DISTANT ELECTRONIC CLASSROOM

UTILIZING LOW BANDWIDTH INTERNET

By
Irfan Hameed

Submitted to Faculty of Computer Science
National University of Science and Technology Rawalpindi,
In partial fulfillment for the requirements of an M.S. degree
April 2002

Acknowledgements

I wish to thank Almighty Allah Who enabled me to complete this thesis work. I also express my deepest appreciation to all those whose sincere help and encouragement made the present work a reality.

I gratefully acknowledge the help and guidance provided by my thesis advisor Dr. Romana Aziz. Her essential supervision, advice and valuable guidance, proved an asset in the completion of this thesis work. I am deeply indebted to her for her encouragement and continued help during execution of this thesis work.

My very special thanks are extended to the members of my guidance committee namely Air Cdre Waqar Sami Khan, Lt. Col. Muhammad Nadeem and Mr. Rizwan Ahmad Khan for their very useful suggestions and critical reviews that assisted me in widening the perspective of this thesis.

I deeply treasure the Parental support and forbearance that I received from my family.

I would also like to extend my thanks to all faculty members of the Department of Computer Science for their cooperation and healthy suggestions throughout my academic development at Military College of Signals Rawalpindi, a prestigious NUST campus.

Abstract

The Internet is fast becoming the vehicle of choice for disseminating learning across distances. The main aim of this thesis work is to achieve a realization of Classroom Environment, by combining asynchronous and synchronous teaching-learning contents and activities, through the Internet. The thesis work is divided into two main parts. First part is development of Lecture Preparation Application for the teacher. It makes it easy to divide complete Lecture into Sections and allows rapid integration of Text, Pictures and Sound in required format. It provides facilities of importing Text and Figures through Scanner and allows formatting of Lecture content. Provision for recording and compressing Sound narration for each Screen is available. The teacher can easily preview and navigate through complete Lecture during its development.

Second part is the development of Server Side Software that provides basic logging facilities to the teachers and the students of different Subjects. The teacher can easily upload all Lecture content without any need of manually selecting the files. Activation of Lecture offers different Modes of Classroom, thus it is possible to deliver prepared Lecture content with or without Live Query and Response with Blackboard content, with or without Live Audio and Video facility. It also provides student's status monitoring. After live class is over, the teacher has the facility to combine prepared Lecture material with interactive Live Class's content, thus producing a new reference material. Server Side Software manages attendance record of the students and provides browsing of Lecture material with merged Queries, after Live Class Sessions.

Proper use of Object Oriented technology requires the development of Analysis and Design specifications using some CASE tool. Therefore "Rational Rose" is used for Visual Modeling of our Software. Programming is done for this thesis based Project by writing code as well as assembling and integrating existing

and custom developed software components. Our UML model of Software will make it easy to maintain the complete system.

Distant Electronic Classroom software has its direct application in Distance Education programs. The complete Software Implementation is designed in such a way that during its maintenance phase a few updates will be required to take advantage of increased bandwidth availability, in Pakistan, in future. This realization effectively eliminates many weak and negative aspects of a Distant Education institution, especially a Virtual University.

Contents

Chapter 1

Concept of Distant Electronic Classroom

Chapter 2
Lecture Preparation Application

Chapter 3

Analysis and Design of Lecture Preparation Application

Chapter 4

Server Side Software

Chapter 5

Analysis and Design of Server Side Software

Chapter 6
Overview of Thesis: Methodology Applied and The Project

List of Figures

Chapter 1

Concept of Distant Electronic Classroom

1.1 Introduction

The Internet has quickened the way we communicate. Access to Internet is becoming more widespread, and its capabilities of delivering multimedia are improving almost daily. With such credentials the Internet is fast becoming the vehicle of choice for disseminating learning across distances. This newfound medium of instruction will enhance the existing Distance Education capabilities by making radical changes and a total reshaping of the process.

Our aim is to achieve a realization of classroom environment utilizing Low Bandwidth Internet that is widely available now. It will combine some asynchronous as well as synchronous contents and activities. This realization will add a new dimension to Distance Learning by seamlessly integrating almost every activity, and much of the teacher-learner interaction that occurs in a typical University Lecture, by using different streams of content presenting an easily accessible interface.

If all of the Lecture materials are distributed to the students, it might seem that there is a reduction in the need to take notes on the part of the students and to further process the contents. The attribute of copy-ability

may actually reduce interactivity and lead the student to disengage from attending to and processing the Lecture because they know that the notes have been copied to their workspace and that they will be available at a later time.

Similar needs and aspirations will be experienced by the students and the teachers as in the formal system. The same interpersonal needs and skills must be understood and supported in the emerging environment. The students will continue to learn and the teachers will continue to teach. Teaching- learning is an eternal human activity. A few things cannot be replaced by technology; nor would one want to. We just want to fill a significant need for those who are in remote locations.

At present some work has been done in Online Universities, which covers how to setup a basic University infrastructure with different departments. Part of these setups involves Lectures for the students, which is usually provided in the form of prepared downloadable slides. But as educationists say a book can’t replace a teacher, therefore there is need to integrate asynchronous and synchronous contents. This in simple terms means combining prepared Lecture contents, illustrations etc. with live interactivity. It is here that the need for this project originates.[28,29]

Many educational institutions around the world, including Pakistan where Distance Education is already practiced are in direct need of such

Distance Education learning facility. Therefore it is really a need-based project. It facilitates learning by the students at their locations while providing to them *True Interactive Class Environment*.

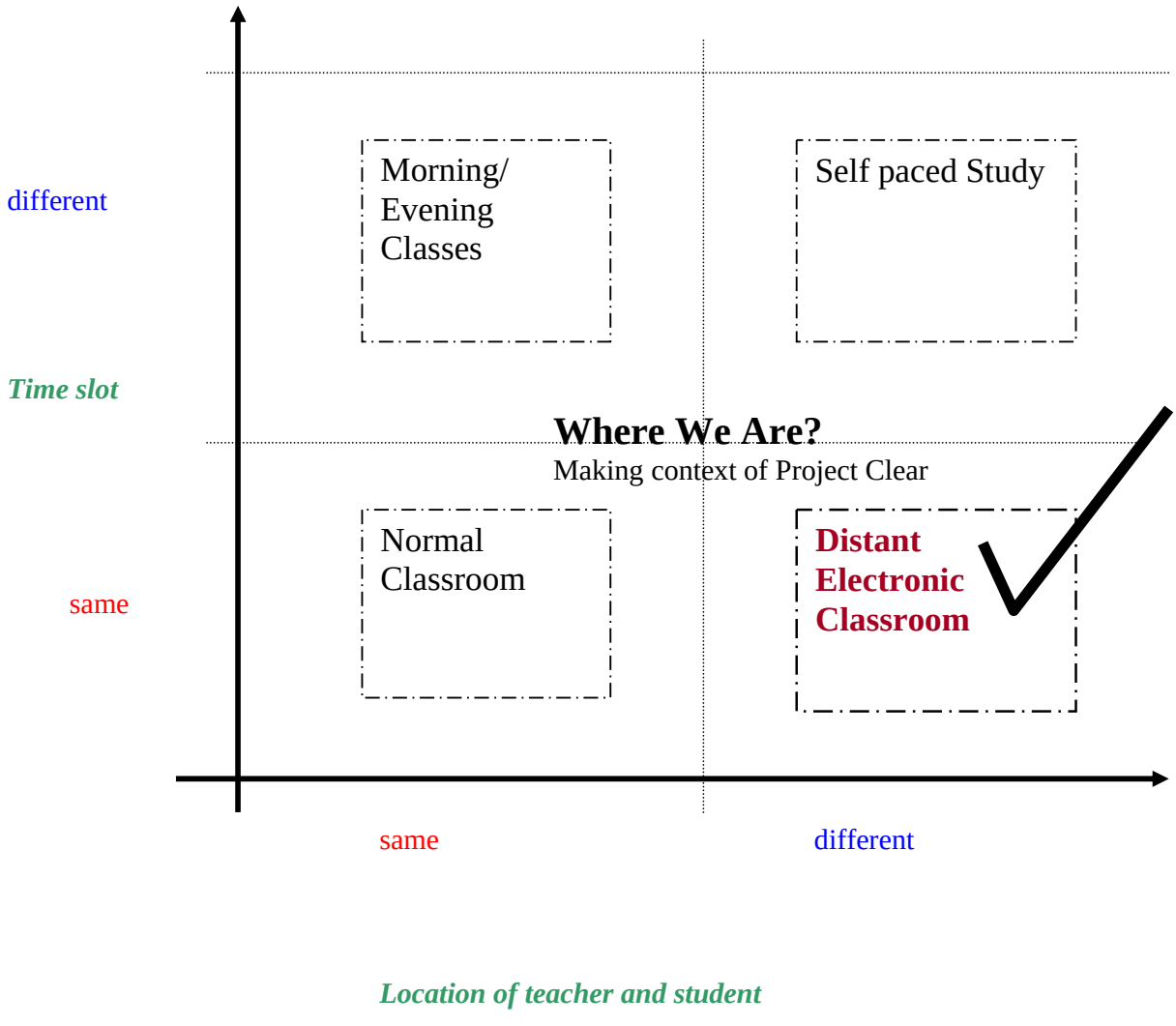

Fig 1.1: Illustration of different possibilities of educational system.

1.2 Project Specifications

1.2.1 Software for Lecture Content Development

- A software that will enable the teacher to rapidly integrate Text, Pictures and Sound in required format for this system, by providing a 'Template'.
- Template will divide Lecture into Sections, for display on Browser Screens.
- Each Section may include one or more Text, Pictures and Sound.
- Provision for recording Sound by the teacher.
- Either use scanned Text and Pictures, or use from a saved file.
- Apply formatting to above content.
- Convert all material to specific form that will be used by The University Server during Live Class Session.
- This software using a user-friendly interface will provide all above functionality.

1.2.2 Powerful Interaction

- To increase student-to-teacher interaction and collaboration.

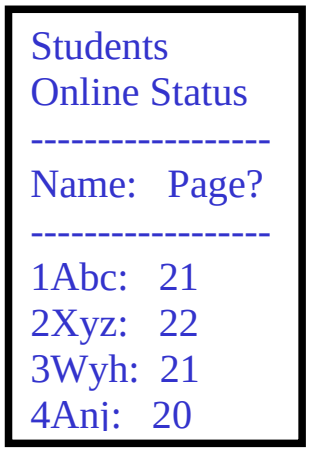

Fig 1.2: Students Online

– It does not have the same disruptive effect as on the class nor does it draw attention to the one asking the question.

– Interaction will keep the student's attention focused on the Lecture rather than on unrelated thoughts.

– The teacher knows how many and which students are lagging.

1.2.3 Automatic Preparation of Excellent Future Reference

– It will combine prepared Lecture contents with dynamic real time contents, without any need of post editing labor neither on the part of the teacher nor the student.

– After Live Class Session, interactive Queries and Responses will be embedded into proper location in the Lecture material.

– It can be printed and saved for future reference.

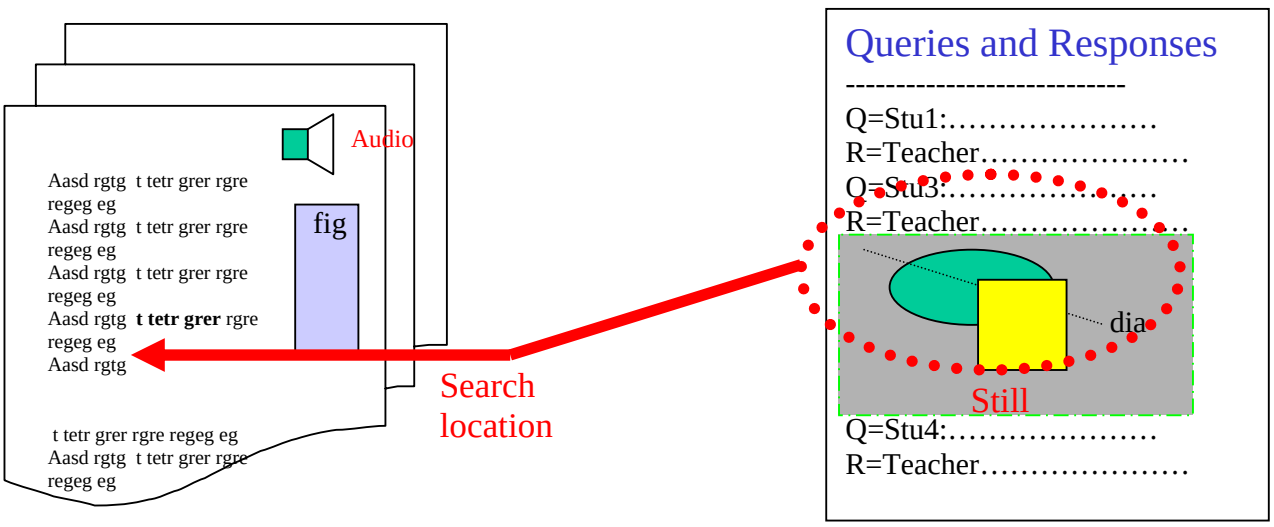

Fig 1.3: Merging Queries into Lecture material.

1.2.4 Live Video Frames

The students will ask Queries and the teacher's Response will be either textual or involve working out problems on the Blackboard. It is most effective and helpful method of teaching allowing the teacher to show and explain each step along the way. [25]

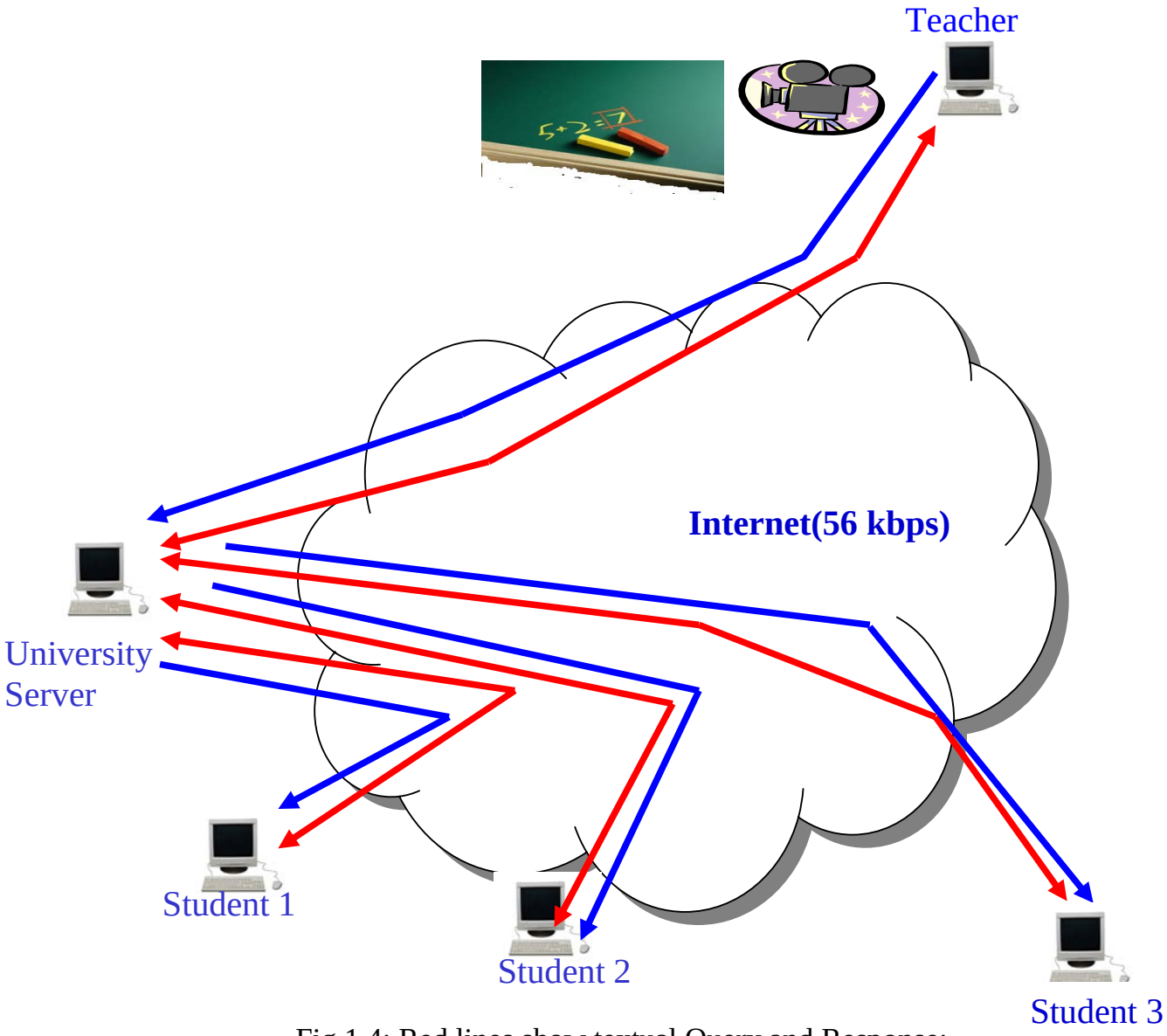

Fig 1.4: Red lines show textual Query and Response:
blue lines show Board Images

1.2.5 Ease of Receive ability and Accessibility

- Using simple Browser on the client side.
- No special software is required by the student.
- The student sees the teacher's Lecture notes accessible on Screen two feet away.

1.2.6 Ensure Regularity

A central Server where the students will log on through the Internet, prior to gaining access to the Live Electronic Class Lecture. Log on is necessary to ensure regularity by recording their presence in a Database, thus enabling automatic update of *Attendance Record*. The teacher's Interface also displays the students *Online* at a particular instant.

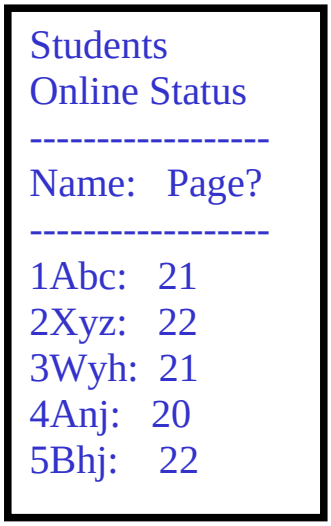

Fig 1.5: Status Monitoring

1.2.7 Enhance and enrich the learning with Integrated Learning Environment

To provide the students with Integrated Learning Environment and structured education in the new media. The Electronic Classroom will not change the content of education but it will change the medium and in doing so it will enhance and enrich the learning and communication process.

Online Lecture can be made attractive and self expressing by utilizing some animations, graphics, pictures as relevant to the Subject.

1.2.8 Providing Geographical Flexibility

Both the students and the teachers can log on from their home location, or directly from within University Campus. This would also facilitate in utilizing services of a teacher who is even poles apart from the University.

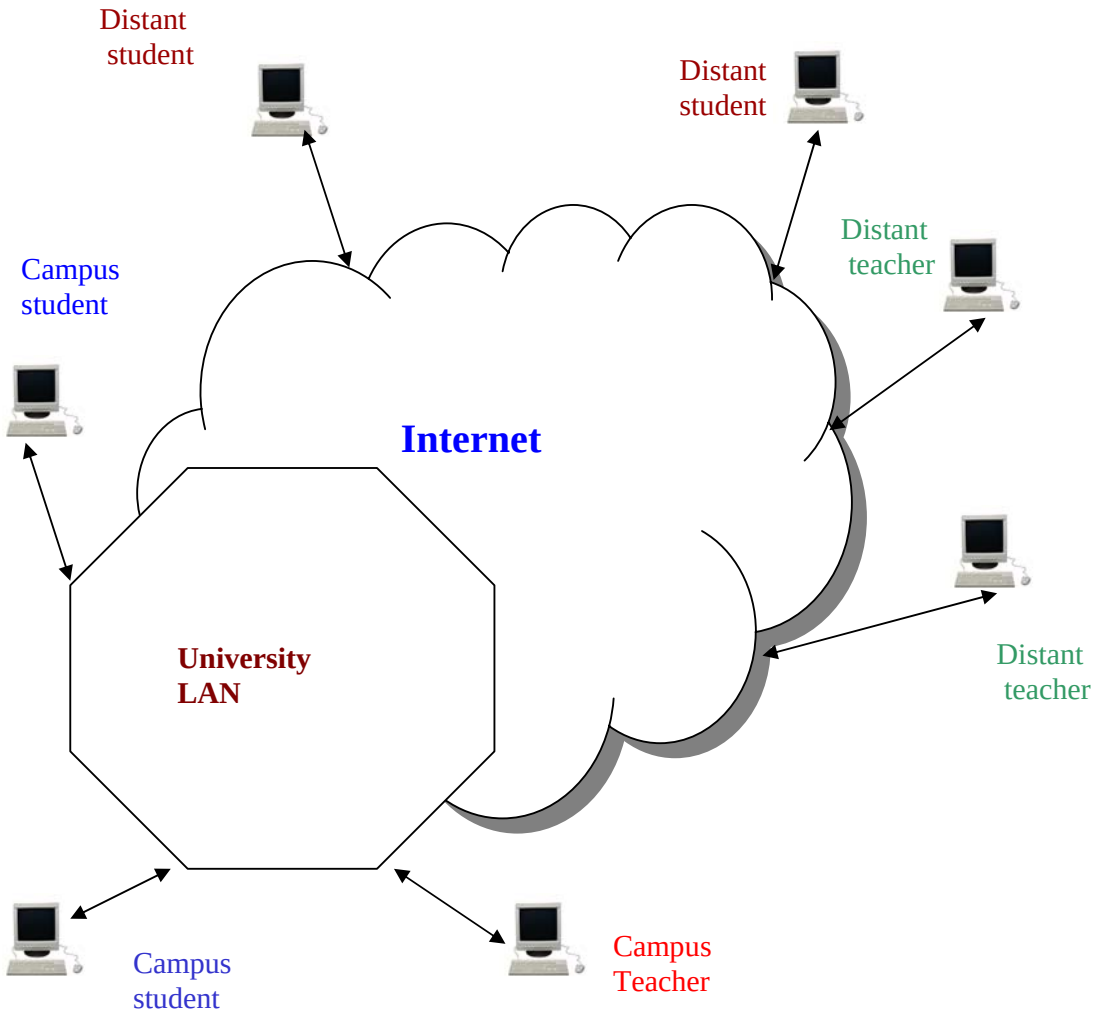

Fig 1.6: Geographically flexible system.

1.2.9 Navigating through activities

There must be a way to navigate among the activities like reading, listening and viewing others interactions, queering and getting Response.

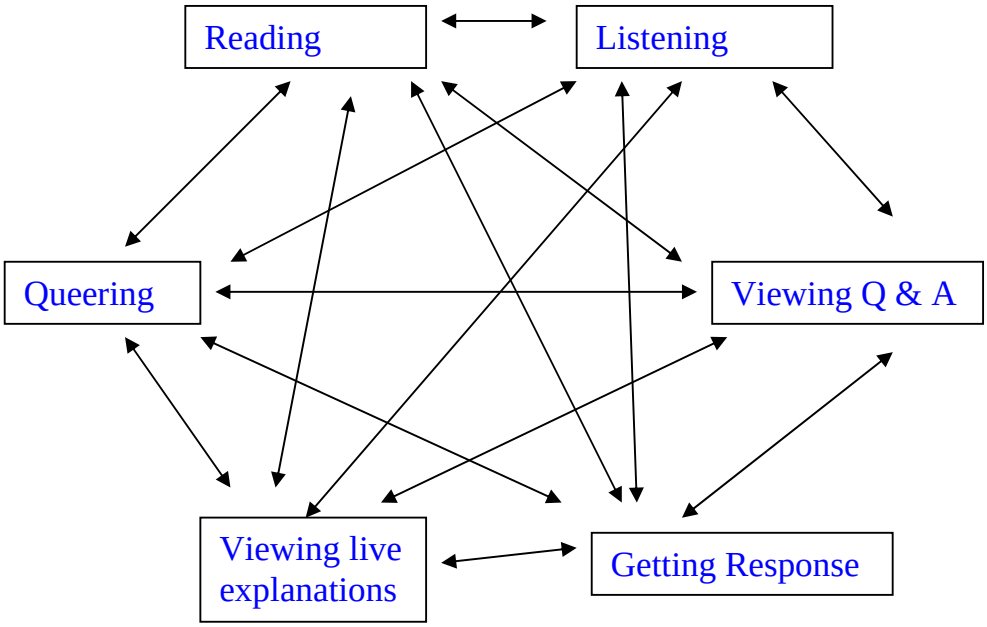

Fig 1.7: Easy Navigation

1.2.10 Fast and Flexible System Development

This realization is targeted to work through low bandwidth 56 kbps Internet. There might be a need for some optimizations, which will reveal once problem domain will be analyzed completely or perhaps later during Design and Implementation stages.

1.3 Solidifying Required Functionalities

Some of the functionalities aimed to achieve include:

- ✓ Support rapid development of Lecture material by the teacher.
- ✓ Powerful interaction of the teacher and the students while avoiding any disruptive effect and keeping the students attention focused.
- ✓ Tracking the student's progress through Lecture by the teacher.
- ✓ Live explanations by the teacher using Blackboard demonstrations through live still Pictures.
- ✓ Automatic preparation of excellent future reference after interactive Class Session. Thus to make it available afterwards.
- ✓ Providing an integrated learning environment and providing means of navigation through different activities.
- ✓ Ensures regularity of the students by maintaining attendance record.

1.3.1 What if Available Internet Bandwidth Increases?

Asynchronous content will apply as it is, while there might be some improvement in synchronous content. Therefore this Implementation will not become obsolete in future.

1.4 What we will prepare

1.4.1 A teacher's software for Lecture content development

1. A software that will enable the teacher to rapidly integrate Text, Pictures and Sound in required format for this system, by providing a 'Template'.
2. Template will divide Lecture into Sections, for display on Browser Screens.
3. Each Section may include one or more Text, Pictures and Sound.
4. Provision for recording Sound by the teacher.
5. Ability to use scanner for Text and Pictures, or use from some saved file.
6. Apply formatting to above content.
7. Convert all material to specific form that will be used by the University Server during live interactive Class Session.
8. All above functionalities are to be provided by this Software using a user-friendly interface.

1.4.2 Server Side Software

1. Showing the student's online status at particular instant.
2. Automatically maintain Database record of attendance.

3. Automatic tracking of the student's progress through Lecture material during live interactive Class Session.
4. Support live interactive Class Session that will use prepared Lecture material and add interactivity to it in the form of Queries and Responses.
5. Being able to deliver some fixed limited amount of Text, Figures and Audio of prepared Lecture material to a client at a time.
6. Depending on the teacher's choice for a particular Query, Response will be either in textual format or involve working out problems on the Board, which is taken through Camera and delivered in the form of still Pictures.[26]
7. Being able to handle multiple Queries at a time.
8. If class time is over unanswered Queries will be responded through email.
9. After live Class Session, prepared Lecture contents will be combined with dynamic real time contents, without any need of post editing labor, neither on the part of the teacher nor the student. Interactive Queries and Responses will be embedded into proper location in the material that will be available through The University Server afterwards for future reference.

10. We are working to make such a realization in distant environment. But our teacher being a human can only answer sequentially. He cannot work in parallel. Therefore assuming a single Query takes 1 minute on average (either using text or pictorial illustration) then we can say that for a class of about 2 hours, we can provide service to typically 30-40 students, each student asking 4 Queries on average (There is no limit on number of Silent Students).

1.4.3 Clients will use Browser only.

No special software will be required by the students.

1.5 Hardware

(i) Hardware Required for Developing the System

Computer System

- P-III 733MHz (or above), 256 RAM,
- Dialup Internet Connection,
- Scanner, Camera,
- Microphone, Speakers.

(ii) Hardware Required for Commissioning the System:

1. A computer for the instructor, with Internet connection.
2. A Blackoard for the teacher for presenting information.

3. A computer for each student, with Internet connection.
4. Powerful Server with fixed IP address assigned.
5. Scanner, Camera and Microphone for the teacher.[26]
6. Speakers for both the teacher and the students.

1.6 Software Tools

- Rational Rose [11]
- Visual Basic 6 [5]
- VB Script 5 [4]
- Internet Explorer 5.0
- Active Server Pages / HTML Pages [1,2], [3,7]
- Windows 2000 Platform [10]
- SQL Server 2000 [6]
- Microsoft Access 2000
- Real Video Server 8.0 [19-22,32]
- Internet Information Server 5.0

1.7 Dividing Our Work

It is always better to divide work into parts that are completed one by one instead of delving into intricate details all at once. This strategy often simplifies complexities and aids in accomplishing aggregate task quickly, utilizing step-by-step refinement approach. This is a basic strategy in developing large software projects. [9,11]

This thesis work is divided into two main parts. First part is the Analysis, Design and Implementation of the Application that will be used for Lecture preparation. Second part is the Server Side Software Analysis, Design and Implementation that include provision of live interaction capabilities as required.

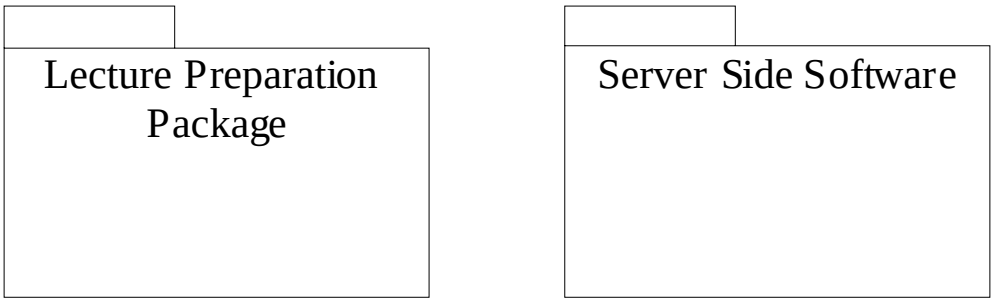

Fig 1.8: Two Parts of the Software developed during thesis work

1.8 Actors Generalization

Following figure shows a generalization relationship between actors of the system we are aiming to develop. Note, this relationship is shown here in the beginning, as it is same for both parts of the software.

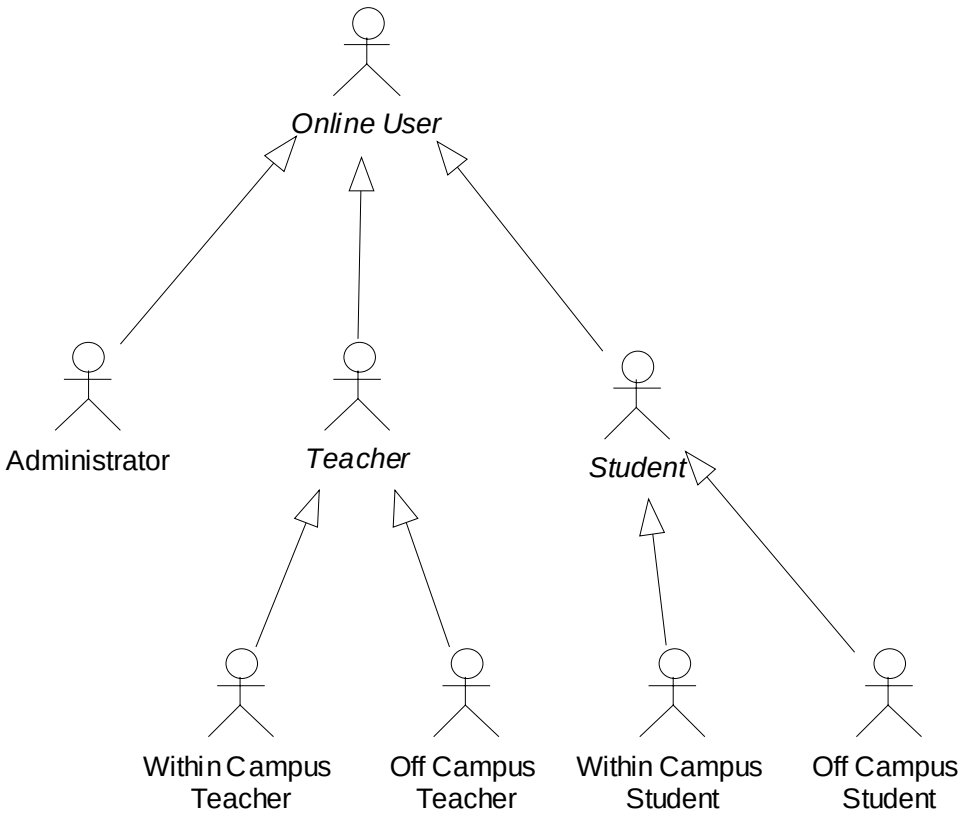

Fig 1.9: Generalization Relationships between Actors.

Chapter 2

Lecture Preparation Application

2.1 Need For Lecture Material

We are developing Distant Electronic Classroom therefore it is necessary that first of all some teaching material (Text, Pictures etc) should be prepared that is available during Class Session. Generally in a normal multimedia classroom the teacher prepares Lecture material prior to taking class. It helps in better delivering concepts using slides with Figures plus associated Text. Here we are developing system for Distance Education, therefore in addition to Figures and Text, we also need to include related Audio narration by the teacher. This will be audible to the students when they will be attending Live Class Session as well as afterwards.

2.2 Offline Lecture Preparation

It is most appropriate to assume that our teacher when he is outside the University will prepare his Lecture materials without being connected to the Internet. This is a good economic consideration, especially for our country. Only when a Lecture is prepared completely and even reviewed, then it is uploaded to the University Server. The Server Side Software must automate and simplify process of uploading Lecture and related Audio and Picture files.

2.3 Defining Lecture Format

As mentioned earlier our prepared Lecture will include Text, Figures and recorded Audio content. Our software should be capable of dividing Lecture into specific Sections each with own “Section Title”. It is possible that not all Sections are of equal length. To circumvent this problem each Section will be divided into a number of relevant Screens. Only single Screen content will be delivered to the student’s Browser at one time. This will also facilitate in fast download of content during Live Class Session.

It is not necessary that each Screen must include Text, Figure and Audio. This is just a general Template we are designing for our Lecture. It is possible that a Screen may contain a Figure only, or it may comprise of Text content only.

2.4 Ease of Lecture Preparation

One of the main objectives of the Software we are developing for Lecture preparation is that it must be user friendly. We are expecting that users with little computer experience may use our Application. Therefore user interface of our Application should provide sufficient guidance to a new user. It should allow him to rapidly prepare Lecture material.

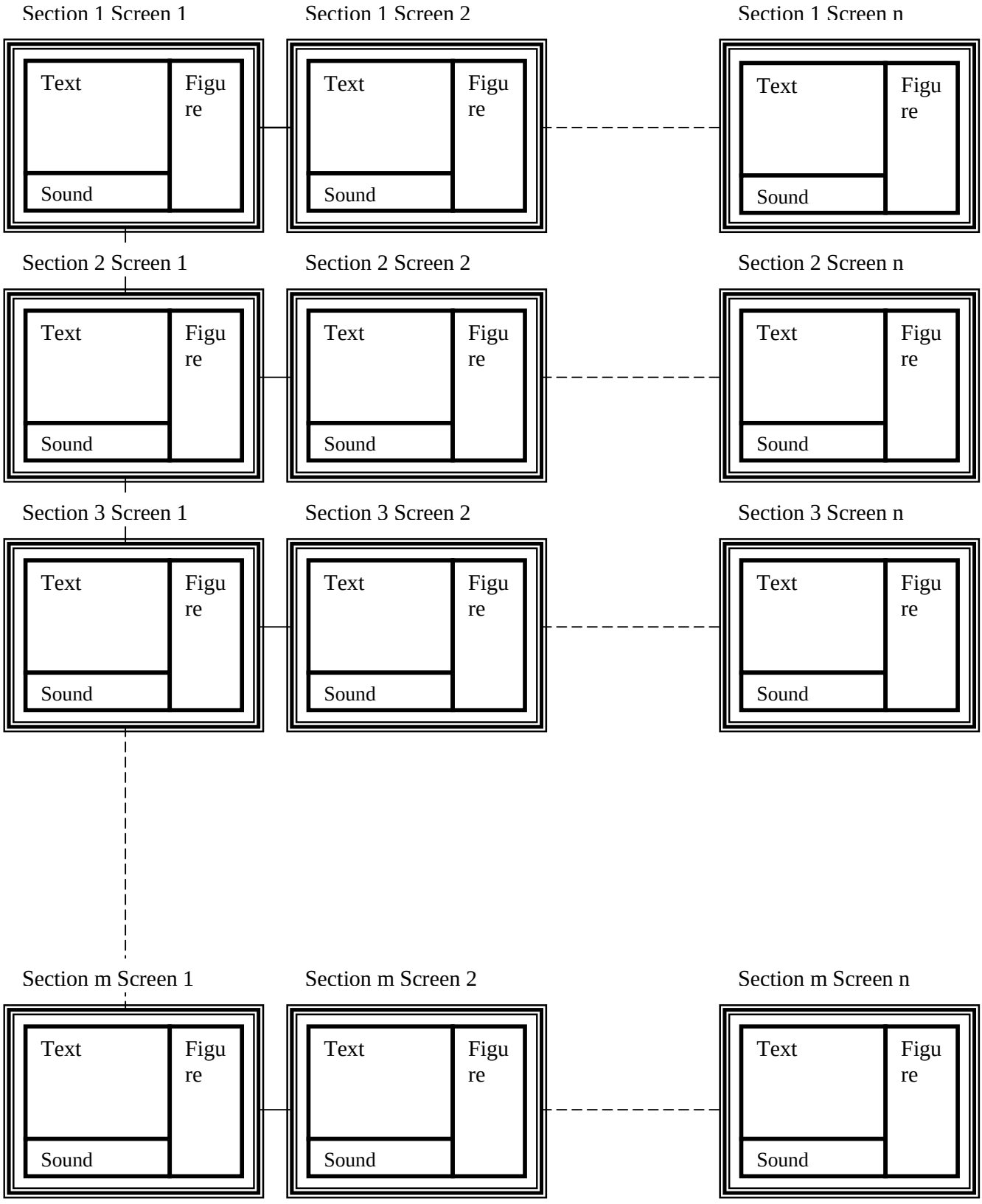

Fig 2.1: General Layout of Template

2.5 Scanning Facility

The developed Application should be capable of importing Text and Figures using Scanner.

2.6 Maintaining record of all related files

Application should take care of where ever Picture and Sound files are stored on the client's computer. Because it may become a problem when user will be transferring data to the main University Server. It should not be a responsibility of user to maintain a record of where different files are.

2.7 Recording and Compressing Audio

Our application will provide facility to the Teacher, to record voice for each Screen in either any or all Sections of a specific Lecture. Plus this recorded Voice would be compressed using some Encoder, so that it is small in size and does not consume much of available Internet bandwidth.[23,24]

2.8 Previewing the Prepared Lecture

There must be some means to preview the whole prepared Lecture material including all the content, actually through Browser, so as to give an idea of how it looks when actually it is be delivered to the students. This would facilitate the teacher to view early during development of Lecture materials the actual output. This functionality should be available in offline mode.

2.9 Formatting Content

Our Application will allow the teacher to apply basic formatting to the Text. Simple Text formatting includes setting font name, font color, and font size, bold, italic and underline. Plus there must be provision for adjusting size and position of Figures.

2.10 Easy Navigation Through Lecture Material During Development

Our Application would make it easy to move forward and backward through all Sections and through all Screens of a particular Section. Also it should provide facility to add any number of Sections and their extended Screens.

2.11 Maximizing Reuse

It is a fact that utilizing reused components, task of developing robust systems becomes easy and straightforward. Therefore throughout development of our Software reuse, either Black box or White box will be applied.[15]

Chapter 3

Analysis and Design of

Lecture Preparation Application

3.1 Use Case Hierarchy For Lecture Preparation Application

Number	Name	Type
ULB10000	Prepare Lecture	(Package)
ULB10100	Generic Functionalities	(Sub Package)
ULB10101	Create Lecture	(Concrete)
ULB10102	Create New Using Template	(Abstract)
ULB10103	Change Existing Lecture	(Abstract)
ULB10104	Save Contents	(Concrete)
ULB10105	Browse Different Parts	(Concrete)
ULB10106	Upload	(Concrete)
ULB10107	Preview	(Concrete)
ULB10200	Manipulate Section Titles	(Sub Package)
ULB10201	Add Section Title	(Concrete)
ULB10202	Rename Section Title	(Concrete)
ULB10203	Remove Section Title	(Concrete)
ULB10204	Insert in between	(Abstract)

ULB10205	Remove All	(Concrete)
ULB10300	Add and Format Text	(Sub Package)
ULB10301	Type Text	(Concrete)
ULB10302	Set Font name and Color	(Concrete)
ULB10303	Set Bold Italic and Underline	(Concrete)
ULB10304	Import Text	(Concrete)
ULB10305	Scan	(Abstract)
ULB10400	Add and Format Figures	(Sub Package)
ULB10401	Insert Figure	(Concrete)
ULB10402	Remove Figure	(Concrete)
ULB10403	Set Size and Position	(Concrete)
ULB10404	Edit Figure	(Concrete)
ULB10305	Scan	(Abstract)
ULB10500	Add Voice	(Sub Package)
ULB10501	Record Audio	(Concrete)
ULB10502	Play Audio	(Concrete)
ULB10503	Insert Audio	(Concrete)
ULB10504	Compress	(Abstract)

Ref [11]

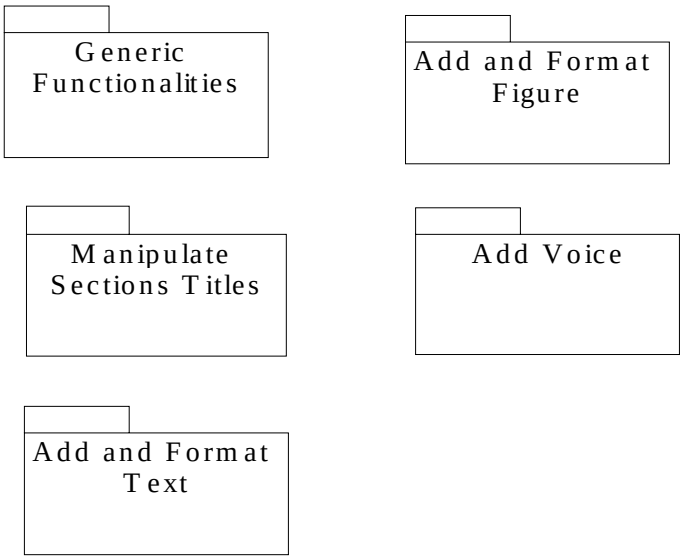

Fig 3.1: Sub Packages of Prepare Lecture Package

3.2 Generic Functionalities Sub Package

3.2.1 Use Case Diagram

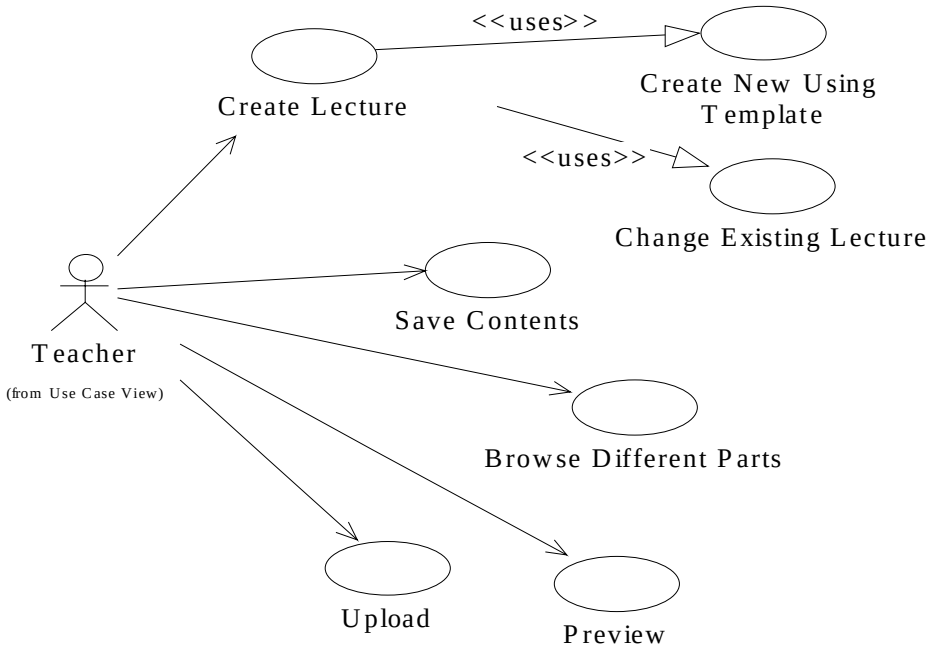

Fig 3.2: Generic Functionalities Sub Package Use Case Diagram

3.2.2 Use Case Specifications and Sequence diagrams

Use Case Name: Create New Using Template

Use Case Number: ULB10102

Description: Create New Using Template generates new Lecture file.

Purpose: This use case is used to automatically create new Lecture file using a Template.

Related Use Cases: Create Lecture , Open Existing Lecture , Preview Save Contents , Browse different Parts , Upload

Primary Actors: Teacher

Type of Actor: Human

UC Spec: Body Section

Main-Course:

Create Lecture use case is used to generate files that are used for storing data. This use case uses functionalities of two abstract use cases.

If a new Lecture file is to be created then “Create New Using Template” use case is used.

If some earlier saved Lecture file is to be changed than “Change Existing Lecture” uses case is used.

Actor action	System response
1) Teacher commands to create Lecture.	
	2) System reads file names already present.
	3) Prompt user for new file name with some proposed file name as default.
4) Teacher enters new file name or selects default.	
	5) System create new file using Template.
	6) The use case ends.

UC Spec: Trailer Section

Question: What is a Template?

Answer: Template is a file, which has some structure for storing data including Text, Audio and Figures.

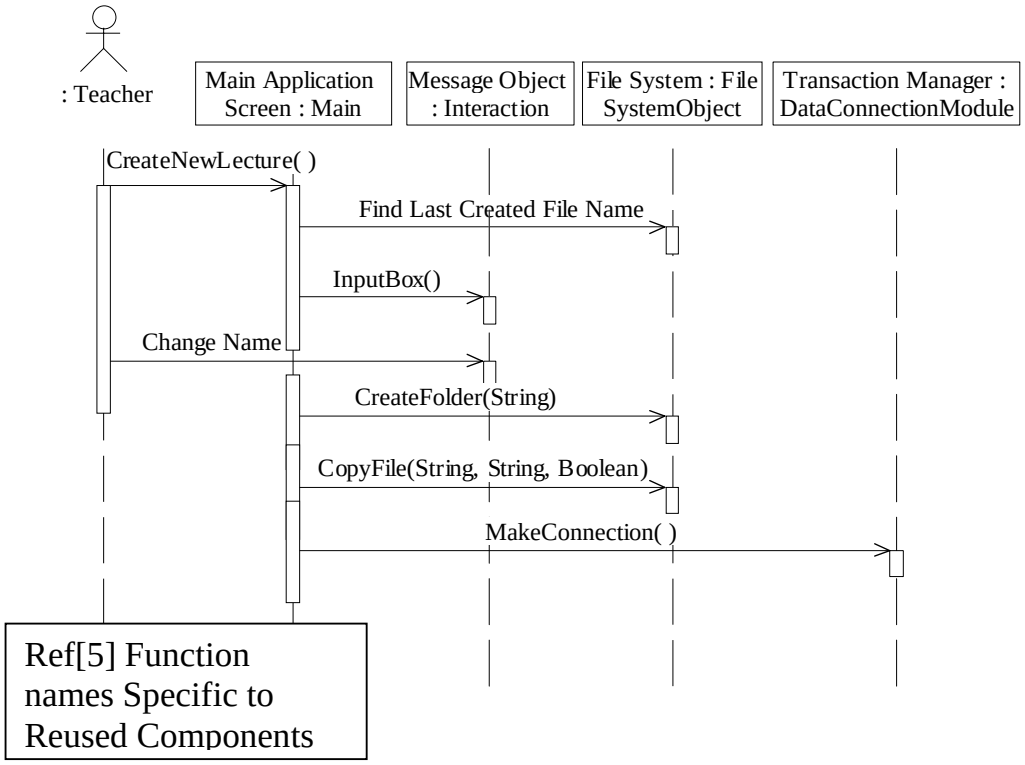

Fig 3.3: Create New Using Template Sequence Diagram

Use Case Name: Change Existing Lecture

Use Case Number: ULB10103

Description: Change Existing Lecture opens an existing Lecture file.

Purpose: This use case is used to load contents from existing Lecture file and thus enable modification.

Related Use Cases: Create Lecture , Create New Using Template , Upload Save Contents , Browse different Parts , Preview

Primary Actors: Teacher

Type of Actor: Human

UC Spec: Body Section

Actor action	System response
1) Teacher commands to open Lecture file.	2) System checks files present and asks user for selection.
3) Teacher makes selection of file.	4) System loads selected file. Its stored data is now available for modification.
	5) System displays previous editing date.
	6) The use case ends.

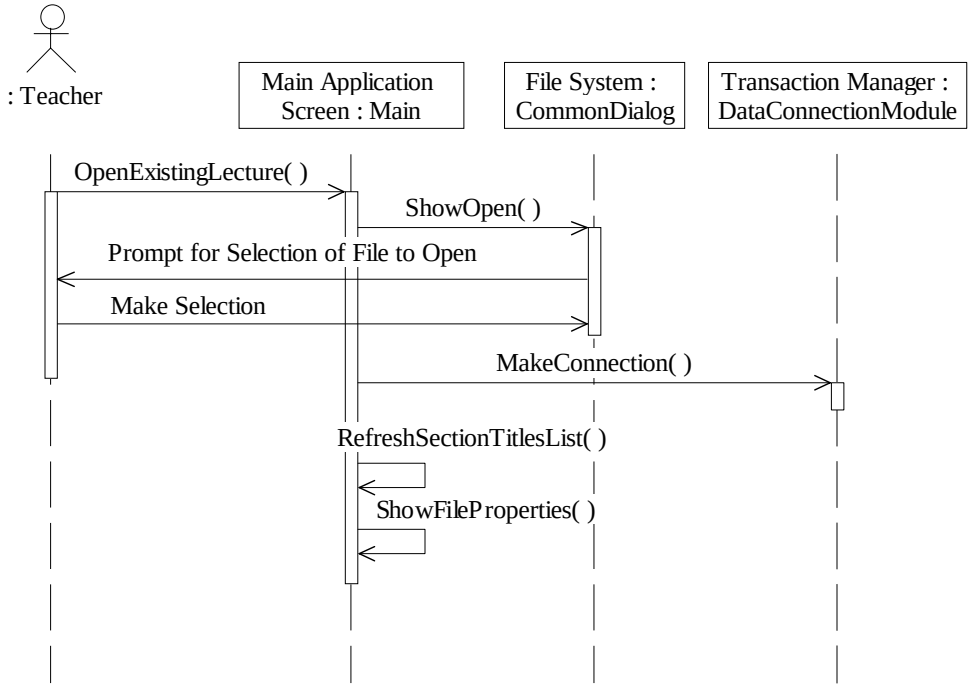

Fig 3.4: Change Existing Lecture Sequence Diagram

Use Case Name: Save Contents

Use Case Number: ULB10104

Description: Save Contents saves contents to some file.

Purpose: This use case is used to save Sections content including Text, Audio and Figures to some specific file.

Related Use Cases: Create Lecture , Create New Using Template Change Existing Lecture

Primary Actors: Teacher

UC Spec: Body Section

Teacher commands to save data.

System saves all Section Titles, their individual data including Text data, associated Voice data and Pictures.

System may automatically save data after some specific intervals.

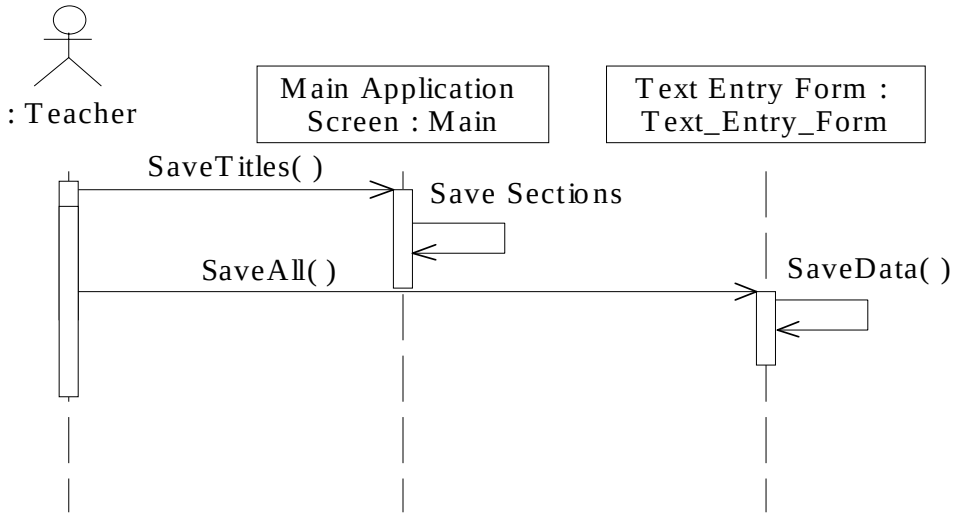

Fig 3.5: Save Contents Sequence Diagram

Use Case Name: Browse Different Parts

Use Case Number: ULB10105

Description: Browse Different Parts use case is used to load next or previous Section.

Purpose: Facilitates easy movement between Sections.

Related Use Cases: Rename Section Title , Remove Section Title

Primary Actors: Teacher

Type of Actor: Human

UC Spec: Body Section

Actor action	System response
1) Teacher commands to go to next Section.	2) System first saves current Section data. 3) System loads next Section's data, and displays it.
4) Teacher commands to go to previous Section.	5) System first saves current Section's data. 6) System loads previous Section's data and displays it. 7) The use case ends.

UC Spec: Trailer Section

Question: How many Sections are supported?

Answer: Initially we will design for 1 to 100 Sections.

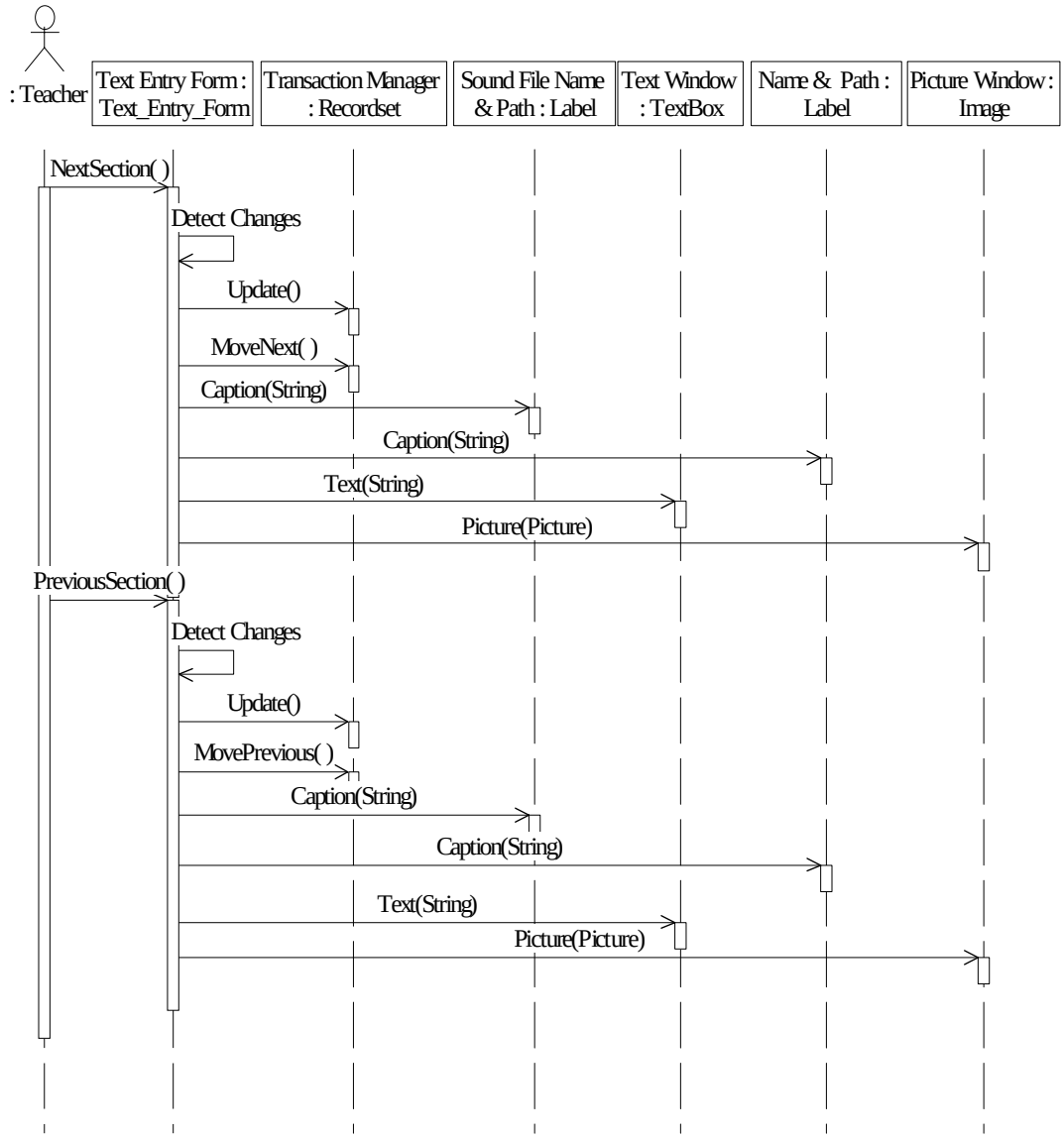

Fig 3.6: Browse Different Parts Sequence Diagram

Use Case Name: Upload

Use Case Number: ULB10106

Description: Upload use case transfers Lecture data files to the Server.

Purpose: It is necessary to transfer whole Lecture contents to the Server before Lecture Session is started through the Internet.

Related Use Cases: Create New Using Template , Open Existing Lecture , Save Contents , Browse different Parts , Preview

Primary Actors: Teacher

Technical Requirements: Dial up Connection is available.

UC Spec: Body Section

Pre-condition: Lecture is ready.

Computer is connected to the Internet.

Main-Course:

Teacher commands to upload.

System starts Internet Browser and opens specific web site.

Teacher authenticates himself.

Teacher selects Lecture to upload.

System starts transferring data.

After transfer Use case ends

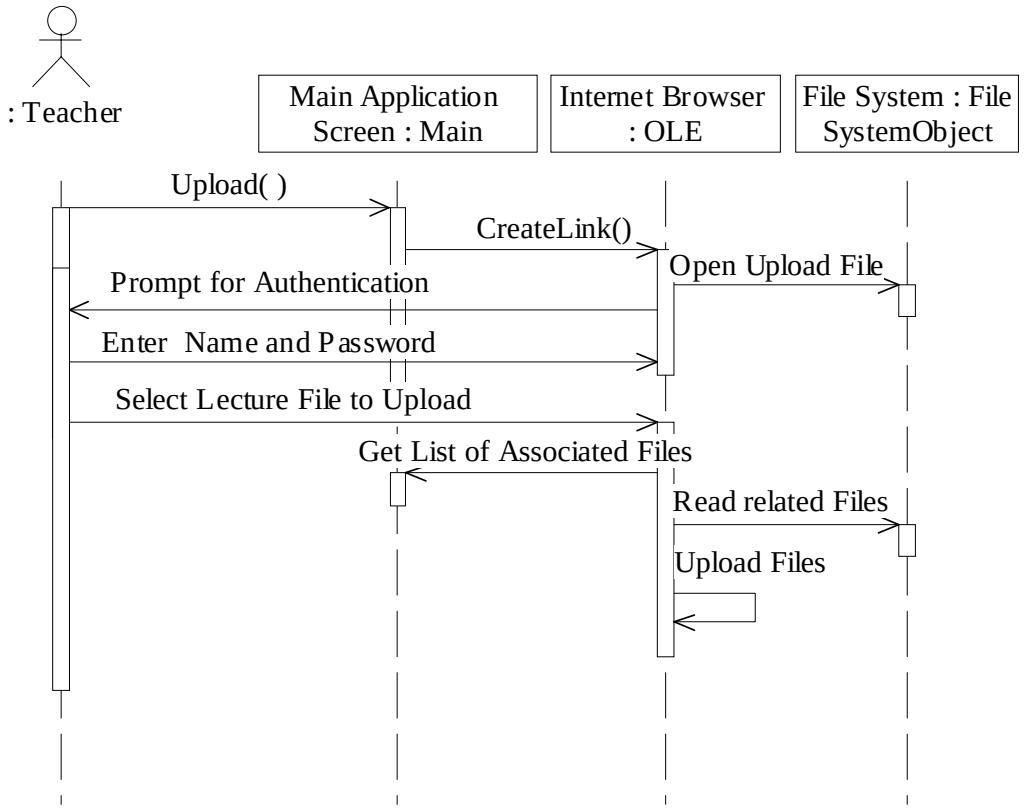

Fig 3.7: Upload Sequence Diagram

Use Case Name: Preview

Use Case Number: ULB10107

Description: Preview use case is used to generate a preview of Lecture during its development.

Purpose: Enables to view Screens as will be seen during interactive class.

Related Use Cases: Create New Using Template , Open Existing Lecture Save Contents , Browse different Parts Upload

Primary Actors: Teacher

UC Spec: Body Section

Pre-condition: Some Section data is available.

Main-Course:

Teacher commands to preview during development any time.

System generates web pages in specific format. After generating files, system loads first Screen in an The Internet Browser.

Teacher browses through all Sections.

Teacher closes Internet Browser after he has finished previewing.

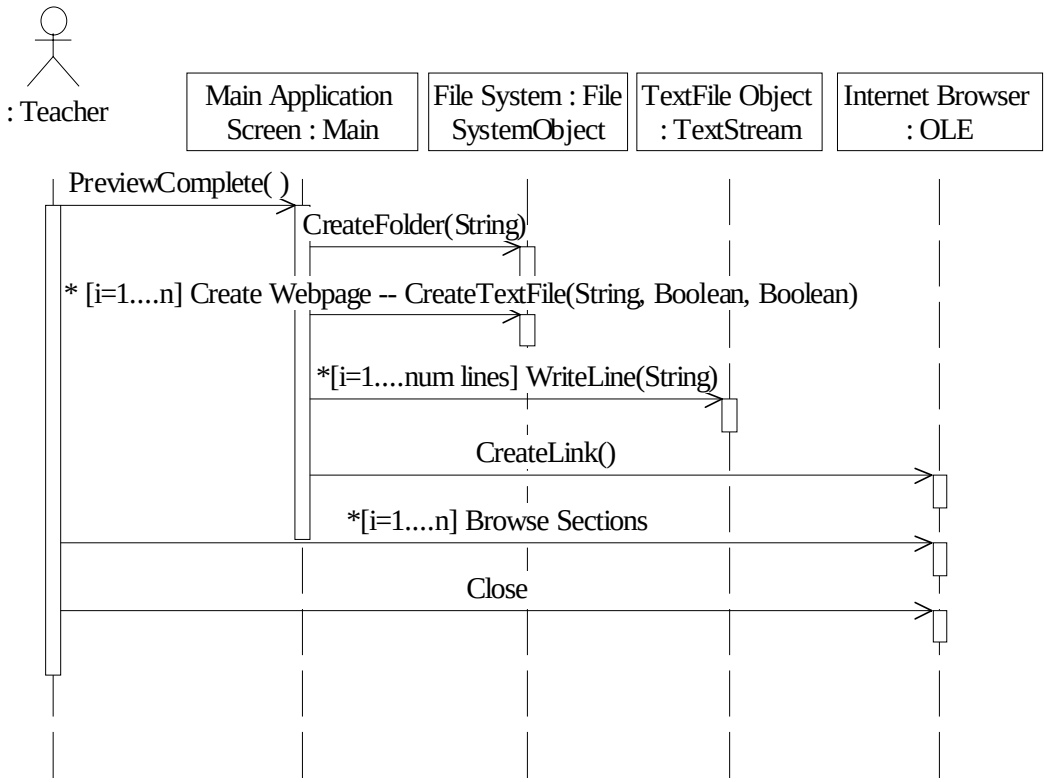

Fig 3.8: Preview Sequence Diagram

3.3 Manipulate Section Titles Sub Package

3.3.1 Use Case Diagram

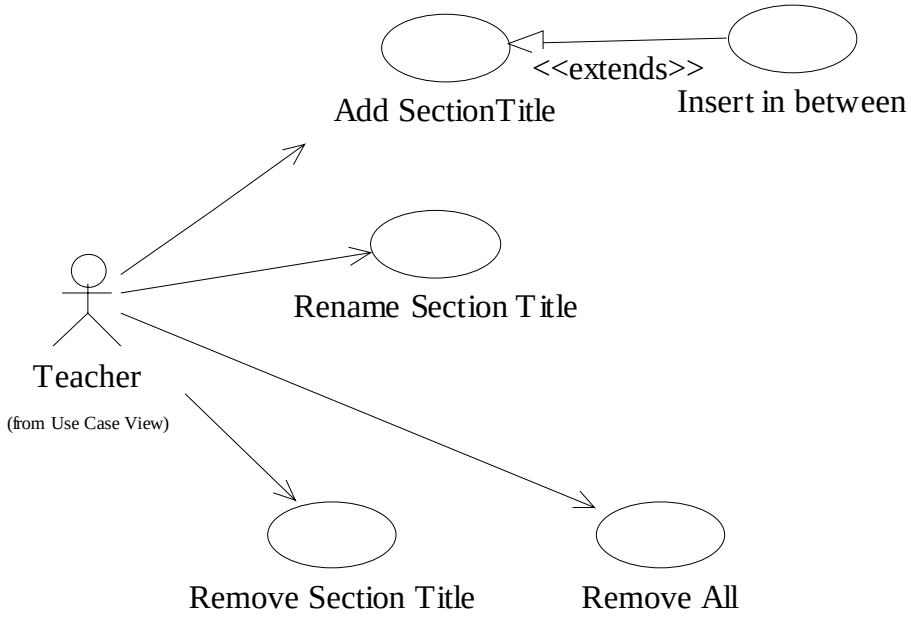

Fig 3.9: Manipulate Section Titles Sub Package Use Case Diagram

3.3.2 Use Case Specifications and Sequence diagrams

Use Case Name: Add Section Title

Use Case Number: ULB10201

Description: Add Section Title use case is used to Add New Titles.

Purpose: Facilitates easy entry of Section Headings.

Related Use Cases: Rename Section Title , Remove Section Title

Primary Actors: Teacher

Technical Requirements: Section Titles in English language alphabets. Section Titles must have at least one distinct character to facilitate differentiating them.

UC Spec: Body Section

Actor action	System response
1) Teacher types new Section Title.	2) System adds it to list in background and displays Section number for this Section Title and name of title for user convenience at each step.
3) Teacher adds as many Titles as required.	4) Titles list populates accordingly.
	5) Total number Sections is also displayed for convenience.
	6) The use case ends.

UC Spec: Trailer Section

Question: How many Sections are supported?

Answer: Initially we will design for 1 to 100 Sections.

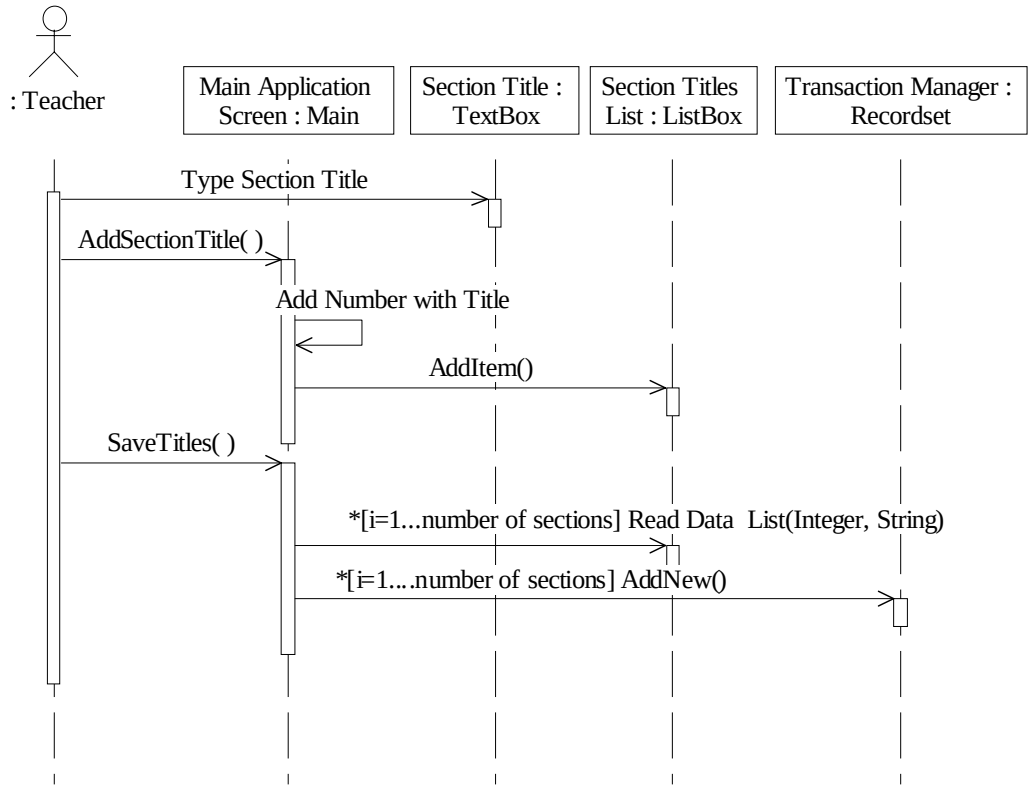

Fig 3.10: Add Section Title Sequence Diagram

Use Case Name: Rename Section Title

Use Case Number: ULB10202

Description: Rename Section Title use case is used to change name of an already entered Section Title.

Purpose: Provision for changing Section Heading at any time.

Related Use Cases: Add Section Title , Remove Section Title

Primary Actors: Teacher

Technical Requirements: Changed Section Title must be different from others.

UC Spec: Body Section

Pre-condition: At least one Section Heading is present before changing its name (a bit of common sense).

Actor action	System response
1) Teacher selects Section Title to be renamed.	2) Application highlights it to differentiate from rest.
3) Teacher enters changed title name.	4) Application updates list of titles.
	5) The use case ends.

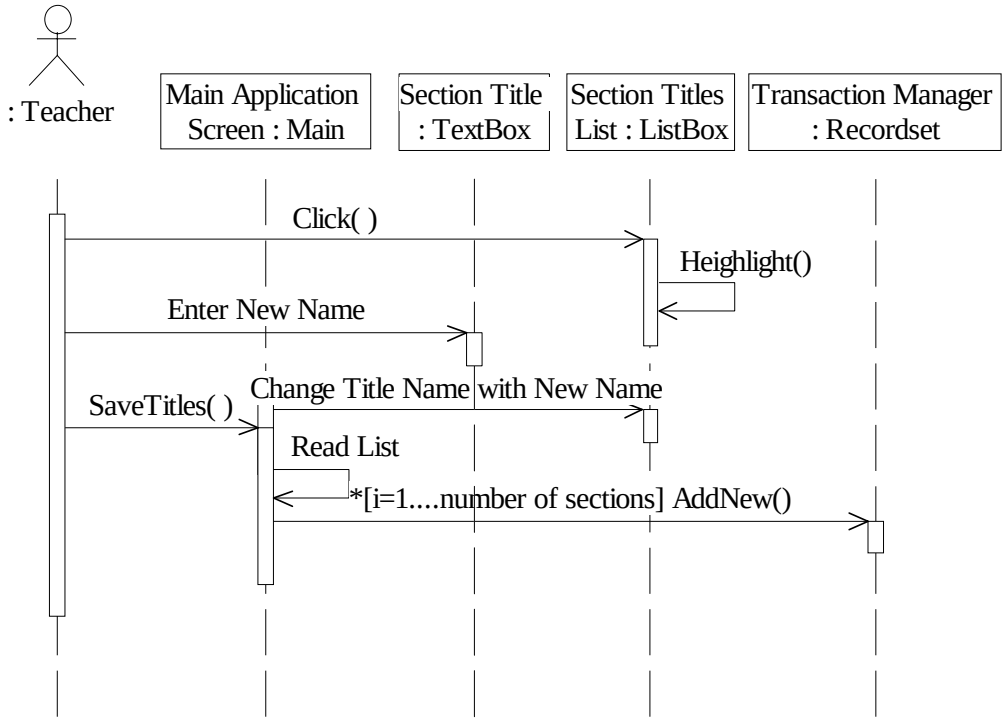

Fig 3.11: Rename Section Title Sequence Diagram

Use Case Name: Remove Section Title

Use Case Number: ULB10203

Description: Remove Section Title use case is used to remove any Section Title. Also complete contents of that Section are removed.

Purpose: Provides facility of removing any Section form Lecture.

Related Use Cases: Add Section Title , Rename Section Title

Primary Actors: Teacher

UC Spec: Body Section

Pre-condition: At Least one Section Heading is present before deleting.

Main-Course:

Actor action	System response
1) Teacher selects Section Title to be removed.	2) Application highlights it to differentiate from rest.
3) Teacher commands to remove.	**4)** Application prompts for confirmation, so as to avoid accidental removal.
5) Teacher confirms removal.	6) Titles list shrinks with each removal accordingly.
	7) The use case ends.

UC Spec: Trailer Section

Question: How to secure from accidental removal of any Section?

Answer: System warning to user confirms user's decision.

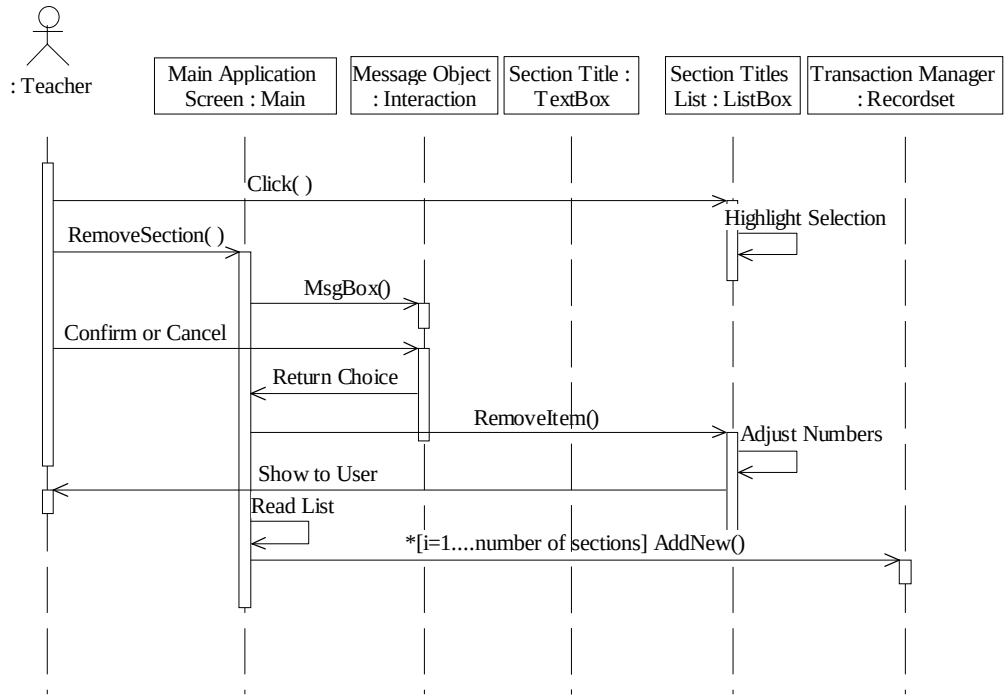

Fig 3.12: Remove Section Title Sequence Diagram

Use Case Name: Insert in between

Use Case Number: ULB10204

Description: Insert in between use case is used to add new titles in between others.

Purpose: Inserting new Sections.

Related Use Cases: Add Section Title

Primary Actors: Teacher

UC Spec: Body Section

Main-Course:

Actor action	System response
1) Teacher selects position for new Section Title.	
2) Teacher types new title to be inserted.	
3) Teacher commands to add.	4) New Title is added at desired location in list.
	5) The use case ends.

UC Spec: Trailer Section

Question: How Section numbers are affected after inserting new Sections?

Answer: New inserted title takes number of old one, and system takes account of this. All related contents of other Sections are not disturbed by this operation. This operation can be done at any stage of preparation.

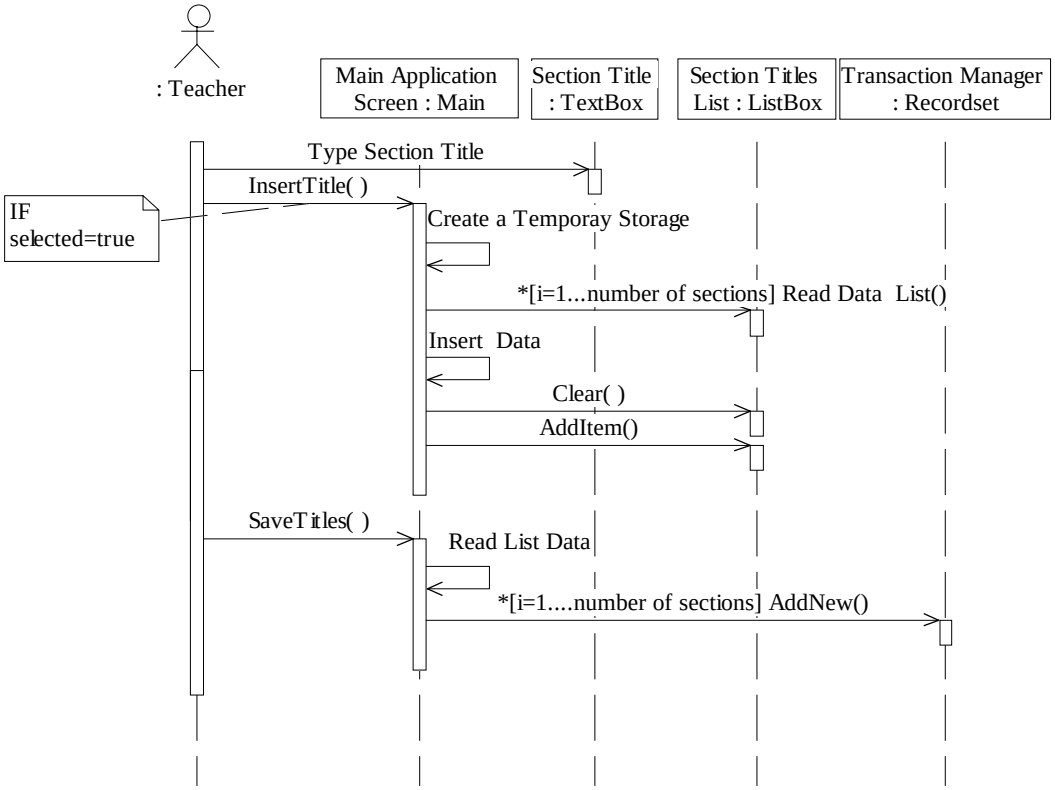

Fig 3.13: Insert in Between Sequence Diagram

Use Case Name: Remove all

Use Case Number: ULB10205

Description: Remove all use case is used to remove all Section Titles and their contents as well.

Purpose: To delete all Sections.

Related Use Cases: Remove Section Title

Primary Actors: Teacher

UC Spec: Body Section

Pre-condition: There must be at least one Section developed so far.

Main-Course:

Actor action	System response
1) Teacher commands to remove all Sections.	2) System prompts for confirmation because it is a critical operation
3) Teacher confirms operation.	4) System does the job.
	5) The use case ends.

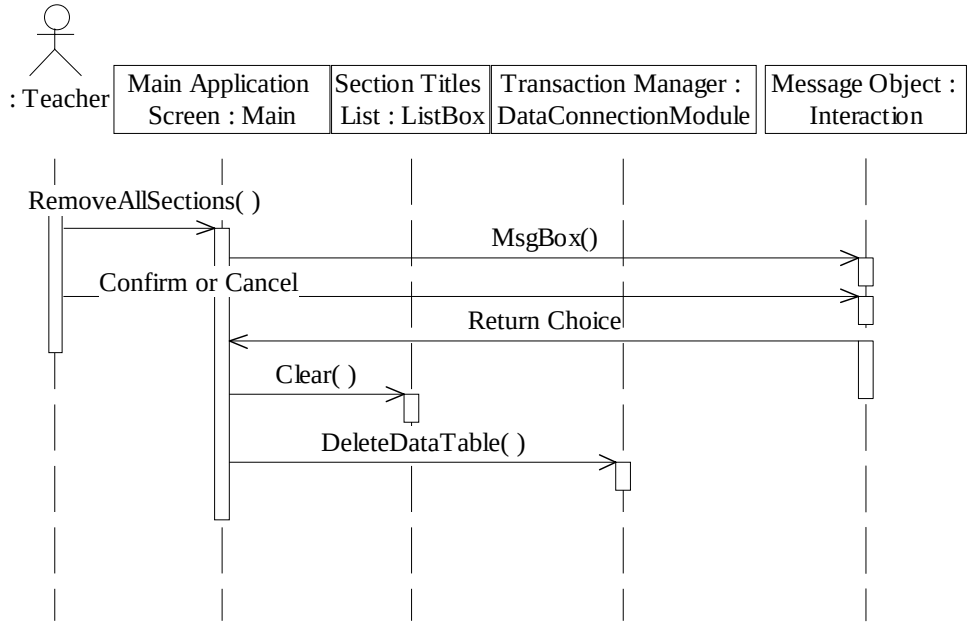

Fig 3.14: Remove All Sequence Diagram

3.4 Add and format Text Sub Package

3.4.1 Use case Diagram

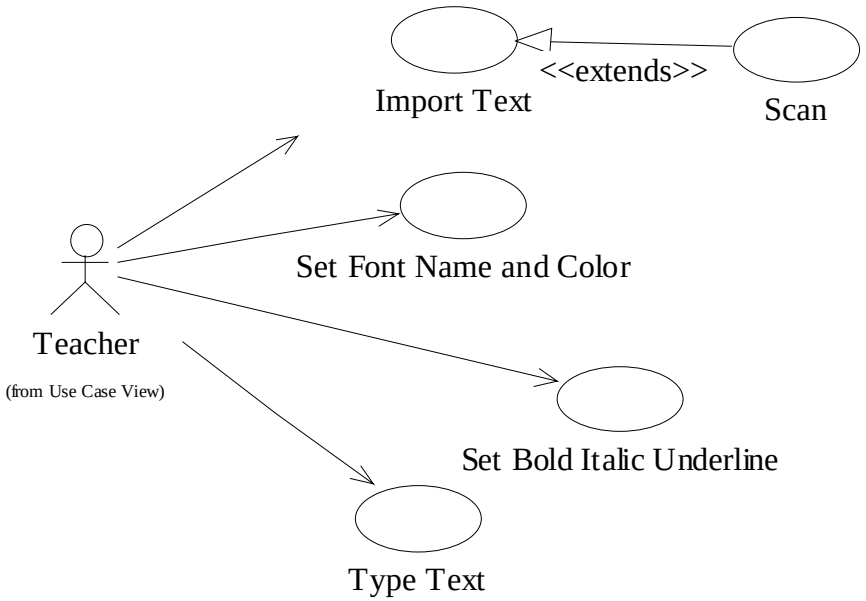

Fig 3.15: Add and Format Text Sub Package Use Case Diagram

3.4.2 Use Case Specifications and Sequence Diagrams

Use Case Name: Type Text

Use Case Number: ULB10301

Description: Type Text use case is used to type Text by user through keyboard.

Purpose: Prepare Text data for Lecture.

Related Use Cases: Set Font name and Color , Set Bold Italic , Underline , Import Text

Primary Actors: Teacher

Technical Requirements: User will use keyboard for entering Text data.

UC Spec: Body Section

Main-Course:

Actor action	System response
1) Teacher starts a Section.	2) System provides facility of entering alphabets, numbers and other special characters.
3) Teacher types Text.	4) System stores all data.
	5) The use case ends.

UC Spec: Trailer Section

Question:	How much Text can be entered in a specific Section?
Answer:	It is better to divide Text in chunks that fit in one Browser window on client side so as to avoid scrolling. Therefore each Section's Text data will be divided into different Browser Screens for convenience.

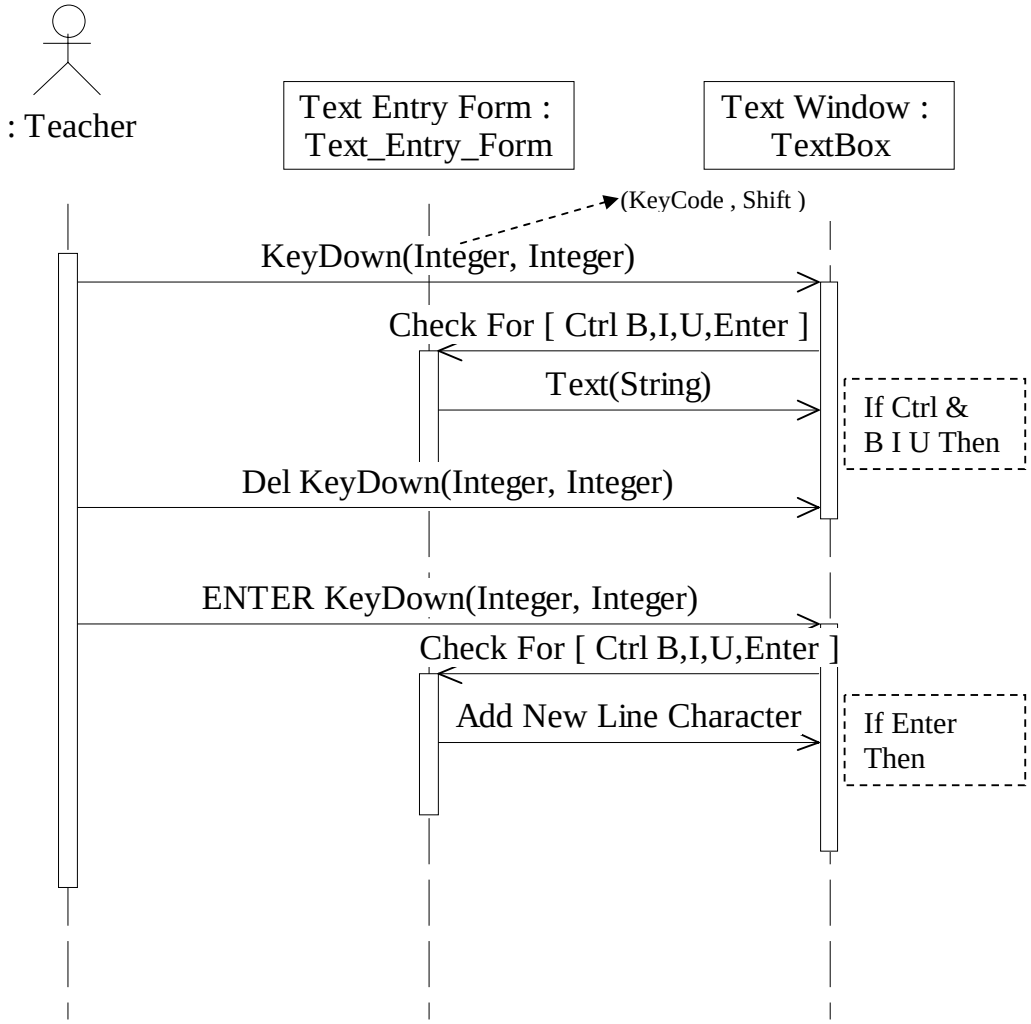

Fig 3.16: Type Text Sequence Diagram

Use Case Name: Set Font name Color

Use Case Number: ULB10302

Description: This use case is used to select different font names and respective colors of Text.

Purpose: To format Text data.

Related Use Cases: Type Text , Set Bold Italic Underline , Import Text

Primary Actors: Teacher

Technical Requirements: There is a need to use some color conversion technique because we are dealing with system and web based Browser applications. User will use mouse and/or keyboard for selections.

UC Spec: Body Section

Pre-condition: Some fonts must be installed in the system.

Actor action	System response
1) Teacher opens font selection control.	
2) Teacher selects specific font name and selects its color.	3) System sets Text with a specific font and color.
	4) The use case ends.

UC Spec: Trailer Section

Question: When font selection can be done?

Answer: This selection can be done at any time prior or after editing, or at any time.

Question: Why “Color Conversion” is required?

Answer: This requirement arises when we are dealing with standalone applications on a system and web based Browser applications.

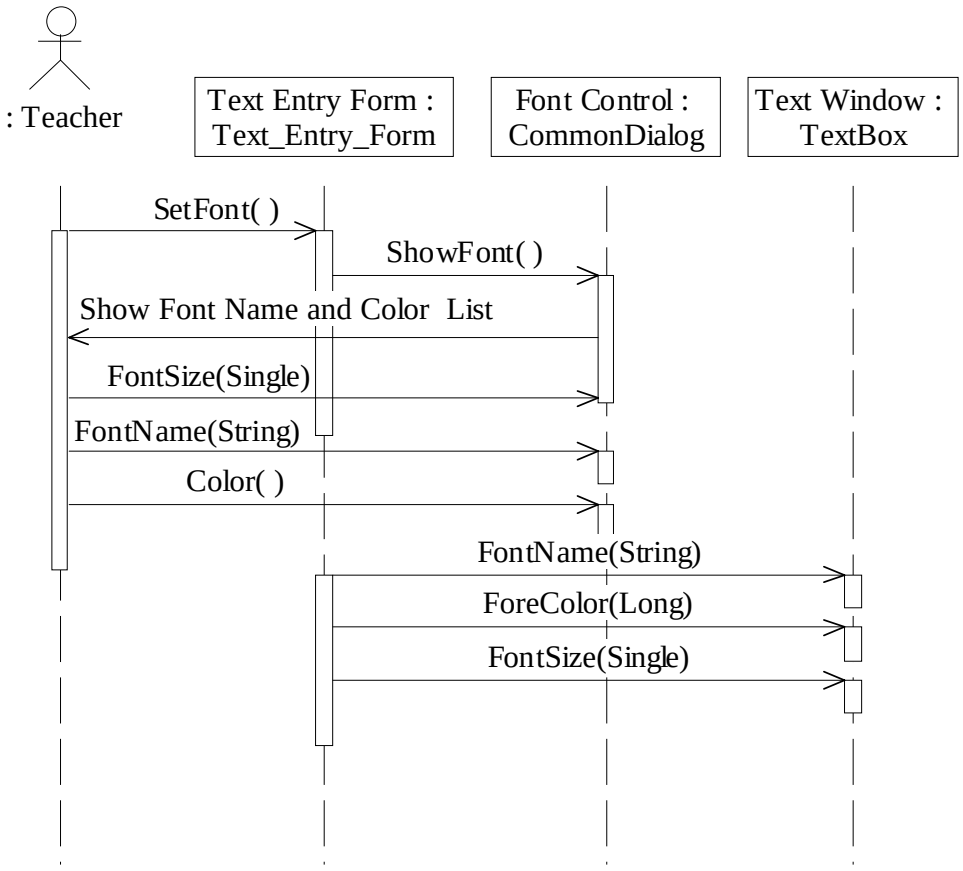

Fig 3.17: Set Font Name and Color Sequence Diagram

Use Case Name: Bold Italic Underline

Use Case Number: ULB10303

Description: This use case is used to select bold and/or italic and/or underline formatting of whole and/or specific words.

Purpose: To make Text more appealing.

Related Use Cases: Type Text , Set Font name and Color , Import Text

Primary Actors: Teacher

Technical Requirements: User will use mouse and/or keyboard for bold, italic and underline selections of words in Text data.

UC Spec: Body Section

Main-Course:

Actor action	System response
1) Teacher selects a specific word/phrase/paragraph(s).	**2)** System highlights the selection.
3) Teacher makes selection of bold and/or italic and/or underline through keyboard shortcuts or use mouse at his own convenience.	4) System appropriately makes bold and/or italic and/or underline formatting.
	5) The use case ends.

UC Spec: Trailer Section

Question: Whether it is possible to apply bold italic and underline setting at once to some content?

Answer: Yes, this can be done.

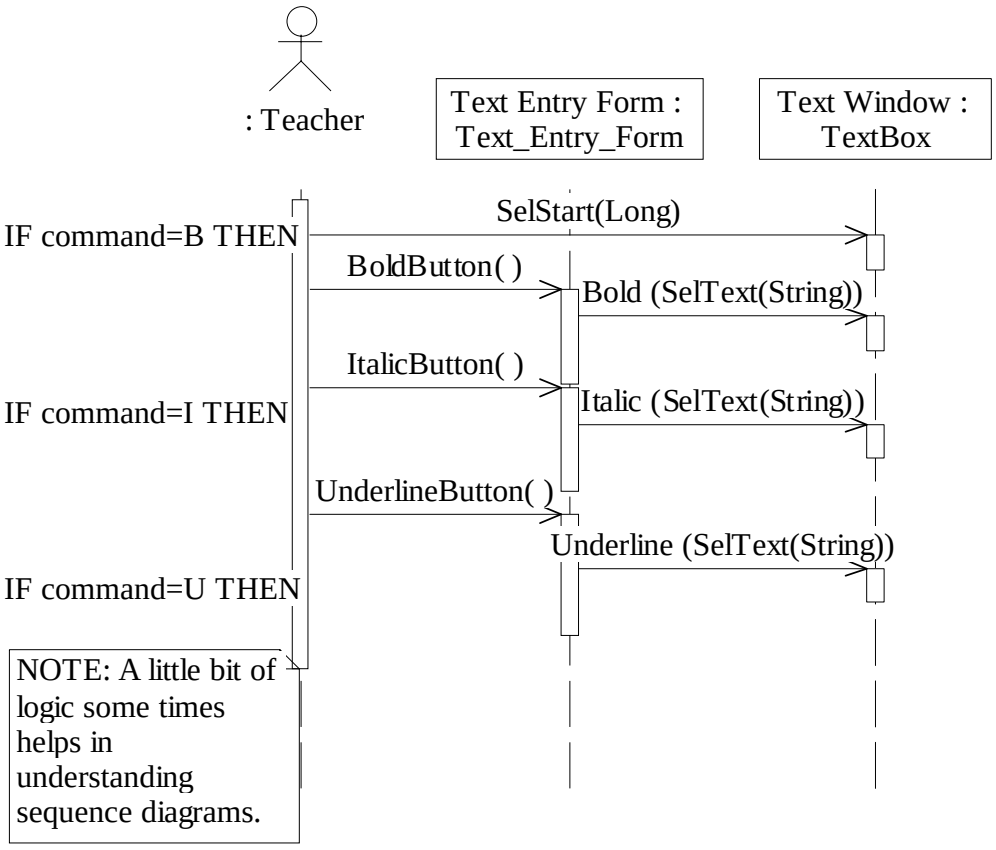

Fig 3.18: Set Bold Italic Underline Sequence Diagram

Use Case Name: Import Text

Use Case Number: ULB10304

Description: Import Text use case is used to import Text that is not in electronic form.

Purpose: Facilitating Text data entry from some book or paper.

Related Use Cases: Set Font name and Color , Set Bold Italic Underline Add Text

Primary Actors: Teacher

Technical Requirements: This requires some Optical Character Recognition.

Type of Actor: Human

UC Spec: Body Section

Pre-condition: (Scanner is available).

Main-Course:

Actor action	System response
1) Teacher commands to import some Text data.	2) System uses OCR application to import Text data.
	3) Imported data is adjusted in between.
	4) The use case ends.

UC Spec: Trailer Section

Question: Why to import Text?

Answer: Importing Text through Scanner leads to fast development and thus saving of time.

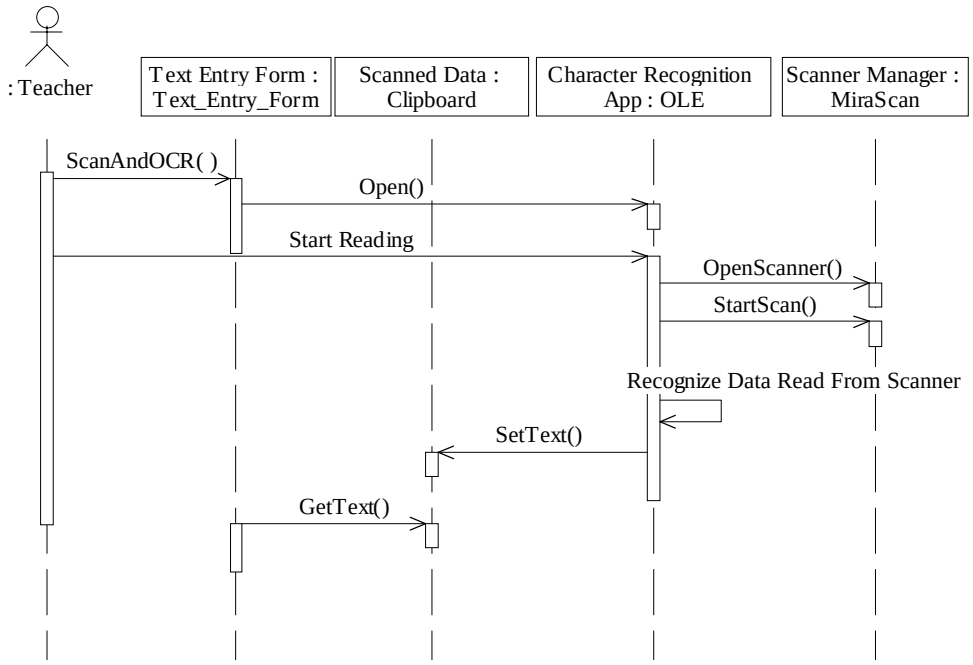

Fig 3.19: Import Text Sequence Diagram

Use Case Name: Scan

Use Case Number: ULB10305

Description: Scan use case is used to enable use of Scanner.

Purpose: It is required for scanning Text and/or Figures data directly from some book or paper/card etc.

Related Use Cases: Import Text

Primary Actors: Teacher

Technical Requirements: Scanner is installed in the system with some OCR application.

Type of Actor: Human

UC Spec: Body Section

Pre-condition: Scanner is online and ready.

Main-Course:

Actor action	System response
1) Teacher places book or paper that is to be scanned.	
2) Teacher commands to start scan.	3) System uses Scanner to read data.
	4) If data is textual then Scanner data is read through OCR application.
	5) The use case ends.

UC Spec: Trailer Section

Question: What is OCR?

Answer: OCR stands for Optical Character Recognition. It is required to identify alphanumeric characters from scanned Text. This enables editing of imported Text also.

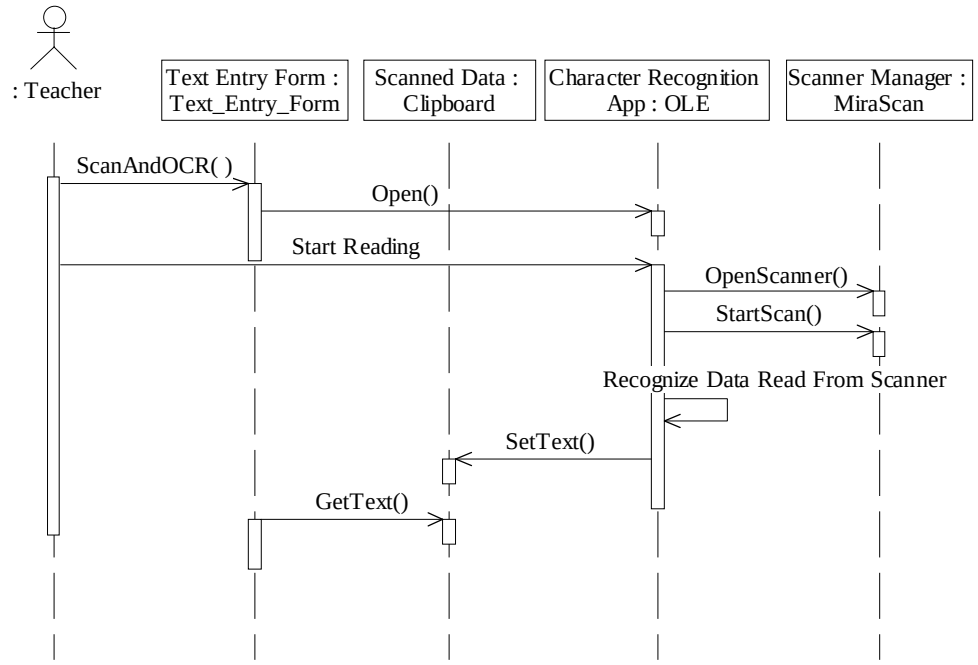

Fig 3.20: Scan Sequence Diagram

3.5 Add Voice Sub Package

3.5.1 Use case Diagram

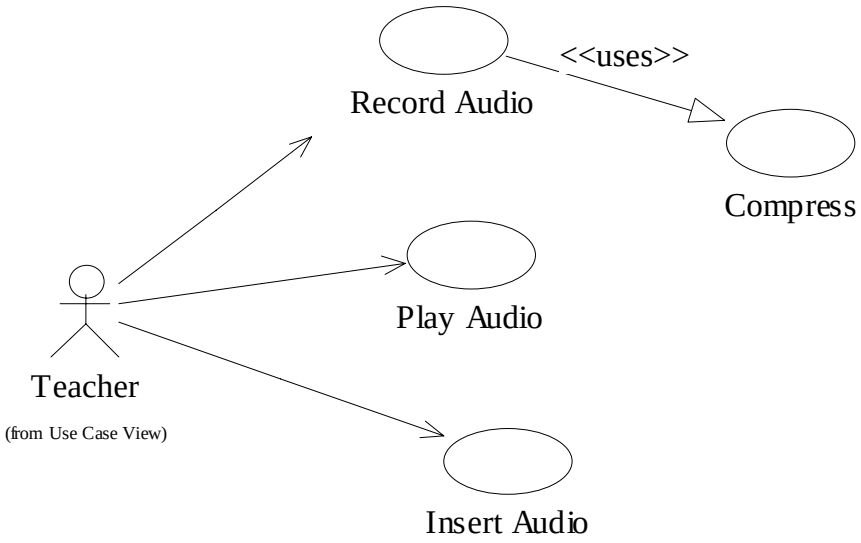

Fig 3.21: Add Voice Sub Package Use Case Diagram

3.5.2 Use Case Specifications and Sequence Diagrams

Use Case Name: Record Audio

Use Case Number: ULB10501

Description: Record Audio use case is used for recording Voice.

Purpose: Prepare Audio files related to Text and Figure data.

Related Use Cases: Play Audio , Insert Audio , Compress

Primary Actors: Teacher

Technical Requirements: Requires Microphone and Speakers.

UC Spec: Body Section

Main course:

Actor action	System response
1) Teacher commands to record Voice.	**3)** System starts recording program.
	4) Show settings to the User.
4) Teacher may change settings.	
5) Teacher speaks in Mic for some duration and then stop recording.	**6)** System stores data in some available supported format.
	7) Compress use case is then used to compress Audio file.
	8) The use case ends.

UC Spec: Trailer Section

Question: What is the length of Audio that can be recorded?

Answer: Basically there is no limit, but it is better to limit Audio content to related material only. After all it will share our 56 kbps pipe during the time slot when it is delivered. Thus for each Browser Screen, that will be delivered to client Screen, comprising of one or two paragraphs may sufficiently require 2,3 minutes Audio.

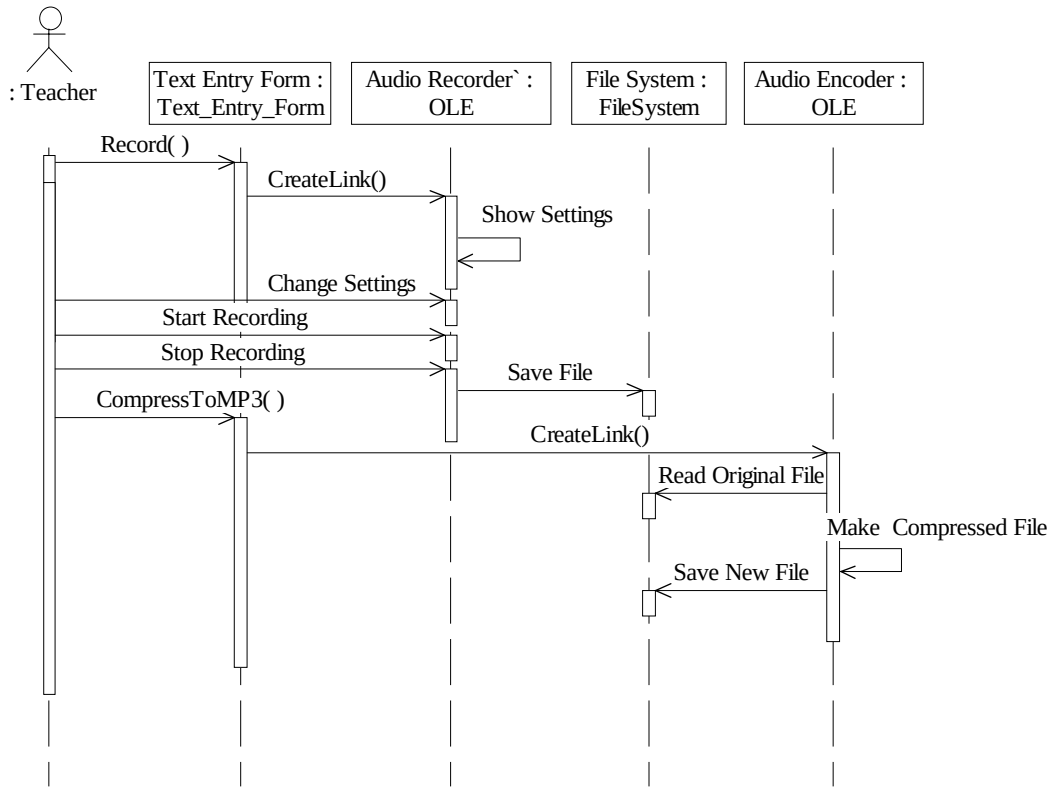

Fig 3.22: Record Audio Sequence Diagram

Use Case Name: Play Audio

Use Case Number: ULB10502

Description: Play Audio use case is used to play Audio content.

Purpose: For listening to added Audio content during development process.

Related Use Cases: Record Audio , Insert Audio , Compress

Primary Actors: Teacher

Technical Requirements: Requires Speakers.

UC Spec: Body Section

Pre-condition: Speakers are switched on. It's better to turn off Microphone during play.

Main-Course:

Actor action	System response
1) Teacher commands to play earlier added compressed recorded Voice.	2) System starts playing Audio file through Speakers.
3) Teacher may decide to stop playing Audio at any time.	4) System stops playing Audio. 5) The use case ends.

UC Spec: Trailer Section

Question: Why to turn off Microphone?

Answer: To eliminate any background environment noise and also to stop re-entering Sound from Speakers to Microphone, which can suddenly start producing amplified noise due to positive feedback effect.

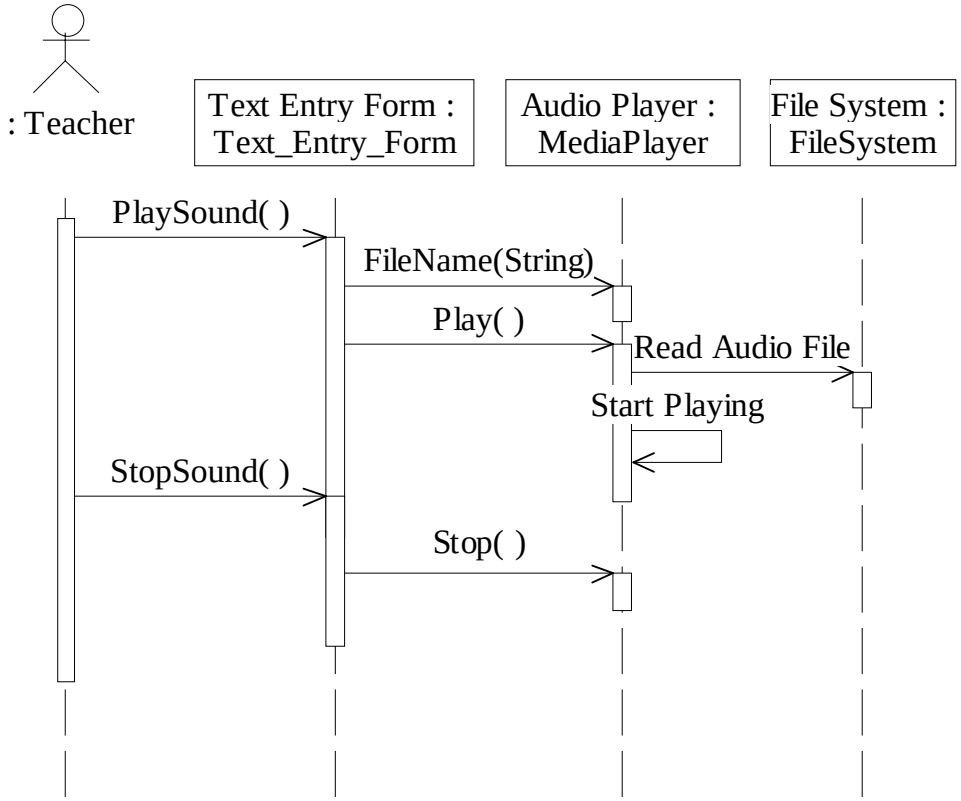

Fig 3.23: Play Audio Sequence Diagram

Use Case Name: Insert Audio

Use Case Number: ULB10503

Description: Insert Audio use case is used to add specific Audio file to some Text.

Purpose: To attach a prerecorded or newly recorded Audio narration.

Related Use Cases: Record Audio , Play Audio , Compress

Primary Actors: Teacher

UC Spec: Body Section

Pre-condition: Some Audio content is available on system.

Main-Course:

Actor action	System response
1) Teacher selects some Audio file from available ones.	2) System manages to attach it with Section data. 3) Display added Audio File name and its location. 4) The use case ends.

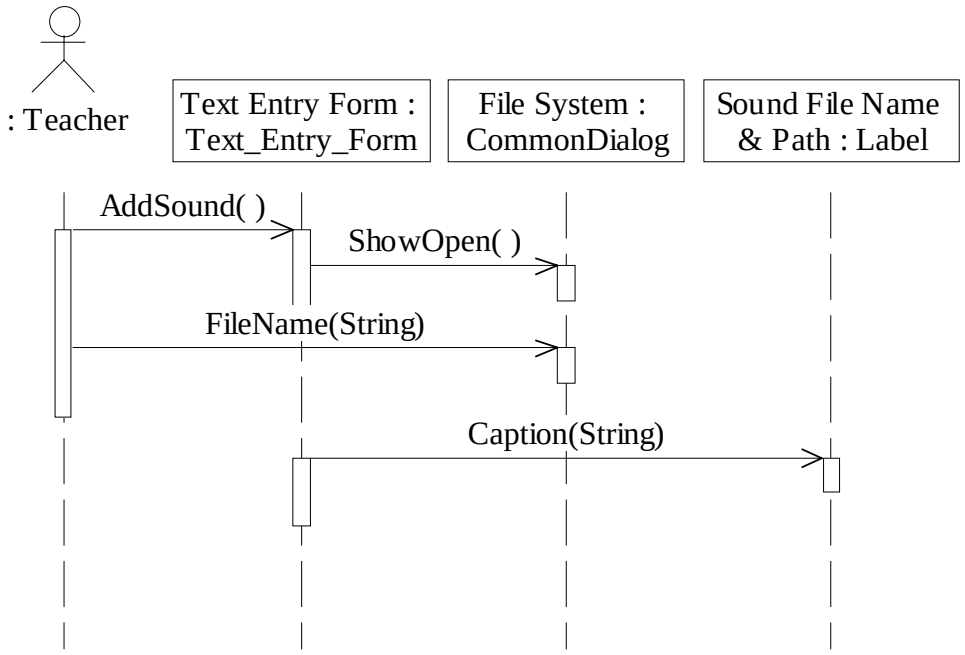

Fig 3.24: Insert Audio Sequence Diagram

Use Case Name: Compress

Use Case Number: ULB10504

Description: Compress use case is used to compress recorded Audio.

Purpose: Compression shrinks size of Audio files.

Related Use Cases: Record Audio , Play Audio , Insert Audio

Primary Actors: Teacher

UC Spec: Body Section

Pre-condition: Some uncompressed Audio content is available on system.

Main-Course:

This use case is an abstract use case. It has a <<uses>> relationship with Record Audio use case. It is used to read a recorded Audio file and then produce a compressed version of original file. It utilizes ‘Mp3’ or some other Audio encoding technique. Compressed file produced is much smaller in size approx 10:1.

UC Spec: Trailer Section

Question:	Why Audio compression is required?
Answer:	Compression of Audio files is required to enable efficient delivery through 56 kbps the Internet bandwidth. We will try to use Encoder that utilizes 8 kbit/s bit rate and 11025Hz frequency, thus achieving small compressed file size.

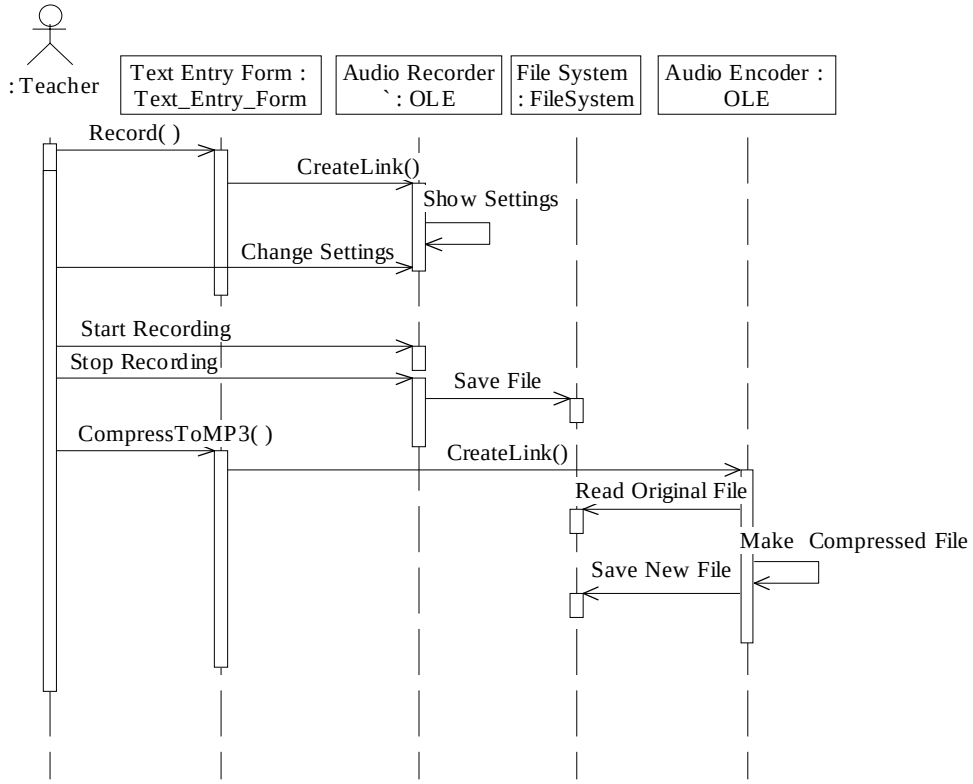

Fig 3.25: Compress Sequence Diagram

3.6 Add and Format Figures Sub Package

3.6.1 Use Case Diagram

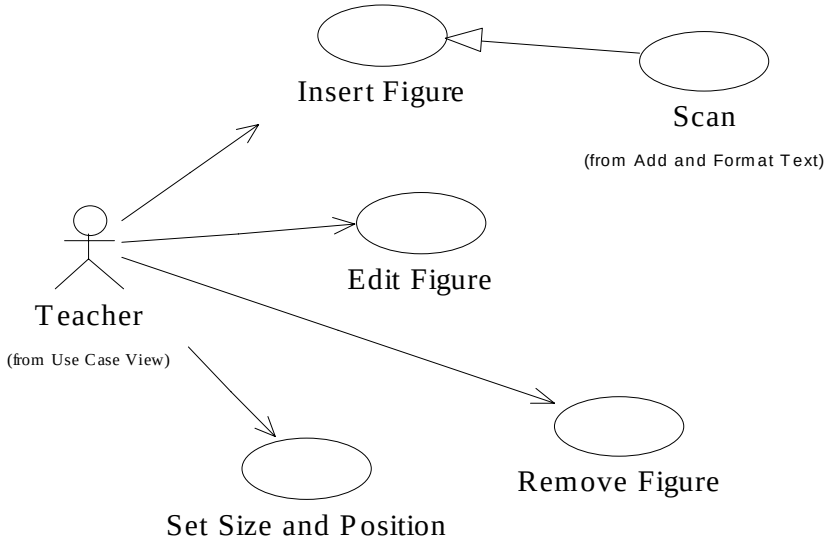

Fig 3.26: Add and Format Figures Sub Package Use Case Diagram

3.6.2 Use Case Specifications and Sequence Diagrams

Use Case Name: Insert Figure

Use Case Number: ULB10401

Description: Insert Figure use case is used to add specific Figure with some Text.

Purpose: To attach a scanned or otherwise available Figure in a Section.

Related Use Cases: Remove Figure , Set Size and Position , Edit Figure

Primary Actors: Teacher

UC Spec: Body Section

Pre-condition: Some Pictures/Images must be available on system.

Main-Course:

Actor action	System response
1) Teacher chooses to add a Figure.	**2)** System show available Picture files.
3) Select some Picture file from available ones.	4) System manages to attach image with Section data.
	6) The use case ends.

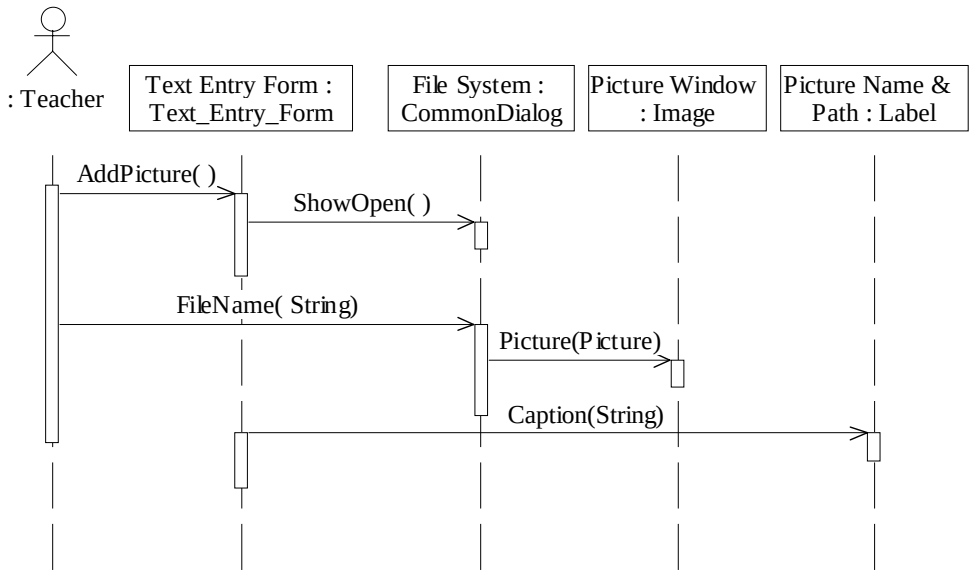

Fig 3.27: Insert Figure Sequence Diagram

Use Case Name: Remove Figure

Use Case Number: ULB10402

Description: Remove Figure use case is used to remove a Figure.

Purpose: To detach an attached Figure from a Section.

Related Use Cases: Insert Figure , Set Size and Position , Edit Figure , Scan

Primary Actors: Teacher

UC Spec: Body Section

Actor action	System response
1) Teacher commands to remove some Figure.	2) System removes Figure from Section data and clears Picture window.
	3) The use case ends.

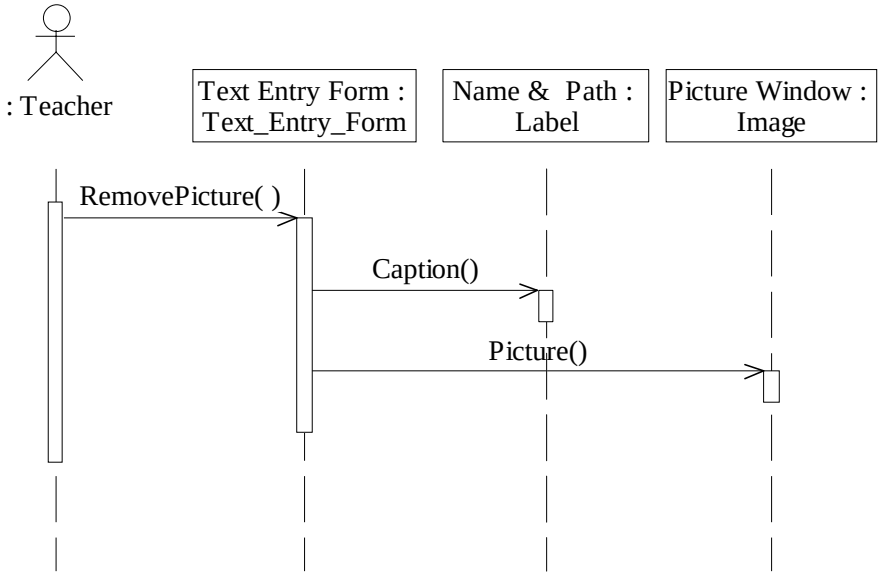

Fig 3.28: Remove Figure Sequence Diagram

Use Case Name: Set Size and Position

Use Case Number: ULB10403

Description: Set Size and Position use case is used to set the size of Figure and it's position.

Purpose: Enables resizing of original Figure as well as it position as it is delivered on Browser Screen.

Related Use Cases: Remove Figure , Insert Figure , Edit Figure , Scan

Primary Actors: Teacher

UC Spec: Body Section

Actor action	System response
1) Teacher sets height and width of Figure.	**2)** System makes adjustments and stores new values.
3) Teacher sets position of Figure.	**4)** System makes adjustments and stores new values.
	5) The use case ends.

UC Spec: Trailer Section

Question: What is meant by “position on Browser Screen”?

Answer: Position is left, right, top or bottom of Text.

Question: Is it possible to set height separately from width?

Answer: Yes, it’s possible. It facilitates reshaping of Figure also.

Question: How to measure height and width?

Answer: It’s better to measure these values in terms of percentage ratios of original Figure. Also it is more convenient for an ordinary user to conceptually visualize double size, half size, ¾ size instead of number of pixels, because users

don’t know every time, how much is the size in pixel or in some other units such as cm, inches etc.

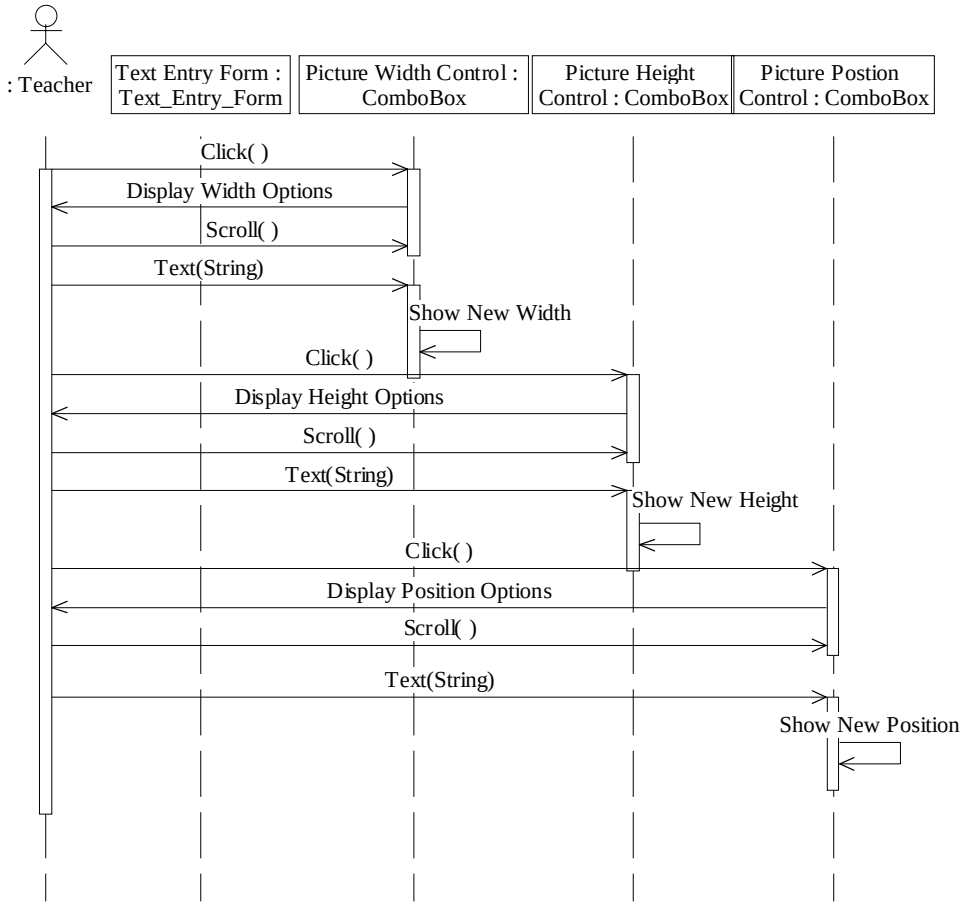

Fig 3.29: Set Size and Position Sequence Diagram

Use Case Name: Edit Figure

Use Case Number: ULB10404

Description: Edit Figure use case is used to make corrections and amend transformations to a Figure.

Purpose: Enables modifying a Figure as required by user. This facilitates changing contents of a scanned Figure

Related Use Cases: Remove Figure , Insert Figure , Set Size and Position , Scan

Primary Actors: Teacher

Secondary Actors: None

UC Spec: Body Section

Main-Course:

Actor action	System response
1) Teacher after adding a Figure through Insert Figure use case decides to modify it.	2) System opens the Figure for editing purpose.
3) Teacher makes changes to Figure using different editing tools such as lines, circles, shapes.	4) System saves changes.
	5) The use case ends.

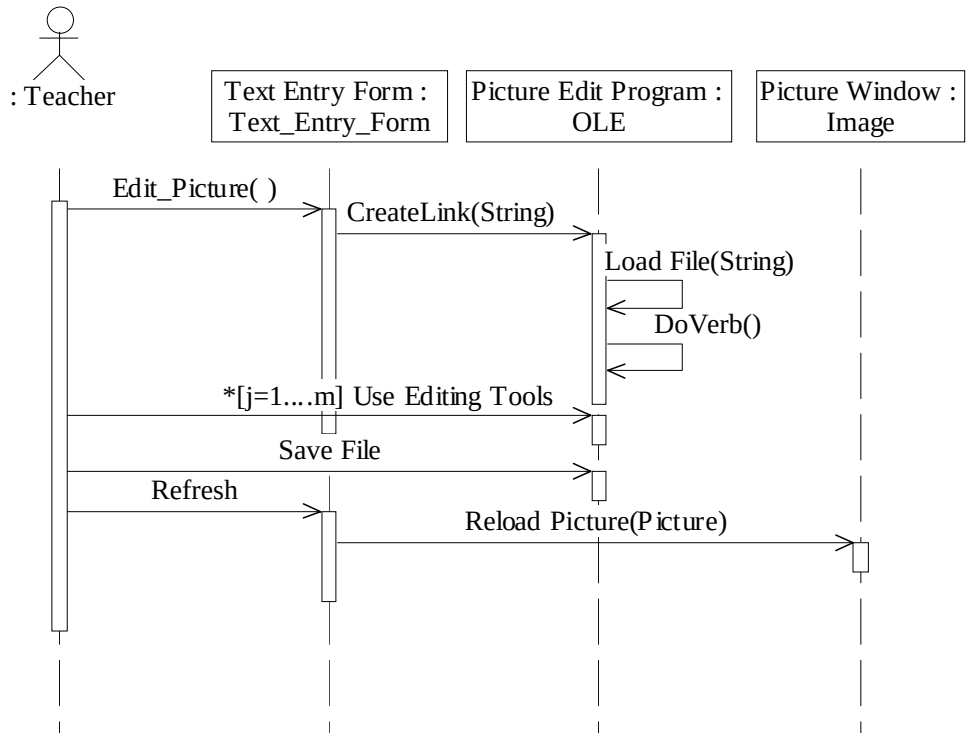

Fig 3.30: Edit Figure Sequence Diagram

3.7 Class Diagrams

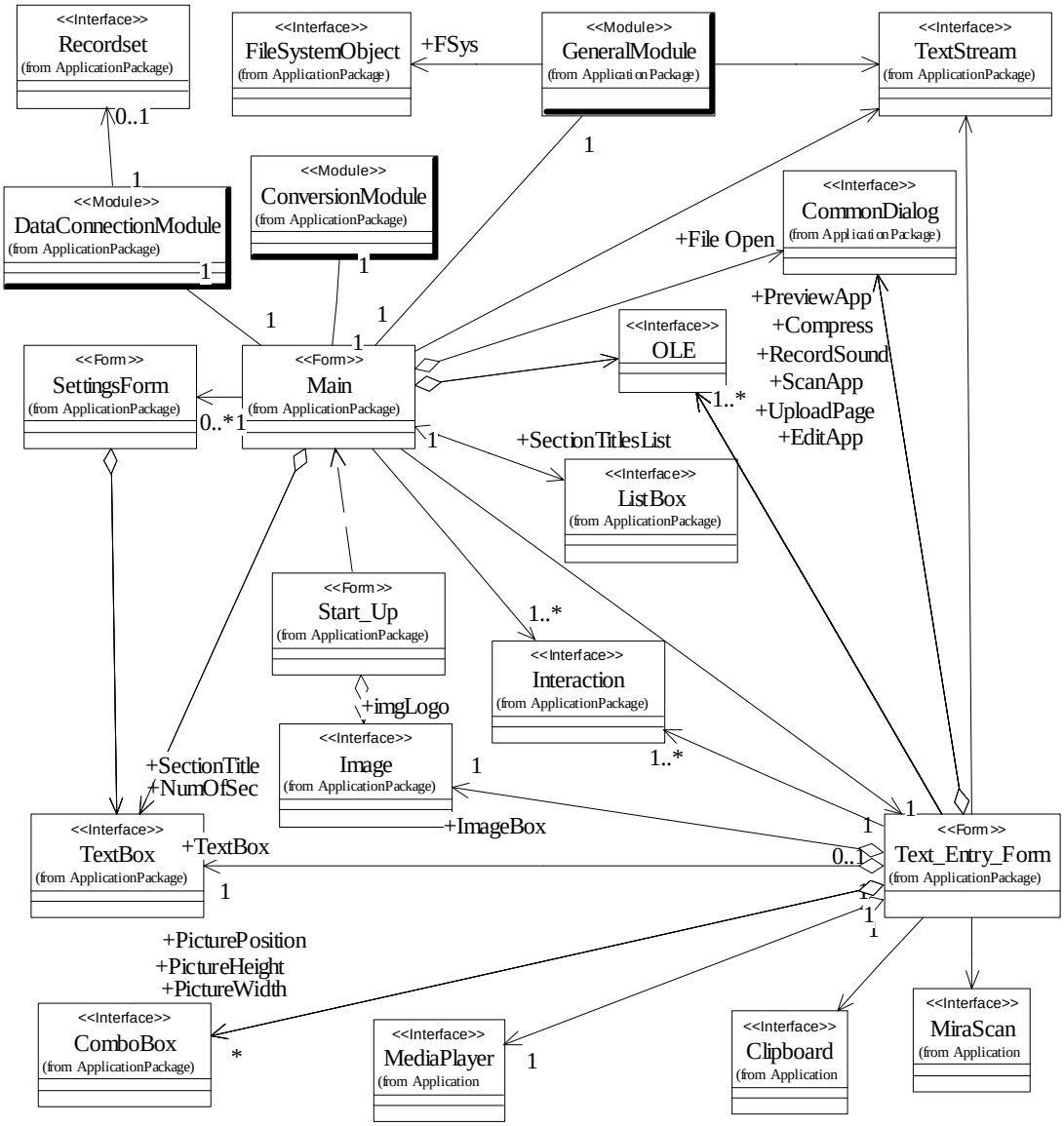

Fig 3.31: Main Class Interaction Diagram (Lecture Preparation Application)

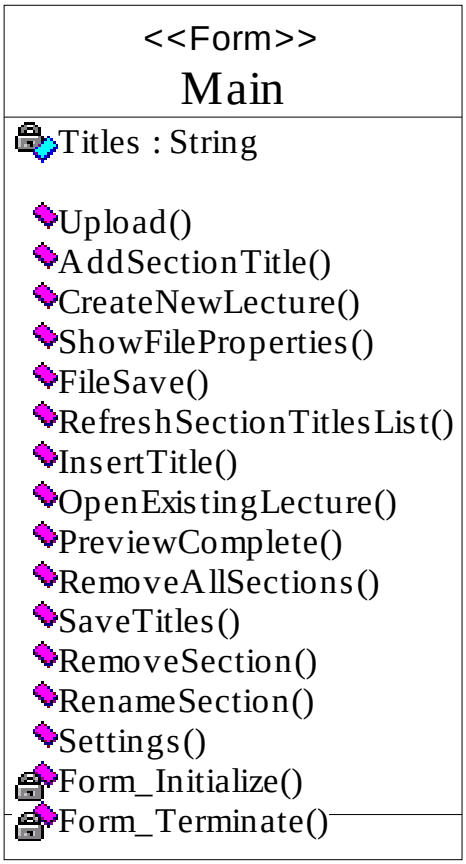

Fig 3.32: Class "Main" Stereotype <<Form>> Attributes and Operations

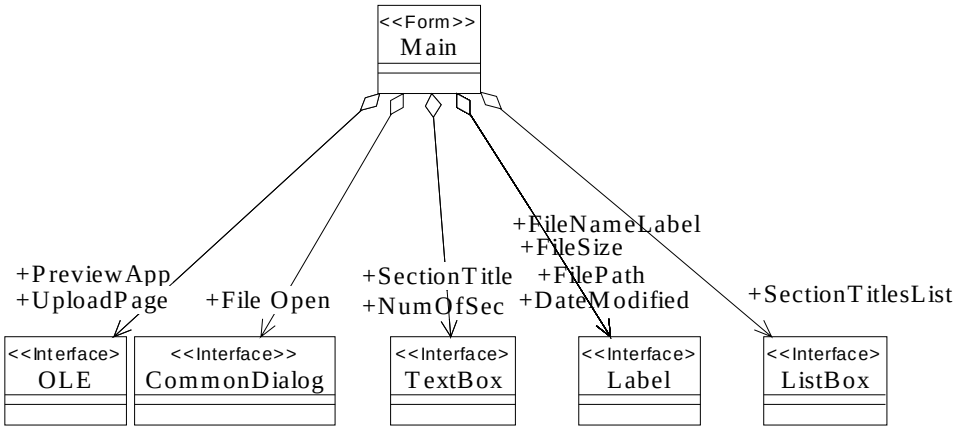

Fig 3.33: Aggregation Relationships of Class: Main Form with other classes

Fig 3.34: Class "Text Entry Form" Attributes and Operations

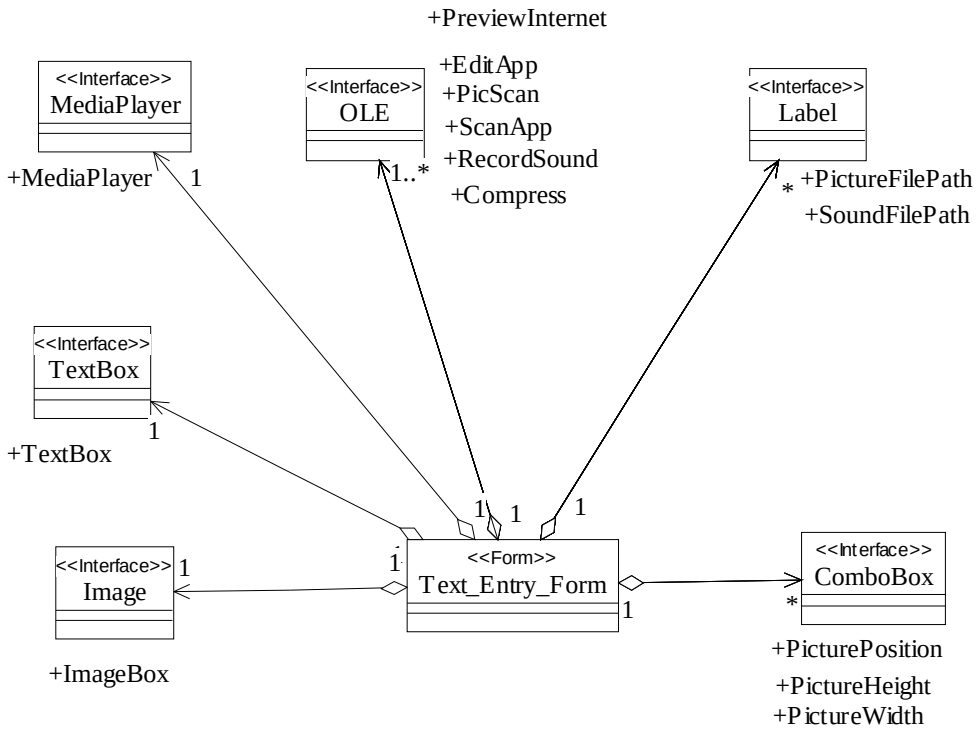

Fig 3.35: Aggregation relationships of 'Text Entry Form' Class

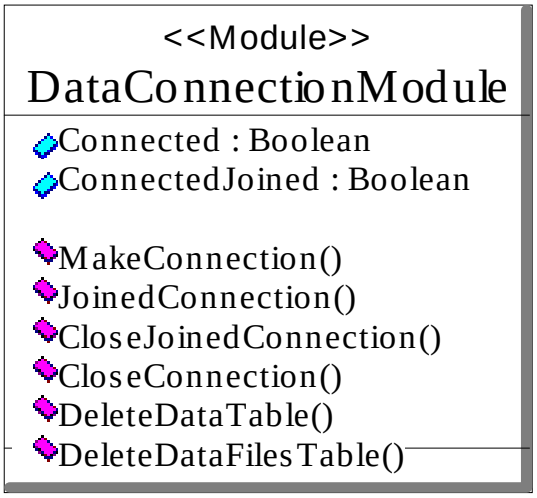

Fig 3.36: Data Connection Module

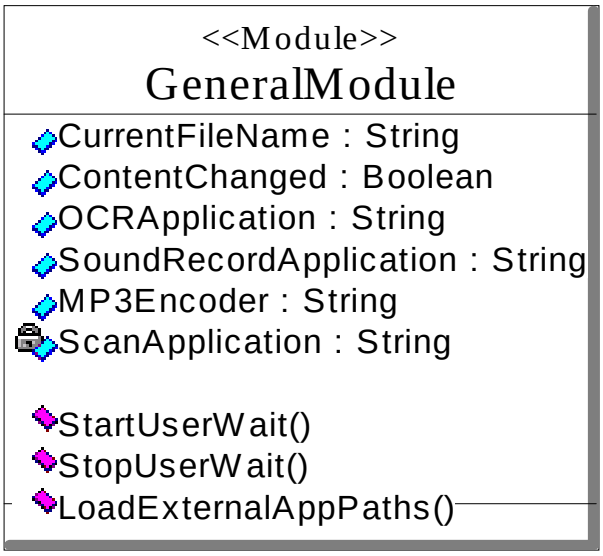

Fig 3.37: General Module

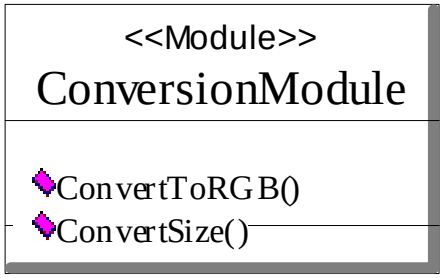

Fig 3.38: Conversion module

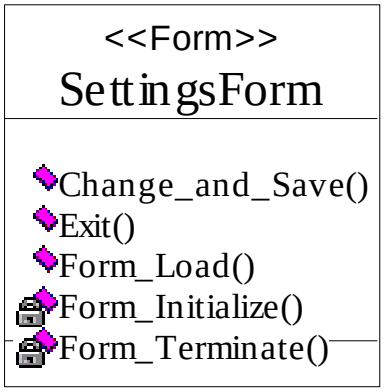

Fig 3.39: Class Settings Form

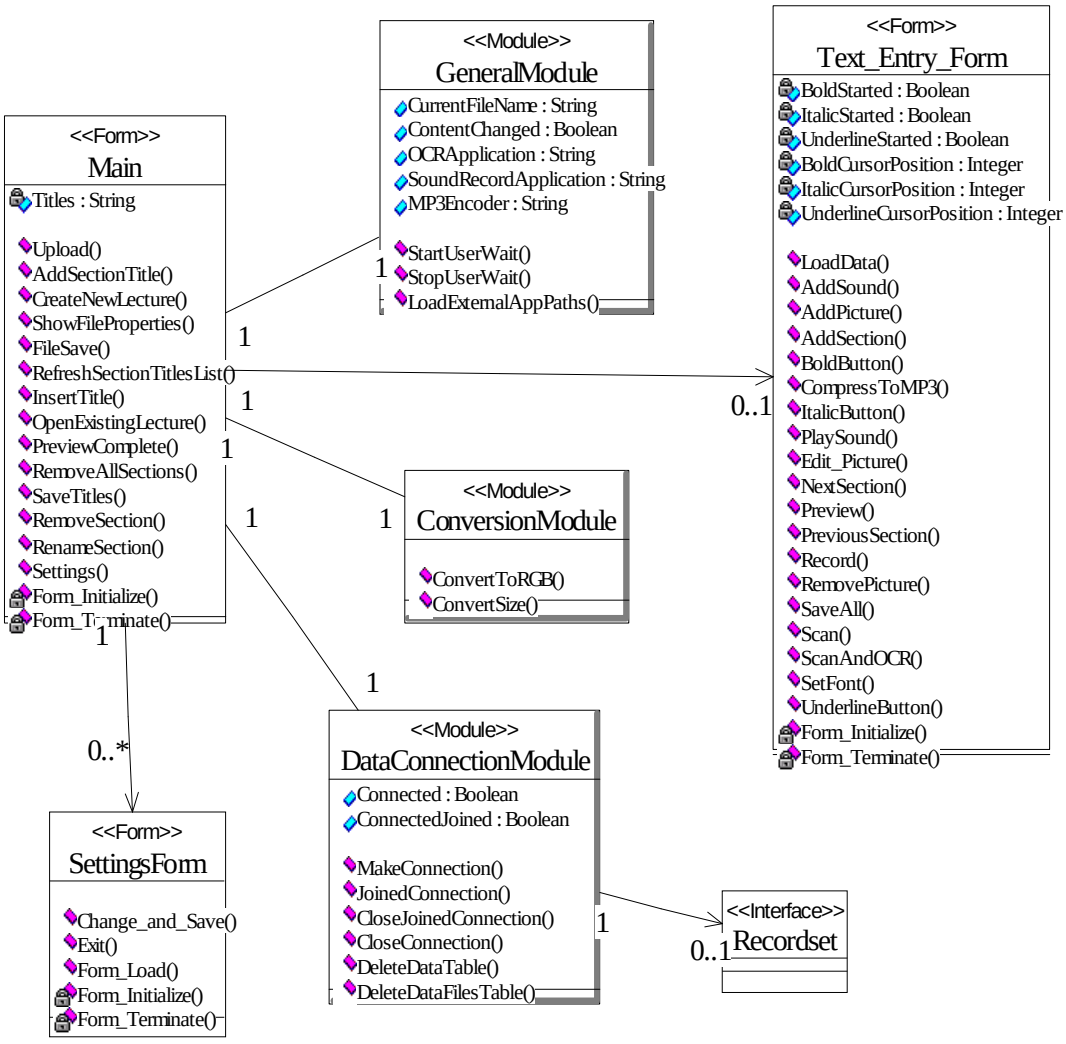

Fig 3.40: Six Important classes in detail and their relationships

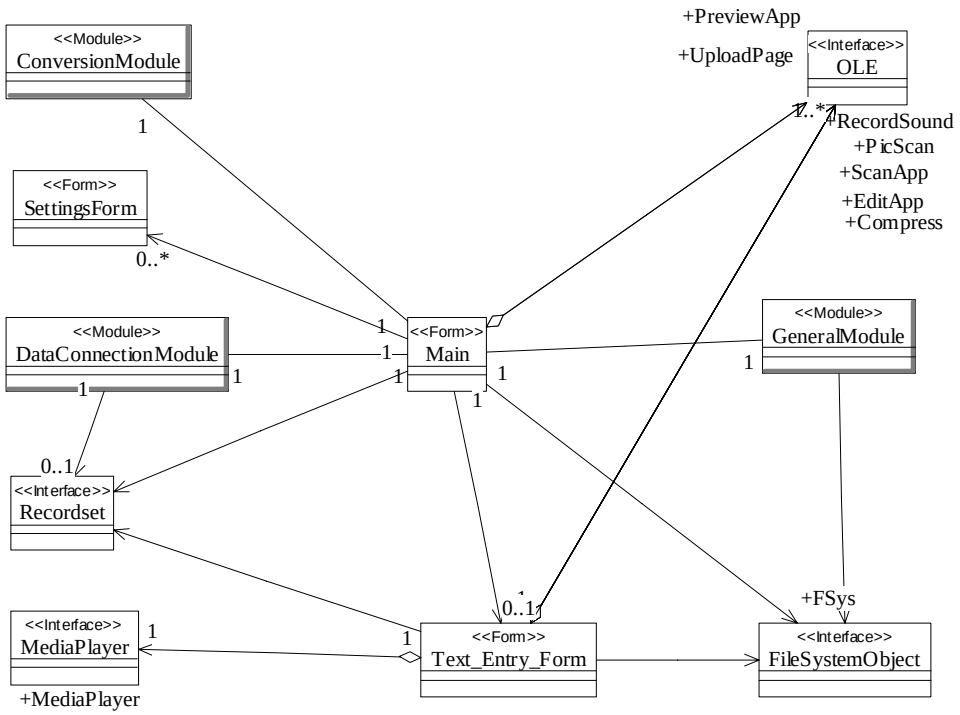

Fig 3.41: Relationships between important classes

3.8 Component Diagram

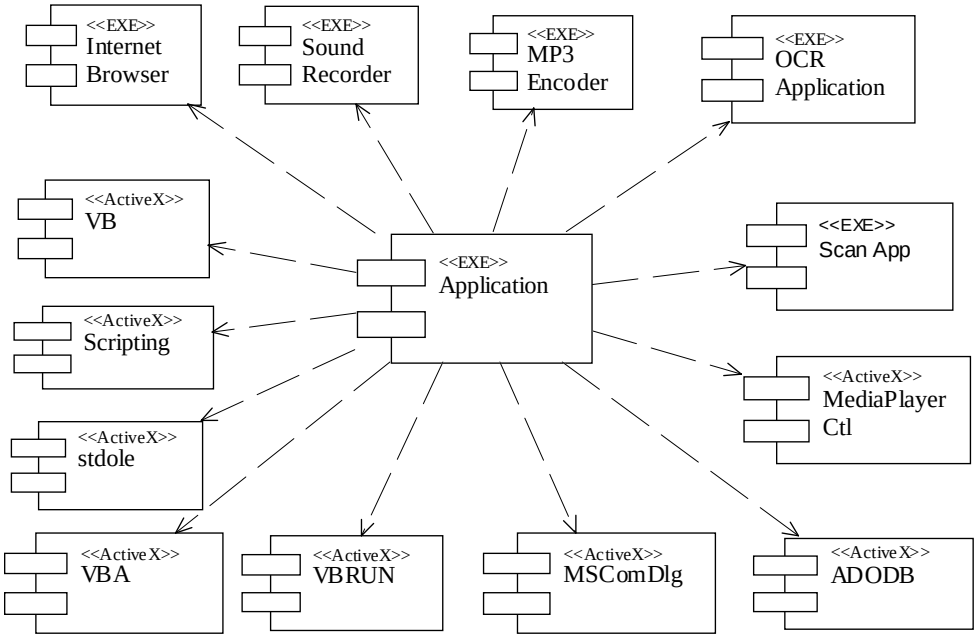

Fig 3.42: Component Diagram of Lecture Preparation Application

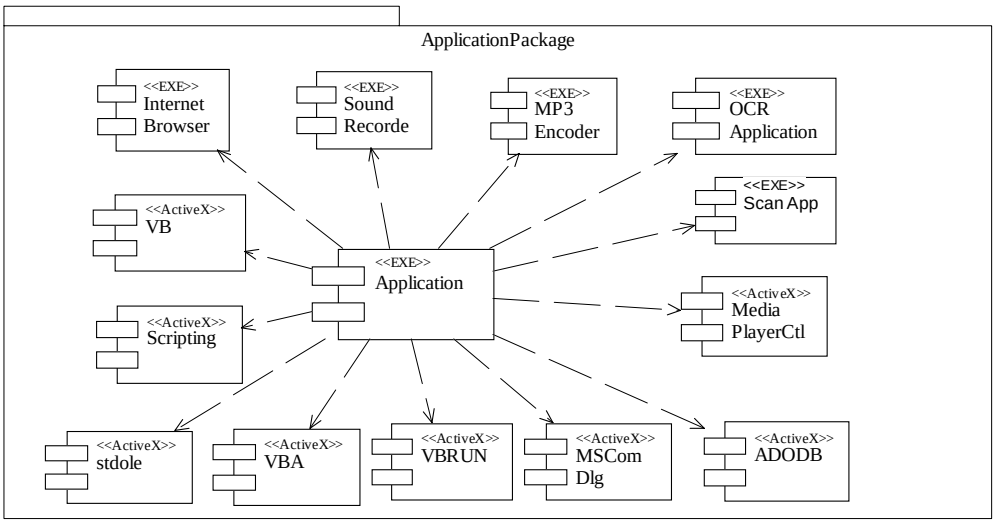

Fig 3.43: Application Package

Chapter 4

Server Side Software

4.1 Purpose of Server Side Software

When the teacher has prepared Lecture on his computer then it must be uploaded to a central University Server prior to conducting Live Class Session. Our Server Side Software will make it easy to automate the process of uploading Lecture and related content. The teacher can upload more than one Lecture simultaneously. The teacher has authority of activating any one of the Lectures in a particular course. The activated Lecture will be available during the next Live Class Session.

4.2 Simultaneous Availability of All Functionalities

Our Distant Classroom is proposed to offer a number of functionalities simultaneously. These include, being able to deliver prepared Lecture material in specific size chunks, Live Query and Response, including Blackboard content, monitor status of online users, and Live Audio and Video Session. Therefore Server Side Software should be capable of providing all these functionalities simultaneously to the users.

4.3 Modes of Classroom

As mentioned above, there are different functionalities that are provided simultaneously. To generalize usage it is better to have a number of

distinct class modes that are available. In a Class Session it may not always be necessary to include Live Audio and Video Session, or it is not necessary to include Live Query and Response facility. Similarly the teacher may like to deliver Lecture only using Live Audio and Video Session without any prepared Lecture material. Therefore to be more generic our Server Side Software should provide option of selecting Class Modes. A number of Class Modes can be defined easily, for example:

- Text, Voice and Figures Only (All Recorded)
- Mode 1 plus Live Query and Response with Blackboard Content
- Mode 2 plus additional Live Video and Audio Session
- Mode 3 with simultaneous Live Video and Audio
- Live Audio and Video Session only

4.4 Uploading Lecture and Related Content

Uploading Lecture material means that the teacher after preparing the Lecture material will transfer it to the Server. Now there are some issues to be resolved. First of all some data file will be required to hold textual content plus information about all other related Picture and Sound files. Now if we assume that upload process is to be done by selecting files then the user will soon become annoyed because a single Lecture may comprise 25-30 Picture and Sound files or even more. It is very difficult to remember individual locations of all files, and then manually select each file.

In order to simplify this process our Software should be capable of uploading main Lecture file and its related Picture and Sound files automatically, without intervention of the user. But this will require client side file system access to be provided to Server Side Software. First of all main file will be uploaded and the Server Software will prepare a list of related files which will be uploaded automatically without manual user selection. [17,18]

4.5 Activating Lecture

The teacher can upload more than one Lecture sequentially. But how the Server Side Software will know which Lecture is to be delivered during Live Class Session. There must be some discrimination between Lectures; an active Lecture and an inactive Lecture. The teacher has authority of activating any one of the Lectures in a particular Course. Thus he will determine which Lecture will be available during the coming Live Class Session.

4.6 Role of DB Server

There will be a Central Database Server on the Server Side. This Server will hold a separate Database for each Subject. Information which is stored here will comprise of list of all Lectures uploaded for a particular Subject, active Lecture content, Live Query and Response content, registered

users information, users online status and activation information. There is a need of fast Database Server to provide good service to online users.

4.7 Optimization: Making Fast Access

When any specific Lecture after being uploaded is activated its contents are transferred to Central Database Server. But to provide fast access, it is better that standard HTML/ASP pages are generated on Server file system for Lecture material in addition. Also for the same reason of providing comparatively faster access, we are not storing Picture and Sound data in the Database. This is a good optimization technique because we are constrained by low bandwidth Internet. We have to be very conscious in deciding the content and it's delivery format throughout development process.

4.8 Live Audio and Video Session

Live Audio and Video is a facility for the teacher to deliver Audio and Video content complementary to the Lecture material or in a fully separate Session. There must be some options for example:

- Audio only
- Video only
- Audio and Video (Combined)

4.9 Need For Automatic Announcement Of Live Audio Video Session

Another necessary requirement is to provide such functionality to the teacher that he can announce when he desires to start Live Audio and Video Session during a Class Session. This announcement should reach all the students so that they divert their attention to Live Audio and Video Session. Without this functionality the students will have no indication as to when they have to attend Live Audio and Video Session.

4.10 Query and Response

Query and Response is an integral feature of our Distant Electronic Classroom. This functionality will provide the students capability of asking their typical new questions relevant to a particular Section. The teacher will answer question in textual format and/or with Blackboard content taken through Camera. All students viewing a particular Section will also view related Queries relevant to that particular Section. As and when a student desires he/she can view textual and/or Blackboard demonstration in Response.

4.11 Live Teacher's Board Content

In Response to a particular Query a teacher may want to answer by drawing a sketch/graph etc on Board as is common in our normal classrooms. Now our Software should be capable of automatically selecting new Blackboard content file and sending it to the main Server from where it

becomes viewable by all desirous users. Therefore there must be some means to simplify and automate this process.

4.12 Merging Live Interaction with Lecture Content

When the live interactive Class Session will be over, the teacher if he wishes may need to combine the live interactive content with prepared Lecture material. The prepared Lecture material includes Text, Figures and recorded Audio that was developed earlier by the teacher. Live interactive content will include Query and Response in textual form plus Picture of the Board content. There must be some means to distinguish at which location in the Lecture the Query was raised, so that when the teacher orders to merge it with the content each Query and it's Response is placed appropriately in between relevant Section boundaries.

4.13 Browsing Old Lectures

The students may desire to view Lectures after the Class Session time is over. Therefore all the Lectures must be available for browsing by the students at any time. Here one more issue arises; that two separate Lecture formats should be available for browsing, one with Queries and Responses and the second without such Queries and Responses.

4.14 Monitoring Status of Students

Our Server Side Software should provide facility to the teacher to monitor when a student comes online. In addition there must be some

indication about where in the Lecture material a student is viewing contents at a particular instant. This will facilitate the teacher in conducting Live Class Session.

4.15 General Purpose Architecture Development For All Subjects.

We are developing general-purpose Distant Electronic Classroom Software. Therefore our Server Side Software should be capable of maintaining a number of subject's data simultaneously on the Database Server as well as on file system. Specific folders for each Subject and it's Lecture(s) will be generated and maintained by our Software. This process should be automated by the Server Side Software because there may arise serious issue of similar Picture and Sound file names, issue of separating contents of one Lecture form other, and issue of maintaining a record of where relevant content of a particular Lecture of a specific Subject lies. All these issues will become troublesome, once teachers will start uploading a number of Lectures, if these are not taken care early.

4.16 Registering New Students

We will assume that it is the responsibility of the Administrator to register new users. Therefore we have to provide some suitable interface for the Administrator for this purpose. Administrator will register a new user in any Subject, his user type; as Teacher, Student or Silent Student.

4.17 Subject Contents on Server

Lecture file with related Picture and Sound files will be uploaded to the Server. When a particular Lecture is activated its contents are transferred to Database Server and web pages are also generated for faster access. After class is over, a merged version of Lecture is generated. Therefore Lecture content on Server Side will consist of Database content, Picture and Sound files, Web Pages and Board images. Therefore Lecture content will be scattered on the Server at different locations during Live Class Session but will be combined and modified afterwards.

4.18 Role of Video Server

Live Audio and Video Session requires a Video Server on Server Side. Our Software should be capable of acquiring Live Audio and Video content taken through the Camera, processed at the teacher's computer and then routed to the Server. Audio and Video content that arrives at the Server is streamed to all connected clients. The clients receive the media in real time, and without having to wait for clips to be downloaded. [19]

4.19 Dual use of Camera

Camera connected with the teacher's computer will serve two purposes. First it will be used to take static Board content during Live Query and Response. Secondly Camera is used to take live Video during Live Audio and Video Session.

Chapter 5

Analysis and Design of Server Side Software

5.1 Use Case Hierarchy {Server Side Software}

ULB20000	Server Side Package	(Package)
ULB20100	Common Functionalities	(Sub Package)
ULB20101	Log in	(Concrete)
ULB20102	Join Class	(Concrete)
ULB20103	View Prepared Lecture	(Concrete)
ULB20104	Exit	(Concrete)
ULB20105	Browse Old Lectures	(Concrete)
ULB20106	Log out	(Concrete)
ULB20200	Functionalities For Teacher	(Sub Package)
ULB20201	Reply To Query	(Concrete)
ULB20202	Start Live Audio & Video Session	(Concrete)
ULB20203	Monitor Status of Online Users	(Concrete)
ULB20204	Upload Lecture	(Concrete)
ULB20205	Activate Lecture	(Concrete)
ULB20300	Functionalities For Student	(Sub Package)
ULB20301	Ask A Query	(Concrete)
ULB20302	View Live Query & Response	(Concrete)
ULB20303	Attend Live Audio & Video Session	(Concrete)
ULB20400	Functionalities For Administrator	(Sub Package)
ULB20401	Register Users	(Concrete)

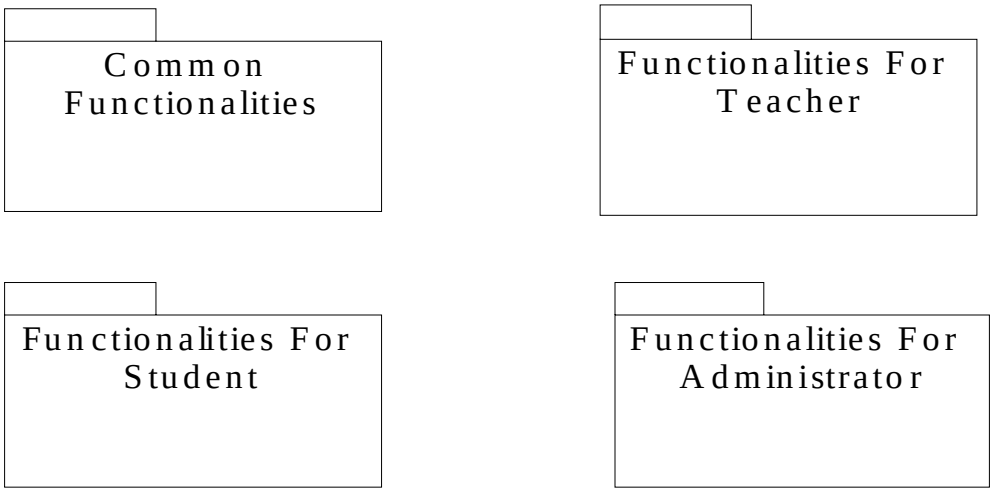

Fig 5.1: Server Side Software Functionalities

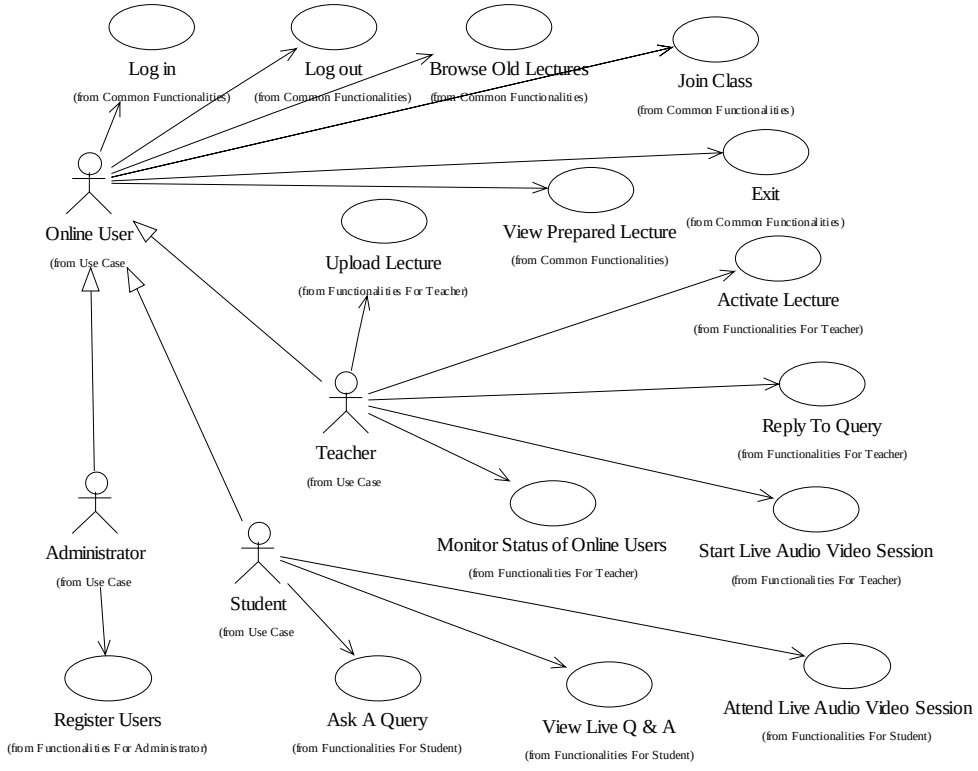

Fig 5.2: Server Side Software Main Use Case Diagram

5.2 Common Functionalities

5.2.1 Use Case Diagram

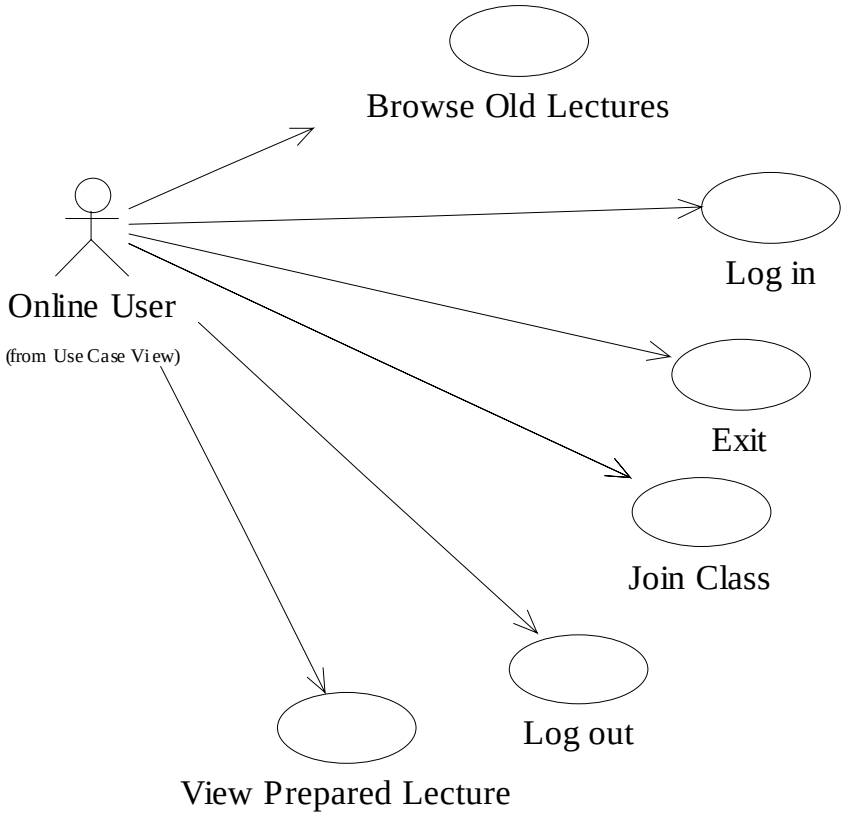

Fig 5.3: Common Functionalities Use Case Diagram

5.2.2 Use Case Specifications and Sequence Diagrams

Use Case Name: Log in

Use Case Number: ULB20101

Description: Log in use case is used to log in to system.

Purpose: Checks whether a user has authentication to enter a particular Subject area.

Related Use Cases: Log out

Primary Actors: Student, Teacher, and Administrator

UC Spec: Body Section

Actor action	System Response
1) User launches web Browser.	
2) User types address in location bar of web Browser.	3) System displays '*default page*'
4) User enters his user name, password , subject and submits information.	5) System authenticates user. 6) If user is an authenticated user then system opens student Screen for the particular Subject. Else alternate flow A1 7) The use case ends.

Alternate Flow A1

Actor action	System Response
	1) If user is not an authenticated user then system redisplays '*default page*' plus it displays an '*error message*'. 2) The use case ends.

UC Spec: Trailer Section

Question: Why Subject is to be entered when logging?

Answer: We are developing a general Classroom that is expected to support multiple Subjects after it is deployed.

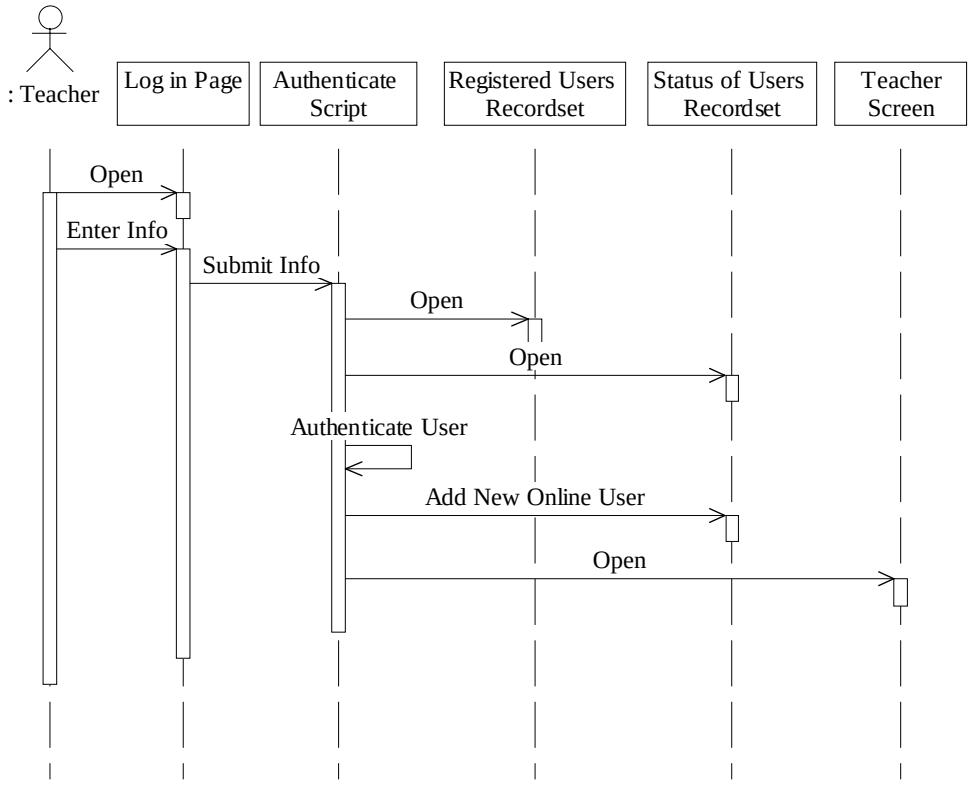

Fig 5.4: Sequence Diagram for Logging Teacher

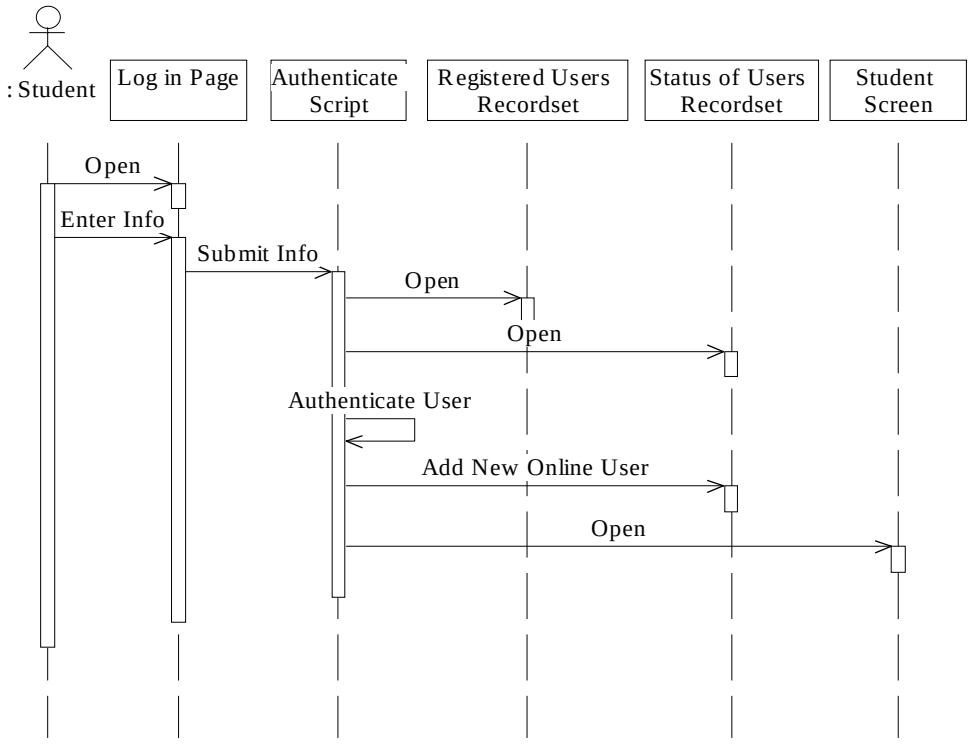

Fig 5.5: Sequence Diagram for Logging Student

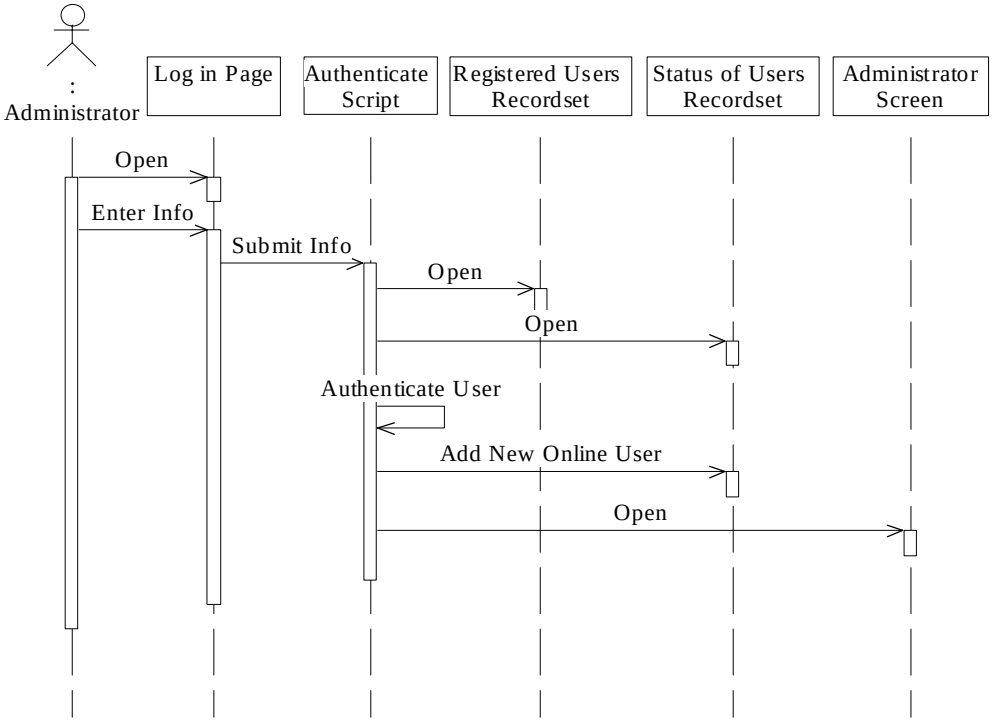

Fig 5.6: Sequence Diagram for Logging Administrator

Use Case Name: Join Class

Use Case Number: ULB20102

Description: Join Class use case is used to join an active Classroom.

Purpose: Users after logging to Server have opportunity of attending a Live Class Session by entering Classroom.

Related Use Cases: Exit

Primary Actors: Student, Teacher

UC Spec: Body Section

Pre-condition: User is Logged in and a Class Session is being conducted.

Main-Course:

Actor action	System Response
1) User commands to join Classroom.	3) System opens Classroom designed for specific user in preset mode. 4) System opens Lecture window, Live Queries window, Ask New Query window. 4) The use case ends.

UC Spec: Trailer Section

Question: What is a Classroom Mode?

Answer: To generalize usage it is better to have a number of distinct Class Modes that are available. A Class Session may not always necessary include Live Audio and Video Session, or it is not necessary to include Live Query and Response facility. We are designing five modes:

- Text, Voice and Figures only (All recorded)
- Mode 1 plus Live Query and Response with BlackBoard Content
- Mode 2 plus additional Live Video and Audio Session
- Mode 3 with simultaneous Live Video and Audio
- Live Audio and Video Session only

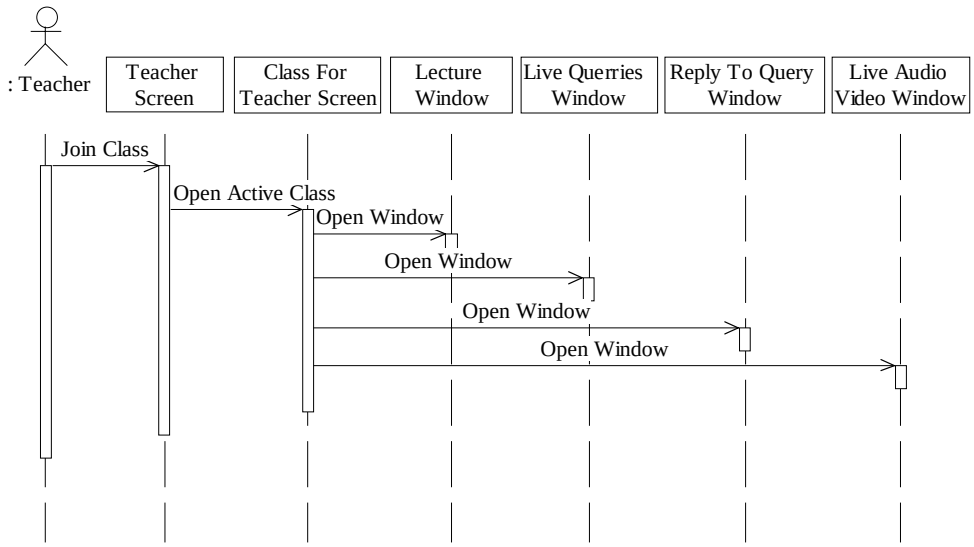

Fig 5.7: Sequence Diagram For Join Class For Teacher

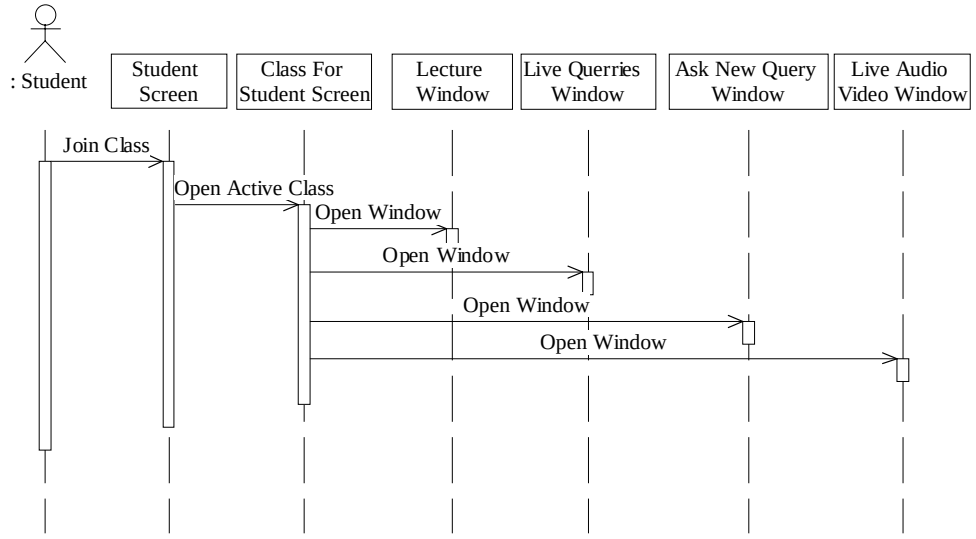

Fig 5.8: Sequence Diagram For Join Class For Student

Use Case Name: View Prepared Lecture

Use Case Number: ULB20103

Description: View Prepared Lecture use case is used to display prepared Lecture material.

Purpose: This use case is for properly displaying Section Screens to users in fixed amount of data chunks including Text, Pictures and recorded Audio.

Related Use Cases: Ask A Query , View Live Query and Response

Primary Actors: Student, Teacher

UC Spec: Body Section

Main-Course:

Actor action	System Response
1) User joins Live Class.	2) System opens active Classroom in a specific Subject. 3) System displays prepared Lecture material with limited. amount of Text, Picture and recorded Audio.
4) Users reads and listens to prepared Lecture. User browse forward and backward through Lecture material.	5) System displays *next* or *previous* content and records location of user. 6) The use case ends.

UC Spec: Trailer Section

Question: Why location of a student is recorded?

Answer: There is a need to maintain status of a student's progress through Lecture material to provide information to the teacher.

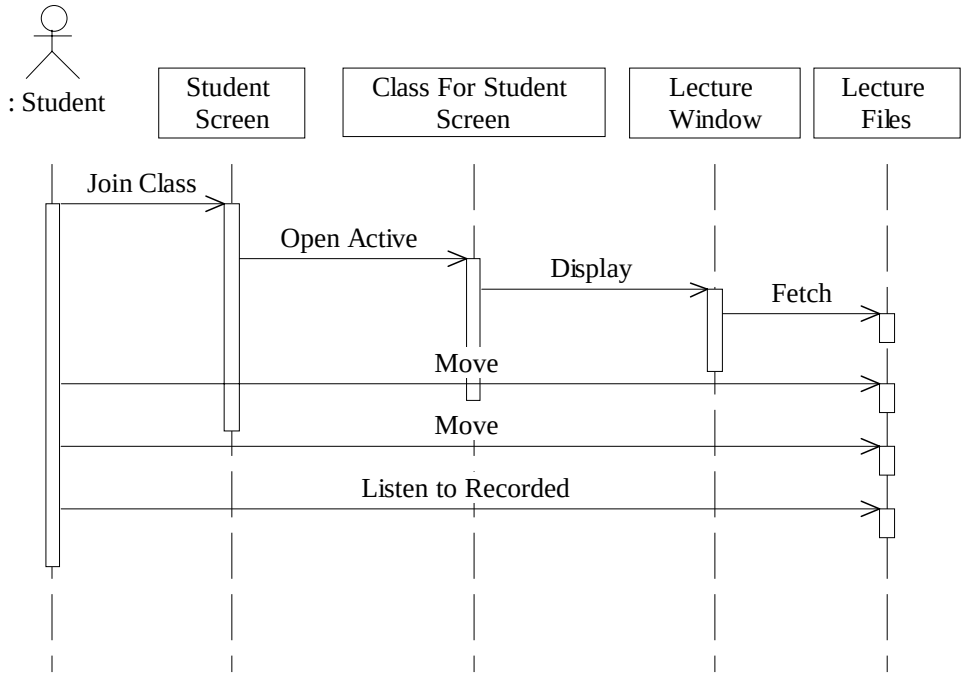

Fig 5.9: Sequence Diagram For View Prepared Lecture in Classroom

Use Case Name: Exit

Use Case Number: ULB20104

Description: Exit use case is used to exit user from the Classroom.

Purpose: Allows user to leave the Classroom at his own will.

Related Use Cases: Join Class

Primary Actors: Student, Teacher

UC Spec: Body Section

Pre-condition: Class is being held.

Main-Course:

Actor action	System Response
1) User commands to exit from Classroom.	2) System close Classroom window. 3) System shows specific user's Screen (that offers other functionalities for user). 4) The use case ends.

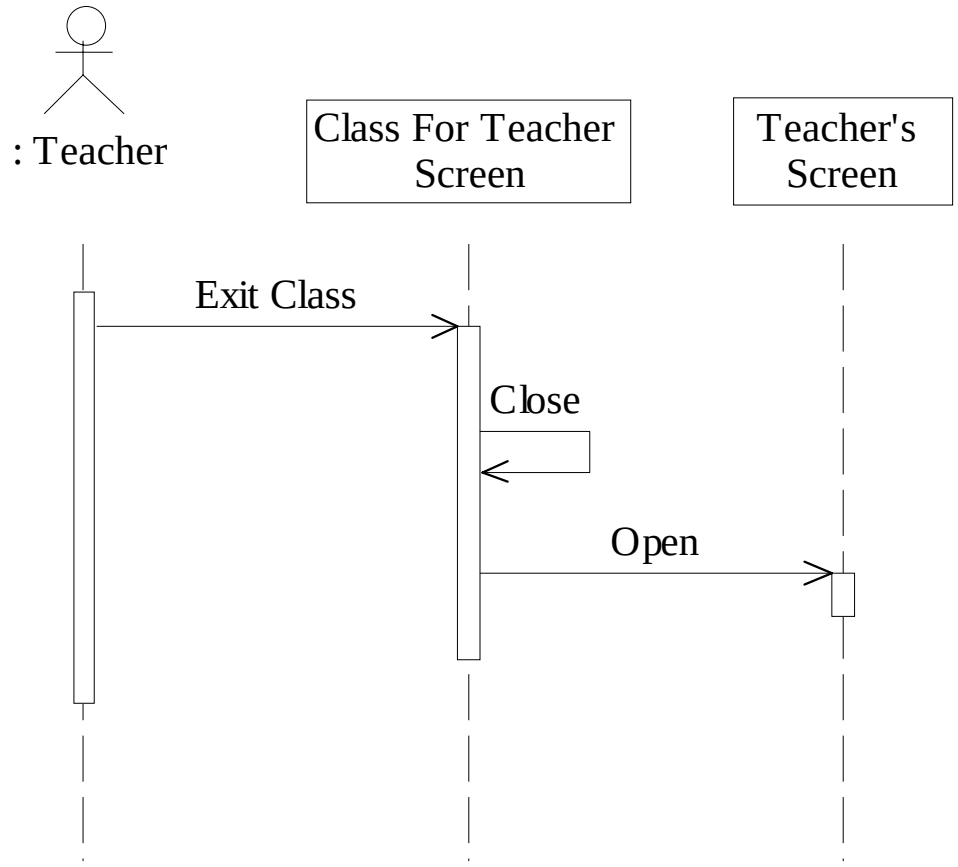

Fig 5.10: Sequence Diagram for Exiting Teacher From Classroom

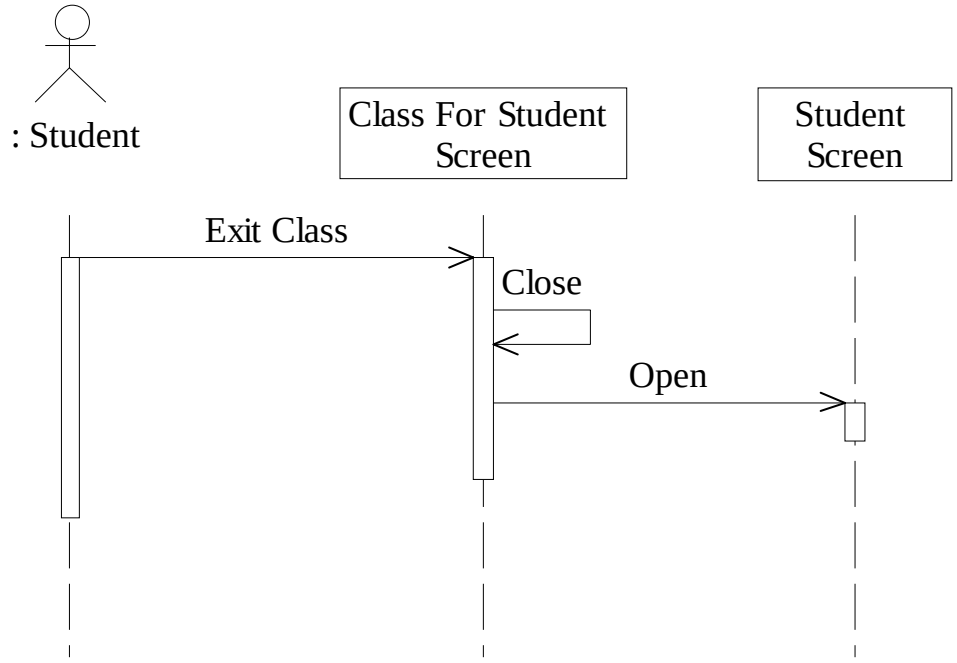

Fig 5.11: Sequence Diagram for Exiting Student From Classroom

Use Case Name: Browse Old Lectures

Use Case Number: ULB20300

Description: Browse Old Lectures use case provides functionality of browsing all Lectures.

Purpose: To provide facility to users to view all Lectures. This facilitates viewing Lecture material with or without Queries when there is no Class being conducted.

Related Use Cases: View Prepared Lecture

Primary Actors: Student, Teacher

UC Spec: Body Section

Pre-condition: Lecture material is properly loaded on Server and is available for viewing.

Main-Course:

Actor action	System Response
1) User logs in to a particular Subject main page.	
2) User commands to view Lectures.	3) System shows list of Lectures and option of with or without Queries.
4) User selects a Lecture to view. User selects option of Queries content.	5) If 'without Queries' option is selected then the system deliver Lecture content only.
	6) The use case ends.

Alternate Flow A1

1) If User selects with ‘Queries option’ system shows Lecture material intertwined with Queries and Responses in all Sections.
2) System also shows black Board images in case a specific Response involves it.

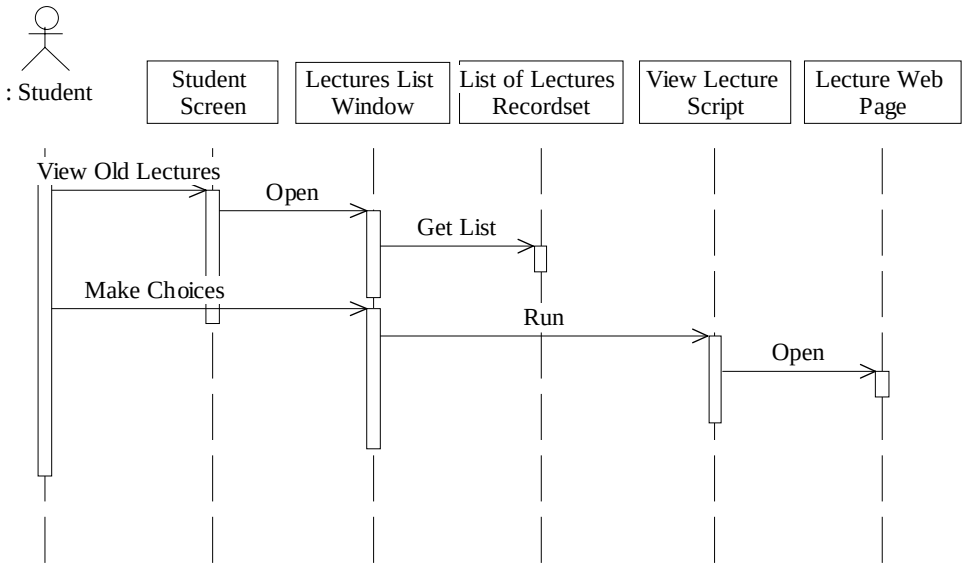

Fig 5.12: Sequence Diagram for Browse Old Lectures

UC Spec: Header Section

Use Case Name: Log out

Use Case Number: ULB20106

Description: Log out use case is used to log out from system.

Purpose: Allows user to log out from system. User name is removed from on line users list.

Related Use Cases: Log out

Primary Actors: Student, Teacher, and Administrator

UC Spec: Body Section

Pre-condition: User is logged in

Main course:

Actor action	System Response
1) User commands to log out from within the student Screen.	2) System finds user entry in online user's list, and removes it.
	3) System shows logout message to user.
	4) The use case ends.

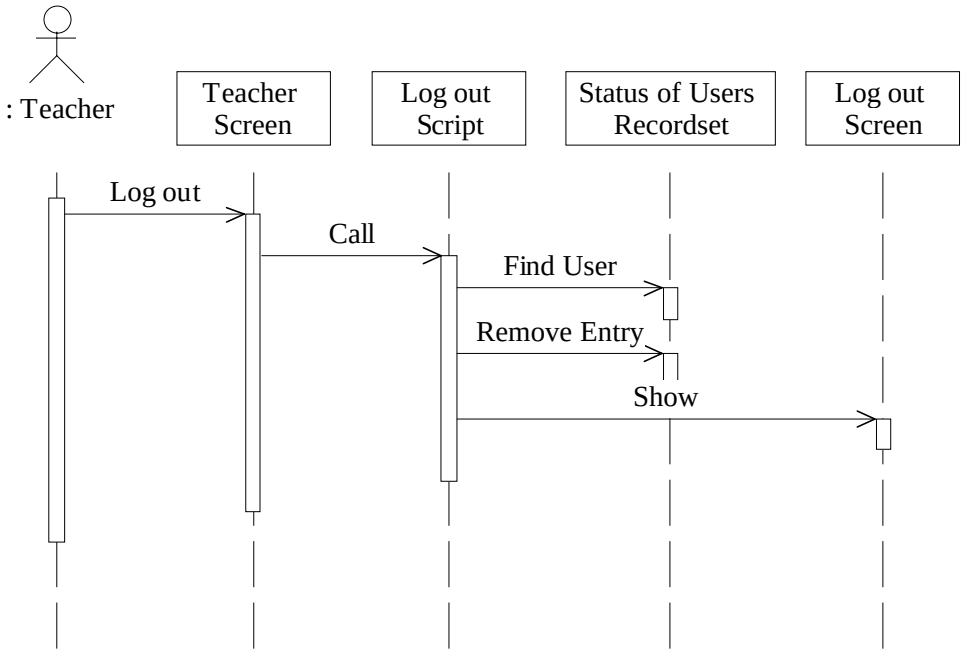

Fig 5.13: Sequence Diagram for Logging out Teacher

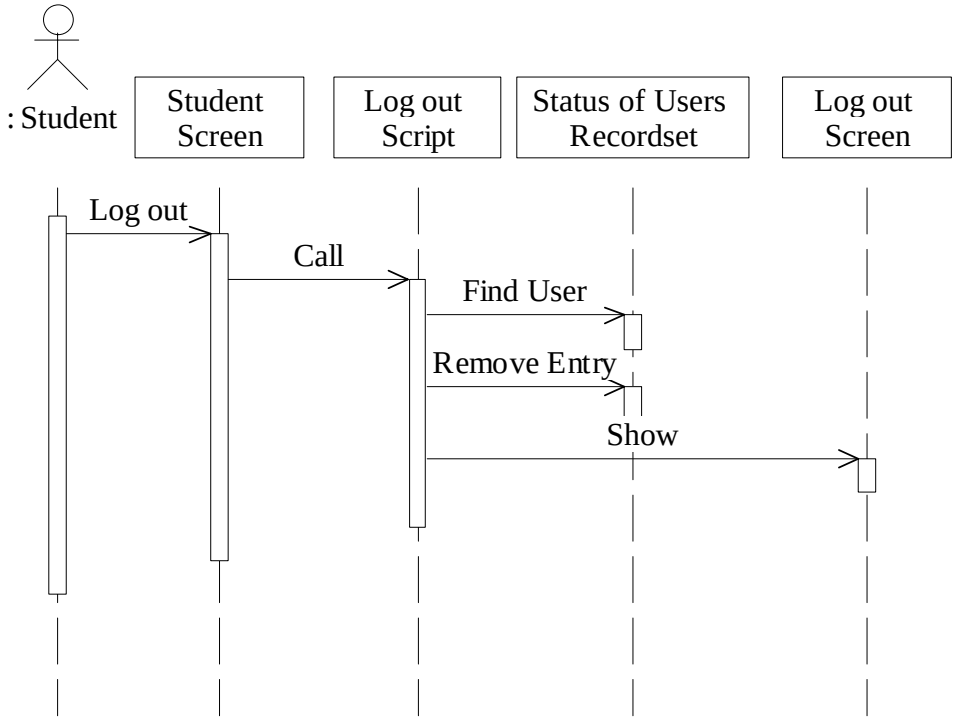

Fig 5.14: Sequence Diagram for Logging out Student

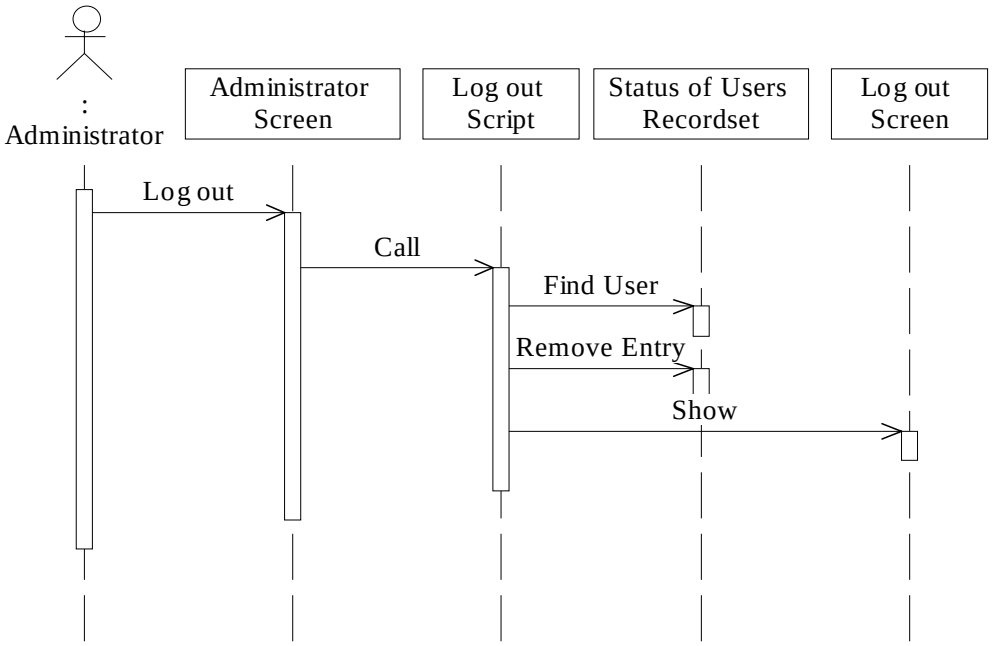

Fig 5.15: Sequence Diagram for Logging out Administrator

5.3 Functionalities for Teacher

5.3.1 Use Case Diagram

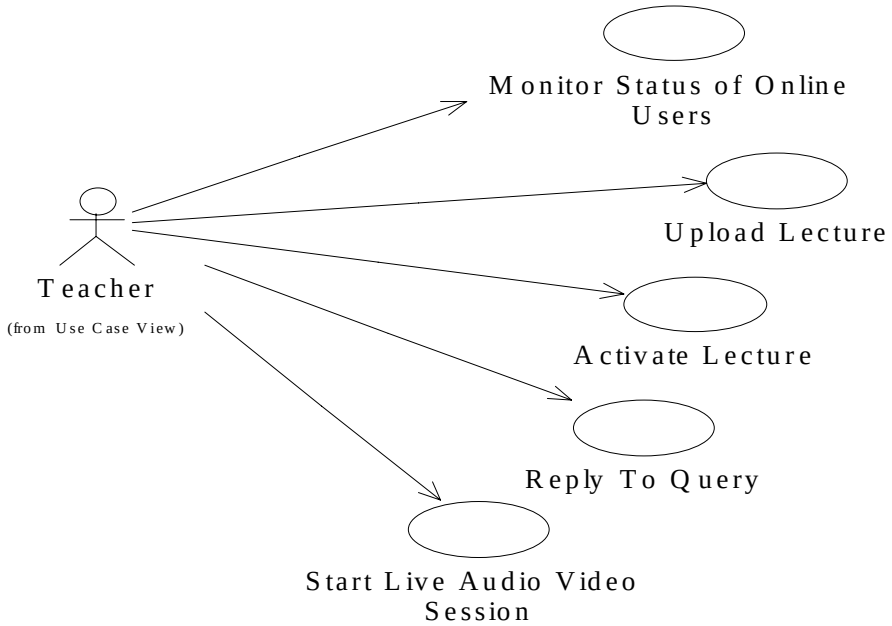

Fig 5.16: Use Case Diagram Functionalities for Teacher

5.3.2 Use case Specifications and Sequence diagrams

Use Case Name: Reply To Query

Use Case Number: ULB20201

Description: Reply To Query use case is used to reply to some Query.

Purpose: Teacher may reply to a particular Query using either textual answer only, or he may draw Figure on Board.

Related Use Cases: Ask A Query

Primary Actors: Teacher

UC Spec: Body Section

Main Course:

Actor action	System Response
1) Teacher joins Live Class.	2) System opens Classroom for teacher. 3) System displays Response window.
4) Teacher types textual answer. 5) Teacher may command to include Board's Image if he has drawn some Figure.	6) System adds textual and/or Image Response to Query Response data. 7) The use case ends.

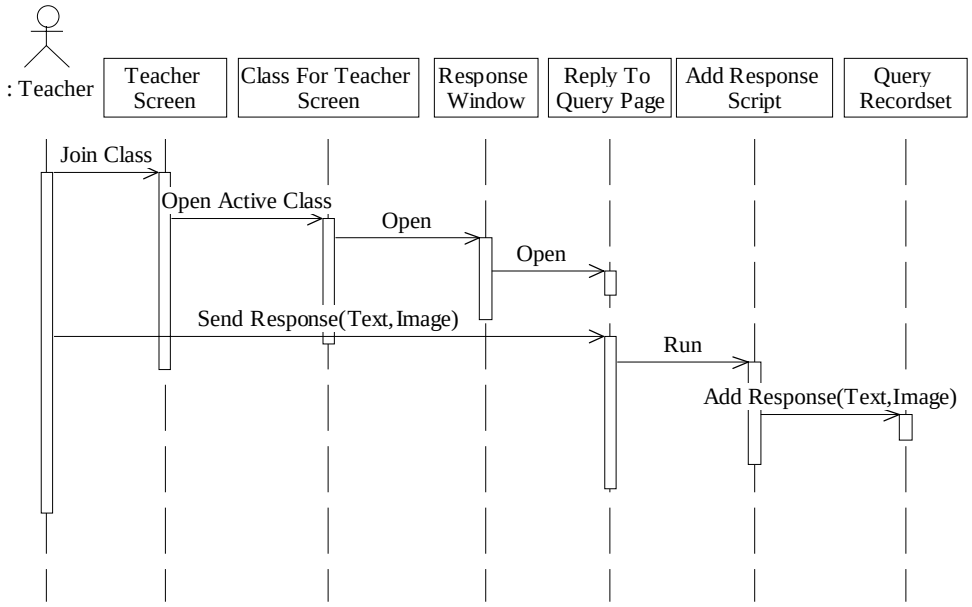

Fig 5.17: Reply to Query Sequence Diagram

Use Case Name: Start Live Audio & Video Session

Use Case Number: ULB20202

Description: The teacher uses Start Live Audio & Video Session use case to start a Live Audio and Video Session.

Purpose: Live Audio and Video is a facility for the teacher to deliver Audio and Video content complementary to the Lecture material or as a fully separate Session. Another necessary requirement is to provide such functionality to the teacher that he can announce when he desires to start Live Audio and Video Session during class.

Related Use Cases: Attend Live Audio and Video Session

Primary Actors: Teacher

UC Spec: Body Section

Main-Course:

Actor action	System Response
1) Teacher joins class.	2) System opens active Classroom for teacher.
3) Teacher commands to start Live Audio and Video Session.	4) System opens Video window. 5) System loads Transmission Control.
6) Teacher sets Audio and Video options. 7) Teacher commands to start transmission.	8) System announce start of Live Audio and Video Session to students. 9) The use case ends.

UC Spec: Trailer Section

Question: Why Live Audio and Video Session is announced.?

Answer: Without this functionality the students will have no indication as to when they have to attend Live Audio and Video Session.

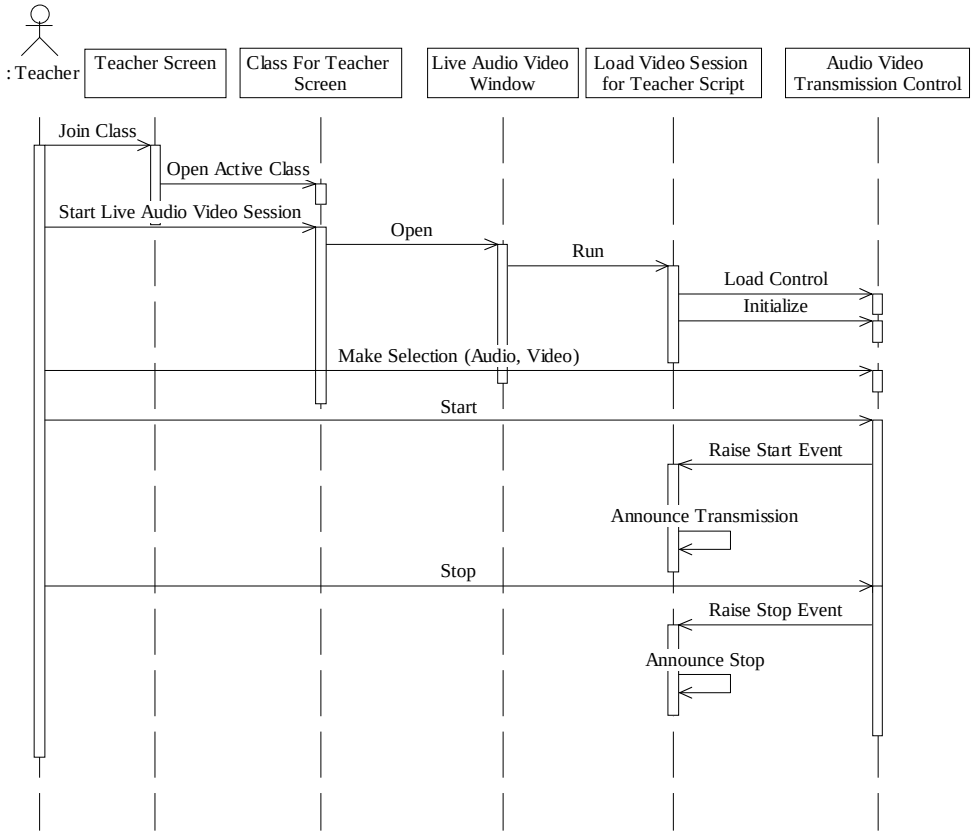

Fig 5.18: Sequence Diagram For Live Audio and Video Transmission

Use Case Name: Monitor Status of Online Users

Use Case Number: ULB20203

Description: Monitor Status of Online Users use case is used to view online user's status.

Purpose: It is the facility for the teacher to view which students are attending Live Class Session. Also location of the student in current Lecture Sections is indicated.

Primary Actors: Teacher

UC Spec: Body Section

Main-Course:

Actor action	System Response
1) Teacher joins Class.	2) System opens Classroom for teacher.
	3) System displays online users and their status.
	4) System automatically updates this view after some preset time.
	5) The use case ends.

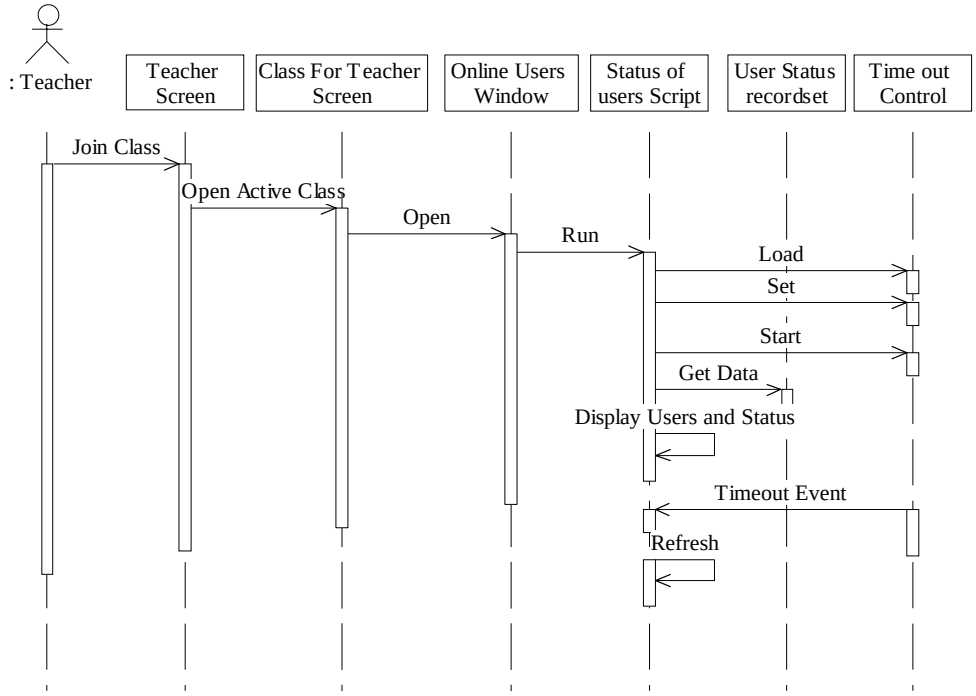

Fig 5.19: Sequence Diagram For On line Users status monitoring

Use Case Name: Upload Lecture

Use Case Number: ULB20204

Description: Upload Lecture use case is used to upload Lecture data.

Purpose: Teacher after previewing Lecture will transfer it to the Server.

Related Use Cases: Activate Lecture

Primary Actor: Teacher

UC Spec: Body Section

Main Course:

Actor action	System Response
1) Teacher logs in.	2) System display teacher Screen.
	4) System opens upload Lecture window.
3) Teacher commands to upload Lecture.	6) System transfers Lecture data file to Server.
5) Teacher selects Lecture file on his computer.	7) System updates Lecture's list.
	8) System finds information about related Picture and Sound files. Automatic upload from client computer is done.
	9) The use case ends.

UC Spec: Trailer Section

Question: What are *related files*?

Answer: Related files are the Picture and Sound files that are necessary for

a specific Lecture.

Question: Why *automatic upload* of related files is required?

Answer: It is expected that there are a number of Picture and Sound files that constitute a Lecture. Therefore manually selecting and uploading these may consume much time; it is prone to mistakes in selection by user and after all it is quite a dull task.

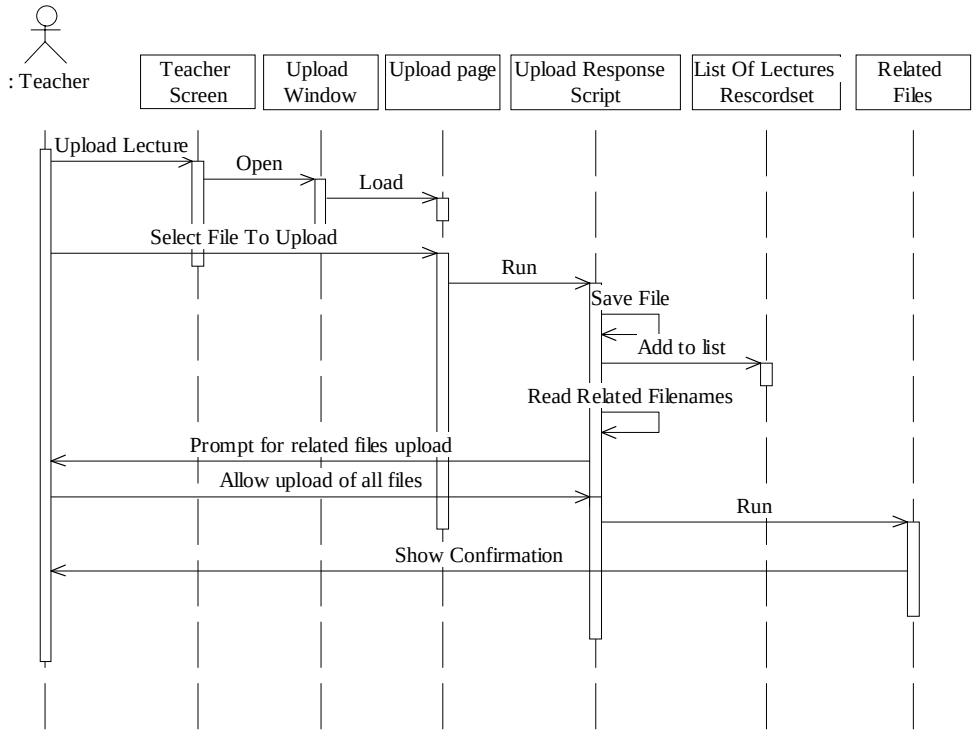

Fig 5.20: Sequence Diagram For Upload Lecture

Use Case Name: Activate Lecture

Use Case Number: ULB20205

Description: Activate Lecture use case is used to activate a particular Lecture for use in Classroom.

Purpose: Teacher after uploading Lecture files needs to activate it. He can upload a number of Lectures and then activate any one of them for a particular Class Session.

Related Use Cases: Upload Lecture

Primary Actors: Teacher

UC Spec: Body Section

Pre-condition: Some Lecture(s) has previously been uploaded.

Main-Course:

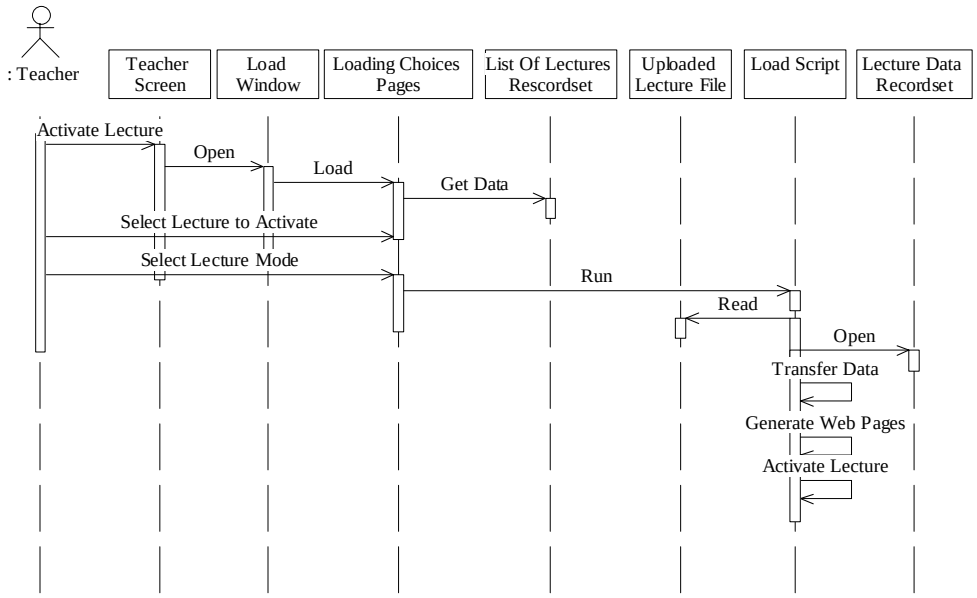

Fig 5.21: Sequence Diagram For Activating Lecture

Actor action	System Response
1) Teacher from teacher's Main Screen commands to open Activation window.	2) System opens Activation window and displays list of all available Lectures that are ready to be activated. 3) System also provides different options for Class Mode.
4) Teacher selects a particular Lecture for activation, and makes selection of its mode.	5) System transfer data to main DB Server. Web pages for faster access of Lecture data are generated. 6) Class is now in ready state. 7) The use case ends.

UC Spec: Trailer Section

Question: Why *Web Page* are generated?

Answer: To provide fast access to Lecture material, it is better that standard HTML/ASP pages are generated on Server file system for Lecture material, in addition.

5.4 Functionalities For Student

5.4.1 Use Case Diagram

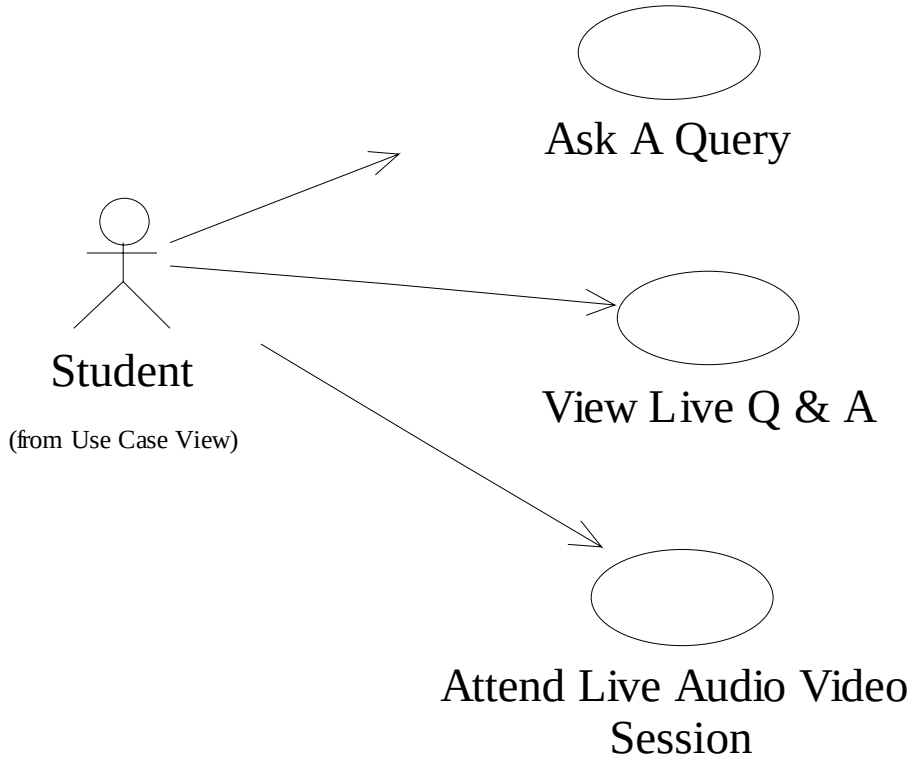

Fig 5.22: Functionalities for Student Use Case Diagram

5.4.2 Use Case Specifications and Sequence Diagrams

Use Case Name: Ask A Query

Use Case Number: ULB20301

Description: As A Query use case is used to ask new Queries.

Purpose: To provide facility to the students to ask questions during Live Class Session.

Related Use Cases: Reply To Query , View Prepared Lecture

Primary Actors: Student

UC Spec: Body Section

Pre-condition: Class is being held.

Main-Course:

Actor action	System Response
1) Student views prepared Lecture content.	
2) At some specific point he may want to ask a question.	
3) Student types his Query.	4) Query is sent to main Server.
	5) Query is added to Query data and location in Lecture where it was raised is also recorded.
	6) The use case ends.

UC Spec: Trailer Section

Question: Why *location of a Query* is recorded?

Answer: Location of a Query is important because it is used to distinguish Queries; which Queries are to be shown to the students viewing a particular Section, and also when merging Queries into Lecture material.

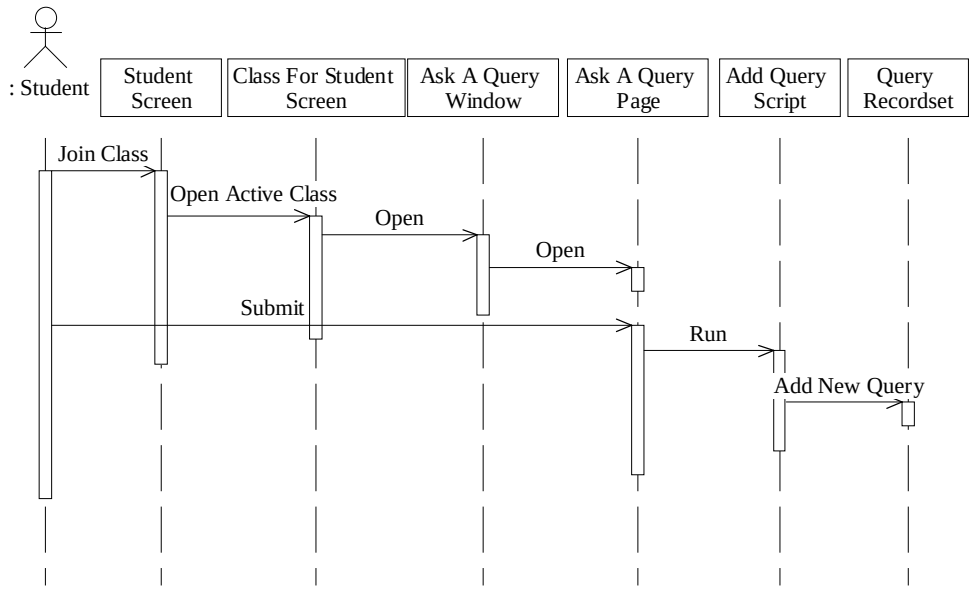

Fig 5.23: Sequence Diagram For Ask A Query

Use Case Name: View Live Query & Response

Use Case Number: ULB20302

Description: View Live Query & Response use case provides functionality of displaying Live Query and Response.

Purpose: It allows the students to view Live Queries relevant to particular Lecture Section they are going through at some particular instant. Answer to a Query is displayed on desired request made by the student.

Related Use Cases: Ask A Query

Primary Actors: Student

UC Spec: Body Section

Main-Course:

Actor action	System Response
1) Student joins Class.	
2) Student reads and listens to prepared Lecture.	3) System checks location of student in Lecture, and displays relevant Queries that are raised by other students.
4) Student commands to view answer to some specific Query.	5) System displays Answer window, displaying teacher's Board content, where applicable.
	6) The use case ends.

UC Spec: Trailer Section

Question: Why teacher's Board content is not always displayed in the Answer window?

Answer: Some answers may only have textual content without any Image of the teacher's Board.

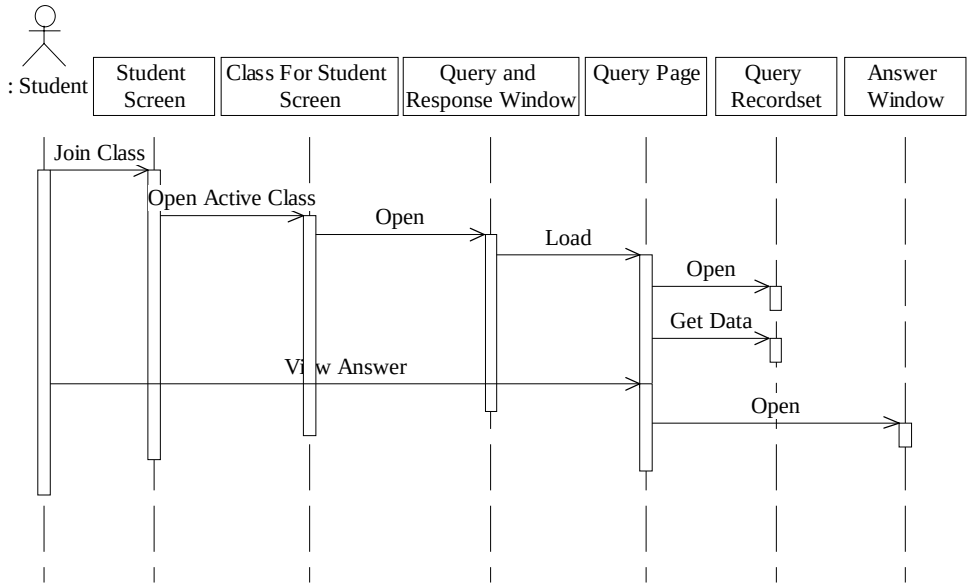

Fig 5.24: Sequence Diagram For View Live Query and Response

Use Case Name: Attend Live Audio And Video Session

Use Case Number: ULB20303

Description: Attend Live Audio And Video Session use case is used to attend Live Audio and Video Session.

Purpose: To notify the students about Live Audio and Video Session and provide facility to the students for attending it.

Related Use Cases: Start Live Audio And Video Session

Primary Actors: Student

UC Spec: Body Section

Pre-condition: Live Class Session is being held.

Main-Course:

Actor action	System Response
	1) System indicates that Live Audio and Video Session is being started by teacher.
2) Student commands to attend Live Audio and Video Session.	3) Live Video window is displayed.
	4) Live Video and Audio streams start receiving.
	5) Live Audio and Video Session stops when teacher ends it from his side.
6) Student close window.	7) The use case ends.

UC Spec: Trailer Section

Question: Why *notification* to the students is used?

Answer: Notification means announcement of Live Audio and Video Session. It is necessary because a student may be involved in viewing Lecture material and Live Query and Response. There is no other method of intimating the

student that now the teacher wants to start Live Audio and Video Session.

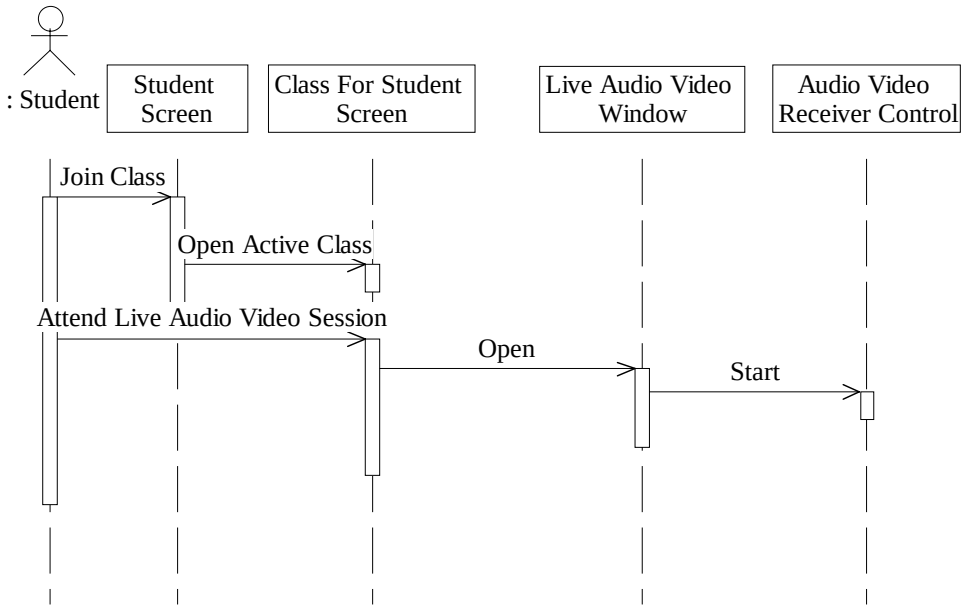

Fig 5.25 Sequence Diagram For Live Audio and Video For Student

5.5 Functionalities For Administrator

5.5.1 Use Case Diagram

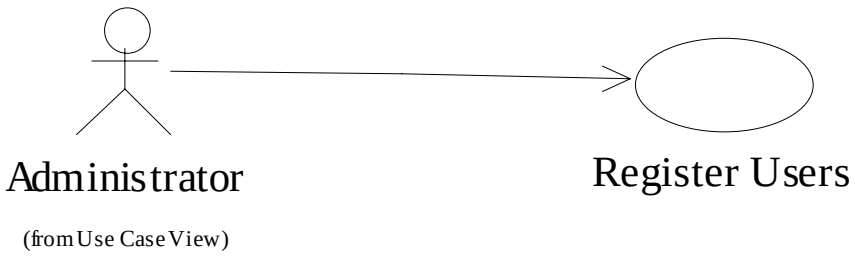

Fig 5.26 Functionalities For Administrator

5.5.2 Use Case Specification and Sequence Diagram

Use Case Name: Register Users

Use Case Number: ULB20401

Description: Register Users use case is used to register new users to a particular Course's Distant Electronic Classroom.

Purpose: It is an administrative functionality. It's purpose is to provide an on line interface of adding new students, by connecting to the Main Server from any where.

Primary Actors: Administrator

UC Spec: Body Section

Main-Course:

Actor action	System Response
1) Administrator logs on.	2) System displays administrator page.
3) Administrator commands to open Register User Screen.	4) System displays register user Screen.
5) Administrator adds new user's information and submits it.	6) System updates record.
	7) The use case ends.

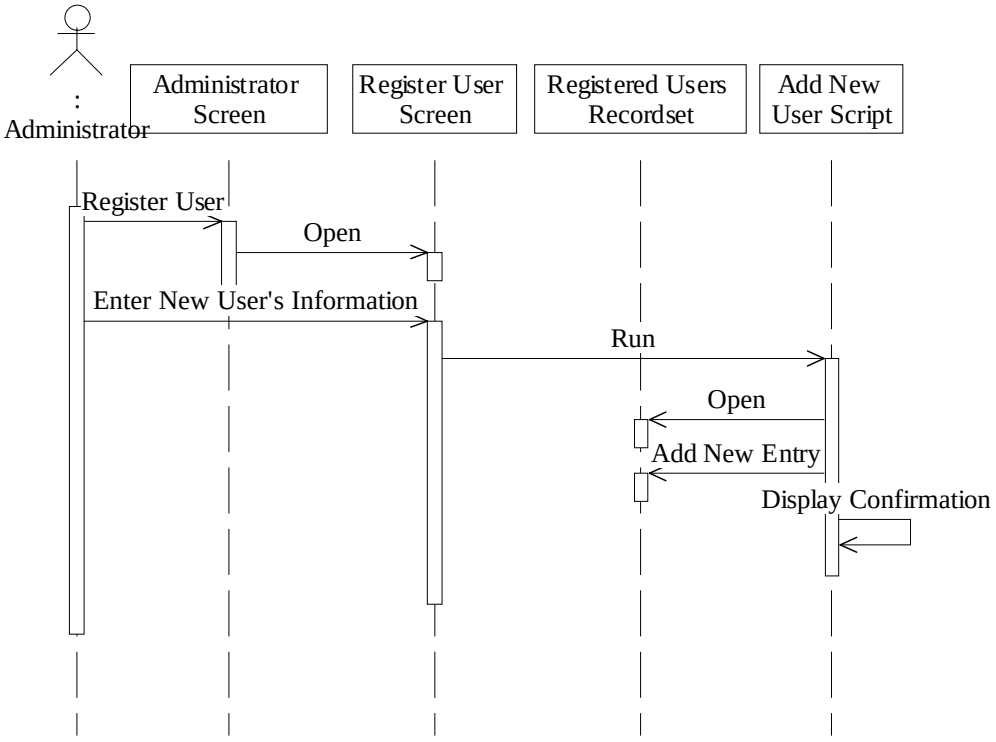

Fig 5.27 Sequence Diagram for Register User

5.6 Server Side Software Class Diagrams

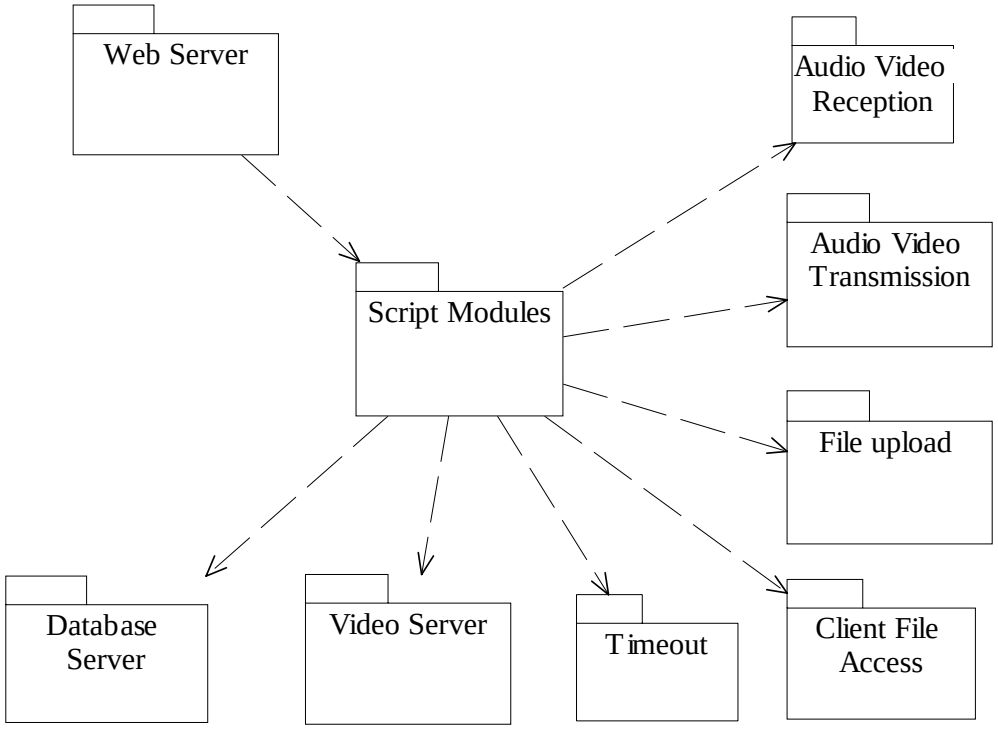

Fig 5.28 Server Side Software Packages

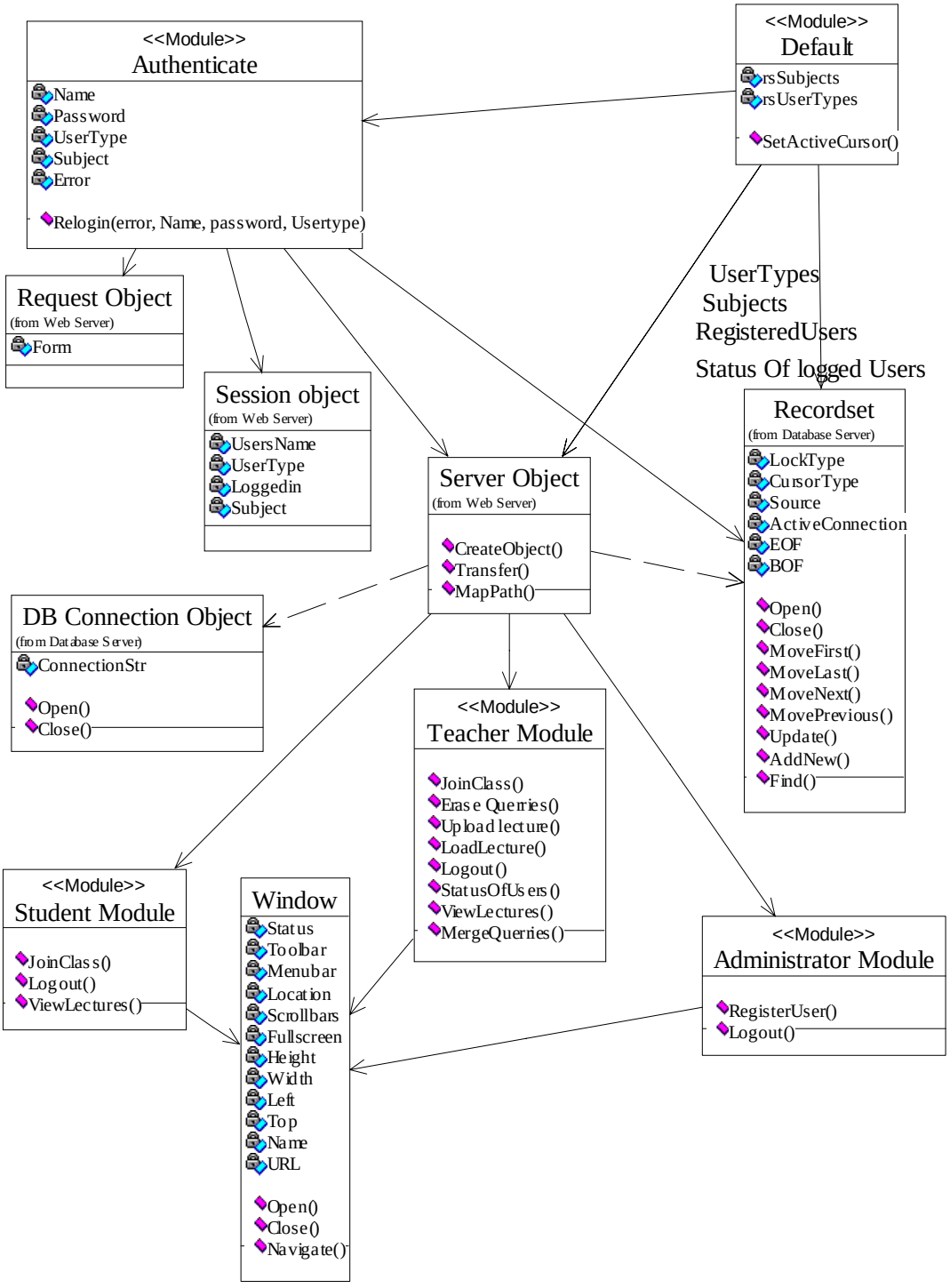

Fig 5.29: Server Side Software Class Diagram 1

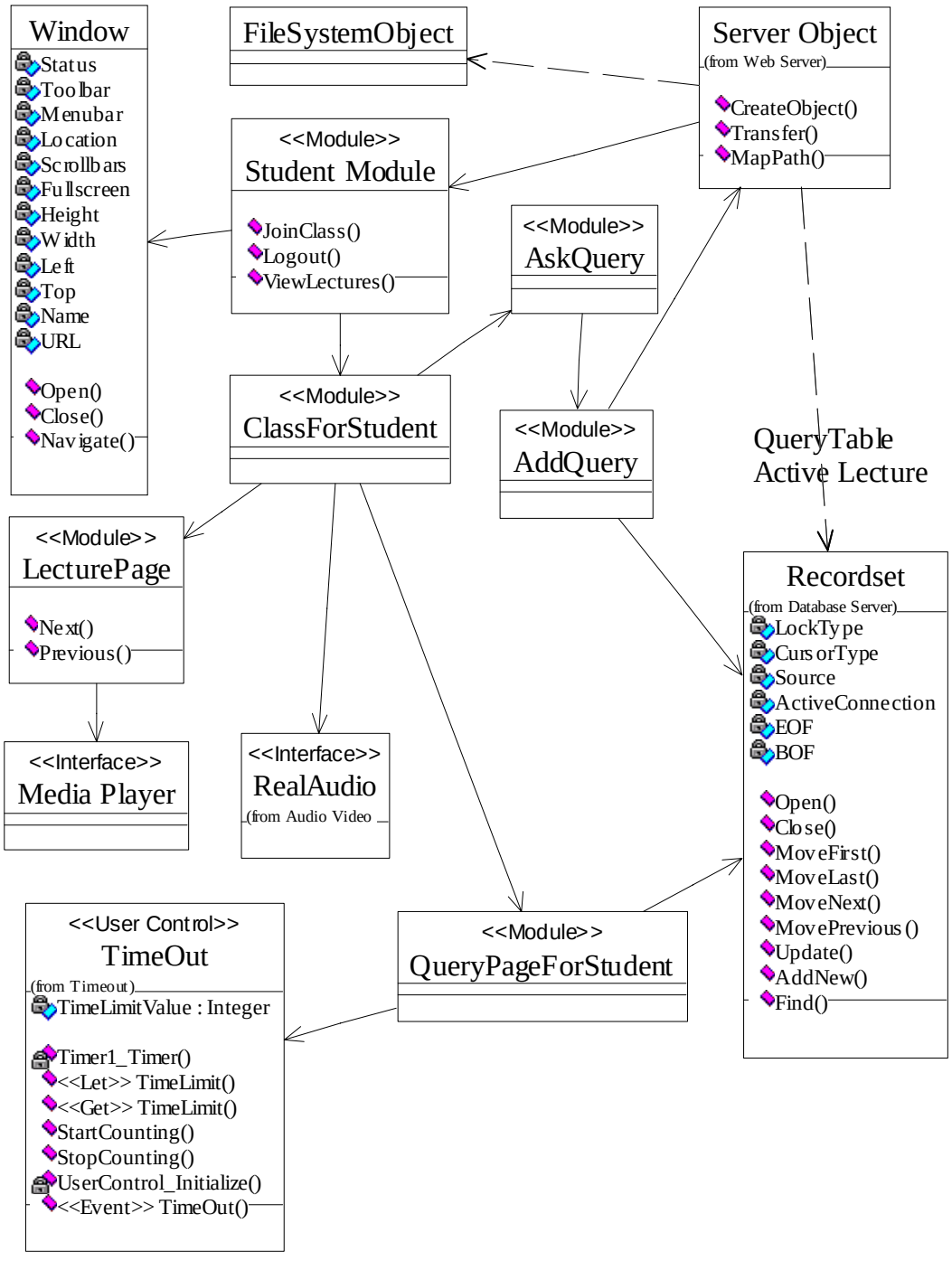

Fig 5.30: Server Side Software Class Diagram 2

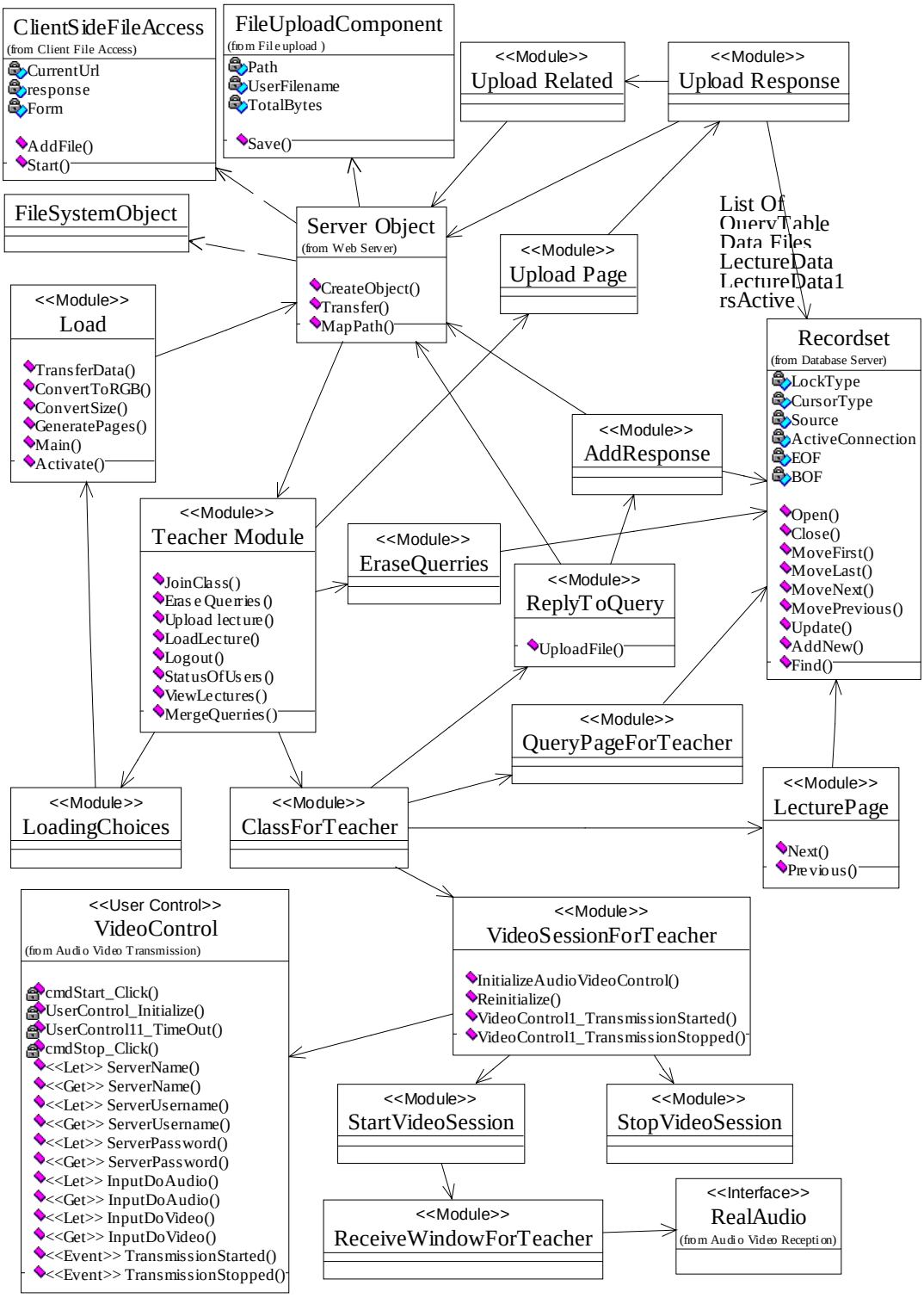

Fig 5.31: Server Side Software Class Diagram 3

5.7 Server Side Software Component Diagram

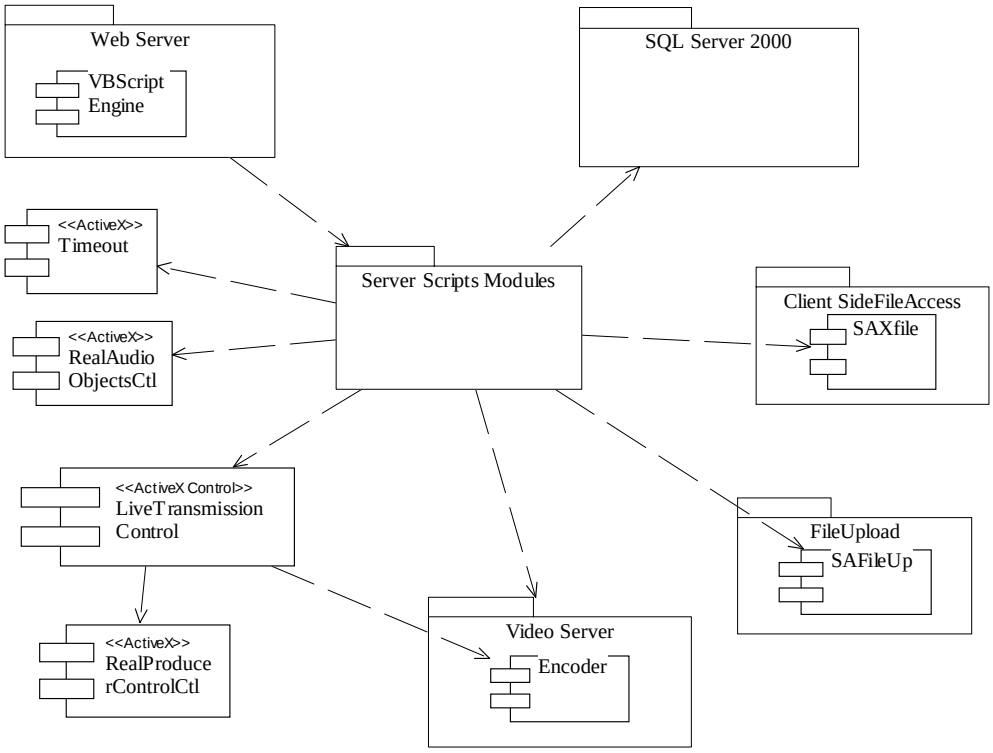

Fig 5.32: Server Side Software Component Diagram

5.8 Complete Deployment Diagram

Client Side and Server Side Software

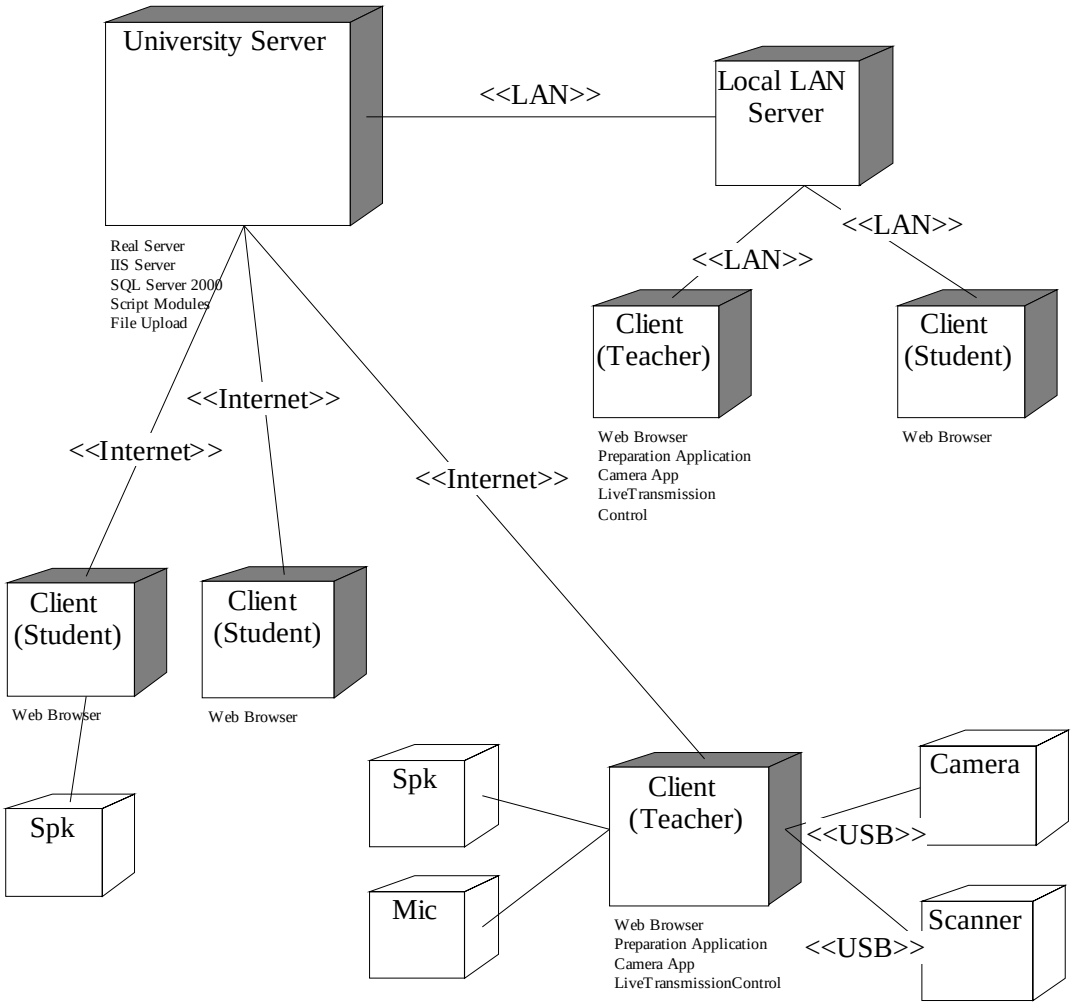

Fig 5.33: Complete Deployment Diagram

Chapter 6
Overview of Thesis: Methodology Applied and The Project

6.1 Object Oriented Software Paradigm

The development of the Software for "Distant Electronic Classroom Utilizing Low Bandwidth Internet" is done in object Oriented paradigm. Object Oriented analysis is concerned with developing Software Engineering requirements and specifications, expressed as a system's object model that is composed of a population of interacting objects. The idea that a system can be viewed as a population of interacting objects, each of which is an atomic bundle of data and functionality, is the foundation of object technology and is used for the development of complex systems. [34]

Designing of our system is done in two phases. At first, the *high level design* deals with the decomposition of the system into large complex objects. The second phase is called *low level design*. In this second phase, attributes and methods are specified at the level of individual objects. Here some reuse of Object Oriented products is done. As an outcome of the Object Oriented software paradigm our developed Software is *maintainable, reusable* and *productive*.

6.2 How UML Has Simplified Complete Software Development Task?

> The Unified Modeling Language (UML) is considered the de facto standard object modeling language in the industry. The UML is the evolution of early approaches to Object Oriented Analysis and Design. Based on the seminal works by Grady Booch, Ivar Jacobson, and Jim Rumbaugh in the Booch, OOSE, and Object Modeling Technique (OMT) methods, these three Rational experts have provided the basis for the next generation of visual software modeling. The UML details the application modeling language for:

- Business Process Modeling with Use Cases
- Class and Object Modeling
- Component Modeling
- Distribution and Deployment Modeling [33]

6.2.1 UML Standardization

UML makes it easier to develop and manage high quality applications in a reasonable amount of time. UML is the information technology industry's version of a blueprint. It is a method of describing system's architecture in detail. Using this blue print it becomes much easier to build and maintain a system, and to ensure that the system will hold up to requirement changes. UML standardizes representation of the Object Oriented Analysis and Design.

6.2.2 Use of CASE Tool: UML Modeling

Proper use of the Object Oriented technology requires the development of analysis and design specifications using some *CASE tools* to support both the drawing of objects and the description of the relationships between objects. For the development of our Software we have done UML

modeling in CASE tool entitled ***"Rational Rose"***. Modeling is the designing of software applications before coding. Modeling is an essential part of software projects and is helpful. Using modeling it is easier to *verify correctness* of our system before implementation in code, otherwise changes are difficult and expensive to make.

UML helps to specify, visualize, and document models of software systems, including their structure and design, in a way that meets all of these requirements. CASE tool makes it easier to analyze future application's requirements and design a solution that meets them. As the complexity of systems increase, so does the importance of good modeling techniques. There are many additional factors of a *project's success*, but having a rigorous modeling language standard is one essential ingredient.

6.2.3 Benefits of Well Designed Architecture

Our UML model of the Software will make it easy to *maintain* the complete system, and *quickly find and fix a bug* that shows up after long time. A well designed architecture benefits any program. Another benefit of structure is that it enables *code reuse*. Design time is the easiest time to structure an application as a collection of self-contained modules or components.

6.3 Component Based Software Development

Component Based Software Development focuses on building large software systems by integrating previously existing software components. At the foundation of this approach is the assumption that certain parts of large software systems reappear with sufficient regularity, thus common parts should be written once, rather than many times, and that common systems should be assembled through reuse rather than rewritten over and over. [34]

Component based systems encompass both *Commercial Off the Shelf (COTS) products* and *custom made components*. We have utilized component based software development in our project. Some of the components are just used as commercial off the shelf components while we have also designed some specific custom components. Programming is done for this project by writing code as well as assembling and integrating existing and custom developed software components. In contrast to traditional development, component integration is the centerpiece of the approach. In our Software development project, integration is a key consideration in the decision whether to acquire, reuse, or build the components.

Our developed Software is composed of two parts, ***Lecture Preparation Application*** and ***Server Side Software***. Use of Component Technology has significantly reduced *development time and costs*, than the traditional method of building systems "from scratch".

6.4 What We Have Achieved?

6.4.1 Lecture Preparation Application

Lecture Preparation Application is developed for the teacher. Our developed Software is ready to be deployed on any Windows based Personal Computer. Our Application makes it easy to divide complete Lecture into Sections and Sections into subsequent Screens. Each Section has its own Section Title. After adding Section Titles our application provides facilities of adding Text by typing or using a Scanner. Basic formatting facilities are available. Provision for recording and compressing Sound narration for each Screen is available. Figures relevant to textual material can be added in each Screen. Here also scanning facility is available. The teacher can preview complete Lecture during its development in a web Browser. Navigation between different Screens and Sections is possible. Complete Lecture consists of *main Lecture data file* plus *associated Picture and Sound files.* After preparation of Lecture material next task is to upload it to the Server.

6.4.2 Server Side Software

Server Side Software provides basic logging facilities to the teacher and the students in different subjects. The teacher after logging to the Server can upload Lecture to the Server. Upload process is made easy by enabling automatic upload of main Lecture data file and associated Sound and Picture files. User doesn't need to manually select all files. More than one Lecture in a specific Subject can be uploaded at one time.

After uploading Lecture our Server Side Software provides facility of activating the Lecture. In this process a specific mode of classroom is selected. These modes make it possible to select prepared Lecture content (Text, Voice and Figures), with or without Live Query and Response with Blackboard content, with or without Live Audio and Video facility. When Lecture is activated it is uploaded to SQL Server. Also web pages are generated for fast access by users.

Server Side Software integrates different facilities during Live Class Session. It provides student's status monitoring, prepared Lecture material browsing in specific Screen chunks, Live Query and Response regarding each Screen. The teacher can answer textually and/or use the blackboard. When the teacher starts Live Audio and Video Session, automatic announcement to all the students is made.

After Live Class is over, the teacher has the facility to combine prepared Lecture material with live interactive content thus producing a new reference material. Server Side Software manages attendance record of the students. It provides browsing of Lecture material with or without merged Queries, after class Sessions.

6.5 Software Testing

Software goes through two stages of release testing: Alpha (in-house) and Beta (client-site). Alpha testing is usually where the system is tested to identify bugs. Alpha testing of the developed the Software has been done. It includes tests at the Software interface to demonstrate that each function is operational, that is, *black box testing*. Secondly *glass box (white box) testing* has been done to assure that internal operations perform according to specifications; logical paths, loops execution and procedural details have been checked.

In addition, *inspection* that is close examination of Specification, Design and Code, has been done which has resulted in many early defects detection, prevention and isolation. Code reading has also detected many defects.

Now our developed Software is ready for beta testing. This involves testing the system or software in the environment in which it will be eventually used, formally by a group of potential end-users. But this requires that this Software is installed in a proper working environment and used by a

number of users. An item in beta test is mostly working, but still considered under test.[15]

6.6 Maintenance of The Software

The developed Software has been thoroughly tested and has met the requirements. Now once the system has been in operation for any length of time, it is simply a matter of maintaining it. This may involve routine upgrades or refurbishment. Maintenance of Distant Electronic Classroom Software is required at the teacher's computer and at Server Side computer system. After some period of time, when the teachers will use Lecture Preparation Application Software, some *hidden bugs* in the code may reveal, in addition there might be some *new suggestions* to improve the Software. This may lead to a new iteration of complete development of Lecture preparation application or just a few changes in Code and Specifications. Hence new version may be developed and distributed to the teachers thereafter.

The ostensible purpose of documentation developed in UML is to aid in understanding what the system does, and for the maintenance programmer how the system does it. Typically the tools and techniques used for maintenance are those that were used to develop the system.

6.7 Areas of Application

Distant Electronic Classroom Software has its direct application in *Distance Learning* programs. It is an economical solution for Allama Iqbal Open University (AIOU) for utilizing synchronous method to deliver education. Also, National University of Science and Technology (NUST) can use this Software to effectively combine services of the teachers in different campuses to take classes from distance. [28,29]

Recently Government of Pakistan has started a new project 'Virtual University'. The idea of the Virtual University is to extend the quality education to remote areas of Pakistan, where qualified faculty is not available. VU plans to deliver the contents via the television broadcast and the reverse interaction will be accomplished with the help of the Internet. This has a ***minus part*** with the poor student teacher interaction along with missing environment of the classroom. The VU is starting its operations by collaborating with private sector partners to achieve the classroom environment. Our Distant Electronic Classroom Software efficiently *eliminates both minus points*, that is, it provides real time student teacher interaction, as well as provides a Classroom environment to users while they are at home or at any specific location. [27]

6.8 Future Enhancement Possibilities

Once our Software has been installed, users will inevitably find ways to improve the existing system to meet ever-changing needs. It is important

to note the life cycle process is not linear. Rather, it is an ***iterative process***, with the development of new systems taking place during the life cycle of an existing system. [33]

First realization of this Software is programmed to work through 56 kbps Internet. In future, when available bandwidth increases in Pakistan, for example through ISDN connections, then live transmission control designed for Live Audio Video Session can be easily updated to take advantage of increased bandwidth. Also the implementation that provides Query and Response involving live Blackboard content can also take advantage of increased bandwidth. Thus the complete Software implementation is designed in such a way that during its maintenance phase a few updates will enable it to take advantage of new technology.

6.9 Conclusion

Software developed during this thesis work has enhanced the existing Distance Education capabilities by making radical changes and a total reshaping of the process. This Software has achieved realization of Classroom environment utilizing Low Bandwidth Internet. It efficiently provides means to develop asynchronous content using Lecture preparation application, and adds live interaction to it during Live Classroom. Using this Software it is possible to provide a true Classroom

environment in Distant Leaning system and thus this realization has added a new dimension to Distance Education.

Once deployed in a typical environment this Software provides user friendly interface. Therefore this Software implementation provides ease of use to users from other fields than just supporting computer professionals only. Students and teachers in all learning-teaching professions can effectively utilize this Software. As mentioned earlier this realization effectively eliminates many weak and negative aspects of a Distance Education institution, especially a Virtual University.[27]