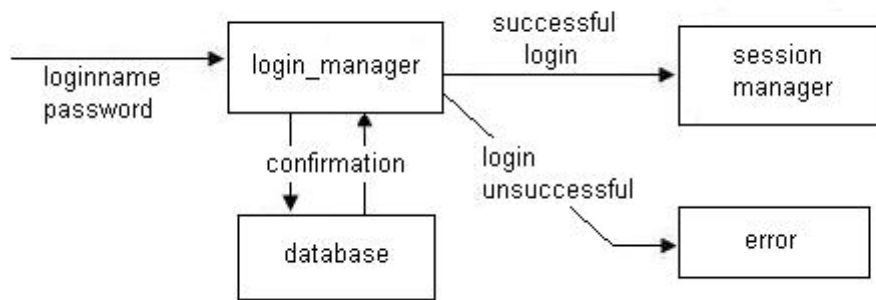
- **Narrative:** Receives state requests from the objects as input. It holds the mouse position and buttons' states and sends these data to those objects.
- **Diagram:**



- **Interface:** This component interfaces with all of the components.
- **Functions:** GetRequest(), SendState()
- **Issues:** None.
- **Constraints:** None.

login_manager

- **Narrative:** Receives the login name and password from the user. It gives authentication or not. It sends the successful logins to the session_manager object.
- **Diagram:**



- **Interface:** This component interfaces with the database, session_manager and error objects.
- **Functions:** ConnectDatabaseServer(), ConfirmPassword()
- **Issues:** If the login is unsuccessful, there will be error.
- **Constraints:** If the user is already online, then the login manager prevents logging in with the same userID and password.

session_manager

- **Narrative:** Receives the successful logins from the login manager as the input. It sends the online users data to the online_user object.
- **Diagram:**



- **Interface:** This component interfaces with the login_manager and online_users object.
- **Functions:** GetOnlineUsers(), SendOnlineUsers()
- **Issues:** None.
- **Constraints:** None.

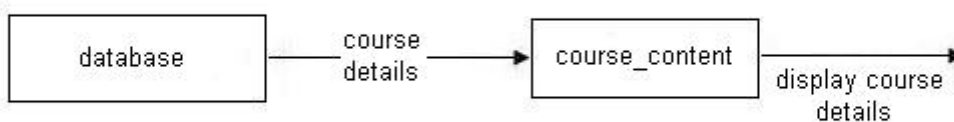
newsgroup

- **Narrative:** Receives no input. This is the newsgroup on the web.

- **Diagram:** None.
- **Interface:** None (It Works independently).
- **Functions:** None.
- **Issues:** None.
- **Constraints:** None.

course_content

- **Narrative:** Receives the course details from the database as input. It displays the course content.
- **Diagram:**



- **Interface:** This component interfaces with the database component.
- **Functions:** ConnectDatabaseServer(), DisplayCourseDetails()
- **Issues:** None.
- **Constraints:** None.

Software Interface Description

The interface that we have chosen to use will be designed in Microsoft Visual C# .Net. The interface will utilize many of the common controls included in Visual C# .Net, and the majority of windows-based applications. The interface will be a graphical user interface that provides the instructor to give lectures efficiently and that provides the user to follow the lectures easily. For a more detailed description of the interface, please refer to the User Interface Design section found below.

4. USER INTERFACE DESIGN

Description of the User Interface

There are two types of user interfaces: instructor interface and student interface. These interfaces show all the contents (player, multimediasboard, whiteboard, etc) simultaneously. In the instructor interface broadcasting process is started by using the “broadcast” command in the main menu or context menu. There is a question queue window in this interface for intrustor to see the questions asked by the students. In the student interface, all the content sent by the multimedia server is displayed to the student. The student can chat with the other students or ask questions using the chat window. Moreover, there is a help window in both of the interfaces.

Login Interface

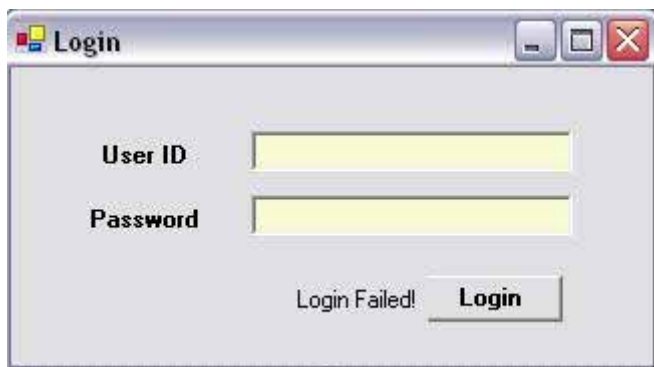
The users (instructor and the students) log in the system by entering their userID and password in the login window.



A screenshot of a Windows-style login window titled "Login". It features two input fields: "User ID" containing the text "e1234567" and "Password" containing a series of asterisks. A "Login" button is positioned below the fields.

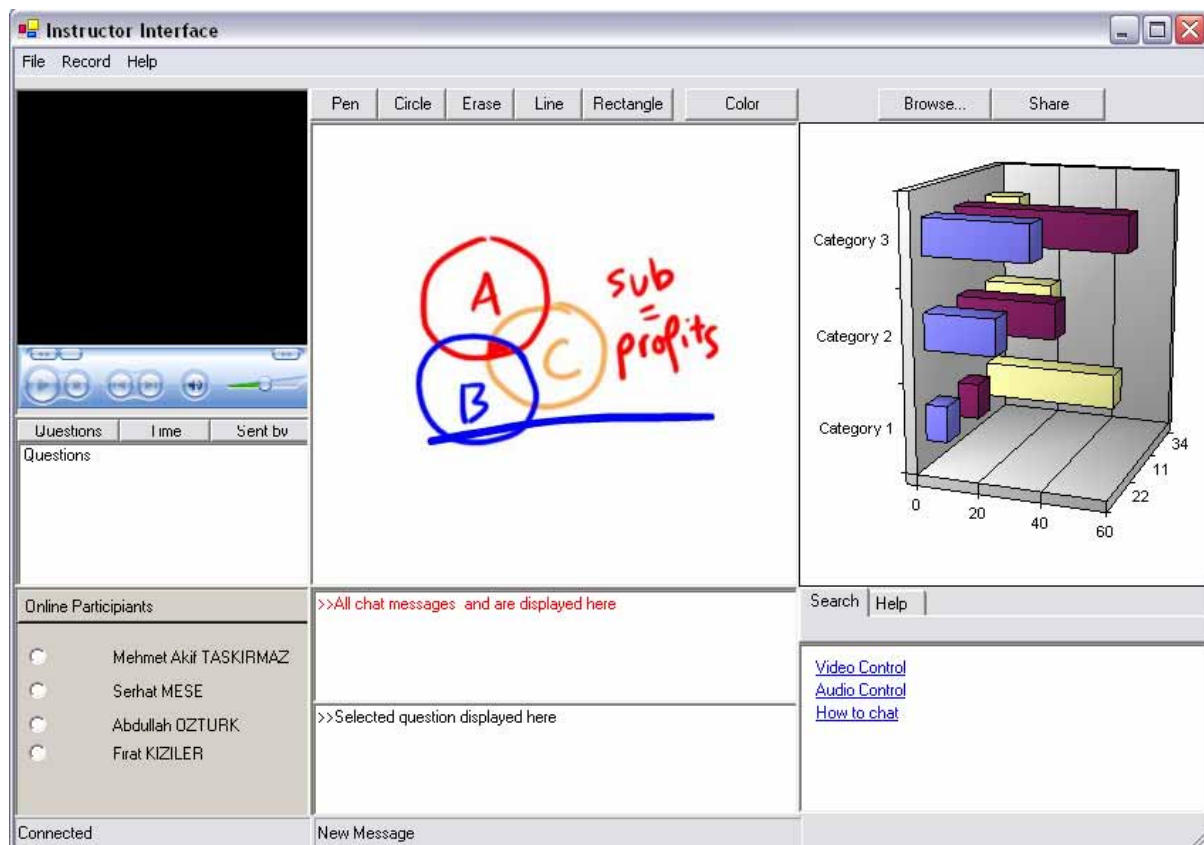
Login Failed Interface

If login process fails, the login failed interface appears. The reason (wrong userID or password, the user is already online, database connection failure, etc.) for the failure is displayed in the interface.



A screenshot of the same "Login" window, but with the "User ID" and "Password" fields empty. The "Login" button is now disabled and has the text "Login Failed!" displayed next to it.

The Instructor Interface



Player

Displays the audiovisual streams coming from the media server. The instructor sees his/her image in this part.

Multimediboard Window

The instructor uses this part to show and send the multimedia content to the students.

Whiteboard Window

The instructor uses this part as the chalkboard in a real classroom [3].

Chat Window

The instructor uses this part to see the questions asked by the students and all the public chat messages.

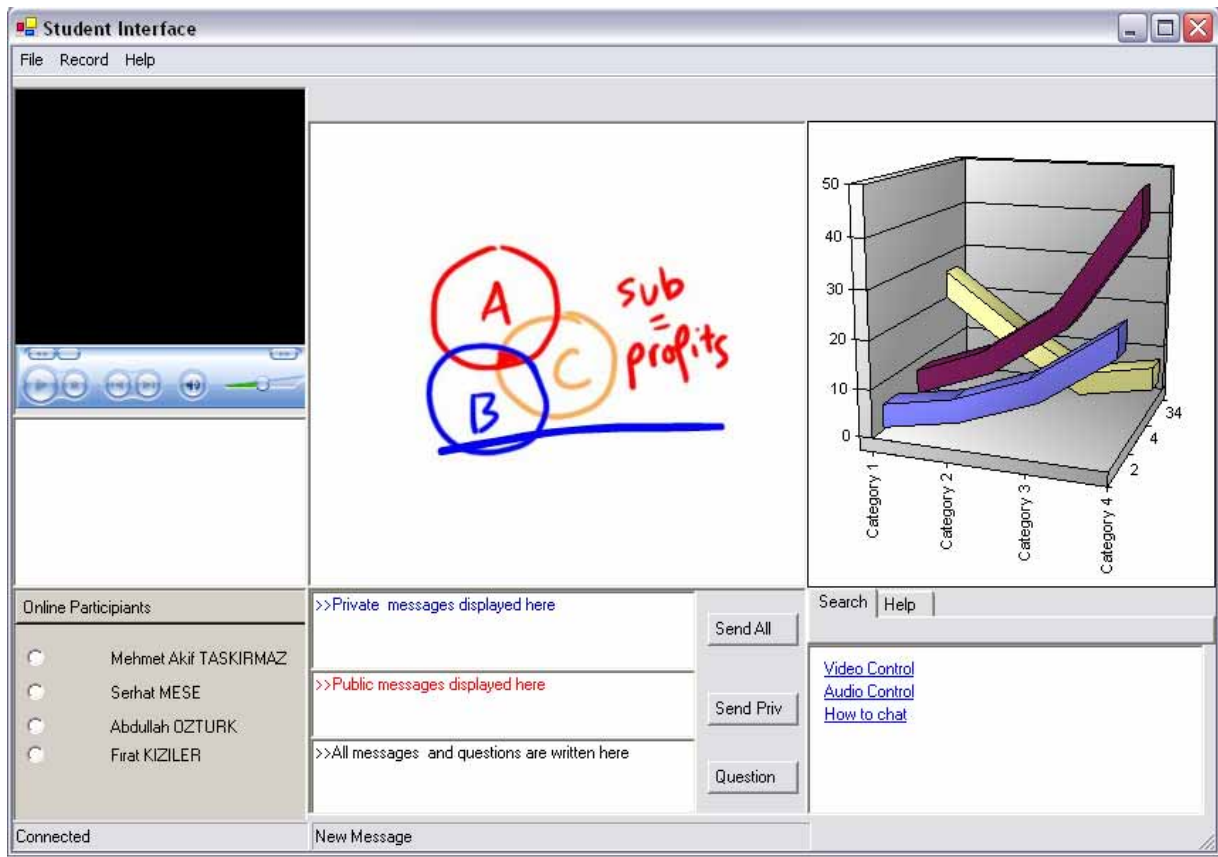
Online Users Window

The instructor uses this part to see the online users.

Help Window

The instructor uses this part for getting help about the usage of the application. It consists of two parts: dynamic help and search.

The Student Interface



Player

Displays the audiovisual streams coming from the media server.

Multimediboard Window

Displays the multimediboard streams coming from the media server.

Whiteboard Window

Displays the whiteboard streams coming from the media server.

Chat Window

Used for chatting and asking questions.

Online Users Window

Used to see the online users.

Help Window

Used for getting help about the usage of the application. It consists of two parts: dynamic help and search.

Components Available

Microsoft Visual C#.Net is a programming language that allows the user to create complex applications for Windows, without all of the overhead required using other languages. It allows the user to pick from a list of practically thousands of controls, and draw them on the screen. These controls then have certain events, methods, and properties that can be set. When a particular event fires, the code associated with that event is executed. See the MSDN Library for Microsoft C#.Net for a complete list of all components available, including each component's properties, methods and events.

5. RESTRICTIONS, LIMITATIONS AND CONSTRAINTS

Performance/Behaviour Issues

The server side of the VCT can only work on Microsoft Windows 2000/2003 server operating systems because Windows Media Server is distributed only with these operating systems.

The client side of the VCT is designed to be compatible with Microsoft Windows 2000/XP/2003 operating systems (earlier versions will not be supported).

The reason for choosing DirectX 8.1 or higher is that the GDI+ Libraries (which is used for creating interfaces) requires at least DirectX 8.1.

Program Limitations

- VCT works in only full-screen mode at 16-bit or higher color depth.
- The resolution of the player is 232 by 200.
- All the streams are in wmv format.
- The students do not have permissions to change the contents of the whiteboard, multimediaboard and the player.
- The user can not connect again immediately. Connections are refreshed every five minutes.
- VCT works on LAN.

6. TESTING ISSUES

Classes of Tests

Unit Testing

Individual components (database, error, help, ...) will be tested separately. All components can be tested through the object handler. All unit testing will be done in Whit Box fashion.

Integration Testing

Combined components will be tested as a whole. To maintain maximum control over the testing criteria, all data files will be made specifically for testing purposes.

High-Order Testing

The High-Order testing will be performed on the complete, integrated system.

Expected Software Response

General Hardware Requirements

- Processor: PIII 733 MHz
- System Memory: 256 MB
- Video Memory: 32 MB

Identification Of Critical Components

AudioVisual Streaming

The audiovisual data should be transmitted continuously without errors. Windows Media Server performs this operation.

Synchronization of Components

The synchronization between the three main components (player, whiteboard and multimediaboard) should be provided. This is the most critical part in this project. The broadcast object performs this task.

The most critical parts of VCT implementation is to provide synchronization between audio and video streams and to supply multimedia content, which includes content of whiteboard and multimedia-board, to students exactly as produced on the instructor side.

Unpredictable network errors while transmissions of audio and video data packets lead to latency between audio and video while processing data packets on students side. For real-time streaming data, using

strategy which will ignore loss of some data packets, especially for audio and video, will not affect the performance of audio-visual processing on the student side. This obstacle will be overcome by using Windows Media Server.

On the other hand, transmission of multimedia content requires an approach will provide almost-lossless transmissions of data packets over network. Namely, retransmission of data packets which were not received by the student-side is necessary. This does not consume bandwidth as much as retransmission of large video frames over network.

7. GLOSSARY

American Standard Code for Information Interexchange (ASCII)	A computer language used to convert letters, numbers, and control codes into a digital code understood by most computers.
Application Service Provider (ASP)	A specialized form of an Internet service provider (ISP) that allows a company to have a software application hosted via a rental fee. An ASP sells access to a “packaged application” on a fee basis. ASPs provide IT operations expertise (offering the necessary application functionality, hardware, database and networking services, etc.) and frequently also business operation expertise in a particular market niche or in a particular functional area (such as HR or logistics management).
Asynchronous	Communication in which interaction between parties does not take place simultaneously.
Asynchronous Transfer Mode (ATM)	A high bandwidth, low delay, packet-like switching and multi-plexing technique.
Audio Bridge	A device used in audioconferencing that connects multiple telephone lines.
Audioconferencing	Voice only connection of more than two sites using standard telephone lines.
Backbone	A primary communication path connecting multiple users.
Band	A range of frequencies.
Bandwidth	Information carrying capacity of a communication channel.
Browser	Software that allows you to find and see information on the Internet.
Codec (COder/DECoder)	Device used to convert analog signals to digital signals for transmission and reconvert signals upon reception at the remote site while allowing for the signal to be compressed for less expensive transmission.
Collaborative Tools	Allow learners to work with others via e-mail,

threaded discussions or chat. In some cases, collaboration is used to facilitate team-based projects. Collaborative tools can sometimes provide the ability to host moderated discussion groups, where students and instructors can collaborate on course-related materials or assignments in an asynchronous environment. In addition, real-time synchronous chat allows learners to communicate with their peers and instructors, emulating a physical classroom setting.

Compressed Video

Television signals transmitted with much less than the usual bit rate. The lower bit rates typically involve some compromise in picture quality, particularly when there is rapid motion on the screen.

Computer-Based Training (CBT)

Course or educational material presented on a computer, primarily via CDROM or floppy disk. Unlike Web-based training, computer-based training does not require a computer connected to a network and does typically not provide links to learning resources outside of the course.

Desktop Videoconferencing

Videoconferencing on a personal computer.

Dial-Up Teleconference

Using public telephone lines for communication links among various locations.

Distance Education

The process of providing instruction when students and instructors are separated by physical distance, and technology, often in tandem with face-toface communication, is used to bridge the gap.

Distance Learning

The desired outcome of distance education.

Echo Cancellation

The process of eliminating the acoustic echo in a videoconferencing room.

E-Learning/Technology-Based Learning

Covers a wide set of applications and processes such as Web-based learning, computer-based learning, virtual classrooms, and digital collaboration. It includes the delivery of content via Internet, intranet/extranet (LAN/WAN), audio/video tape, satellite broadcast, interactive TV, and CD-ROM.

Fiber Optic Cable

Glass fiber that is used for laser transmission of video, audio, and/or data.

File Transfer Protocol

A protocol that allows you to move files from a

(FTP)	distant computer to a local computer using a network like the Internet.
Full Motion Video	Signal which allows transmission of complete action taking place at the origination site.
Fully Interactive Video	(Two way interactive video) Two sites interact with audio and video as if they were colocated.
Host	A network computer that can receive information from other computers.
Instructional Television Fixed Service (ITFS)	Microwave-based, high-frequency television used in educational program delivery.
Integrated Services Digital Network (ISDN)	A telecommunications standard allowing communications channels to carry voice, video, and data simultaneously.
Interactive Media	Frequency assignment that allows for a two-way interaction or exchange of information.
Internet-Based Training/Web-Based Training (WBT)/Online Training	Delivery of educational content via a web browser over the public Internet, a private intranet, or an extranet (LAN/WAN). Internet-based training provides links to learning resources outside of the course, such as references, e-mail, bulletin boards, and discussion groups. It provides the advantages of computer-based training (CBT) while retaining advantages of instructor-led training. The term Internet-based training is used synonymously with Web-based training and online training.
Internet Protocol (IP):	The international standard for addressing and sending data via the Internet.
Local Area Network (LAN)	Two or more local computers that are physically connected.
Learning Management System (LMS):	Internet-based software that deploys, manages, tracks and reports on interaction between a) the learner and the content, and b) the learner and the instructor. In particular, training management systems perform student registration, track learner progress, record test scores, and indicate course completions, and finally allow instructors/trainers to assess the performance of their students. Learning management systems administer and track both online and classroom-based learning events, as well as other training processes.

Learning Portal	A Web site that offers learners or organizations consolidated access to learning and training resources from multiple sources. Learning portals can be grouped into content consolidation portals, embedded technology portals, internal portals, community & collaboration portals, and affiliation portals. Operators of learning portals are also called content aggregators, distributors or hosts.
Learning Service Provider (LSP)	An LSP is a specialized type of ASP offering learning management and training delivery software on a hosted/rental basis via diverse business models. There are four different types of LSPs: 1) full service LSPs (customizing, implementing, and hosting a complete software solution via a private network); 2) content specific LSPs (licensing content to an organization and providing a level of learning management services to the buyer); 3) tool specific LSPs (licensing and hosting their specific system to an organization); and 4) portal LSPs (hosting a portal site and bundle the learning system in the background). LSPs also include value-added resellers (VAR) and companies providing certification and testing services, online collaboration services, media production and delivery services, and online tutoring.
Multimedia	Any document which uses multiple forms of communication, such as text, audio, and/or video.
Multi-Point Control Unit (MCU)	Computerized switching system which allows point-to-multipoint videoconferencing.
Network	A series of points connected by communication channels in different locations.
Protocol	A formal set of standards, rules, or formats for exchanging data that assures uniformity between computers and applications.
Server	A computer with a special service function on a network, generally receiving and connecting incoming information traffic.
Slow Scan Converter	Transmitter/receiver of still video over narrow band channels. In real time, camera subjects must remain still for highest resolution.

Synchronous	Communication in which interaction between participants is simultaneous.
Teleconferencing	Two way electronic communication between two or more groups in separate locations via audio, video, and/or computer systems.
Technology-Based Training	Includes the delivery of content via Internet, intranet/extranet (LAN/WAN), satellite broadcast, audio/video tape, interactive TV, and CD-ROM. Technology-based training includes computer-based training (CBT) and Web-based training (WBT).
Training Management Systems	See Learning Management Systems.
Transmission Control Protocol (TCP)	A protocol which makes sure that packets of data are shipped and received in the intended order.
Video Teleconferencing	A teleconference including two way video.
Web-Based Training (WBT)	See Internet-Based Training

Note

The following resources were reviewed and consulted in the preparation of this publication:

- Glossary:
<http://152.30.11.86/DEER/Houghton/Committees/distancelearn/GlossaryDistEd.html>
- Glossary of Terms. <http://www.ctcnet.com/tips/glossary.htm>
- Newton, H. (1991). Newton's telecom dictionary.
- Telecom Library Inc: New York.
- Reed, J. (1996). Videoconferencing for learning glossary.
<http://www.kn.pacbell.com/wired/vidconf/glossary.html>.
- The EdWeb Dictionary. <http://k12.cnidr.org:90/dic.html>
- Willis, B. (Ed.) (1994). Distance education: Strategies and tools. Educational Technology Publications, Inc.: Englewood Cliffs, N. J.

8. REFERENCES

1. Moore, M.G. & Thompson, M.M., with Quigley, A.B., Clark, G.C., & Goff, G.G. (1990). *The effects of distance learning: A summary of the literature. Research Monograph No. 2.* University Park, PA: The Pennsylvania State University, American Center for the Study of Distance Education. (ED 330 321)
2. Verduin, J.R. & Clark, T.A. (1991). *Distance education: The foundations of effective practice.* San Francisco, CA: Jossey-Bass Publishers.
3. <http://www.tc.cornell.edu/~dwyer/>