

CENG 491 SENIOR PROJECT

Requirement Analysis Report

InnovationTech

Ashhan Ünal 1298397

Birsen Omay 1298058

Gülistan Doğan 1297738

Mesude Eser 1297803

Songul Çeçen 1250109

TABLE OF CONTENTS

TABLE OF CONTENTS	1
1. INTRODUCTION.....	2
1.1 OVERVIEW.....	2
1.2 USER OBJECTIVES	2
1.3 LITERATURE SURVEY	2
1.3.1 LearnLinc	3
1.3.2 Centra 7	3
1.3.3 HorizonLive	4
1.4 MAIN GOALS	4
2. IDENTIFICATION OF SYSTEM.....	5
2.1 VIDEO AND AUDIO CONFERENCING	5
2.2 TEXT CHAT	5
2.3 WHITEBOARD	5
2.4 STUDENT ACTIONS	5
2.5 INSTRUCTOR ACTIONS	5
3. TECHNICAL ANALYSIS	6
3.1 ARCHITECTURE OF InNoClass	6
3.2 DEVELOPMENT REQUIREMENTS	6
3.2.1 Programming Languages.....	6
3.2.2 Database Server.....	7
3.2.3 Web Server.....	8
3.2.4 Video Server.....	8
3.3 SOFTWARE AND HARDWARE REQUIREMENTS FOR USERS	8
4. SYSTEM MODELING.....	9
4.1 USE CASE MODEL.....	9
4.2 DATA MODEL.....	10
4.2.1 Entity Relationship Diagram.....	10
4.2.2 Data Dictionary	11
4.3 FUNCTIONAL MODEL	12
4.3.1 Data Flow Diagram	12
4.3.2 Control Flow Diagram	14
4.4 BEHAVIORAL MODEL.....	15
4.4.1 State Transition Diagram	15
5. COST ANALYSIS AND SCHEDULING.....	15
5.1 TIME AND EFFORT ESTIMATION	15
5.2 SCHEDULE	17
6. MANAGEMENT DECISIONS	18
6.1 RISKS ANALYSIS AND MANAGEMENT	18
6.2 SOFTWARE CONFIGURATION MANAGEMENT (SCM)	18
6.3 SOFTWARE QUALITY ASSURANCE (SQA)	18
7. METHODOLOGY.....	19
7.1 PROCESS MODEL	19
7.2 TEAM ORGANIZATION	19
8. REFERENCES.....	20

1. INTRODUCTION

1.1 OVERVIEW

The idea of virtual classrooms is becoming more and more popular with the advances in technology. They offer more attractive teaching environments than the traditional classes have. Newer and faster computers combined with high resolution graphics, video and audio make electronic delivery possible for training and communication of many types. Thus, today long distances, large number of students, time problems and costs related with these are no longer problems for the users of virtual classrooms.

1.2 USER OBJECTIVES

In order to get more opinion about user expectations of virtual classrooms, we visited Tes Group. Tes group is the distributor of Interwise Enterprise Communication Platform. According to the questionnaires they made, the most common user expectation is the ease of use. Students want a simple interface that will let them to use all virtual classroom capabilities efficiently and to reach the course material after the lecture. Instructors want to create materials easily. Also, the users request that the system should not require specific hardware to run. [1]

1.3 LITERATURE SURVEY

Since virtual classroom is a popular concept, there are many vendors who want to get their share from this market. In this section, five of them are evaluated for this part. The criterion for evaluation is as follows:

- Platform supported
- Browser or client based
- Bandwidth requirement
- Hosting available
- Primary setting
- Synchronous features
- Feature gaps

1.3.1 LearnLinc

Platforms supported	Windows, Unix, Mac
Browser or client based	Browser
Bandwidth requirement	56K
Hosting available	No
Primary setting	Corporate
Synchronous features	Chat Presentation Whiteboard Annotations Polling Application sharing Audio conferencing Video conferencing
Asynchronous features	Archiving and playback

[2]

1.3.2 Centra 7

Platforms supported	Windows
Browser or client based	Thin web client
Bandwidth requirement	56K
Hosting available	Yes
Primary setting	Corporate and academic
Synchronous features	Chat Presentation Whiteboard Annotations Polling Web touring Audio conferencing Video conferencing Support for multiple presenters Breakout groups

Asynchronous features	Archiving and playback
Gaps	Application and desktop sharing, asynchronous discussions, runs only on Windows platforms

[3]

1.3.3 HorizonLive

Platforms supported	Windows, Unix, Mac
Browser or client based	Browser
Bandwidth requirement	56K
Hosting available	Yes
Primary setting	Corporate and academic
Synchronous features	Chat Presentation Whiteboard Annotations Polling Application sharing Audio conferencing Video conferencing Breakout groups
Asynchronous features	Archiving and playback
Gaps	Web touring, mood indicators, sharing control, asynchronous discussions

[4]

1.4 MAIN GOALS

While searching the market, we see that most of the products have high prices intended for large enterprises and are not customizable. So our group, InnovationTech, aims to develop a virtual classroom, InNoClass offering the necessary features and serving the needs of public. The users will get what they need in an easy and efficient way. There won't be any unnecessary features that will mix the minds. InNoClass, while being simple to understand and use, will also let you to use it in different areas.

2. IDENTIFICATION OF SYSTEM

The features of InNoClass will be described under five titles.

2.1 VIDEO AND AUDIO CONFERENCING

Using this feature, students will be able to see the instructor within the interface. Both the instructor and the students will communicate synchronously using speakers, headsets and microphones. Both sides will be able to tune their own audio levels using audio check facility.

2.2 TEXT CHAT

Students and instructor can exchange ideas with each other via text chat. Public messages will be sent to all students and a student will be able to chat with the instructor or another student privately. Also, sub-group chat, that enables three or more students to discuss a specific issue will be possible.

2.3 WHITEBOARD

The whiteboard is the place from which the information will be presented to the students. The instructor will be able to insert text or draw figures on it. It will offer multiple colors, shapes and sizes that can be cut, pasted, moved or aligned together. PowerPoint slides, PDF documents and different formats of files will be loaded and edited on the whiteboard. Multimedia content such as Flash, Shockwave, video and image files will be supported. The instructor can use web touring facility on the whiteboard to take students to any web site during a lecture. The contents of the whiteboard will be saved and printed by the students with the permission of the instructor.

2.4 STUDENT ACTIONS

Students will have the ability to access their personal profile information such as applying the preferred settings. They can participate to the lecture by handraising and showing their emotions. They will be able to download the course material whenever they have permission.

2.5 INSTRUCTOR ACTIONS

During a lecture, the instructor will always maintain the floor control by giving the floor to any student. The student who takes the floor will use the audio conferencing. The instructor will test the students and receive instant feedback by creating true/false or multiple choice questions during a lecture.

3. TECHNICAL ANALYSIS

3.1 ARCHITECTURE OF InNoClass

The architecture of InNoClass consists of three main components as shown in Figure 3.1. One of them is the web server which displays static or dynamic web pages. Second is the database server used for storing instructors' and students' personal and login information. The database server also stores lecture notes, video and audio files. For streaming video and audio, a video server will be used.

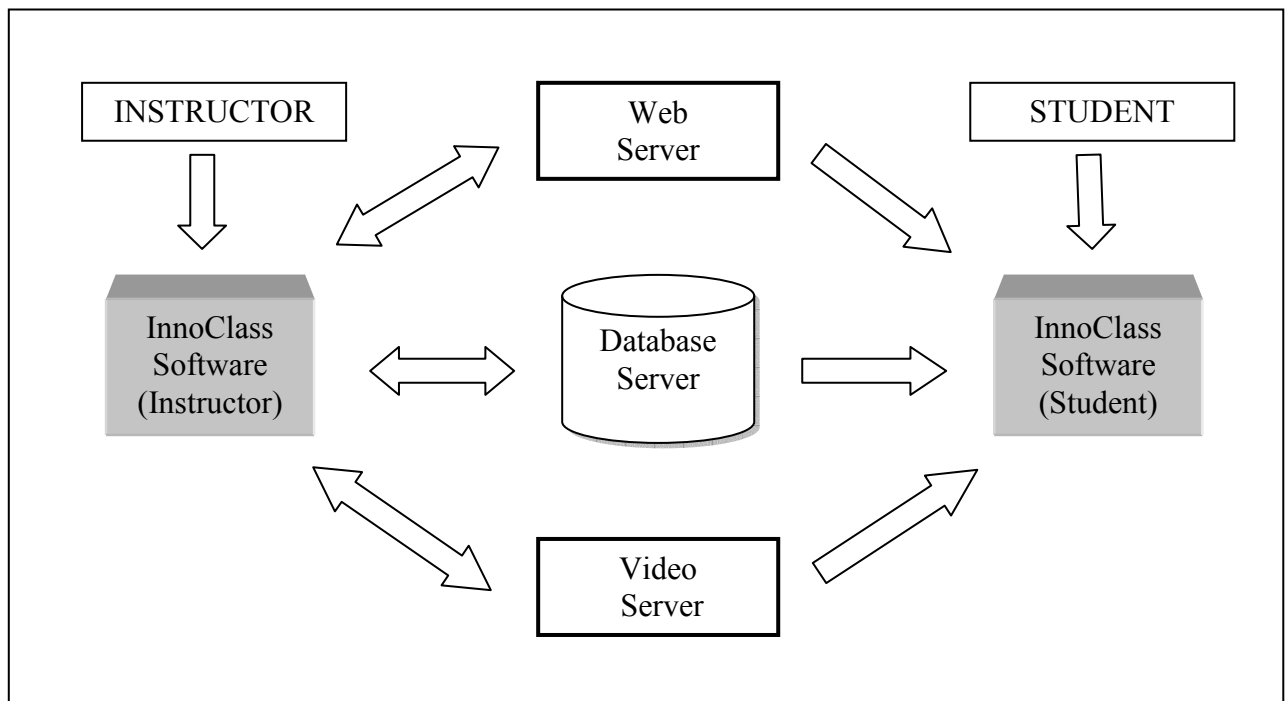


Figure 3.1
Architecture of InNoClass

3.2 DEVELOPMENT REQUIREMENTS

3.2.1 Programming Languages

While deciding on development software of our project, we had a dilemma between Java and C# since these are the two dominant platforms in the web services area. After a broad search on these two platforms, we saw that both of them have their own superiorities such as while Java supports for multiple operating systems, C# supports for multiple languages. But we found Java more suitable for our system because of the following reasons:

- Java runs on different environments (Windows, Macintosh, Unix) whereas C# runs only on Microsoft environment.

- Java has strict error checking, exception handling which makes sure that no item is forgotten but in C#, exceptions are ignored easily.
- Java avoids poor programming process but C# facilitates this.
- Java has an extensive library of high quality even it has more libraries than Windows. This feature of Java makes it easier to write programs

For building web application, we considered two alternatives PHP and JSP. Although PHP is also a good web application development tool and cooperate with MySQL, considering the performance, JSP is far better than PHP. JSP has a powerful Java Database Connectivity (JDBC) API supported, but PHP does not provide a uniform database connective interface. We will use Tomcat JSP Engine as the JSP development environment. According [5], JSP offers the following features:

- Write once, run anywhere properties
- High quality tool support
- Separation of roles
- Reuse of components and tag libraries
- Separation of dynamic and static content
- Support for scripting and actions
- Web access layer for N-tier enterprise application architecture

3.2.2 Database Server

From a variety of database servers, we decided to build our database applications with MySQL. While searching for an appropriate database server, we found out that although other database servers such as Oracle, MSSQL are very powerful and qualified, their costs are very high. MySQL serves our needs without any payment and besides, provides the following capabilities:

- MySQL is a fairly well featured and fast database system which can handle most standard SQL constructs. It supports a broad subset of the ANSISQL 99 syntax so it will be easy for us to use it.
- Connection to MySQL database server can be supplied from all of the major platforms using Java.
- MySQL allows changes to structure of tables while running, which will be useful for synchronous applications like our system.
- MySQL requires less hardware resources which will prevent additional cost and rework.

3.2.3 Web Server

We chose Apache among various web servers since it works compatibly with Java based projects. Additionally, Apache has the following features:

- Apache handles JDBC application program interface efficiently.
- Apache supports multiple platforms.
- Apache is open source and it can be updated which permits adding functionalities as it becomes available.

3.2.4 Video Server

There are two primary methods to follow for streaming audio and video content. First method is the web server and the associated HTTP and TCP protocols resulting in a slow transmission that wastes bandwidth. Second method is the video server approach which uses a streaming video server specialized to audio/video streaming task. We decided to follow the second approach by using Real Media server which is a powerful video server running on different platforms providing these advantages:

- Qualified audio and video streaming
- Efficient use of network bandwidth
- Qualified transmission of synchronized multimedia presentations and interactive video presentations.

3.3 SOFTWARE AND HARDWARE REQUIREMENTS FOR USERS

- Operating System : Linux , Unix , Sun , Windows
- Internet Explorer : 5.x , 6.x , Netscape 4.5x, 4.7x , 7.x
- CPU : P350 MHZ
- Memory : 128 MB
- Disk : 40 MB
- Network : 28.8 kpbs, Network Adapter
- Monitor : 18 bit colors
- Sound Card
- Headset with microphone

4. SYSTEM MODELING

4.1 USE CASE MODEL

The work done in our system can be described with the help of the actors and their use cases. InNoClass consists of two main actors.

- Student User

Students are people who will benefit from the learning platform of InNoClass. They will log in to system and they can use the lectures of InNoClass which are determined by the instructor.

- Instructor User

Instructors are people who give lectures. They have some extra privileges on the system.

Use cases for each of the actor described above are listed below.

- Use cases for student user

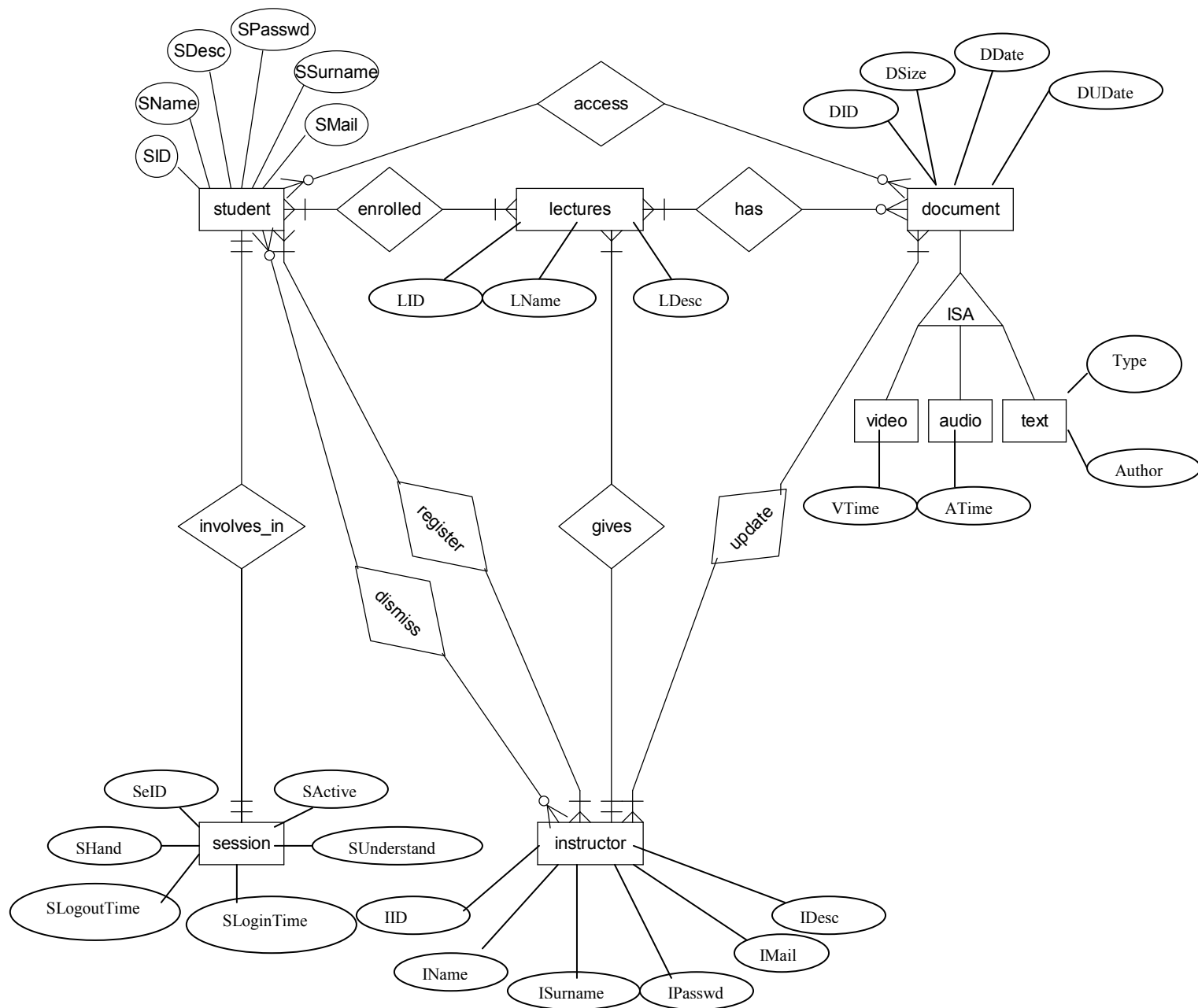
- Logs in entering a user name and password.
- Starts a process selecting from a list of predefined actions.
- Monitors the processes that are active and already done.
- Approves/rejects an activity(text chat, handraising, etc)

- Use cases for instructors

- Create user/user groups
- Designs/changes the route of processes
- Forms documents needed for lectures.
- Terminates a process that is active.
- Monitor system status.

4.2 DATA MODEL

4.2.1 Entity Relationship Diagram



4.2.2 Data Dictionary

Student

SID: INTEGER (Student's identity number)

SName: VARCHAR (30) (Student's name)

SSurname: VARCHAR (30) (Student's surname)

SPasswd: VARCHAR (12) (Student's password to enter the system)

SMail: VARCHAR (50) (Student's mail address)

SDescr: VARCHAR (100) (Extra info about student)

Instructor

IID: INTEGER (Instructor's identity number)

IName: VARCHAR (30) (Instructor's name)

ISurname: VARCHAR (30) (Instructor's surname)

IPasswd: VARCHAR (12) (Instructor's password)

IMail: VARCHAR (50) (Instructor's mail address)

IDescr: VARCHAR (100) (Extra info about instructor)

Document

DID: INTEGER (Document ID)

DSize: INTEGER (Document size)

DDate: DATE (Document insertion date)

DUDate: DATE (Document last update date)

Video

VTime: INTEGER (Video Time Length)

Audio

ATime: INTEGER (Audio Time Length)

Text

TType: CHAR (5) (Type of document e.g. PDF, PowerPoint, Word and Excel)

TAutor: CHAR (30) (Author of the document)

Lecture

LID: INTEGER (Lecture ID)

LName: VARCHAR (40) (Lecture name)

LDescr: VARCHAR (100) (Info about lecture)

Session

SeID: INTEGER (Session ID)

SActive: LOGICAL (If student is participating the class TRUE, else FALSE)

SHand: LOGICAL (If student raises hand TRUE else FALSE)

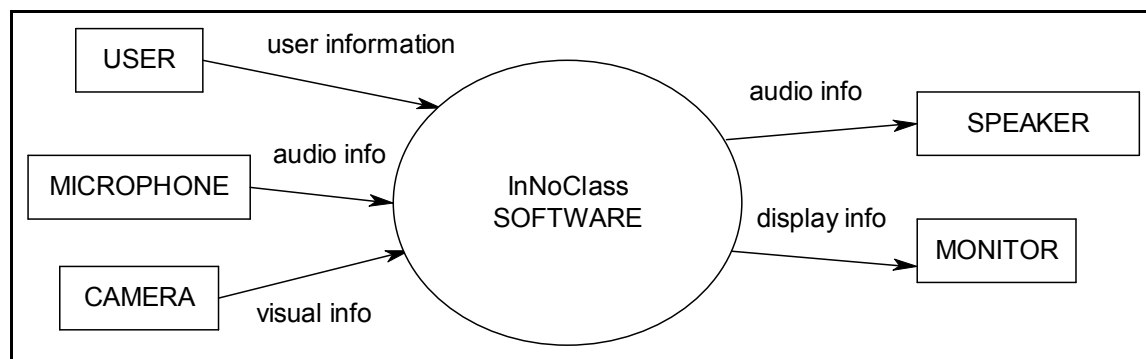
SLoginTime: DATE (time student logs in)

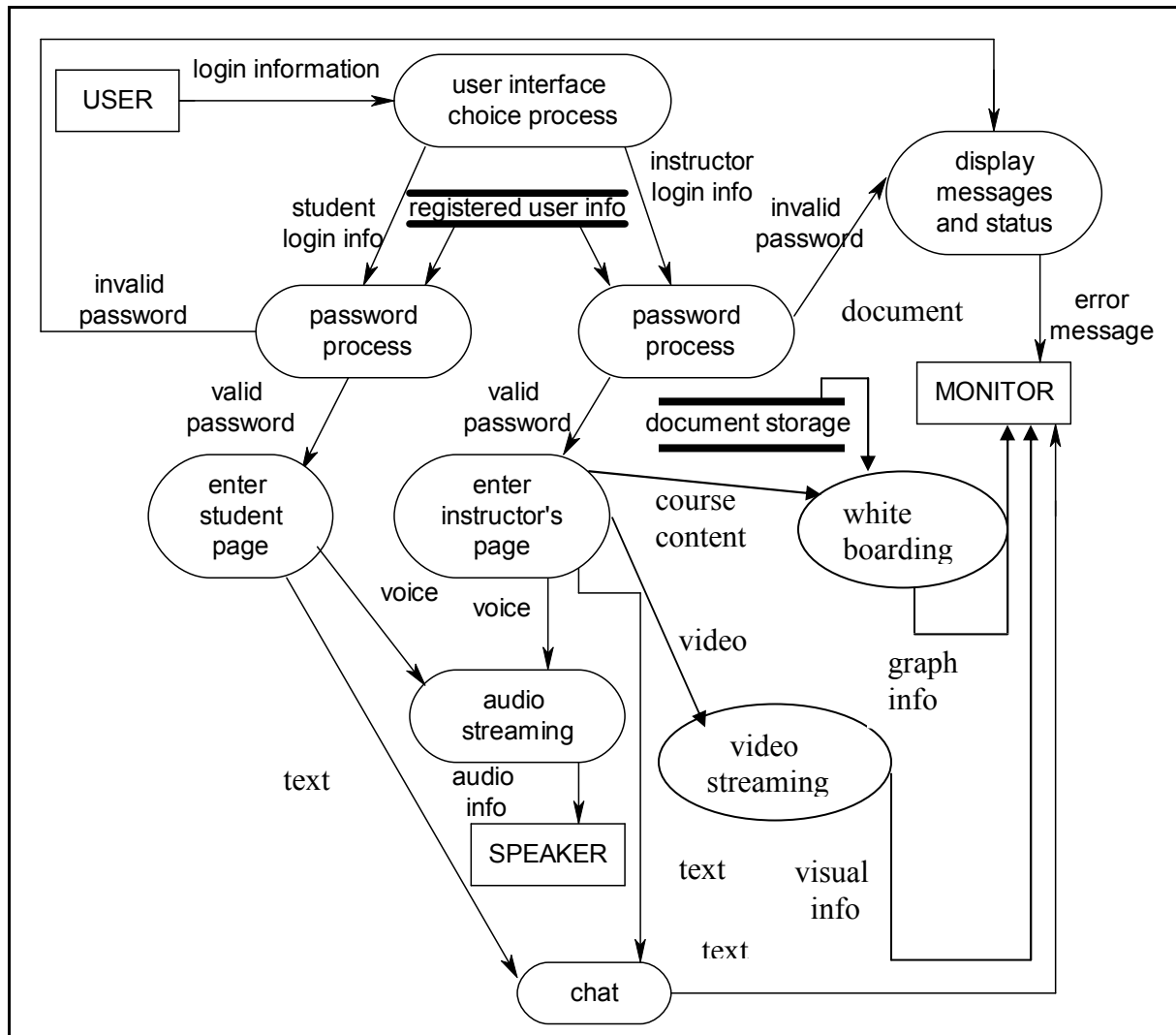
SLogoutTime: DATE (time student logs out)

4.3 FUNCTIONAL MODEL

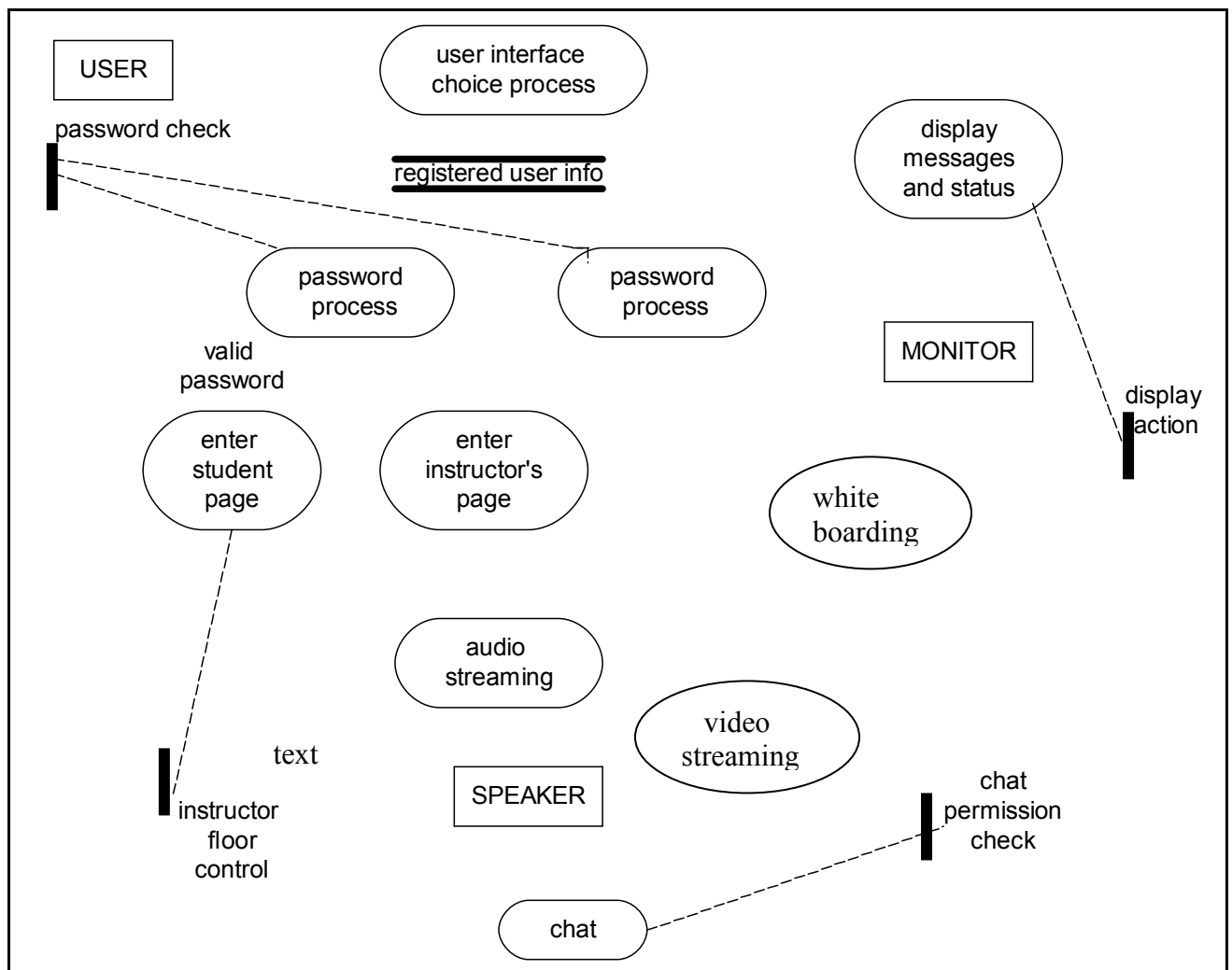
4.3.1 Data Flow Diagram

Level 0:



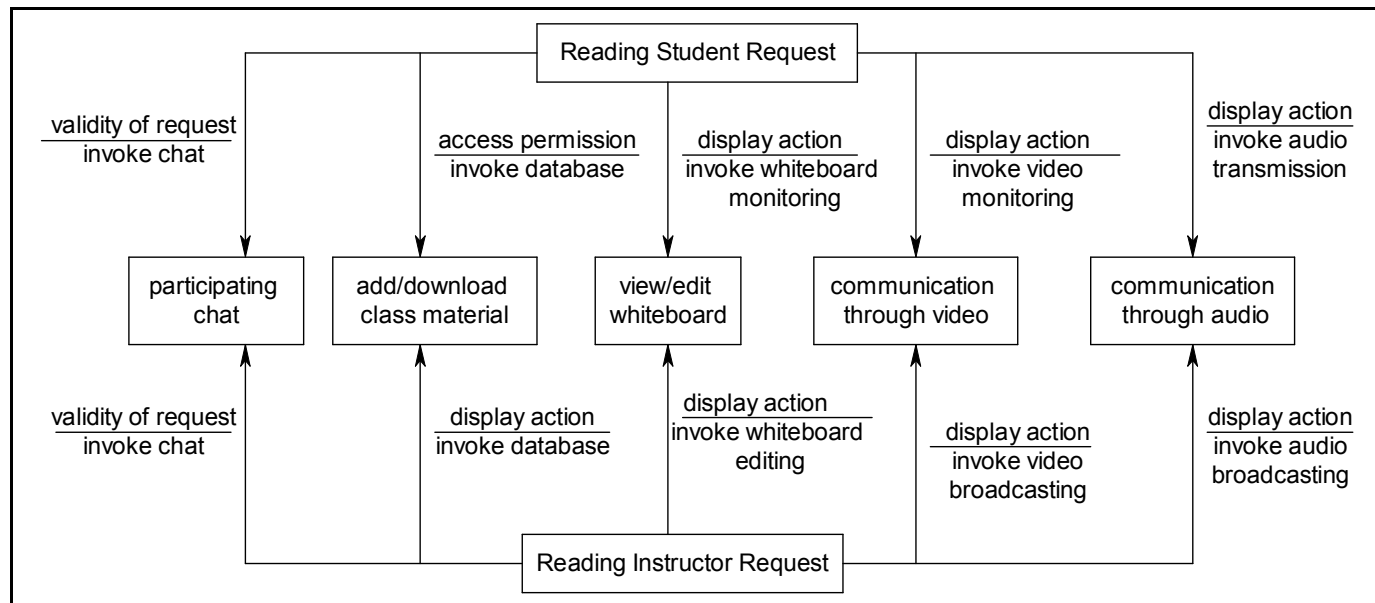
Level 1:

4.3.2 Control Flow Diagram



4.4 BEHAVIORAL MODEL

4.4.1 State Transition Diagram



5. COST ANALYSIS AND SCHEDULING

5.1 TIME AND EFFORT ESTIMATION

There are several ways to do estimation. We did our estimations according to FP oriented calculation.

	optimistic	likely	pessimistic	estimated count	weight	FP count
Number of inputs	22	25	27	24	4	96
Number of outputs	13	16	18	16	5	80
Number of inquires	15	17	20	17	4	68
Number of files	4	4	5	4	10	40
Number of external interfaces	5	5	6	5	7	35

Count total						319
--------------------	--	--	--	--	--	------------

Inputs: Password, chat sent, video sent (3), audio sent (3), question, handraising, understanding, whiteboard (10), floor giving

Number of inputs: 22

Outputs: Result of password entry(accepted/rejected, 2), requesting floor(accepted/ rejected, 2), changing audio settings, video broadcasting, editing whiteboard, polling, quizzing, participating chat, downloading course material.

Number of outputs: 13

Files: Audio, video, documentation, tables

Number of files: 4

External Interfaces: Login interface, instructor interface, student interface, document Interface (2)

Number of external interfaces: 5

1	Does the system require reliable back up and recovery?	5
2	Are data communications required?	5
3	Are there distributed processing functions?	1
4	Is performance critical?	3
5	Will system run in an existing, heavily utilized operational environment?	3
6	Does the system require online data entry?	5
7	Is the system designed for input transactions over multiple screens?	1
8	Are the master files updated online?	5
9	Are the inputs, outputs, files or inquires complex?	2
10	Is the internal processing complex?	4
11	Is the code designed to be reusable?	5
12	Are conversion and installation included in the design?	5
13	Is the system designed for multiple installations in different organizations?	4
14	Is the system designed to facilitate change?	4
	Complexity adjustment factor	51

$$\Sigma F_i = 51$$

$$\text{Count Total} = 319$$

$$FP = \text{Count Total} \times (0.65 + 0.01 \times \Sigma F_i)$$

$$= 319 \times (0.65 + 0.01 \times 51)$$

$$= 370$$

Since five people are working in our team, and we have seven months to complete the project, we have to show a performance of about 10 fp/m totally. This means that each member of the team should have a productivity of 2 fp/m.

As java will be used in the implementation part of the project, we are going to have a LOC value of $370 \times 53 = 19610$. Since our project is large, we are unfamiliar with this topic and there are severe time constraints, embedded COCOMO model [5] will be used.

$$\text{Effort} = 3.6 \times \text{KLOC}^{1.2} = 3.6 \times (19.61)^{1.2} = 136 \text{ pm}$$

$$\text{Duration} = 2.5 * \text{EFFORT}^{0.32} = 12 \text{ months}$$

According to these measures, we need five extra months to complete the project. Considering this result, in order to keep with the schedule, we should improve our performance.

5.2 SCHEDULE

Start Date	End Date	TASK	2004			2005				
			Oct	Nov	Dec	Jan	Feb	Mar	Apr	May
25.9.04	6.10.04	I- Project Topic Determination								
		II- Proposal Phase								
7.10.04	15.10.04	1. Survey on project features								
		MS: Proposal is finished.								
		III- Requirement Analysis Phase								
15.10.04	31.10.04	1. Problem identification								
23.10.04	25.11.04	2. Survey on tools								
23.10.04	19.11.04	3. Having meetings with company								
1.11.04	25.11.04	4. Comparison of tools								
15.11.04	7.12.04	5. Decision of development tools								
26.11.04	8.12.04	6. Requirement definiton								
		MS: Requirement analysis is done								
		IV- Design and Prototype Phase								
9.12.04	23.12.04	1. Database design								
17.12.04	13.1.05	2. Network design								
17.12.04	13.1.05	3. Multimedia design								
24.12.04	13.1.05	4. Graphical design								
31.12.04	13.1.05	5. GUI interface								
		MS: Design & prototype are finished								
16.1.05	12.5.05	V- Implementation Phase								
27.4.05	31.5.05	VI- Test Phase								

6. MANAGEMENT DECISIONS

6.1 RISKS ANALYSIS AND MANAGEMENT

It is nearly impossible to develop a system with no defect. By taking the right plan for managing the risk, the defects and error rates of the software can be reduced. In the meetings we identified several risks in advance. These risks have been documented below and will be reviewed on a regular basis to ensure that the project is not adversely affected. We will give more consideration to the risks related with incompatibility of development tools. Furthermore, potential risks will present themselves during the project and will be identified and discussed as they appear.

- Product Size Risk (The wrong estimation of LOC or FP)
- Process Risk (Ill defined process & poor quality)
- Technology Risk (The new technology to the project)
- Team Member Risk (Team inexperience)
- Development Environment Risk (Incompatible tools)

6.2 SOFTWARE CONFIGURATION MANAGEMENT (SCM)

SCM is primarily focused on the identification and control of the changes made during the software development. In our project, if one of the team members lodge a claim of modification in some part of the project, by the approval of the other team members, the necessary changes will be adapted. In the decision phase, Software Quality Activities will be applied. The configuration system tool (CVS) is not used at this stage. When it is in use, the version control will be done appropriately to keep the changes.

6.3 SOFTWARE QUALITY ASSURANCE (SQA)

SQA is necessary to complete the tasks accurately without the waste of time, effort and materials. To achieve this, SQA provides mechanisms for paying close attention. In our project, we take account the quality in our weekly meetings. Quality will be assured with short coding and long testing phase which will prevent rework and decrease debug time in continuing phases.

7. METHODOLOGY

7.1 PROCESS MODEL

The application is to be developed using an evolutionary software process model. Spiral model provides the greatest efficiency for the development of large scale systems like our project. This life cycle allows each function to be completed prior to the next function being commenced. We will be able to show our improvements at each rotation. This model allows software to evolve as the process progresses so we will better understand risks at each evolutionary level. Since any little failure in the software may result in serious hazards, it is very important to identify the possible errors in every stage by direct consideration at all steps of the project we will have the opportunity to reduce risks before they become problematic.

7.2 TEAM ORGANIZATION

We decided to organize our project team as control decentralized (CD). CD team organization is the most appropriate structure for our software because a defined leader will be responsible for ensuring that all tasks are completed in a timely manner but there will not be a strict hierarchy between team members. Besides the vertical communication along the control hierarchy, there will also be a horizontal communication between group members although each team member will be allocated a particular role within the project team, it is expected that each team member will assist the other team members where the need arises.

8. REFERENCES

- [1] Tes Group, “Bir Uzaktan Eğitim Sistemi Nasıl Değerlendirilir”, 2003
- [2] “iLinc Communications: Web and Audio Conferencing Software for Collaboration”
<http://www.learnlinc.com>
- [3] “Centra Web Conferencing, Online Meeting and Virtual Classroom Applications”
<http://www.centra.com>
- [4] “Horizon Wimba Open Campus”
<http://www.horizonlive.com>
- [5] “JSP 1.2 Specification Proposed Final Draft 2, Sun Microsystems, Inc, April 27,2001”
<http://java.sun.com/products/jsp/index.jsp>
- [6] “Software Estimation”
http://www.spc.ca/resources/metrics/software_estimation.pdf