

# **KHAZAR UNIVERSITY**

**School:** Architecture, engineering and applied sciences

**Department:** Computer Science

**Major:** Computer Engineering and Management

## **MASTER THESIS**

**Title:** **Development of web-based application for distant learning system.**

**MS Student:** Jamil Gasimov

**Supervisor:** PhD. Sakhavat Talibov

December - 2009

# KHAZAR UNIVERSITY

**School:** Architecture, engineering and applied sciences

**Department:** Computer Science

**Major:** Computer Engineering and Management

## MASTER THESIS

**Title:** Development of web-based application for distant learning system.

**MS Student:** Jamil Gasimov

**Supervisor:** PhD. Sakhavat Talibov

December - 2009

## **KHAZAR UNIVERSITY**

**School:** Architecture, engineering and applied sciences

**Department:** Computer Science

**Major:** Computer Engineering and Management

### **MASTER THESIS**

**Title:** **Development of web-based application for distant learning system.**

**MS Student:** Jamil Gasimov

**Supervisor:** PhD. Sakhavat Talibov

December - 2009

## **KHAZAR UNIVERSITY**

**School:** Architecture, engineering and applied sciences

**Department:** Computer Science

**Major:** Computer Engineering and Management

### **MASTER THESIS**

**Title:** Development of web-based application for distant learning system.

**MS Student:** Jamil Gasimov

**Supervisor:** PhD. Sakhavat Talibov

December - 2009

**Table of contents**

Introduction	2
Chapter I Concept, principles and features of distance learning	
1.1 Basic concepts	4
1.2 Principles of distance learning	5
1.3 Features of distance learning	8
1.4 Use of the World Wide Web in remote training	9
Chapter II The review of means of remote training	
2.1 Requirements to system electronic textbook	13
2.2. The requirements for test formation:	13
2.3 Requirements to the development environment of system of electronic book	18
2.4 Tool Book II Assistant	18
2.5 Lotus LearningSpace	22
2.6 eXtensible Distance Learning System	25
Chapter III Development of interactive web-based applications	
3.1 The Nature of Web-Based Applications	42
3.2 Client-Server Web-Based Application Architecture	42
3.3 Tools for creating web-based applications	42

3.4 Efficiency and cost	46
Chapter IV Description of the “Khazar University” System	
4.1 Server	47
4.2 Users	48
4.3 User interface	54
Conclusion	56
Literature	57
Appendix A	58
Appendix B	59
Appendix C	64

## **Introduction**

One of the most actively discussed forms of services in last years in sphere of education is a study with use of global network Internet or distance learning. There are various ideas and opinions on expediency of such form of training, and sometimes they are absolutely opposite to each other. It is normal if to consider, that recently almost all types of trainings at which even the part of materials handed out for independent studying was considered as a distance learning. It also didn't matter what type carrier were used: paper or magnetic.

Most recently, the tendency to greater clarity was outlined. Distance learning, are even more often connected with the closed system of training, at which the basic means of dialogue, training and transfer of the information is the Internet. The web-technology developed in 1991 becomes that environment where distance learning is most naturally and effectively entered, not excepting uses of other means, for example, e-mail, instant messaging, etc.

Being under the form in intermediate position between full-time tuition and correspondence, remote training is the phenomenon absolutely different, not reduced to first two. Specially developed environment should provide the full tooling, allowing to train individually, to provide all information support according to curricula, testing and self-testing, system of final control actions and etc. There are quite enough means for this purpose, considering a variety of already created computer technologies. Here again, probably, already soon there will be a necessity for standardization of approaches.

The present work puts before itself two purposes: to systematize the requirements to distance learning systems, and principles by which it is necessary to be guided at creation of such systems, and also to offer own development implementing one of the distance learning subsystems, opening opportunity of asynchronous dialog between the teacher and student by the Internet.

The basic results of work are classification of scientific and didactic principles of distance learning, the description of its features, the analysis of the basic distance learning systems and development of own software having practical value for high schools.

Novelty of results consists that:

- In development are used MySQL database system and PHP language, which give ample opportunities and widely widespread on many platforms;
- The object model of programming are used, noticeably raising readership of the source text and facilitating system addition by applying of new components.
- New system for generating examination sets of questions depending on their complexity.

Results application level: the designed system can be implemented on a test basis at faculty of Computer Engineering of Khazar University.

Practical value: results can be used in high schools as practical guidance for distance learning programs design. "Khazar" system can be applied as ready development.

The summary of chapters.

The first chapter is devoted to classification of distance learning principles and features.

In the second chapter the comparative analysis and characteristics of some means of remote training is resulted.

In the third chapter I described some common principles of developing web-based applications and counted several development tools.

The fourth chapter is devoted to the description of own development of "Khazar" for asynchronous dialogue of teachers and students.



## **CHAPTER I. Concept, principles and features of distance learning.**

### **Basic concepts**

*Training* is the purposeful, regular, organized process of arms by knowledge, skills, and *education* is a result of training, upbringing and development of the person.

*Distance learning* (DL) is the type of reception of education, alongside with full-time tuition and correspondence education at which in educational process the best traditional and innovative methods are used, means and the forms of learning based on computer and telecommunication technologies.

Basis of educational process at distance learning formed by purposeful and controllable intensive independent work of a trainee which can study in a place convenient for, under the individual schedule, having at itself makes the complete set of special materials of training and the coordinated opportunity of contact to the teacher by phone, electronic and usual mail, and also full-time tuition.

Distance learning represents purposeful interactive, asynchronous process of interaction of subjects and objects of training among themselves and with means of training, and process of training is indifferent to their spatial arrangement. Educational process passes in specific pedagogical system which elements are subsystems: the purposes of training, the content of training, methods of training, means of training, organizational forms of training, educational – material, financial and economic, standard-legal.

In an education system distance learning meets a humanity principle according to which nobody should be deprived an opportunity to study by reason poverty, geographical or time isolation, social vulnerability and impossibility to visit educational institutions by virtue of physical defects or industrial and private affairs occupation. Being consequence of objective process of informatization of a society and formation and incorporating the best features of other education forms, distance learning will enter into XXI century as the most perspective, synthetic, humanistic, integrated form of reception of education.

There are also other treatments of distance training and education reflecting variety of approaches to their understanding:

*Distance learning* – the special, perfect form combining elements full-time tuition, mixed full-time and correspondence, correspondence and evening training on the basis of new information technologies and systems of multimedia. Modern means of telecommunications and electronic publications allow overcoming lacks of traditional forms of training, keeping thus all of them advantages;

*Distance learning* – a complex of the educational services given to large groups of population in the country and abroad by means of the specialized information educational environment, based on means of an exchange of the educational information on distance (satellite TV, radio, computer communication, etc.). The information-educational environment distance learning represents the system-organized set of means of data transmission, information resources, reports of interaction, the hardware-software and organizational-methodical maintenance, focused on satisfaction of educational needs of users. Distance learning is one of forms of continuous training which is called to realize human rights on education and reception of the information;

*Distance learning* – the new organization of educational process which are based on a principle of independent training of the student. The environment of training is characterized by that pupils basically, and often and absolutely, are remote from the teacher in distant area and (or) in time, at the same time they have an opportunity at any moment to support dialogue by means of telecommunication.

In our time when telecommunication speeds increase dramatically video conferences and online discussions can be implemented.

## 1.2 Principles of distance learning

### Scientific principles

**Principle of integration of data.** The essence of a principle of integration of data consists in association separate, mutually not connected data in a single whole, therefore all data are given to the user as uniform array of information. Thus search of the interconnected data and their joint processing is facilitated, redundancy of data decreases, process of work with informational arrays becomes simpler.

The principle of integration assumes interdependence (connectivity) of data, and connectivity together with a principle of a composition allows reducing redundancy of data to a minimum.

There are 2 levels of integration: logical and physical. At a logical level variety of data structures are connected among themselves in uniform structure of data.

**The principle of consistency** of data reflects the requirement of adequacy of data in an informational array to a condition of a subject domain: at any moment data should correspond to properties and characteristics of reflected objects that is reached by the control of the entrance information, periodic checks of stored data, restoration of data, etc.

**The principle of independence** of data means such independence of applied programs from the stored data so any changes do not demand correction of these programs. That is reached by introduction of levels of abstraction of data (a principle of multiple levels): the logical level shares on two – external (a level of the user) and conceptual (the general system level of data).

**No redundancy** is a condition of data when each of them is present at an informational array as unique instance. Realization of the given principle in system is promoted by a principle of the integration, allowing to reduce redundancy to a minimum.

**The principle of consistency** of data means semantic conformity between given data.

**The principle of connectivity** of data considers that all data are interconnected, and communications reflect attitudes between objects of a subject domain.

**The principle of centralized management** consists in transfer of all functions of management of data to a uniform group of operating programs – a database management system, that is all the operations connected with access to a database are carried out by a core of management of databases.

**The principle modularity of systems** consists that the information is structured by a modular principle, and, modules are considered, on the one hand, as components of the whole, giving in to rearrangement and interface, on the other hand, - as target functional unit in which the maintenance and technology are incorporated in system of a high level of integrity.

### **Didactic principles**

**Principle of humanity of learning.** This principle is defining in system of continuous intensive training and amplifies with reference to distant learning systems. Its essence consists in an orientation of training and educational process as a whole to the person; in creation of as much as possible favorable conditions for mastering trained socially saved up experience concluded in the content of training; learning and studying of the selected trade for development and display of creative individuality, high civil, moral, intellectual qualities which would provide to it social security, safe and comfortable existence.

**Principle of advancing education:** not only transfer of already saved up knowledge, a cultural and historical heritage, but also formation of consciousness and scientific outlook of a trainee.

**Principle of a priority** of the pedagogical approach at designing educational process in Distant Learning Systems. The essence of the named principle consists that designing distant learning systems is necessary for beginning with development of theoretical concepts, creations of didactic models of those phenomena, which are supposed to be implemented. Experience of a computerization allows improvement, that when the pedagogical party is priority, the system becomes more effective.

**Principle of pedagogical expediency of application of new information technologies.** It demands a pedagogical valuation of efficiency of each step of designing and creation distant learning systems. Therefore on the foreground it is necessary to put

not introduction of techniques, but corresponding substantial filling of training courses and educational services.

**Principle of a choice of the content of training.** The content of education distant learning systems should correspond to normative requirements of the State educational standard and requirements of the market.

**Principle of a security of the information circulating in distant learning systems.** It is necessary to provide if necessary organizational and technical ways of secure and confidential storage, transfer and use of the necessary data, maintenance of its safety at storage, transfers and use.

**Principle of a starting educational level.** Effective training in distant learning systems demands of the certain set of knowledge, skills. For example, for productive training the candidate for study should be a familiar with scientific bases of independent educational work, should have the certain skills of the computer usage, etc.

**Principle of conformity of technologies of training.** Technologies of training should be adequate to models of distant learning. So, in traditional disciplinary models of training as organizational forms of training (kinds of classes) lectures, seminar and practical employment, imitating or business games, laboratory researches, independent work, an industrial practice, course and degree works, the control of mastering of knowledge are used. During creation of distant learning systems there can be new models which should be in case of need included in it. As an example of such new models object-oriented or design-information models can serve. Among organizational forms of training in these models computer conferences, teleconferences, information sessions, teleconsultations, design works and so forth will be used

**Principle of mobility of training.** It consists in creation of information networks, bases and banks of knowledge and data for distant learning, allowing the student to correct or supplement the educational program in a necessary direction at absence of corresponding services in high school where he studies. Thus preservation of the information invariant education providing an opportunity of transition from high school to high school on training in related or other directions is required.

**Principle of consistency of distant learning to existing forms of education.** Projected distant learning system can give necessary social and economic benefit under

condition of if created and introduced information technologies become not an alien element in traditional system of higher education, and will be naturally integrated into it.

### **1.3 Features of distance learning**

**Flexibility.** Trained, basically, do not visit regular classes like lectures, seminars. Everyone can study so much, how much it is personally necessary to gain a development of a course, discipline and reception of necessary knowledge on the chosen specialty.

**Modularity.** In a basis of distant learning programs the modular principle is pawned. Each separate discipline or a number of disciplines which are mastered by trainee, create complete conception about the certain subject domain. It allows forming the curriculum meeting individual or group needs from a set of independent training courses.

**Parallelism.** Training can be conducted at overlapping the basic professional work with study, i.e. “on the job”.

**Long-range action.** The distance from the location trained up to educational institution (under condition of qualitative work of communication) is not an obstacle for effective educational process.

**Asynchrony.** That fact is meant, that during training and a trainee can realize technology of training and the doctrine is independent in time, i.e. under the schedule convenient for everyone and in convenient rate.

**Scope.** This feature sometimes name “mass character”. The quantity trained in distant learning systems is not critical parameter. They have access to many sources of the educational information (to electronic libraries, databases), and also can communicate with each other and with the teacher through communication networks or by means of other means IT.

**Profitability.** Economic efficiency is meant this feature of distant learning. The average valuation of foreign and local educational systems distant learning shows, that they manage approximately on 10-50 % more cheaply, basically due to more effective utilization of the existing educational areas and means of IT, and also representation of more concentrated and unified content of teaching materials and orientations of technologies of distant learning on a plenty of trained and other factors.

**The teacher.** It is a question of a new role of the teacher (in Distant Learning System – the tutor) when such functions are assigned to it, as co-ordination of cognitive process, updating of a taught course, consultation, a management of educational projects, etc. Interaction with trained is carried out, basically, asynchronously by means of mail or communication systems. Full-time contacts are supposed and welcomed as well.

**Trainee.** More precisely, a new role of trainee, or as it is more accepted in system distant learning, the listener. To pass distant learning, from it exclusive self-organizing, diligence and the certain starting educational level is required.

**New information technologies.** In distant learning are used, mainly, Network Information Technologies (computers, audio-video equipment, systems and means of telecommunications, etc.)

#### **1.4 Use of the World wide web in remote training**

The international network Internet gives greater opportunities for institutions of education. Widespread, one of the cheapest, reliable, it gives the richest opportunities for the organization of distance learning. The Internet gives some types of services on the basis of which there is an opportunity of installation of system of support of distance learning.

The most suitable for the organization of remote training is system WWW. WWW it is constructed on technology in which basis the concept of the hypertext is necessary. This system is one of the most flexible means of search in the Internet. Hypertext model WWW can provide the description of any volume for each element. It also has more ample opportunities by way of granting the universal interface. Each article contains references to other messages. The browser will transform the materials received by e-mail, to the hypertext, transforming cross references in hypertext. Hence, you can move between initial articles and articles continuing discussion, and under cross references, using the established communications.

The mentioned advantages of system WWW allow creating a following spectrum of educational systems with access through the Internet:

- Interactive textbooks and manuals;

- Anonymous qualifying and trial tests;
- Tests and examinations for students of remote training;
- The organization of a feedback the teacher-student.

WWW allows to combine text, graphic, sound and video data. Use Java and JavaScript allows to create the applications loaded on a network that removes a problem of updating of the software. Cross platform compatibility of Java provides correct work of curriculums on various platforms without modification in an initial code.

Now is necessary to tell some words about design of the teaching material used during training.

In general, Web gives practically unique opportunities of the creative approach to design of the documents. Any convenient form of representation of a material to the reader can be chosen. It can be photos, schemas, figures, the text, video fragments, etc. All this can be located on page in the any design form. But in this case, when it is a question of reading and consideration of a material with the purpose of its understanding and storing, probably, it is necessary to develop the uniform concept of design of documents. Perhaps, even definition of similarity of the standard which will be used in all without exception cases of the publication of a teaching material will be useful.

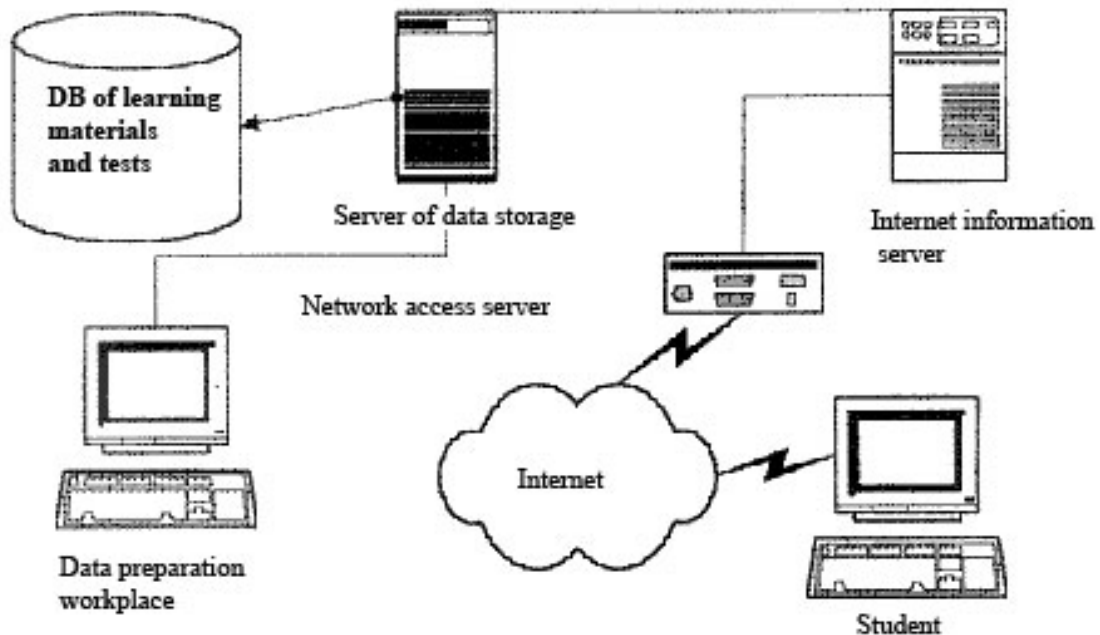
It is already proved, that a degree of understanding information from the screen of the monitor are 25 % less, than when reading from the paper. It causes some simple principles which can help at design and publication.

First of all, it is not necessary to use any graphic representations placed on a background of page. The background, most likely, should be monotonous, but unessential white. If a background is white the text should be written in dark colors, for example, black or dark blue. You should not use a dark background and a light font is will tire eyes of the reader. When selecting sets of fonts that will be used it is necessary to recognize that readership of the text written without serif, is above, than of the text written with serif. Thus it is necessary to refuse completely use of the small sizes of font set.

It is necessary to consider also, that pages will be looked through in systems with the different graphic resolution and depth of color. Probably, it is necessary to use only a palette in 16 colors that will allow excluding distortions at display of the text allocated by color. It also concerns a graphic material (schedules, schemes, diagrams, figures and to



that similar). Exception can be made only for photos where it is necessary to apply a palette at least of 256 colors. It will provide the minimal quality at display of photos in the browser. Besides use of these color palettes, and also the graphic formats supporting compression of the image (such as GIF and JPEG), will allow to reduce total amount of the documents transferred on a network for display.



Now we shall discuss the content of documents. The distance learning is just that case when the content of documents should prevail of the form of its representation. The form of all documents should be as more as possible strict. The page should not contain any superfluous information (graphic or text) which could distract attention of the reader. The zone of reading of the document should be narrower. It can be promoted by creation of space on edges of the document. You should avoid creating documents with horizontal scrolling because it is very uncomfortable to read and study them. The size (amount) of each separate document should not exceed 30-40 KB that will allow to accelerate their loading.

Needless to say, that all teaching materials should be precisely structured and whenever possible should give an opportunity of interactive search among them. Teaching materials can be transferred to the person interested in their studying, by any way. It can be both direct loading and reading of the document from a Web-server, and transfer of

archive of documents for viewing by the program-client without connection to the Internet, and also other ways which will be caused by a technique of educational process.

## **CHAPTER II. The review of means of remote training**

### **2.1 Requirements to system electronic textbook .**

In a basis we shall put the following principles for environment of electronic textbooks.

For effective functioning of the person in electronic training system without dependence from the problem solved by the researcher, special role are playing different methods of visualization of initial data, the intermediate results of processing providing the uniform standard of representation of the current and final information in the form of displays, adequate to visual perception of the person and the received results convenient for unequivocal interpretation. The important requirement of the interface is its intuitivism. It is necessary to notice, that operating elements of the interface should be convenient and appreciable, at the same time they should not distract from the basic maintenance, except for cases when operating elements are the basic maintenance.

The textbook can be structured as the usual book. In the beginning of the textbook there is a description of navigating elements of the textbook, such, as buttons of transition to pages, buttons of a call for the help, content and tables of contents. Then, on page "content" items of the basic sections of a material presented in the textbook, in the form of "hot" words contain, pressing by a mouse on which occurs moving to page specified in content. In the end of each section control questions or a various sort test tasks on the passed material are located. To put marks to answers which show, how the offered

material has been acquired, in the end of the electronic manual probably also to generalize all the received marks and to put a final mark at the passed rate. Depending on requirements for each concrete student, i.e. from a degree of intellectual development, psychological stability and various other personal factors, the teacher or the assistant of the course can suggest to repeat those sections on which the mark can be considered unsatisfactory.

## **2.2. The requirements for test formation:**

There are various kinds of tests, as for testing "closed" form of testing may be used.

Question or statement and also few versions of answers are presented to the user.

The quantity of answers and right answers can be unlimited. Test can be defined as effective in case if it satisfies all requirements:

The main requirements are following:

- reliability
- validity
- discriminativity

Lets discuss and define all terms which need to be understood very good in order tests to be not only constructed in right manner but also could be used properly.

### **Reliability.**

Reliability of the test is a characteristics of the methods of creating tests that reflexes accuracy of the psychodiagnostic measurements and also stability of test result against the influence of foreign random factors.

Reliability can be of several types:

1. Retesting reliability. When results that are achieved during repetitive researches of the group of people using the same tests for some period of time are being compared;

2. Reliability of the parts of the test. Is being calculated according to analysis of the stability of the results of particular munity of the test cases or different parts of the test.

Retesting reliability. When test can not achieve the same result for a particular person (in condition that this person did not change) in different cases and conditions this means that something is wrong with the test. Methodology of measurement of retesting reliability is very simple. Correlation of the results for a set of experimentators tested using two cases is being calculated. The minimal satisfactory value for retesting reliability is 0,7. Shown discreet coefficient of reliability is conditionally in some means. For projection and some other tests of the personality value of the retesting reliability may be lower. But thia does not influence the diagnostical usefulness of the methods.

Lower indexes make the usage of the test fairly expedient, because standard error of results received for that test would be so big, that interpretation of the results would be doubtful.

### **Coefficient of reliability.**

The average value of correlation for one test or task with all tests and tasks from general majority is called factor of reliability. Square root of reliability factor is correlation of given test or task with true index.

But in practice it's very hard to calculate theoretic meaning of  $r_{ii}$ , because the number of developed tests and tasks is not unlimited. This means that reliability  $r_{ii}$ , of some test may be estimated approximately.

So in practice coefficients of reliabilities are based on correlation of one test with others and this estimate may be inexact. And this means that the most important meaning of test or task correlation with true index also may be estimated inexactly.

### **Reliability and quantity of tests.**

The reliability of test increases according to quantity of tests. From the developer's point of view the increase of reliability in accordance with increase of task quantity is very important. It's always hard to develop big number of valid tasks ( for example such tasks which belong to needed general complex of subjects), therefore if we want to

demonstrate that the reliability of twenty five tasks ( with given average correlation) is big means that the achievement of this goal will have meaning.

Coefficient of association for test reliability evaluation. Tightness of connection between qualitative characteristics X and Y is measured with a help of association coefficient, where X stands for kind of test and Y for results of testing. In the simplest view the formula which shows the calculation of this index is following:

$$r_a = \frac{ad - bc}{\sqrt{(a+b)(c+d)(a+c)(b+d)}}$$

Where a,b,c,d, stand for quantity of correlated groups

Coefficient of association identical to Pirsonov's coefficient of correlation is changed from -1 to +1. The importance can be checked by t-criteria of Styudent. The zero hypotheses, which claims that in general complex the index = 0, is rejected if:

$$t_f = \frac{r_a \sqrt{n-2}}{\sqrt{1-r_a^2}} \geq t_{st}, \quad (1.2)$$

Where

N – the quantity of tested

$r_a$  - factor of association

$t_{st}$  - criteria of Styudent

For accepted importance level ( $\alpha$ ) and number of variation steps  $k = n - 2$ .

As coefficient of association is in direct connection with Pirsonov's criteria  $\chi^2$ , on which it's based, distribution of credible values of criteria  $\chi^2$  is continuous. Qualitative

characteristics are discrete, their numeral values are not distributed continuously. Taking into account such feature, they usually add amendment of Yets for continuous of variation equal to half volume of sampling to the formula. And it becomes as shown below:

$$r_a = \frac{|ad - bc| - 0.5n}{\sqrt{(a+b)(c+d)(a+c)(b+d)}}$$

**Validity.** Let's look through second main characteristic of effective tests – validity. The test is called valid if it measures the thing which it should measure. But such definition does not define satisfactory the exact meaning of validity. In this case we have such question: how can we know that test is measuring the thing it should measure? Actually there are lots of ways to prove the validity of test and each of these ways are in accordance with aspects of its evaluation.

It is said that test is obviously valid if it makes an impression of test that measures what it is mentioned to be measured, especially from the point of view of under tested.

Competitive validity – is validity which is evaluated according to correlation of results of this test with the results of other tests. So if we want to set the competitive validity of some test of intelligence, we will study its' correlation with tests which validity is defined.

Informative validity. Such form of validity is mostly related to achievement tests and can be simply defined in following way. If you can show that the task of test reflects all aspects of researched fields of behavior, the test is actually valid, in case if instructions are presented clearly. Informative validity does not boil down to simple obvious validity which is connected with external environment of task. If during the testing of mathematic skills we want to test ability of multiply indexes in brackets, for example  $(y + 2k)(2y - 3x) = ?$ , it's hard to deny the validity of such test.

Obviously informative validity is useful for the tests in which the meaning of measured parameter is clear, as shown in example.

There were described different ways of determining the validity level of tests; some of them differ from each other very strongly. From discussion should be clear that there cannot be one common metric that could demonstrate validity of a test. For complete identification of validity variety of calculated indicators should be considered. More of it the validity of a test can be in fact guaranteed by logically proven methods of test construction.

### **Discriminativity.**

Discriminativity is the ability of the different test tasks or the whole test to differentiate under testers regarding maximum and minimum result of the test.

Discriminativity is one of the characteristics of effective test. It's true that achievement of satisfactory index distribution level is one of the aims of developer of the tests. Getting through detailed design and planning of test can help to achieve satisfactory level of discriminativity and this is the advantage of test in comparison to another form of tasks. Generally it was discovered that nine scales may be used in evaluation but in inquiry it may be more effective to use only three scales – low mediate, mediate and upper mediate. Discriminativity is measured by Ferguson's delta index and takes the maximum value in equal distribution of indexes ( $\delta = 1$ ).

### **Ferguson's method of test evaluation of test discriminativity.**

Ferguson's coefficient is the relation between index of discriminativity, received for particular test and maximum value of discriminativity which can be provided by this test.

$$\delta = \frac{(n+1) \left( N^2 - \sum_{i=1}^N F_i^2 \right)}{nN^2},$$

Where

N – is the number of under tested

n – is the quantity of tasks in test



$F_i$  - frequency of each index's popularity

$\delta = 0$  when all under testers get the same results, and there is no discriminativity .

$\delta = 1$  during equal distribution of tasks, in which possible expressions of measured properties were totally realized.

### **Requirements to the development environment of system of electronic book**

Easy assimilation and usage of this environment for generation of electronic books is reached due to application of visual technologies and possibility of usage any text and graphic editors by specialist for creation of book's content.

Content should not contain big amount of pictures and other multimedia data.

And the environment for creating new content must be user friendly and simple. It should have appropriate level of security.

Also one of the requirements is that interface for creating and modifying content should be available remotely through the internet. That's why it is good practice to use web based interfaces.

For comfortable work in environment of electronic books generation it's allowed to develop project by separate parts. This helps several specialists to organize their work on book simultaneously. This can save a lot of time.

### **Tool Book II Assistant**

#### **Main features of ToolBook II Assistant.**

The software application Tool Book II assistant is mentioned for creation, usage and spread of learning programs available in local versions, in network version for local networks and internet without any programming.

In Tool Book II Assistant there is an interface managed by templates , which helps to conduct learning step by step. The developers may start their work from special

developed module Book Specialist which will take them through main stages of creation of application.

Video, sound and graphics can be easily added in learning program, by drag and drop of files.

As a result you can get learning application which can be used both during traditional classes with teacher and during distance studying on PC.

The developed learning program can be stored on a server in local network, placed on disc or written on CD-Rom. Final application can be easily edited by another developer simply by adding additional pages of material. The main advantage of such approach is in following; teachers can easily create their own individual programs of distant learning.

The package of Tool Book II has standard menu which is located on the top side of the main window and consists of following points:

“File”

“Edit”

“View”

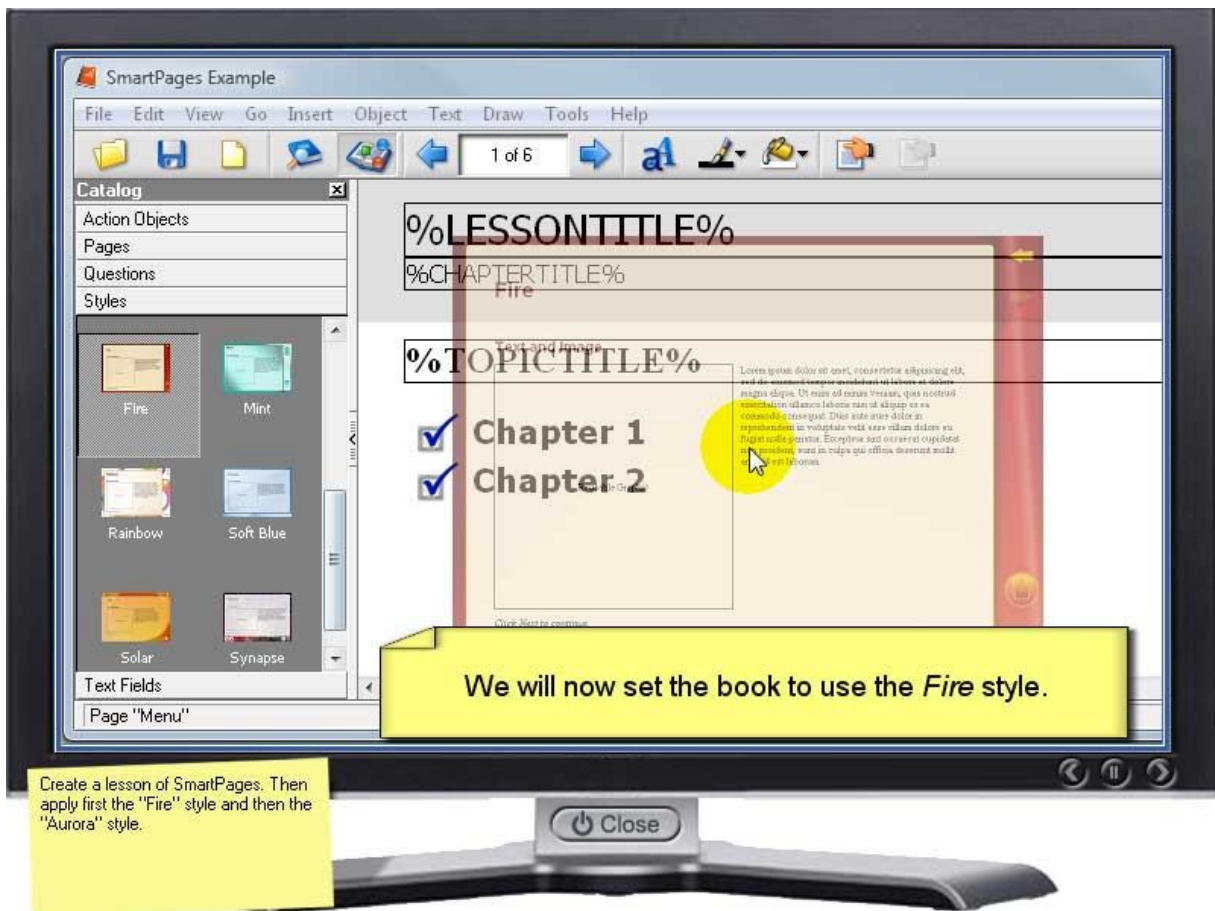
“Page”

“Object”

“Text”

“Draw”

“Help”



### The catalogue of objects in Tool BookII Assistant package.

The usage of object catalogues In Tool Book II Assistant package can be explained by two reasons: first of all it's rational to place such huge variety of objects ( the action of each object is described with a help of Openscript language and user just supplements this action with a help of special dialogue windows and achieves requested result) and graphics in one database;

Secondly the authors of such multimedia package decided to give user opportunity not to programming in language of Open Script, which has been widely used in package Multimedia ToolBook Version 3.0, and such activity helped to create system of visual object oriented programming.

“Graphics” toolbar allows to set graphic image of object in its all positions: normal, inverted, disabled, and marked. This helps user to define individual condition of object in

different situations. Special option “Stretch the graphic” evenly stretches raster image through the whole surface of object.

Toolbar “Limitation” allows manually set sizes and position of object on page and this is distinctive feature of Tool Book Assistant Version 6.0

Toolbar “Font” allows to choose font, style of font, and size of font. Although in package Tool book II Assistant Version 6.0 dialogue window “Letter” is saved, however toolbar “Font” allows to change font properties, without leaving dialogue window of object properties.

Hyper connections in electronic learning books of Tool Book II Assistant Version 6.0 package are created with a help of dialogue window “Hyperlink”. Here you may find what kind of link it is(transition to the page or call of pop up page), final aim of connection( next page, previous page, go back, first page, last page, specific page or e-mail) and effect accompanying hyperlink.

Dialogue window of electronic books Tool Book II Assistant Version 6.0 package properties contains all necessary information about properties of main elements of the book. For example, toolbar “Page” contains unified size of electronic book ( this size is used by default in all pages of the electronic book, but there is possibility to change the size of particular page in dialogue window of background properties if it is necessary).

Organization of the work of controlling objects in Tool Book II.

Objects of control in package of ToolBook II Assistant Version 6.0 – in category of catalogue “Questions” there are different objects, mentioned for control process. For management of mentioned object in Tool Book II Assistant Version 6.0 package is used dialogue window of extended object properties.

Likely the dialogue window of object properties dialogue window of extended object properties uses toolbars (“General properties”, “Answers”, “Evaluation”, “Immediate feedback ” and “Long lasting feedback”)

Tassel “General properties” contains;

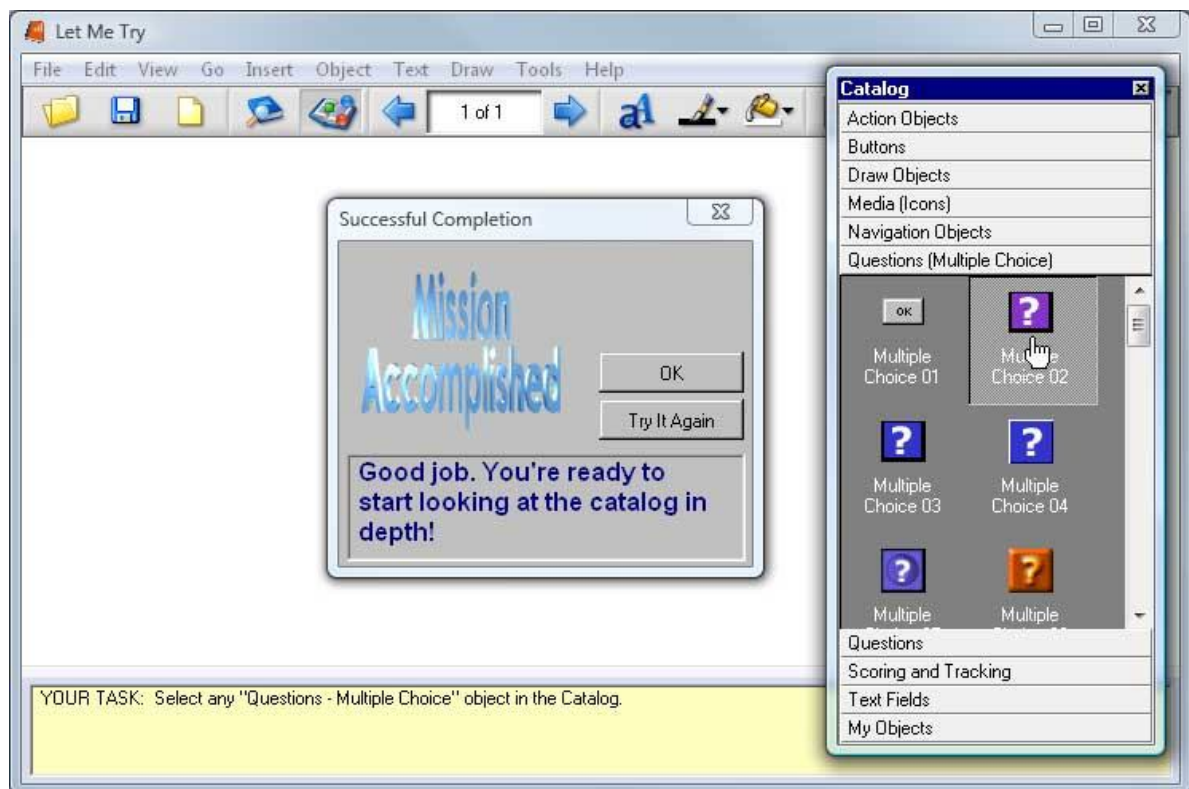
- name of questions
- limitation of interactions
- limitation of time to answer
- limitation of trials

- automatic questions drop
- switching on Internet Support option

Tassel “Answers” contains possible versions of answers; which can be added; edited and deleted, option of accidental question distribution, multiple choices of answers and impossibility to change previously chosen answer.

Tassel “Evaluation” is mentioned for storage of right and wrong answer’s weight and also for delivery of result for tools of evaluation counting.

Tassel “Immediate feedback” and “Long lasting feedback” are used for reaction on right and wrong answers and depending on a result play audio or video spots, outputs messages on particular field of the page or send message by e-mail.



### The usage of Java HTML in package of Tool Book II.

Tool Book II package allows to export electronic learning books as pack of HTML format files for usage in Word Wide Web.

By moving objects described in the language Java from catalogue to the pages of electronic books, by changing their properties and exporting electronic book as files in HTML format, end users can have all advantages of interactive mode which does not depend on operation system.

During exportation of electronic book for usage in World Wide Web, objects supported by Java language and used in electronic books are exported as Java applets. These small applications depend on special code of Java language, which defines their behavior in the Internet.

The conversion of electronic book in HTML format is realized by dialogue window “Export for Web”. Dialogue window “Auto packager” allows creating packed form of electronic book in package Tool Book II Assistant. Options of installation allow choosing one out of five installation methods.

## **2.5 Lotus® LearningSpace™**

Lotus LearningSpace – is integrated environment for working with online study courses and for managing them.

The whole work is done in browser. Learning space supports Web interface and database which is mentioned for organization of courses, work with them and control of work results.

The course consists of set of classes which may be independent, interactive and collective. Independent studies usually include study material and tests. Interactive studies consider visit of lections in virtual classes, and collective studies include participation in debates and chats.

The content of studies may be developed with a help of standard tools of other software developers. Module “Materials” also contain tools for creation of tests and teleconferences, conducted by teachers. After placement of course material in Learning space students may work with it and authors of course may manage it from their Web browsers.

Information about work results of students is kept in relational database, created during installation of Learning Space.

Learning space – is one application with two different interfaces: interface of administrator which is mentioned for management of courses and users and interface of student which is mentioned for work with courses. These interfaces have different addresses.

For example, if Learning Space system is placed on server myserver.company.com, to interface of administrator corresponds URL

*<http://myserver.company.com/LearningSpace5/default.asp?UI=admin>*

Interface of student opens while choosing URL:

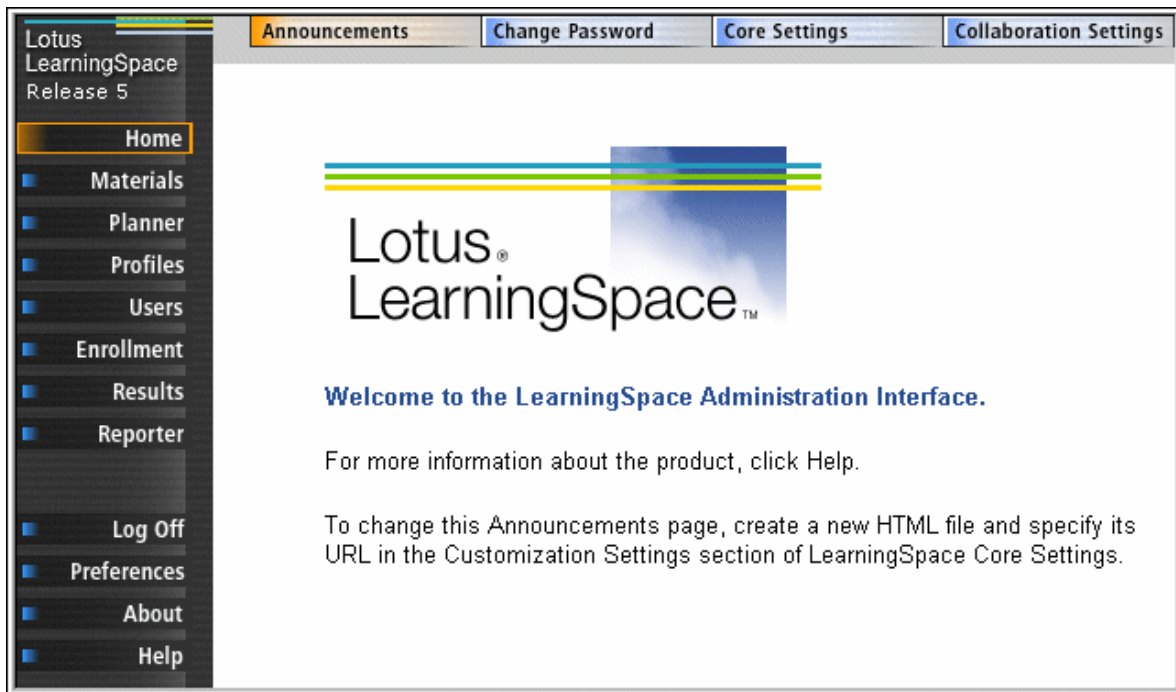
<http://myserver.company.com/LearningSpace5/default.asp?UI=student>

In interface of administrator authors, teachers, registers and administrators of course prepare material, register new students, register them to new courses and follow their results. In interface of student, students may register for new courses, interrupt new courses, and work on tasks of course. Authors use interface of student for testing of created courses. Teachers use interface of students for sending mails, participation in discussions and conduction of teleconferences.

Interface of administrator consists of eight modules located on left side of screen. Each module is responsible for conduction of its own function but changes in properties of one module can affect of what you may see and do in another module.

### **Home.**

This module is used not only for changing user password but also for changing system properties and settings. The page Announcements is opened by default. If you have according rights you may also open pages Core Settings , Collaboration Settings and Domino Settings.



Picture 2. Administrators' interface

### **Materials.**

This module is purposed for creation of different course materials. It Has three main pages : Tests, Teleconferences and Discussions. First two pages have submodes related to different types of created material. Page Tests is opened by default in edition mode.

### **Planner.**

This module is mentioned for changes of course structure and properties of its elements (chapters, classes, and aims). Course structure edition page (the only page of module) is opened by default.

### **Profiles.**

This module is mentioned for setting up the students' and administrators' interfaces user rights and division of courses into categories. Page Profiles is open by default. The second page "Rights" allows changing the rights of certain profiles of users.

### **Users.**



This module is mentioned for addition of user to the system and assignment them particular profile. Page Users is opened by default. The second page – ‘Import’ allows to import at once a group of users from text file of special format.

### **Registration.**

This module is mentioned for registration of students to courses, appointment of teachers and other responsible persons for course and switching on automatic registration. The page registration is opened by default( the only page of module)

### **Results.**

This module is mentioned for review and changes of work results and evaluation of students. The page Results is opened by default (the only page of this module).

### **Reports.**

This module is mentioned for creation of reports about work results and status of student’s registrations and also receipt of information about users connected to the system. The page Reports is opened by default. The second page – ‘Edit’ is mentioned for changes of present reports parameters and creation of new ones.

## **eXtensible Distance Learning System**

This development presents a system for automation (information providing) process of distance learning in different education centers.

The technical realization of system is directed for formulation of integrated decisions. It means the creation of program environment, providing interconnected and agreed decision of different tasks of distance learning process, such as:

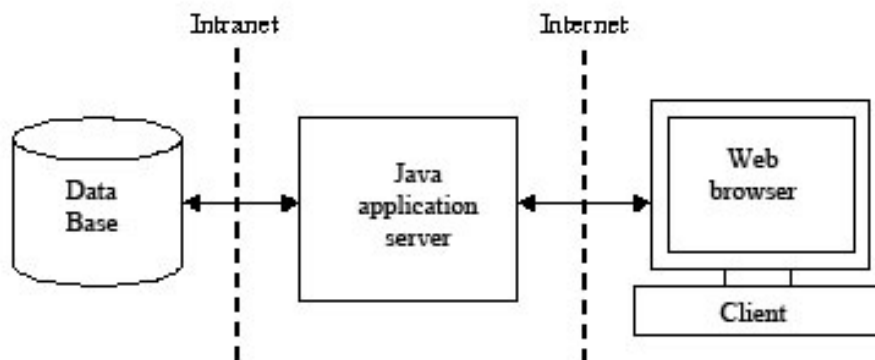
- collection and storage of requested information in different forms of its presentation, and it’s operational update
- giving information to users
- authorization of access to users’ information and realization of multilevel information security system
- administration of system
- effective and fast creation of information resources by users themselves

The system presents all main ways of distance learning organization on Internet:

- providing access to all course materials with a help of internet
- delivery and distribution of course materials
- conducting testing
- providing individual interactive learning courses
- supplying possibility of group work in network
- Storage of system of study information resources ( books, courses, tests and etc) in IMS format in a database , their export and import
- Informing users about current status and results of study process
- Organization of distance learning according to study plans
- Creation of virtual university/class structure (departments, staff and etc)

Architecture of system:

System is built according to architecture – “Client- Server”



The usage of this architecture is standard “de-facto” for information systems of distance learning, as it gives possibilities of remote access of users from internet, combining of different software and hardware solutions in one system, reliable work with huge lack of

information through local and global computer systems, increase productivity of information system by distributing processing data through increasing the capacity of system.

Below you can see more detailed view of system architecture.

There are several types of users. Each type has its particular responsibilities and rights.

The user types are:

Learner – student who takes a course.

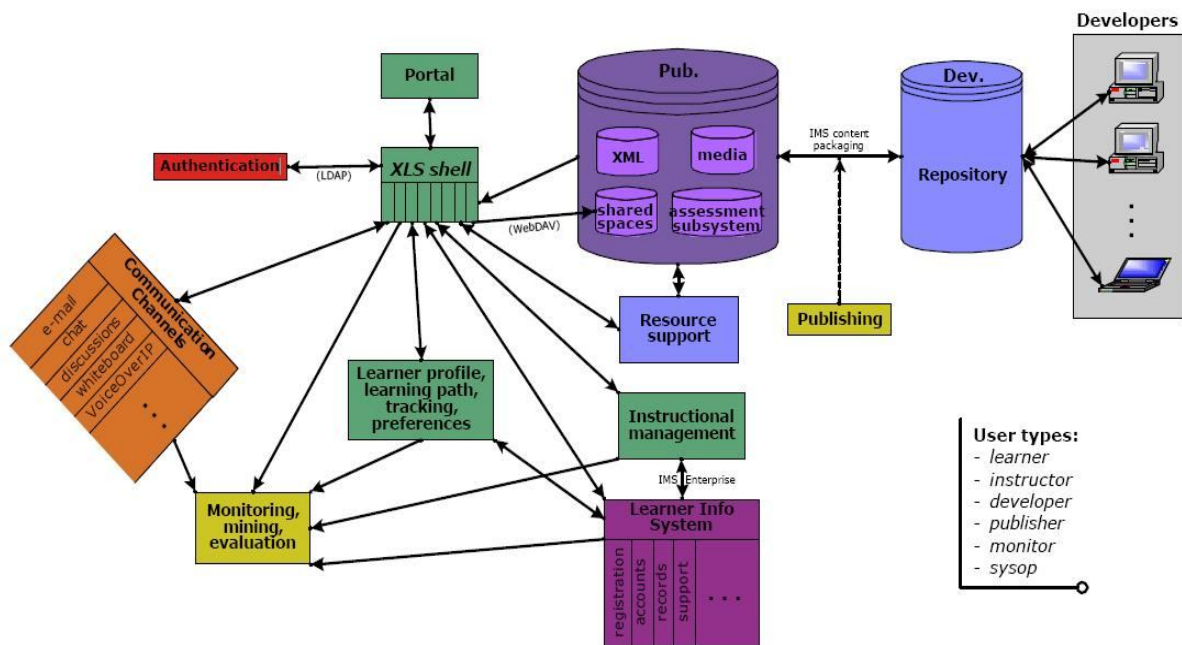
Instructor – person who teaches and leads a course.

Developer – develops new features according to teachers and students requirements.

Publisher – a man who prepares and publishes new learning materials.

Monitor – user that can monitor system activity and load to take appropriate actions if necessary.

Sysop – is system operator. Has full access to the system and performs the role of administrator.

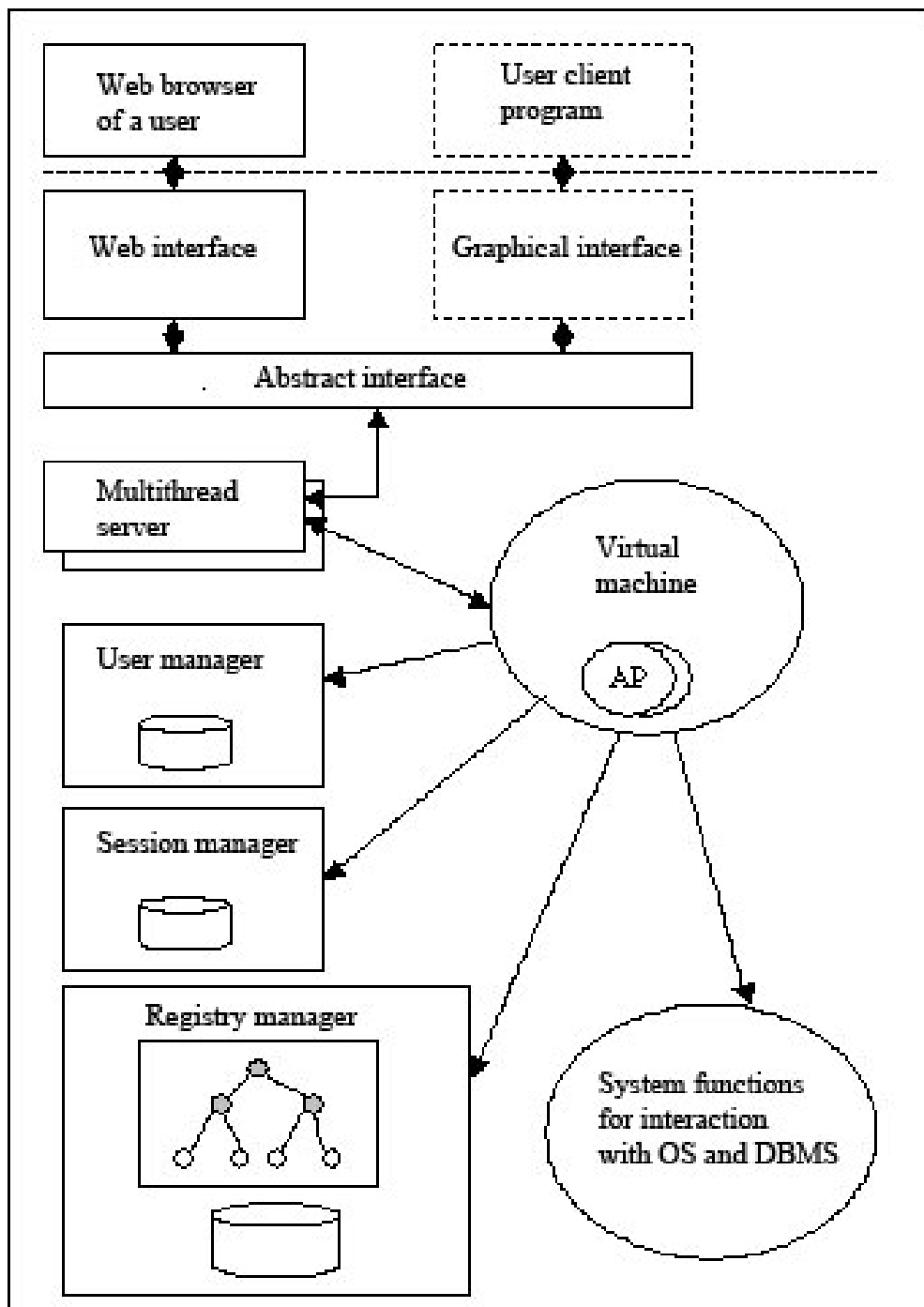


**Server**

Server is structured storage of multimedia informational study resources (such as electronic books, courses, tests) available from Internet

Server of system consists of following components:

1. Interfaces of users ( graphical interface, web-interface – this is the main interface of user) which are mentioned for organization of interaction with client user program (it might be web-browser or other client program).
2. Abstract interface of user- serves as interlayer between particular realization of user interface and system, can be described using XML –derivative interface description language ( for example XUL).
3. Multithread Servlet – component of a system serving for parallel processing of interface requests received from few users at the same time and delivery of this requests to virtual machine.
4. Virtual machine – is the core of the system. There is work of active resources in its context.
5. Registry manager – is used for managing the registry of informational sources of the system.
6. Users manager – is used for managing the users database of the system.
7. Manager of sessions – is used for managing the session records of the users of a system.
8. Component of system functions – contains low-level methods for interaction with operating system's application interfaces (API) and database management systems (DBMS)



For realization of server environment Java is used. It gives such advantages as independence from platform, quickness of development, orientation on network applications and presence of standard libraries of DBMS and Web interfaces support.

**DBMS.**

As Database management system server any well-known server type relational DBMS can be used. It might also depend on scale of systems usage, including multi platform ones. This allows using system on different software and hardware platforms.

If there is large system with many tables with huge amount of data and if system stability and reliability are the main goals you can use Oracle database management system. You can download it from the Oracle website and use it for free for non-commercial purposes. This database provides high level of security and performance. But it can be difficult to administer it for a person without special knowledge.

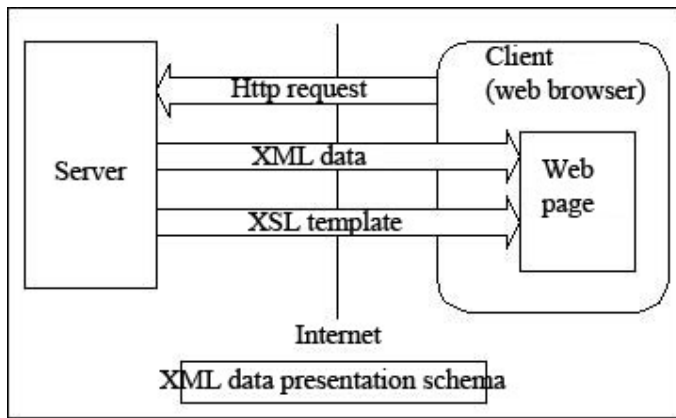
As an alternative MySQL database can be used. It is free and open source product. MySQL is small and robust. It does not have such administrative tools like Oracle. But the main advantage of this DBMS is that it is completely free and it is really fast. It has own mechanisms of archiving data. So all data can be easily backed up and restored if necessary.

### **Ultrathin Client.**

All users get access to the system by using “ultrathin client”, which is presented as web browser.

It makes much easier the development of system (there is no need of writing client’s program) and the usage of system (users can use one of the most popular web browsers, by working with system in usual way for them)

While using as client web browser supporting XML language (for the example Internet Explorer), the system has possibility to output data as combination of XML+ XSL.



Immediate output of information to web browser significantly increase productivity of the system (data comes to web browser as they were kept, and web browser makes conversion to HTML on itself). Therefore we have some kind of distributed computing while most of web browsers on client side perform part of computation during data output.

When components of informational system are geographically far from each other internet is the easiest solution for transportation of information, and this allows using this network for building of different functional information system. Besides that orientation on international standards and typical solutions allows to provide evolution of the system in process of assets perfection, which realize system's functional elements.

### **Storage of data.**

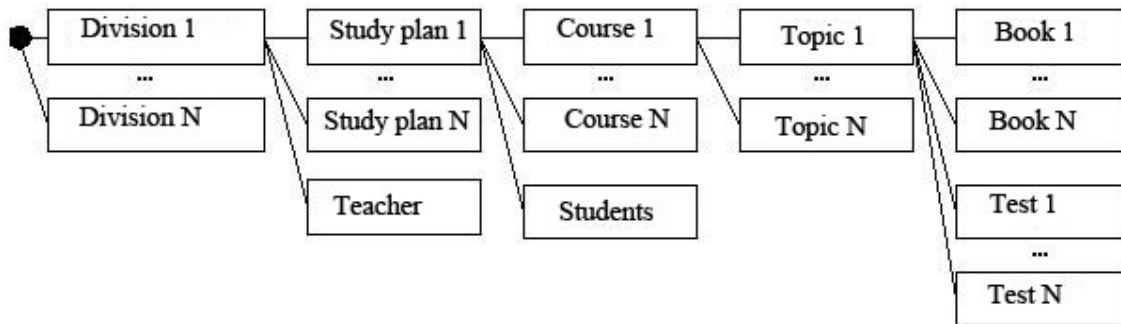
All data of distance education systems includes the variety of informational resources of following types:

- Electronic book
- Test
- Course
- Study plan

These informational resources consist of sub resources (for the example test consists of sections, and sections consist of separate questions and so on). Therefore, most of informational resources are containers containing other resources.

Using such scheme informational resources are performed in the data storage structured as tree according to their type and content.

Below you may see simple example for scheme of distance educational system resources as kind of “Virtual University”:



Such division is naturally catalogued and presents not just organizational and administrative structure of educational center, but also its informational resources.

As storage for tree structured informational resources are used relational database management systems (DBMS) because they are very efficient and reliable. In this meaning DBMS is sublayer and mechanism of tree structured storage of informational resources development (inventory of system) is upper layer.

So new level of abstraction is being discovered – interface, that allows operating with system data on level of separate informational resources. For this case there is special component in the system – registry manager.

Operations over resources are unified and consist of following set:

- creation of resource
- add of resource
- edition of resource
- deleting of resource
- view of resource
- execution of resource
- downloading of resource



These operations perform actions as interface and using that interface the system (environment) interacts with informational resources (filling).

Therefore, in system any new unique type of informational resource may be created which will support unified system interface of access to resources. As a result of unification structure of system does not depend on its filling and these statements are strictly divided.

System can be developed step by step by adding new types of informational resources.

Another important question is storage format of internal informational resources data, it means format of storage of registry nodes data.

There are several possible options of storing data in DBMS:

- records
- serialized objects
- distribution by incidental charts
- XML documents

In this system for storing of data XML format is used and this gives us following advantages:

**1. Possibilities of structuring:**

XML documents can be described as containers that may contain other documents that have various hierarchies. This helps to create complexly structured resources. Physically XML documents represents text file, available for edition.

**2. Possibilities of checking:**

Any XML document may contain description of its grammar, which is studied by special software analyzer that determines if XML document's content corresponds with predicted structure.

Analyzer may also automatically change content and data structure.

**3. Portability:**

XML format may serve as cross platform format of data exchange. And this makes system compatible on syntactic level. Also XML documents serve as universal format for information exchange between different components (chains) of distributed

system. XML's derivative languages allow describing standard communication formats of the data that are acceptable in various subject areas or domains.

For systems of distance learning international standard IMS (Instructional Management System) is accepted. The support of this format infers system of data compatibility on semantic level.

#### **CHAPTER 4. Different ways of displaying data.**

After sending data on user's machine, XML allows data displaying in different ways. Unlike HTML which describes data representation, XML describes data itself.

While data is separated from presentation XML reflects information on screen of each user individually, depending on configuration of client's machine and a number of other criteria.

This means that XML allows separately to describe data structure without assigning it to method of displaying and to define form of data presentation irrespective of its specific content.

So XML language is the best solution for informational systems, operating structured data, in particular for Distant Learning systems. Nowadays for using of XML technology it's possible to use either currently available relational DBMS or special XML oriented DBMS.

#### **Object representation of informational resources.**

For achievement of such informational system characteristics as modularity, versatility, expandability, openness it's requested the creation of unified informational filling and system's functional component support program environment.

Any informational resource of a system (study material, course, test) is presented by object in data base (storehouse of informational resources) and processed by system in common way.

XML document of database contains informational part of system resource and procedural part of resource is defined by according class (in this system Java classes are used). Such resource may be mentioned as active.

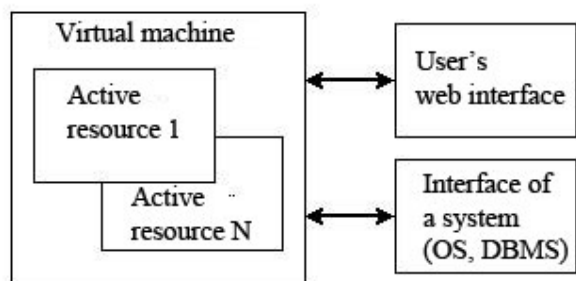
Let's view the definition of active resource. Each class of the resource has according standard methods, describing its interface of interaction with the system. It means that system if it's necessary calls interface methods which are realized in class related to

concrete resource type. So resource processes its own data by itself according to resources internal structure. Typical operations of interface are described below:

- Creation of resource
- Change (edition) of resource
- View of resource
- Execution of resource

Under active resources in common cases we will understand the combination of data and code for processing of this data.

Active resources are executed under management of specialized virtual machine responsible for parallel synchronized execution of active resources, providing their access to system's functions and to user's interface, management of active resource execution context.



Let's list other properties of active resources:

- Active resources are functional components independent from each other
- It's realized as class on the object oriented language
- Class is inherited from abstract class of active resource with unitized methods
- Active resources may be over used and created according to template
- Each active resource is executed in separate stream of operational system and virtual machine watches for its context (operational variables)

Special type of active resource is multiuser active resource that allows the common usage of its data.

Multiuser resources are mostly mentioned for such system components as business game, forums and chats. Multiuser resources allow conducting group form of studying.

System is developed according to principle of open systems.

### **Standards of open systems.**

The main point of open technologies is in formation of environment which includes software solutions, hardware components, communication facilities, interfaces, data formats and protocols, providing cross platform portability, interaction and scalability of applications and data.

Aggregate of described characteristics may be reached by usage of developing, public and acknowledged standards of informational technologies products which are components of open system environment.

Let's list the main standards which are used in system:

**XML** – is the markup language which describes the whole class of data objects, also known as XML documents. This language is mostly used as environment for description of other languages' grammar and for control of correct documents preparations.

XML's design goals emphasize simplicity, generality, and usability over the Internet. It is a textual data format, with strong support via Unicode for the languages of the world. Although XML's design focuses on documents, it is widely used for the representation of arbitrary data structures, for example in web services.

There are a variety of programming interfaces which software developers may use to access XML data, and several schema systems designed to aid in the definition of XML-based languages.

**JAVA**– it's a platform of applications development, which provides portable interoperated programming language with high performance, which is simple and object-oriented and execution environment.

Java-applications may work anywhere, where software of Java virtual machine is installed. Nowadays there is variety of Java virtual machine implementations for all operating systems. For java it's also developed the interfaces for all database

management systems, interfaces of remote procedures call. Many web servers, database servers and application servers are developed using JAVA technologies. So we can see that Java platform is the most convenient way of distributed open informational system's realization.

Combination of XML and Java technologies as declarative and procedural components respectively allows creating universal model of open system's program development.

Modern system of distance learning should have possibilities to use study materials, prepared in other systems and provide its own materials for external usage. And in this case system should support conventional open standard of study material storage. As that standard is used looking forward international standard of IMS organization.

**IMS** (instructional management system – system of teaching management ) is a project which plays an important role in creation of standards for learning through internet. This project is a result of joint effort of 29 manufactures', government structures' and educational centers' representatives, directed on creation of standard package in five areas – metadata, content, management, profiles and external interfaces. The most important standards are metadata standards (description of study material data structure) and profile standards (description of data structure about system users).

Application of standard format for user's profiles allows making them portable so that profile will follow after user from system to system. Personal information and information about properties could serve as a base for creation of social networks in virtual environment and help the users of such networks to discover more about each other. Such information would be also useful for tuning and management of study environment.

IMS project proposed standards for metadata of contents of courses and educational activities. These standards support portability and second usage of objects for electronic learning and introduce structure of information description about objects. In these descriptions there may be such data as object of ridicule, educational level, copyright owner, and description of data format itself. This standard is described on XML language. Usage of IMS standard increases compatibility and openness of system from syntactic level up to semantic level.

**Testing technology.**

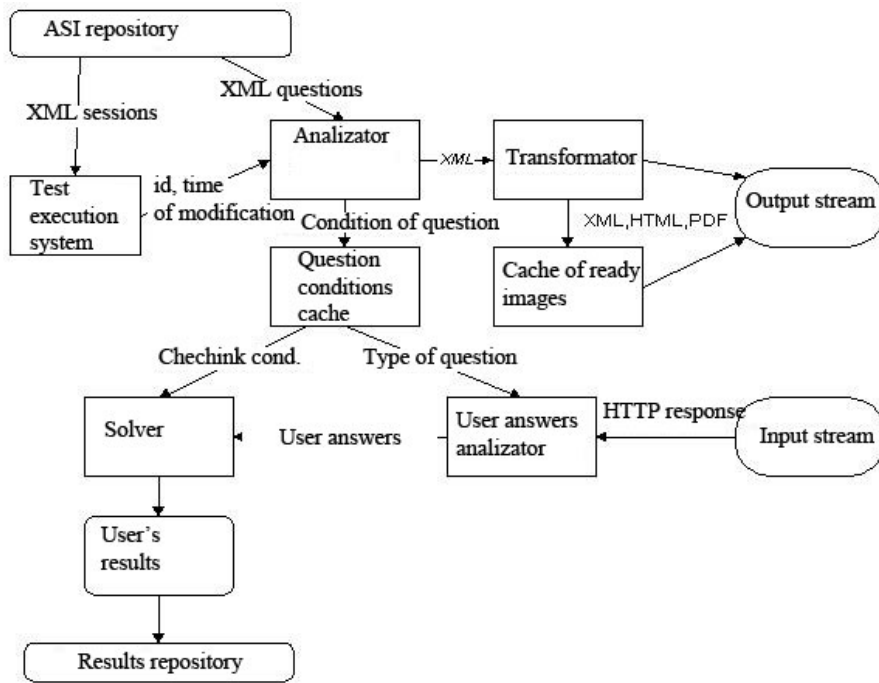
One of the most necessary components of distance learning system obviously is a testing component. In this system it's developed according to principals of active resource.

As presentation format of tests following specifications have been accepted: IMS Q&TI (Question & Test Interoperability). QTI specification is standard of tests import and export between different learning systems. QTI model is comfortable for organization of testing subsystem by following reasons:

- QTI specification is not static format of standardized types of questions presentation, but is extensible language of test representation allowing due to combination of elements to get required type of questions and tests.
- Q&TI specification is described in XML language
- In Q&TI specification there is standard table of this format support level, consequently gradual transition to this format and exchange of tests between learning systems on shown level is possible.
- Q&TI has its own rich setting assortment and also has quality of expandability; it means that it can add its own elements and materials which are not supported by standard.

Hierarchical representation of tests in Q&TI allows division of tests on sections according to test topic.

The subsystem of test execution is implemented on presentation level of Q&TI language with usage of all language's base elements, which allows conducting testing according to all popular test types, and also according to their various combinations. Support of two material's type is realized: text and graphics.



Effectiveness of realization is reached due to usage of XML+XSL combination. This allows after insignificant processing of test data by XML analyzer to take it out in storage format and construction of user interface is produced by web browser according to XSL template. In system is used caching of newly constructed interface forms and caching of users answers checking conditions.

Therefore while conducting mass testing everything that subsystem does is putting cached data to web browser out.

### User interface.

Internet with a help of usual web browser is the main way of interaction between Distant Learning System and user. In system complete web interface is implemented. It means that browser serves as ultrathin universal client of the system.

This method is the most suitable for organization of distance learning system because user does not need any special software application. Usage of such system is similar to usage of internet – easy and accessible.

The usage of web browser provides platform-independence and mobility for user.

Www is built on technology which is based on hypertext. This technology allows describing objects of any volume and provides universal and accessible interface to the system.

Actually, www provides practically unique possibilities of creative approach in creating documents design and form. Any comfortable way for presentation of materials to end user can be provided. It may be photos, graphics, pictures, texts, video fragments and etc.

System server dynamically generates user friendly unified user web interface with a help of servlets and can be integrated with web browser.

System proposes to user interactive interaction with system. It means www not just provides requested materials to users, but also helps to conduct interaction between users, and also between users and system of distant learning.

The user gets access to system after providing username and password. After this he gets access to his unified personal environment on the screen, i.e. to everything which is available on current stage of learning process.

For student it is:

- Individual learning plan, containing static study materials (lections, books) and interactive study materials (tests, business games).
- Study materials available to everyone
- Different forms of interaction with other participants of learning process: distance seminars, conferences, control works, laboratory works, chats with other students, thematic chats, personal consulting with teacher.
- The results of his own works

Teacher has following possibilities:

- Access to all study plans
- Access to all study materials
- Creation of study material



- Work with courses conducted by him/her: creation of these courses, modification, consulting of students.
- Work with group of students; view and evaluation of students' works.

### **Information security.**

“Open system” does not mean that it's not protected and everyone can get access to kept information. Safety of personal and confident information, presenting governmental, commercial and personal secret is requested and one of the main requirements of any informational system and is reached due to implementation of information security standards.

System contains multilevel system of information protection and consists of following components:

- Protection of DBMS
- Protection from illegal access
- Protection of web interface

Identification of user is realized with a help of name and password. Authorization to resource access is also supported in system. For authorization of users matrix scheme is affordable method. For each category of users only fixed set of functions is defined and available. The whole informational reption is contained in DBMS.

For protection of client's web browsers logical connection with system's server, principle of protected session of user based on cookie is used.

Recording of session helps to restore user's data quickly after possible failure of system or network.

Parameters of interface managing commands are being encrypted. Possibility of user and resource administration is implemented in system. System supports group and personal distinction of access to information on informational server, protocol and audit of user actions, remote administration of information node.

### **Increasing productivity.**

Main optimization of productivity can be reached by usage of system of multilevel caching ( by reading) of resources in memory of server resources of system, recording of users sessions, security tables, settings of system and derivative information. This way helps to reach maximum productivity of server hardware platform.

As such system is multithread it's effective to use multiprocessor hardware platforms.

During outputting XML blocks directly on the screen in web browser productivity of the system dramatically increases because of distributed processing of inferring data.

### **Application of system.**

System is mentioned for work in typical software and hardware environment that presents in every education center and does not request any original decisions. But at the same time, if it become necessary there are possibilities to install and implement the system on wide assortment of hardware and software solutions that is on behalf of educational center.

The architecture of the system was designed taking into account possibility of modeling study process in different educational institutions. System is mentioned for usage in educational purposes and allows totally automate system of distant learning. Also there is possibility to use "free of charge" program platform (including operating system, DBMS, development tools (Linux+MySQL+JAVA2+Jbuilder4)).

In conclusion let's finalize the main advantages of this system compared to analogue ones: universality of information filling, modularity of informational resources, cross platform compatibility, high level of scalability, orientation to Internet network usage, openness of architecture, IMS support, totally on web interface, high productivity.

## **CHAPTER III. Development of interactive web-based applications.**

### **The Nature of Web-Based Applications**

There are actually several different kinds of web-based applications, but almost every kind can be categorized into one of two general architectures: client-server applications and client applets.

In the client-server architecture, a browser plays the role of the client and displays content that the server generates. The server is responsible for the processing while the client displays the interface.

In the client applet architecture, the entire application is downloaded to the browser and runs locally. The application may periodically contact external servers for data, but the bulk of the processing is performed by the client.

Both architectures have strengths and weaknesses and neither is appropriate for every application.

Since the work is about writing client-server applications, this chapter focuses on that architecture.

### **3.2 Client-Server Web-Based Application Architecture**

Typically, five components compose the client-server web-based application architecture: the user, the browser, the web server, the application and the database. These components are combined in series such that each component interacts with the next component in the series. In some implementations, the web server and application are actually combined into a single component. In others, the web server, application and database are all three combined into a single component. So variations on the theme exist, but fundamentally the architecture is the same.

A point worth mentioning here is that the interactions between the different components are highly transient in nature. Each component makes a request then waits for a response or waits for a request then sends a response; with relatively long periods of idle time between activities. This behavior is fundamentally different from the behavior of other types of applications and has a significant effect on application design.

### **3.3 Tools for creating web-based applications**

There are several popular languages for creating web-enabled applications. They differ in platforms they can be run on, language structure, tasks they can carry out.

When deciding whether to go with PHP, ASP, ColdFusion, or something else, the most important matter to consider is who will be working on your website now and in the future.

#### **ASP**

Active server pages can be written in VBScript, an easy-to-learn scripting language that uses a syntax similar to Visual Basic's. JavaScript, Perl, and even Python can be used to

code ASP pages. Furthermore, with the extensive integration capabilities of Microsoft's .NET Framework, you can create powerful ASP pages with expanded libraries.

ASP's most distinguishing benefit is its ability to use COM objects. As with everything else in ASP, using COM objects is incredibly easy. One line of code is all it takes to create an instance of a COM object. From there, you can use the object however you see fit, calling its methods, setting its properties. This creates two awesome benefits: First, you can use the same robust COM objects that you use in Visual Basic or Visual C++ on your ASP pages; second, you can create your own COM objects for use on your ASP pages.

The benefits of COM objects can't be fully appreciated until you've made use of Microsoft's ActiveX data objects (ADO), a combination of robust COM objects that are used for data access. With ADO, which can be used in Visual Basic or Visual C++ programs as well as ASP pages, you can connect to just about any data store, from an SQL 7.0 database to an Excel spreadsheet. And it doesn't stop there; ADO can use almost anything as a data store. Of course, the standard ODBC-compliant databases are all usable.

## **ColdFusion**

Its overall simplicity and tag-based syntax make it easy to learn. It has a powerful IDE that can help you get productive quickly, and it can scale to handle even the biggest commercial websites. And while there's nothing you can do in ColdFusion that you couldn't duplicate in ASP or PHP, ColdFusion lets you work faster and better.

The reason for this is simple: unlike other tools, ColdFusion wasn't patched together out of existing technologies. It was designed from the ground up to be a Web application platform. Because the creators of ColdFusion weren't saddled with the wreckage of past paradigms, they were free to find the best way to solve the problems Web developers face. This allowed them to streamline aspects of ColdFusion that were clunky in other technologies.

## **JSP**

JSP utilizes servlets to create a dynamic scripting language where variable content, often retrieved from a database, can be imbedded within HTML. It's conceptually similar to ASP but comes with a major advantage: where ASP uses VBScript or Jscript, JSP unleashes the power of the Java language.

When a client machine requests a JSP page for the first time, the server automatically builds, compiles, and starts a background Java servlet that in turn generates an HTML page. That HTML page is sent to the client's browser for display. From then on, whenever the JSP page is accessed, the servlet is there as Java byte code in the Web server's memory, ready to query a database and serve up the HTML immediately. With ASP, the code needs to be reinterpreted for every client request, which slows down the process of page generation.

## **Perl**

Perl is a "Practical Extraction and Report Language" freely available for Unix, Windows, MS/DOS, Macintosh, OS/2, Amiga, and other operating systems. Perl has powerful text-manipulation functions. It eclectically combines features and purposes of many command languages. Perl has enjoyed recent popularity for programming World Wide Web electronic forms and generally as glue and gateway between systems, databases, and users.

It was first developed by Larry Wall, a linguist working as a systems administrator for NASA in the late 1980s, as a way to make report processing easier. Since then, it has moved into a large number of roles: automating system administration, acting as glue between different computer systems; and, of course, being one of the most popular languages for CGI programming on the Web.

Main feature of Perl are Regular Expressions. They make text manipulations much easier. For example, the following regular expression validates e-mail address:

```
$email = "someone@somewhere.net";
```

```
if ($email =~ /^[\.0-9a-z-]+@((([0-9a-z][0-9a-z-]*[0-9a-z])[a-z0-9])\.)+[a-z]{2,3}$/) {
```

```
    print "E-mail entered is correct";
```

```
} else {  
  
    print "E-mail entered is incorrect"  
  
}
```

## **PHP**

PHP is a language for easily building dynamic web pages. It provides an easier way to accomplish web related programming tasks, which are accomplished only with difficulty in more complex and powerful languages, such as Perl or C. It is ideally suited to the web because PHP scripts live inside web pages right along with the HTML tags and content. For that reason, PHP is called an embedded scripting language. Developers can embed programs in their web pages, making the dynamic. They can treat programs just like web pages. PHP pages can contain both regular HTML and PHP code. This allows you to develop web applications quickly. However, unlike some web scripting languages, PHP makes a clear distinction between sections of PHP code and sections of the HTML document. When the web server fills a request for a PHP enabled page, it first looks through the page content for sections of PHP code and executes any it finds. Any normal HTML sections are passed to the browser without any changes. This means that you can freely mix snippets of program into a web page anywhere.

### **Why Use PHP?**

Here are a few good reasons to choose PHP for enabling interactive content on the web site (besides being open source):

- Because it uses similar syntax and constructs, knowledge of PHP can help you in learning the C language.
- The data types and structures of PHP are easy to use and understand, PHP knows what you mean and can convert types automatically.
- You don't have to know any special commands to compile the program; it runs right in your web browser.
- You don't have to know everything there is to know about PHP to start writing useful programs.

PHP serves as a «wrapper» for many standard C libraries, which are easily compiled into the language giving it the flexibility to respond more rapidly to changes in web technology or trends.

Things you can do in PHP can be done in other languages, but PHP was designed to work in a web context, so things that are difficult or tedious for the applications programmer to «roll their own» in Perl are easy in PHP. PHP enabled web sites can be deployed with amazing rapidity, due to its being tuned for dynamic pages and database backend.

### **3.4 Efficiency and cost**

Efficiency of development is proportional to the cost of deployment.

Due to the transient nature of web-based applications, a very simple mistake to make is underestimating the capacity of an application to consume system resources. When developing and testing a web-based application, even the most inefficient implementation will run very quickly compared to other types of applications. You might be inclined to think then that since the components of an application are so small, take so little time to run, and require so little memory; that you can write them in whatever language is easiest to learn, run them on whatever old computer you have lying around and use lots of fancy graphics on every page. If the application is rarely run and resides on a machine which hosts other rarely run applications, then you might be right. If the application is destined to get a lot of hits though, then it's possibly the worst mistake you can make. While individual hits on an application may consume undetectable amounts of system resources, hundreds of hits per second can grow that insignificant quantity to monstrous proportions. The choices of server and network hardware, operating system, file system, languages, API's, scaling techniques and countless other factors should be driven in no small way by the analysis of the level of activity that the application is likely to receive.

In the end, throwing in more hardware to improve the performance of an improperly designed application will reach the point of diminishing returns; and it will have become costly than long before that.



## **CHAPTER IV. Description of the “Khazar University” System.**

Current system represents the automation system of managing tasks, materials and questions and further students' knowledge checking by the teacher based on the tasks that he solves.

System is developed using technology “server-thin client” where server part consists of web server and database server. And client side can be any web browser that supports HTML 4.0 and CSS2.

### **4.1 Server**

As the web server Apache2 was chosen. This choice is based on highly platform independency of Apache server, it's security opportunities, flexibility of settings and that it is free for usage.

Apache server is the most popular web server. There are a lot of documentation and ready instructions for installing, setting up, tuning and further support of the Apache server.

As the scripting language PHP5 was chosen, because of its availability on many platforms, big opportunities and flexibility. The choice of the operating system does not really affect the workflow.

But for high load systems that need high level of availability and stability it is better to use one of the Unix-like operating systems.

As the database is used server MySQL. First of all this choice was made because of its very high performance and practically full support of the standard SQL-92. Also it is free for non-commercial usage. There are MySQL versions available for many operating systems.

It is easy to install and configure, easy to maintain. MySQL has good documentation and a network of related websites.

When installing all components of the server on the Linux, FreeBSD or any other free operating system it is possible to bring to zero investments to the software part of the new information system.

It is possible to build efficient and reliable information system using only free and open source products. All components can be easily downloaded from internet and all of them are being enhanced all the time. New bugs are being found and fixed, new versions of software are being introduced.

In this case I use software called Denwer. It is a set of programs and modules that are installed during one installation process. This product includes Apache web server, PHP and Perl modules, Mysql database.

Now you do not have to install all required software components separately. You do not have to spend time customizing them and setting them up to work together.

After installing Denwer it creates virtual disk on a system. On this disk are stored program files and sites. User can create aliases, subdomains, place there his scripts, web pages.

Product also contains PHPMyAdmin tool that helps to administer Mysql database from the web interface. Using this graphical tool administrator of the system can view all tables of database and make some actions on them like update, delete and drop.

Database server is used for storage of all information that can be needed for the proper functioning of the system. Here are stored users data, information about the rights of users, groups, subjects, students, bank of tasks and questions and results of the student.

## 4.2 Users

In the system there are three types of users: administrator, teacher and student.

Each user has its personal username and password.

For entering the system users have to fill in their username and password.



The screenshot shows the login interface for the Khazar University Distant Learning System. On the left is the university's logo, a red circle with a yellow star and the text 'KHƏZƏR UNIVERSİTƏSİ 1991 BAKI'. To the right of the logo, the text 'Khazar University' is displayed in blue, with 'Distant learning system' below it in black. The background features a light blue wavy pattern. Below the header, there is a light blue rectangular area containing the login form. The form includes a 'Username:' label above a text input field, a 'Password:' label above another text input field, and a 'Log in' button below the password field.

Administrator has rights to add, delete and modify groups, subjects, teachers, students and give a teacher a group or a student.

When creating a teacher administrator has to specify teachers Full Name, teachers Login, Password, E-mail, phone number, city and which subject he teaches.

Adding new students is a similar process. But instead of Subject you should specify the group to which this student belongs.

One of the main functions of Department Administrator is to share students between teachers.

For this purpose there is menu item "Teacher access". Using this functionality administrator can assign to a teacher some number of students or a whole group depending on his load and a specific subject.

## Give a teacher a group or a student

Teacher ID	Teacher	Groups	Students
0000018	<a href="#">Ali Mamedov</a>	2	10
0000027	<a href="#">Enver Babayev</a>	1	5
0000019	<a href="#">Lala Aliyeva</a>	2	10
0000020	<a href="#">Rashad Nuriyev</a>	1	5
0000032	<a href="#">Sakhavat Talibov</a>	2	10

## Give a teacher a group or a student

### Sakhavat Talibov

 Edit groups   Edit students

Groups	Students
ek16 291	Jamil Gasimov Zaur Gurbanzade Magomed Mamedov Sergey John Doe Sanan Abbasov Imran Mamedov Eldar Hasanov Narmin Guliyeva

## Give a teacher a group or a student

### Sakhavat Talibov

#### Editing students

- Eldar Hasanov
- Imran Mamedov
- Jamil Gasimov
- John Doe
- Magomed Mamedov
- Narmin Guliyeva
- Sanan Abbasov
- Sergey
- Zaur Gurbanzade

Also administrator can view, edit and delete all exams and materials created by teachers. And he can change test and exam results if it is needed.

**Teacher.** When teacher logs in he has access to his set of questions, study materials and list of students which were assigned to him by administrator.

Teacher has rights to add, delete and modify the Study materials, questions for exam and self testing, create exams, view scores of the students.



Welcome, Sakhavat Talibov

Hello, Sakhavat Talibov.  
Welcome to Khazar distant learning system.

- ✓ Students
- ✓ Materials
- ✓ Questions
- ✓ Results
- ✓ Exams
- ✓ Change password
- ✓ Log out



One of the new functionalities added to the program recently is ability to view detailed report about results and answered questions.

## Result details

**Name: Jamil Gasimov**

**Exam: exam1**

Date:

Question	Text	Solution	Answered	Point	
Cont. 6	Suppose that $P(x, y)$ means	$v$ is an uncle of $w$	$v$ is an uncle of $w$	3	✓
Cont. 5	Consider the following pseudocode. $x := 1; i := 1; \text{while } (x \leq 1000) \text{ begin } x := 2x; i := i + 1; \text{end}$ ; What is the value of $i$ at the end of the pseudocode?	5	5	3	✓
Cont. 3	Which of the following algorithms has running time $O(n^2)$ in the worst case but $O(n \log n)$ on average?	Mergesort	Mergesort	2	✓
Cont. 4	Which of the following is the name of the data structure in a compiler that is responsible for managing information about variables and their attributes?	Abstract Syntax tree	Attribute grammar	2	✗
Cont. 2	The object-oriented paradigm includes which of the following properties? I. Encapsulation II. Inheritance III. Recursion	I, II, III	II only	1	✗
Cont. 1	Suppose that a certain software product has a mean time between failures of 10,000 hours and has a mean time to repair of 20 hours. If the product is used by 100 customers, what is its availability?	90%	90%	1	✓
OOP 4	Which of the following is (are) true about virtual memory systems that use pages? I. The virtual address space can be larger than the amount of physical memory. II. Programs must be resident in main memory throughout their execution. III. Pages correspond to semantic characteristics of the program.	I and II	II only	2	✗

**Total score: 9**

Teacher can add materials for studying. Also he can add Topics as sublayer to the materials.

Also teacher has possibility to add questions of two types: exam and test.

Exam questions are not seen to students until teacher creates an exam with these questions. But test questions are visible and student can make test after reading corresponding topic.

In the database table where questions are being stored there is column "tp" that indicates the type of a question.

Tp = 1 is an exam question

Tp = 2 is a test question.

Questions can be 3 level of complexity and can correspond to particular chapter or topic.

When creating the exam teacher must select the start and end date of the exam, chapters that this exam will cover, list of groups and students for whom this exam concern, number of questions from each complexity group and coefficient for every group.

These settings for each exam are stored in a particular table. But there is no predefined set of questions, only parameters of each exam.

# Exams

## Creating new exam

Exam name:

Start date:

End date:

Duration:  mins

	count	koefficient
level 1	<input type="text" value="0"/>	<input type="text" value="0"/>
level 2	<input type="text" value="0"/>	<input type="text" value="0"/>
level 3	<input type="text" value="0"/>	<input type="text" value="0"/>

Subject

- Contemporary Logic Design
  - Preface
  - Chapter1
  - Chapter2
- Object-Oriented Data Structures
  - Chapter1
  - Chapter 2

Groups

- 291
- ek16

Students

- Eldar Hasanov (ek16)
- Imran Mamedov (ek16)
- Jamil Gasimov (ek16)
- John Doe (291)
- Magomed Mamedov (291)
- Narmin Guliyeva (ek16)
- Sanan Abbasov (291)
- Sergey (291)
- Zaur Gurbanzade (ek16)

Exam itself is generated dynamically based on these parameters. Main advantage of this method is that exams are not static. Teacher simply can add new questions, delete old ones and exam will be generated using new questions.

examID	TeacherID	ExamTitle	Difficulty	StartDate	EndDate	c1	k1	c2	k2	c3	k3
1	5	Title	5	2009-06-14 23:12:00	2009-06-14 23:32:00	0	0	0	0	0	0
18	11	Exam 01	3	2009-09-10 00:00:00	2009-09-25 00:00:00	1	10	1	20	1	30
17	11	exam1	5	2009-09-18 00:00:00	2009-09-18 00:00:00	2	1	3	2	4	3
19	11	Exam1	2	2009-09-25 00:00:00	2009-09-30 00:00:00	1	10	1	20	1	30
14	11	test	2	2009-09-02 00:00:00	2009-09-03 00:00:00	1	1	2	2	3	3
15	11	test1	2	2009-09-02 00:00:00	2009-09-03 00:00:00	4	1	5	2	6	3
16	11	test	1	2009-09-03 00:00:00	2009-09-04 00:00:00	2	3	3	3	3	3
21	11	exam4	1	2009-10-06 00:00:00	2009-10-11 00:00:00	2	1	2	2	2	3

'ShowQuestions' function selects question based on input parameters: exam id, complexity level and count of questions.









As you can see there is 'Difficulty' row in this table. This feature is not being used in this version of system. This is for automation of question selection. Based on the difficulty script selects particular number of questions with different complexity level. This method is good when there is predefined number of exam questions and teacher cannot change it.

While creating an exam teacher can specify the duration of the exam in minutes. This information is being added to 'exam' table. When student selects to pass exam counter at the top of the page begins to count down. When time is up html form with exam is being submitted.

After implementing functionality with exam duration I tried to eliminate fraud cases and cheating by students. For the example if there is no mechanism against cheating student can wait until there is one second left and close the window. Then he can pass exam again and timer will count from the beginning. To avoid this cases I insert into the 'examresults' table information with zero points collected. If student closes window it will remain zero points. But if he answers questions and submits the form table is updated with his actual points.

Another new functionality is ability to add pictures to questions and materials.

Teacher can then delete images or add new ones when he edits the question or material topic. All images are stored in the database in a BLOB format.

←T→	imgid	title	imgtype	imgdata	qid
<input type="checkbox"/>  	1	title1	image/jpeg	[BLOB - 47.5 KB]	83
<input type="checkbox"/>  	2	Pic.2	image/gif	[BLOB - 70.0 KB]	84
<input type="checkbox"/>  	5	pic3	image/jpeg	[BLOB - 66.7 KB]	85
<input type="checkbox"/>  	16		image/jpeg	[BLOB - 47.5 KB]	49

There is built in check of type of files and their sizes. Program does not accept non image files and too big images.

User can add up to five images at a time. Number of added input forms change dynamically depending on the selection in dropdown list.



# Questions

## Editing question

Type

---

Subject

---

Complexity

---

Question title:


---

Question text:

---

Answers:   
 Hierarchical  
 Relational  
 Traditional

---

Images:  [delete](#)  
title1 48.6Kb

---

Upload image    
Title     
Title     
Title

This is how it looks in the browser.

Images are listed below the text. You can refer to them during the class. Clicking on the picture opens it in the new window with its actual size.

## Materials

### View material

#### The Centroid

You should already know that the segment connecting the vertex of a triangle to the midpoint of its opposite side is a median. There are three midpoints and three vertices in every triangle, so every triangle has three medians.

The Image1 shows a triangle with one median.

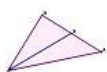
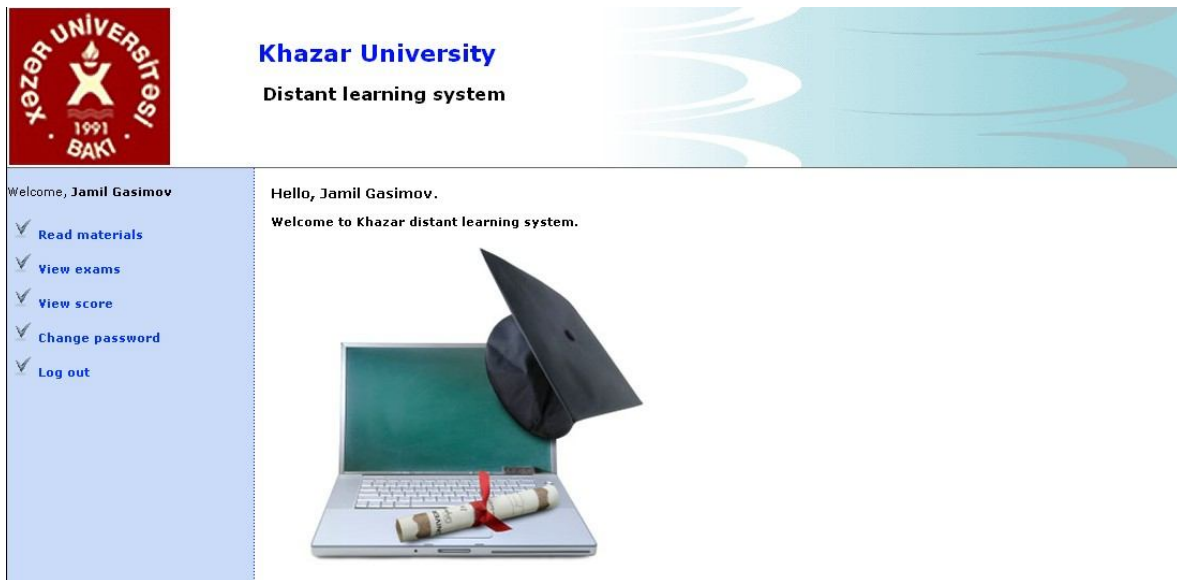


Image1 9.3Kb

[Level up](#)

**Student.** He can read materials, make tests, pass exam, view results of exam.

There are some restrictions on passing exams. Student can pass exam only in a particular time interval that is set by teacher. Student can pass each exam only once.



Let me show how it is implemented in the code. 'GetExams' function gets the list of exams.

It checks which exams are assigned to a student and which materials are used in it.

...

```
if (($dt>$dt1)&($dt<$dt2)&($rows01<1))
```

```
{  
$exams[$i]["Title"]=<a  
href='index.php?link=exams&act=start&id=".$exams[$i]["ID"]."'>".$exams[$i]["Title"  
"]."</a>";}
```

As you can see if system date is more than begin date of an exam and less of end date of an exam and number of rows in results table is null then show exam as link.

# Materials

Title	Subject	Teacher
<a href="#">Chapter 1</a>	Informatics	Sakhavat Talibov
<a href="#">Chapter2</a>	Informatics	Sakhavat Talibov
<a href="#">Chapter3</a>	Informatics	Sakhavat Talibov
<a href="#">mat22</a>	Informatics	Sakhavat Talibov
<a href="#">wqqwwe</a>	Informatics	Sakhavat Talibov
<a href="#">lala1</a>	Geometry	Lala Aliyeva

Materials are divided into chapters and topics. After reading each topic student can make a test and check how he understood the material.

Questions for tests and exams are added by teacher. He also sets the complexity level and to which chapter question refers.

All users can change their passwords but administrator can see and change password of every student and teacher.

## 4.3 User interface.

Interface is build using CSS styles and standard html forms and fields. Also dynamic html was used when creating questions.

Due to usage of object oriented programming principles all main functions and classes are collected in separate files. This helps to change design of pages without changing the program part of the system. You can only edit one or two files, add pictures, change fonts and these changes will affect all pages of the site.

Such principles can avoid making accidentally mistakes in the syntax of program code.

## **Conclusion.**

As a conclusion I would like to summarize performed research in subject area of distant learning. In modern environment innovative technologies, computer and software products must be taken in account while organizing studying process.

In my work I tried to pay attention and discuss a variety of problems related to theoretical bases of distant learning. Also review of some existing distant learning tools was performed.

In the last part I introduced my own solution that provides students and teachers with possibility to interact and share knowledge.

Development of this system can be divided in three parts:

- Planning and building the database structure that would allow holding, updating, editing all necessary information about students, teachers, materials, exams that eliminates data redundancy and inconsistency and allows to access all information with help of standard sql queries.
- Writing the objects and functions using object oriented principles for accessing, manipulating and displaying data. These objects are the functional core of the system.
- Creating user interface for administrators, teachers and students.

Further improvement of this software product consider changes in examination process, using attachments and pictures in learning materials, adding more monitoring possibilities.

The most important results of this work are systematization of requirements that electronic distant learning system should meet and development of the program mentioned above. Performed system allows making first steps in the direction of implementing modern methods of education and further computerization of learning processes in our educational institutions without depending from commercial software of third-party companies.

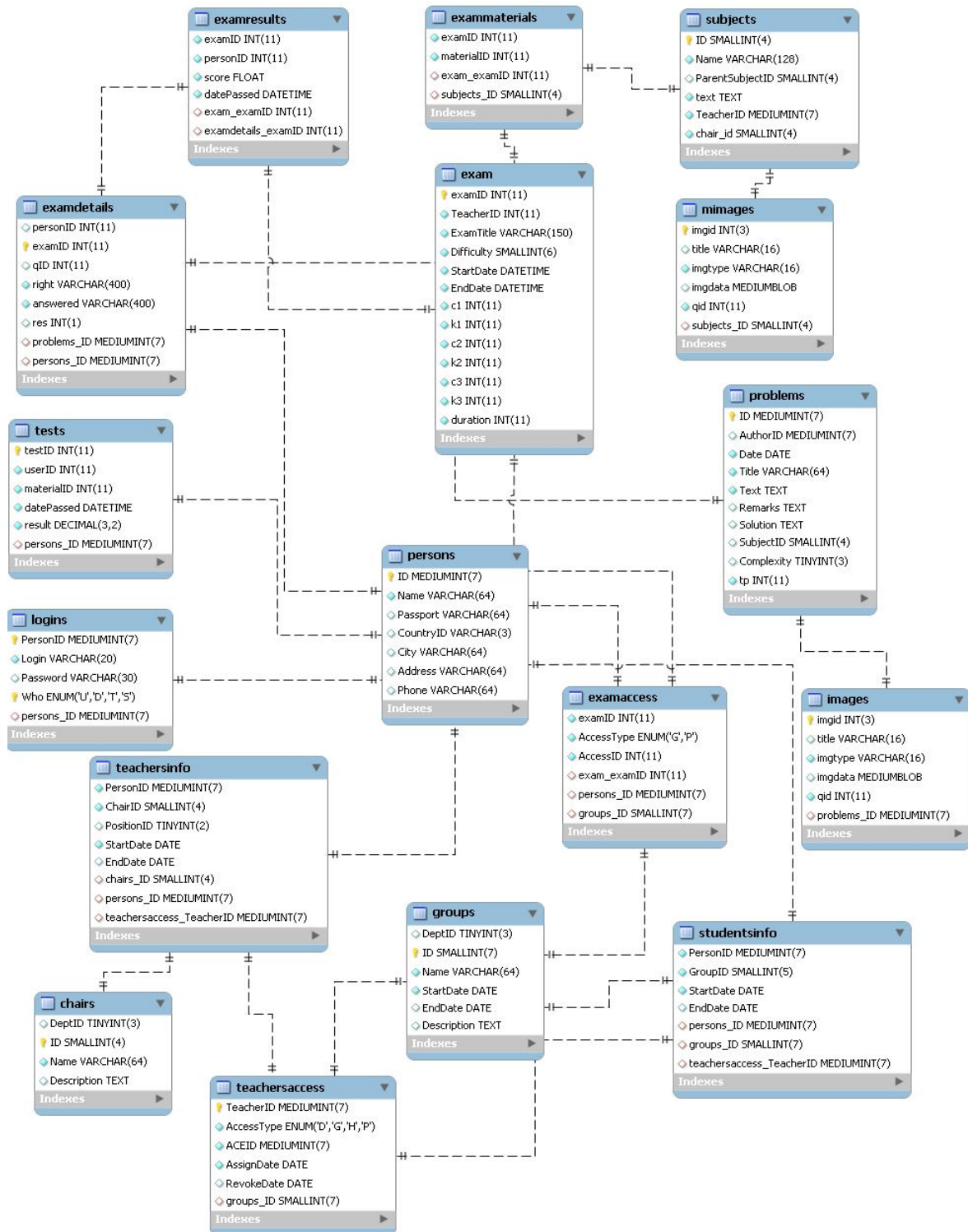
## Literature.

1. Michael R. Simonson, Sharon E. Smaldino, Susan Zvacek. Teaching and Learning at a Distance: Foundations of Distance Education. Prentice Hall, 2002.
2. Marc J. Rosenberg. E-Learning: Strategies for Delivering Knowledge in the Digital Age. McGraw-Hill Trade, 2000.
3. Roger C. Schank. Designing World-Class E-Learning: How IBM, GE, Harvard Business School, And Columbia University Are Succeeding At E-Learning. McGraw-Hill Trade, 2001.
4. William Horton. Designing Web-Based Training: How to Teach Anyone Anything Anywhere Anytime. John Wiley & Sons, 2000.
5. Susan Schor Ko, Steve Rossen, Susan Ko. Teaching Online: A Practical Guide. Houghton Mifflin Co, 2000.
6. Rena M. Palloff, Keith Pratt. The Virtual Student: A Profile and Guide to Working with Online Learners. Jossey-Bass, 2003.
7. Sherif Kamel, Fawzi Albalooshi. Virtual Education: Cases in Learning & Teaching Technologies. IRM Press, 2003.
8. Darin Hartley, Darin E. Hartley. On-Demand Learning: Training in the New Millennium. HRD Press, 2003.
9. Luke Welling, Laura Thomson. PHP and MySQL Web Development. Sams, 2003.
10. Hugh E. Williams, David Lane. Web Database Applications with PHP & MySQL. O'Reilly & Associates, 2002.



# Appendix A

## Tables relationships:



## Appendix B

### Structure of tables:

```
CREATE TABLE chairs (  
    DeptID tinyint(3) unsigned zerofill default NULL,  
    ID smallint(4) unsigned zerofill NOT NULL auto_increment,  
    `Name` varchar(64) NOT NULL default "",  
    Description text,  
    PRIMARY KEY (ID)  
);
```

```
CREATE TABLE exam (  
    examID int(11) NOT NULL auto_increment,  
    TeacherID int(11) NOT NULL default '0',  
    ExamTitle varchar(150) NOT NULL default "",  
    Difficulty smallint(6) NOT NULL default '0',  
    StartDate datetime NOT NULL default '0000-00-00 00:00:00',  
    EndDate datetime NOT NULL default '0000-00-00 00:00:00',  
    c1 int(11) NOT NULL default '0',  
    k1 int(11) NOT NULL default '0',  
    c2 int(11) NOT NULL default '0',  
    k2 int(11) NOT NULL default '0',  
    c3 int(11) NOT NULL default '0',  
    k3 int(11) NOT NULL default '0',  
    duration int(11) NOT NULL,
```



PRIMARY KEY (examID)

);

CREATE TABLE examaccess (

examID int(11) NOT NULL default '0',

AccessType enum('G','P') NOT NULL default 'G',

AccessID int(11) NOT NULL default '0'

);

CREATE TABLE examdetails (

personID int(11) default '0',

examID int(11) default '0',

qID int(11) default '0',

`right` varchar(400) NOT NULL,

answered varchar(400) NOT NULL,

res int(1) default NULL

);

CREATE TABLE exammaterials (

examID int(11) NOT NULL default '0',

materialID int(11) NOT NULL default '0'

);

CREATE TABLE examresults (

examID int(11) NOT NULL default '0',

```
personID int(11) NOT NULL default '0',  
  
score float NOT NULL default '0',  
  
datePassed datetime NOT NULL  
  
);
```

```
CREATE TABLE groups (  
  
  DeptID tinyint(3) unsigned zerofill default NULL,  
  
  ID smallint(7) unsigned zerofill NOT NULL auto_increment,  
  
  `Name` varchar(64) NOT NULL default "",  
  
  StartDate date NOT NULL default '0000-00-00',  
  
  EndDate date default NULL,  
  
  Description text,  
  
  PRIMARY KEY (ID)  
  
);
```

```
CREATE TABLE images (  
  
  imgid int(3) unsigned NOT NULL auto_increment,  
  
  title varchar(16) collate utf8_unicode_ci default NULL,  
  
  imgtype varchar(16) collate utf8_unicode_ci NOT NULL default "",  
  
  imgdata mediumblob,  
  
  qid int(11) NOT NULL,  
  
  PRIMARY KEY (imgid)  
  
);
```

```
CREATE TABLE logins (  
  

```

```
PersonID mediumint(7) unsigned zerofill NOT NULL default '0000000',
Login varchar(20) NOT NULL default "",
`Password` varchar(30) default NULL,
Who enum('U','D','T','S') NOT NULL default 'U',
PRIMARY KEY (PersonID,Who),
UNIQUE KEY Login (Login)
);
```

```
CREATE TABLE mimages (
imgid int(3) unsigned NOT NULL auto_increment,
title varchar(16) collate utf8_unicode_ci default NULL,
imgtype varchar(16) collate utf8_unicode_ci NOT NULL default "",
imgdata mediumblob,
qid int(11) NOT NULL,
PRIMARY KEY (imgid)
);
```

```
CREATE TABLE persons (
ID mediumint(7) unsigned zerofill NOT NULL auto_increment,
`Name` varchar(64) NOT NULL default "",
Passport varchar(64) default NULL,
CountryID varchar(3) default NULL,
City varchar(64) default NULL,
Address varchar(64) default NULL,
Phone varchar(64) default NULL,
```

PRIMARY KEY (ID)

);

CREATE TABLE problems (

ID mediumint(7) unsigned zerofill NOT NULL auto\_increment,

AuthorID mediumint(7) unsigned zerofill default NULL,

`Date` date NOT NULL default '0000-00-00',

Title varchar(64) NOT NULL default "",

`Text` text NOT NULL,

Remarks text,

Solution text,

SubjectID smallint(4) unsigned zerofill default NULL,

Complexity tinyint(3) default NULL,

tp int(11) NOT NULL default '1',

PRIMARY KEY (ID)

);

CREATE TABLE studentsinfo (

PersonID mediumint(7) unsigned zerofill NOT NULL default '0000000',

GroupID smallint(5) unsigned zerofill NOT NULL default '00000',

StartDate date NOT NULL default '0000-00-00',

EndDate date default NULL

);

CREATE TABLE subjects (

```
ID smallint(4) unsigned zerofill NOT NULL auto_increment,  
`Name` varchar(128) NOT NULL default "",  
ParentSubjectID smallint(4) unsigned zerofill default '0000',  
`text` text NOT NULL,  
TeacherID mediumint(7) unsigned zerofill NOT NULL default '0000000',  
chair_id smallint(4) unsigned zerofill NOT NULL default '0000',  
  
PRIMARY KEY (ID)  
  
);
```

```
CREATE TABLE teachersaccess (  
  
TeacherID mediumint(7) unsigned zerofill NOT NULL default '0000000',  
AccessType enum('D','G','H','P') NOT NULL default 'D',  
ACEID mediumint(7) unsigned zerofill NOT NULL default '0000000',  
AssignDate date NOT NULL default '0000-00-00',  
RevokeDate date default NULL,  
  
UNIQUE KEY acc (TeacherID,AccessType,ACEID)  
  
);
```

```
CREATE TABLE teachersinfo (  
  
PersonID mediumint(7) unsigned zerofill NOT NULL default '0000000',  
ChairID smallint(4) unsigned zerofill NOT NULL default '0000',  
PositionID tinyint(2) unsigned zerofill default NULL,  
StartDate date NOT NULL default '0000-00-00',  
EndDate date default NULL  
  
);
```

```
CREATE TABLE tests (  
    testID int(11) NOT NULL auto_increment,  
    userID int(11) NOT NULL,  
    materialID int(11) NOT NULL,  
    datePassed datetime NOT NULL,  
    result decimal(3,2) NOT NULL default '0.00',  
    PRIMARY KEY (testID)  
    ) ENGINE=MyISAM AUTO_INCREMENT=17 DEFAULT CHARSET=utf8;
```

## Appendix C

### Main classes of the system:

#### Deptadmin.class.php

```
<?php
```

```
class DeptAdmin
```

```
{
```

```
    //private
```

```
    var $_id;
```

```
    var $_name;
```

```
    var $_dept_id;
```

```
    var $_dept_name;
```

```
    var $_link;
```

```
    //public
```

```
    function DeptAdmin($id, $link)
```

```
    {
```

```
        /* set private fields */
```

```
        $this->_link = $link;
```

```
        $this->_id = $id;
```

```
        /* get name */
```

```
        $query="SELECT `Name` FROM `persons` WHERE `ID` = ".$this->_id."";  
  
        $result = mysql_query($query, $this->_link);  
  
        $row = mysql_fetch_row($result);  
  
        mysql_free_result($result);  
  
        $this->_name = $row[0];  
  
    }
```

```
function GetID()  
{  
    return $this->_id;  
}
```

```
function GetName()  
{  
    return $this->_name;  
}
```

```
function GetDeptID()  
{  
    return $this->_dept_id;  
}
```

```
function GetDeptName()  
{
```



```

return $this->_dept_name;
}

function AddChair($chair_name)
{
    mysql_query("INSERT INTO `chairs` (`DeptID`, `Name`) VALUES ('$this->_dept_id',
'$chair_name')", $this->_link);

    return mysql_insert_id($this->_link);
}

function GetChairs()
{
    $query="SELECT `ID`, `Name` FROM `chairs` WHERE `DeptID` = '$this-
>_dept_id' ORDER BY `Name`";

    $result = mysql_query($query, $this->_link);

    $chairs = array();

    $row = mysql_fetch_row($result);

    while ($row) {

        $chairs[] = array(

            'id' => $row[0],

            'name' => $row[1]

        );

        $row = mysql_fetch_row($result);
    }
}

```

```

mysql_free_result($result);

return $chairs;

}

function GetChairName($chair_id)
{
    $result = mysql_query("SELECT `Name` FROM `chairs` WHERE `ID` = '$chair_id'",
$this->_link);

    $row = mysql_fetch_row($result);

    mysql_free_result($result);

    return $row[0];
}

function DeleteChair($chair_id)
{
    mysql_query("DELETE FROM `chairs` WHERE `ID` = '$chair_id'", $this->_link);
}

function AddPerson($name, $passport, $city, $phone)
{
    mysql_query("INSERT INTO `persons` (`Name`, `Passport`, `City`, `Phone`) VALUES
('$name', '$passport', '$city', '$phone')", $this->_link);

    return mysql_insert_id($this->_link);
}

```

```

function AddLogin($login, $password, $person_id, $who)
{
    mysql_query("INSERT INTO `logins` (`PersonID`, `Login`, `Password`, `Who`)
VALUES ('$person_id', '$login', '$password', '$who')");
}

function AddTeacherInfo($teacher_id, $chair_id, $start_date, $end_date = NULL)
{
    if ($end_date == NULL)
        mysql_query("INSERT INTO `teachersinfo` (`PersonID`, `ChairID`, `StartDate`)
VALUES ('$teacher_id', '$chair_id', '$start_date')", $this->_link);
    else
        mysql_query("INSERT INTO `teachersinfo` (`PersonID`, `ChairID`, `StartDate`,
`EndDate`) VALUES ('$teacher_id', '$chair_id', '$start_date', '$end_date')", $this->_link);
}

function GetPersonName($person_id)
{
    $result = mysql_query("SELECT `Name` FROM `persons` WHERE `ID` =
'$person_id'", $this->_link);
    $row = mysql_fetch_row($result);
    mysql_free_result($result);
    return $row[0];
}

```

```

function GetTeachers($chair_id = NULL)
{
    if ($chair_id == NULL)

        $query_string = "SELECT t.`ID`, t.`Name`, c.`ID`, c.`Name`, t.`Passport`, t.`City`,
t.`Phone` FROM `persons` AS t, `teachersinfo` AS ti, `chairs` AS c WHERE ti.`PersonID`
= t.`ID` AND CURRENT_DATE BETWEEN ti.`StartDate` AND IFNULL(ti.`EndDate`,
CURRENT_DATE) AND ti.`ChairID` = c.`ID` AND c.`DeptID` = '$this->_dept_id' ORDER
BY t.`Name`";

    else {

        $query_string = "SELECT t.`ID`, t.`Name` FROM `persons` AS t, `teachersinfo` AS ti
WHERE ti.`PersonID` = t.`ID` AND ti.`ChairID` = '$chair_id' AND CURRENT_DATE
BETWEEN ti.`StartDate` AND IFNULL(ti.`EndDate`, CURRENT_DATE) ORDER BY
t.`Name`";

        $current_chair_id = $chair_id;

        $current_chair_name = $this->GetChairName($chair_id);

    };

    $result = mysql_query($query_string, $this->_link);

    $teachers = array();

    $row = mysql_fetch_row($result);

    while ($row) {

        if ($chair_id == NULL) {

            $current_chair_id = $row[2];

            $current_chair_name = $row[3];

        };

        $teachers[] = array(

```

```

        'teacher' => array('id' => $row[0], 'name' => $row[1], 'email' => $row[4], 'city'
=> $row[5], 'phone' => $row[6]),

        'chair' => array('id' => $current_chair_id, 'name' => $current_chair_name)

    );

    $row = mysql_fetch_row($result);

}

mysql_free_result($result);

return $teachers;

}

function DeletePerson($person_id)

{

    mysql_query("DELETE FROM `persons` WHERE `ID` = '$person_id'");

    mysql_query("DELETE FROM `logins` WHERE `PersonID` = '$person_id'");

}

function AddGroup($group_name, $start_date, $end_date = NULL)

{

    if ($end_date == NULL)

        mysql_query("INSERT INTO `groups` (`DeptID`, `Name`, `StartDate`) VALUES
('$this->_dept_id', '$group_name', '$start_date')", $this->_link);

    else

        mysql_query("INSERT INTO `groups` (`DeptID`, `Name`, `StartDate`, `EndDate`)
VALUES ('$this->_dept_id', '$group_name', '$start_date', '$end_date')", $this->_link);

    return mysql_insert_id($this->_link);

}

function DeleteGroup($group_id)

```

```

{
    mysql_query("DELETE FROM `groups` WHERE `ID` = '$group_id'", $this->_link);
}

function AddStudentInfo($student_id, $group_id, $start_date, $end_date = NULL)
{
    if ($end_date == NULL)

        mysql_query("INSERT INTO `studentsinfo` (`PersonID`, `GroupID`, `StartDate`)
VALUES ('$student_id', '$group_id', '$start_date')", $this->_link);

    else

        mysql_query("INSERT INTO `studentsinfo` (`PersonID`, `GroupID`, `StartDate`,
`EndDate`) VALUES ('$student_id', '$group_id', '$start_date', '$end_date')", $this->_link);
}

function GetGroups()
{
    $result = mysql_query("SELECT `ID`, `Name` FROM `groups` WHERE `DeptID` =
'$this->_dept_id' ORDER BY `Name`", $this->_link);

    $groups = array();

    $row = mysql_fetch_row($result);

    while ($row) {

        $groups[] = array(

            'id' => $row[0],

            'name' => $row[1]

        );

        $row = mysql_fetch_row($result);
    }
}

```

```

mysql_free_result($result);

return $groups;

}

function GetGroupName($group_id)

{

    $result = mysql_query("SELECT `Name` FROM `groups` WHERE `ID` = '$group_id'",
$this->_link);

    $row = mysql_fetch_row($result);

    mysql_free_result($result);

    return $row[0];

}

function GetChairTeacher($teacher_id)

{

    $result = mysql_query("SELECT a.Name FROM chairs a, teachersinfo b WHERE a.ID
= b.ChairID and b.PersonID = '$teacher_id'", $this->_link);

    $row = mysql_fetch_row($result);

    mysql_free_result($result);

    return $row[0];

}

function GetStudents($group_id = NULL)

{

    if ($group_id == NULL)

```

```
$query_string = "SELECT s.`ID`, s.`Name`, g.`ID`, g.`Name`, s.`Passport`, s.`City`,  
s.`Phone` FROM `persons` AS s, `studentsinfo` AS si, `groups` AS g WHERE  
si.`PersonID` = s.`ID` AND CURRENT_DATE BETWEEN si.`StartDate` AND  
IFNULL(si.`EndDate`, CURRENT_DATE) AND si.`GroupID` = g.`ID` AND g.`DeptID` =  
'$this->_dept_id' ORDER BY s.`Name`";
```

```
else {
```

```
$query_string = "SELECT s.`ID`, s.`Name` FROM `persons` AS s, `studentsinfo` AS  
si WHERE si.`PersonID` = s.`ID` AND si.`GroupID` = '$group_id' AND CURRENT_DATE  
BETWEEN si.`StartDate` AND IFNULL(si.`EndDate`, CURRENT_DATE) ORDER BY  
s.`Name`";
```

```
$current_group_id = $group_id;
```

```
$current_group_name = $this->GetGroupName($group_id);
```

```
};
```

```
$result = mysql_query($query_string, $this->_link);
```

```
$students = array();
```

```
$row = mysql_fetch_row($result);
```

```
while ($row) {
```

```
if ($group_id == NULL) {
```

```
$current_group_id = $row[2];
```

```
$current_group_name = $row[3];
```

```
};
```

```
$students[] = array(
```

```
'student' => array('id' => $row[0], 'name' => $row[1], 'email' => $row[4], 'city' =>  
$row[5], 'phone' => $row[6]),
```

```
'group' => array('id' => $current_group_id, 'name' => $current_group_name)
```

```
);
```



```
$row = mysql_fetch_row($result);  
  
}  
  
mysql_free_result($result);  
  
return $students;  
  
}
```

```
function GiveTeacherAccess($teacher_id, $access_type, $aceid, $assign_date,  
$revoke_date = NULL)  
  
{  
  
    if ($revoke_date == NULL)  
  
        mysql_query("INSERT INTO `teachersaccess` (`TeacherID`, `AccessType`, `ACEID`,  
`AssignDate`) VALUES ('$teacher_id', '$access_type', '$aceid', '$assign_date')", $this-  
>_link);  
  
        else  
  
            mysql_query("INSERT INTO `teachersaccess` (`TeacherID`, `AccessType`, `ACEID`,  
`AssignDate`, `RevokeDate`) VALUES ('$teacher_id', '$access_type', '$aceid',  
'$assign_date', '$revoke_date')", $this->_link);  
  
}
```

```
function GetExamsAdm()  
  
{  
  
    $query_string="select * from exam order by StartDate desc";  
  
    $result = mysql_query($query_string, $this->_link);  
  
    $exams = array();  
  
    $row = mysql_fetch_array($result);
```

```
while ($row) {  
  
    $groups=array();  
  
    $query_string="select AccessID from examaccess where  
AccessType='G' and examID='".$row["examID"]."'";  
  
    $result1 = mysql_query($query_string, $this->_link);  
  
    while ($row1=mysql_fetch_array($result1))  
  
    {  
  
        $groups[]=$row1["AccessID"];  
  
    }  
  
    $persons=array();  
  
    $query_string="select AccessID from examaccess where  
AccessType='P' and examID='".$row["examID"]."'";  
  
    $result1 = mysql_query($query_string, $this->_link);  
  
    while ($row1=mysql_fetch_array($result1))  
  
    {  
  
        $persons[]=$row1["AccessID"];  
  
    }  
  
    $materials=array();  
  
    $query_string="select materialID from exammaterials where  
examID='".$row["examID"]."'";  
  
    $result1 = mysql_query($query_string, $this->_link);  
  
    while ($row1=mysql_fetch_array($result1))  
  
    {  
  
        $materials[]=$row1["materialID"];  
  
    }  
  
}
```

```

    $exams[] = array(
        'ID' => $row["examID"],
            'Teacher' => $row["TeacherID"],
            'Title' => $row["ExamTitle"],
            'Difficulty' => $row["Difficulty"],
            'StartDate' => $row["StartDate"],
            'EndDate' => $row["EndDate"],
            'Groups' => $groups,
            'Persons' => $persons,
            'Materials' => $materials

        );

        $row = mysql_fetch_array($result);
    };

    mysql_free_result($result);

    return $exams;
}

function GetMaterial($id)
{
    $query_string="select * from subjects where ID='".$id.'" order by name";

    $result = mysql_query($query_string, $this->_link);

    $material = array();

    if ($row = mysql_fetch_array($result))

```

```

        { $str=str_replace("\",\"", $row);
            return $str;
        } else {
            return false;
        }
    }
}

function GetMaterials($parent= '0000')
{
    $query_string="select * from subjects where ParentSubjectID='".$parent.'" order
by name";

    $result = mysql_query($query_string, $this->_link);

    $materials = array();

    $row = mysql_fetch_array($result);

    while ($row) {

        $materials[] = array(

            'ID' => $row["ID"],

            'Name' => str_replace("\",\"", $row["Name"]),

            'Parent' => $row["ParentSubjectID"],

            'Text' => $row["text"],

            'TeacherID' => $row["TeacherID"],

            'ChairID' => $row["chair_id"]

        );

        $row = mysql_fetch_array($result);
    };
};

```

```

mysql_free_result($result);

return $materials;

}

function DeleteMaterial($id)

{

    mysql_query("Delete from `subjects` where id='".$id.'" or ParentSubjectID='".$id.'"',
$this->_link);

    return mysql_insert_id($this->_link);

}

function AddExam($examid, $title, $difficulty, $start, $end, $materials, $groups,
$students, $c1,$c2,$c3, $k1, $k2, $k3)

{

    if ($examid=="0")

    {

        $query_string="insert into exam (TeacherID, ExamTitle, Difficulty,
StartDate, EndDate, c1, c2, c3, k1, k2 , k3) values ('.$this->_id.', ".$title.",
".$difficulty.", ".$start.", ".$end.", ".$c1.", ".$c2.", ".$c3.", ".$k1.", ".$k2.", ".$k3.)";

        $result = mysql_query($query_string, $this->_link);

        $id=mysql_insert_id($this->_link);

    } else {

        $query_string="update exam set ExamTitle='".$title."',
Difficulty='".$difficulty."', StartDate='".$start."', EndDate='".$end."', c1='".$c1."',
c2='".$c2."', c3='".$c3."', k1='".$k1."', k2='".$k2."', k3='".$k3.'" where examID=".$examid;

        $result = mysql_query($query_string, $this->_link);

        $id=$examid;

```

```

}

$query_string="delete from exammaterials where examID='".$id.'";

$sql=mysql_query($query_string);

$query_string="delete from examaccess where examID='".$id.'";

$sql=mysql_query($query_string);

for ($i=0; $i<sizeof($materials); $i++)
{
    $query_string="insert into exammaterials (examID, materialID)
values ('".$id."', '".$materials[$i].')";

    $result = mysql_query($query_string, $this->_link);
}

for ($i=0; $i<sizeof($groups); $i++)
{
    $query_string="insert into examaccess (examID, AccessType,
AccessID) values ('".$id."', 'G', '".$groups[$i].')";

    $result = mysql_query($query_string, $this->_link);
}

for ($i=0; $i<sizeof($students); $i++)
{
    $query_string="insert into examaccess (examID, AccessType,
AccessID) values ('".$id."', 'P', '".$students[$i].')";

    $result = mysql_query($query_string, $this->_link);
}
}

```

```

function DeleteExam($examid)
{
    $query_string="delete from exam where examID='".$examid.'";
    $sql=mysql_query($query_string);
    $query_string="delete from examaccess where examID='".$examid.'";
    $sql=mysql_query($query_string);
    $query_string="delete from exammaterials where examID='".$examid.'";
    $sql=mysql_query($query_string);
    $query_string="delete from examresults where examID='".$examid.'";
    $sql=mysql_query($query_string);
}
}

?>

<script type="text/javascript">
    function confirm_delete(id) {
        var message = "Are you sure? ";
        var return_value = confirm(message);

        if ( return_value === true ) {
//window.navigate("index.php?link=materials&act=del&id=<?=$materials[$i][\"ID\"]?
>");

            //document.write(id);

            //document.write('index.php?link=students&act=del&id='+id);

            window.navigate(id);

```

```
        }  
    }  
</script>
```

### **Teacher.class.php**

```
<?php  
  
class Teacher  
{  
    //private  
    var $_id;  
    var $_name;  
    var $_dept_id;  
    var $_dept_name;  
    var $_chair_id;  
    var $_chair_name;  
  
    var $_link;  
  
    //public  
    function Teacher($id, $link)  
    {
```



```

/* set private fields */

$this->_link = $link;

$this->_id = $id;

/* get name */

$result = mysql_query("SELECT `Name` FROM `persons` WHERE `ID` = '$this->_id'",
$this->_link);

$row = mysql_fetch_row($result);

mysql_free_result($result);

$this->_name = $row[0];

/* get attributes */

$result = mysql_query("SELECT `chairs`.`ID`, `chairs`.`Name` FROM `teachersinfo`
LEFT JOIN `chairs` ON `teachersinfo`.`ChairID` = `chairs`.`ID` WHERE
`teachersinfo`.`PersonID` = '$this->_id' AND `teachersinfo`.`EndDate` IS NULL", $this-
>_link);

$row = mysql_fetch_row($result);

mysql_free_result($result);

$this->_chair_id = $row[0];

$this->_chair_name = $row[1];

}

function GetID()

{

return $this->_id;

}

function GetName()

{

```

```

    return $this->_name;
}

function GetDeptID()
{
    return $this->_dept_id;
}

function GetDeptName()
{
    return $this->_dept_name;
}

function GetChairID()
{
    return $this->_chair_id;
}

function GetChairName()
{
    return $this->_chair_name;
}

function GetProblemsList($subject_id=null, $start, $count)
{
    $query_string = "SELECT DISTINCT `problems`.`ID`, `problems`.`Title`,
`problems`.`Complexity`, `subjects`.`ID`, `subjects`.`Name`, `persons`.`ID`,
`persons`.`Name`, `problems`.`text`, `problems`.`tp`";

    $query_string .= " FROM `problems` LEFT JOIN `subjects` ON `problems`.`SubjectID`
= `subjects`.`ID` LEFT JOIN `persons` ON `problems`.`AuthorID` = `persons`.`ID`";
}

```

```

$query_string .= " WHERE `problems`.`AuthorID` = '$this->_id' ";

if ($subject_id != NULL)

    $query_string .= " AND `subjects`.`ID` = '$subject_id'";

$result = mysql_query($query_string, $this->_link);

$problems_list = array();

$row = mysql_fetch_row($result);

while ($row) {

    $tp="exam";

    if ($row[8]=="2") $tp="test";

    $problems_list[] = array(

        'problem' => array('id' => $row[0], 'title' => $row[1], 'text' => $row[7], 'complexity' =>
$row[2], 'type' => $tp),

        'subject' => array('id' => $row[3], 'name' => $row[4]),

        'author' => array('id' => $row[5], 'name' => $row[6])

    );

    $row = mysql_fetch_row($result);

}

mysql_free_result($result);

return $problems_list;

}

function GetStudentsList()

{

```

```
// $query_string = "SELECT s.`ID`, s.`Name`, g.`ID`, g.`Name`, s.`Passport`, s.`City`,
s.`Phone` FROM `persons` AS s, `studentsinfo` AS si, `groups` AS g WHERE
si.`PersonID` = s.`ID` AND CURRENT_DATE BETWEEN si.`StartDate` AND
IFNULL(si.`EndDate`, CURRENT_DATE) AND si.`GroupID` = g.`ID` AND g.`DeptID` =
'$this->_dept_id' ORDER BY s.`Name`";
```

```
$query_string = "SELECT `persons`.`ID`, `persons`.`Name`, `groups`.`ID`,
`groups`.`Name`, `persons`.`Passport`, `persons`.`City`, `persons`.`Phone`";
```

```
$query_string .= " FROM `persons` JOIN `studentsinfo` ON `persons`.`ID` =
`studentsinfo`.`PersonID` JOIN `groups` ON `studentsinfo`.`GroupID` = `groups`.`ID`
JOIN `teachersaccess` ON `teachersaccess`.`AccessType` = 'P' AND
`teachersaccess`.`ACEID` = `persons`.`ID`";
```

```
$query_string .= " WHERE CURRENT_DATE BETWEEN `studentsinfo`.`StartDate`
AND IFNULL(`studentsinfo`.`EndDate`, CURRENT_DATE) and
`teachersaccess`.`TeacherID` = '$this->_id'";
```

```
$query_string .= " UNION SELECT `persons`.`ID`, `persons`.`Name`, `groups`.`ID`,
`groups`.`Name`, `persons`.`Passport`, `persons`.`City`, `persons`.`Phone`";
```

```
$query_string .= " FROM `persons` JOIN `studentsinfo` ON `persons`.`ID` =
`studentsinfo`.`PersonID` JOIN `groups` ON `studentsinfo`.`GroupID` = `groups`.`ID`
JOIN `teachersaccess` ON `teachersaccess`.`AccessType` = 'G' AND
`teachersaccess`.`ACEID` = `groups`.`ID`";
```

```
$query_string .= " WHERE CURRENT_DATE BETWEEN `studentsinfo`.`StartDate`
AND IFNULL(`studentsinfo`.`EndDate`, CURRENT_DATE) and
`teachersaccess`.`TeacherID` = '$this->_id'ORDER BY 2";
```

```
$result = mysql_query($query_string, $this->_link);
```

```
$students = array();
```

```
$row = mysql_fetch_row($result);
```

```
while ($row) {
```

```
    if ($group_id == NULL) {
```

```

    $current_group_id = $row[2];

    $current_group_name = $row[3];

};

$students[] = array(

    'student' => array('id' => $row[0], 'name' => $row[1], 'email' => $row[4], 'city' =>
$row[5], 'phone' => $row[6]),

    'group' => array('id' => $current_group_id, 'name' => $current_group_name)

);

$row = mysql_fetch_row($result);

}

mysql_free_result($result);

return $students;

}

```

```

function GetStudents($group_type = NULL, $group_id = NULL)

{

if ($group_type != NULL)

switch ($group_type) {

case 'G':

    $current_group_id = $group_id;

    $current_group_name = $this->GetGroupName($group_id);

    $query_string = "SELECT `persons`.`ID`, `persons`.`Name`";

    $query_string .= " FROM `persons`, `studentsinfo` WHERE `persons`.`ID` =
`studentsinfo`.`PersonID`";

```

```
$query_string .= " AND `studentsinfo`.`GroupID` = '$group_id' AND  
CURRENT_DATE BETWEEN `studentsinfo`.`StartDate` AND  
IFNULL(`studentsinfo`.`EndDate`, CURRENT_DATE) ORDER BY 2";
```

```
break;
```

```
case 'H':
```

```
$query_string = "SELECT `persons`.`ID`, `persons`.`Name`, `groups`.`ID`,  
`groups`.`Name`";
```

```
$query_string .= " FROM `persons` JOIN `studentsinfo` ON `persons`.`ID` =  
`studentsinfo`.`PersonID` JOIN `groups` ON `studentsinfo`.`GroupID` = `groups`.`ID`  
JOIN `teachersaccess` ON `teachersaccess`.`AccessType` = 'P' AND  
`teachersaccess`.`ACEID` = `persons`.`ID`";
```

```
$query_string .= " WHERE CURRENT_DATE BETWEEN  
`studentsinfo`.`StartDate` AND IFNULL(`studentsinfo`.`EndDate`, CURRENT_DATE)";
```

```
$query_string .= " UNION SELECT `persons`.`ID`, `persons`.`Name`,  
`groups`.`ID`, `groups`.`Name`";
```

```
$query_string .= " FROM `persons` JOIN `studentsinfo` ON `persons`.`ID` =  
`studentsinfo`.`PersonID` JOIN `groups` ON `studentsinfo`.`GroupID` = `groups`.`ID`  
JOIN `teachersaccess` ON `teachersaccess`.`AccessType` = 'G' AND  
`teachersaccess`.`ACEID` = `groups`.`ID`";
```

```
$query_string .= " WHERE CURRENT_DATE BETWEEN  
`studentsinfo`.`StartDate` AND IFNULL(`studentsinfo`.`EndDate`, CURRENT_DATE)  
AND EXISTS(SELECT `PersonID` FROM `ParticipantsInfo` WHERE `PersonID` =  
`persons`.`ID` AND `CircleID` = '$group_id') ORDER BY 2";
```

```
break;
```

```
}
```

```
else {
```

```
$query_string = "SELECT `persons`.`ID`, `persons`.`Name`, `groups`.`ID`,  
`groups`.`Name`";
```

```
$query_string .= " FROM `persons` JOIN `studentsinfo` ON `persons`.`ID` =
`studentsinfo`.`PersonID` JOIN `groups` ON `studentsinfo`.`GroupID` = `groups`.`ID`
JOIN `teachersaccess` ON `teachersaccess`.`AccessType` = 'P' AND
`teachersaccess`.`ACEID` = `persons`.`ID`";
```

```
$query_string .= " WHERE CURRENT_DATE BETWEEN `studentsinfo`.`StartDate`
AND IFNULL(`studentsinfo`.`EndDate`, CURRENT_DATE) and
`teachersaccess`.`TeacherID` = '$this->_id'";
```

```
$query_string .= " UNION SELECT `persons`.`ID`, `persons`.`Name`, `groups`.`ID`,
`groups`.`Name`";
```

```
$query_string .= " FROM `persons` JOIN `studentsinfo` ON `persons`.`ID` =
`studentsinfo`.`PersonID` JOIN `groups` ON `studentsinfo`.`GroupID` = `groups`.`ID`
JOIN `teachersaccess` ON `teachersaccess`.`AccessType` = 'G' AND
`teachersaccess`.`ACEID` = `groups`.`ID`";
```

```
$query_string .= " WHERE CURRENT_DATE BETWEEN `studentsinfo`.`StartDate`
AND IFNULL(`studentsinfo`.`EndDate`, CURRENT_DATE) and
`teachersaccess`.`TeacherID` = '$this->_id'ORDER BY 2";
```

```
}
```

```
$result = mysql_query($query_string, $this->_link);
```

```
$students = array();
```

```
$row = mysql_fetch_row($result);
```

```
while ($row) {
```

```
    if ($group_type != 'G') {
```

```
        $current_group_id = $row[2];
```

```
        $current_group_name = $row[3];
```

```
    };
```

```
$students[] = array(
```

```
    'student' => array('id' => $row[0], 'name' => $row[1]),
```

```

        'group' => array('id' => $current_group_id, 'name' => $current_group_name)
    );

    $row = mysql_fetch_row($result);

}

mysql_free_result($result);

return $students;

}

function GetGroups()
{
    $result = mysql_query("SELECT g.`ID`, g.`Name` FROM `groups` AS g,
`teachersaccess` AS ta WHERE ta.`TeacherID` = '$this->_id' AND ta.`AccessType` = 'G'
AND ta.`ACEID` = g.`ID` ORDER BY g.`Name`", $this->_link);

    $groups = array();

    $row = mysql_fetch_row($result);

    while ($row) {

        $groups[] = array(

            'id' => $row[0],

            'name' => $row[1]

        );

        $row = mysql_fetch_row($result);

    }

    mysql_free_result($result);

    return $groups;
}

```



```

}

function GetGroupName($group_id)
{
    $result = mysql_query("SELECT `Name` FROM `groups` WHERE `ID` = '$group_id'",
$this->_link);

    $row = mysql_fetch_row($result);

    mysql_free_result($result);

    return $row[0];
}

function GetPersonName($person_id)
{
    $result = mysql_query("SELECT `Name` FROM `persons` WHERE `ID` =
'$person_id'", $this->_link);

    $row = mysql_fetch_row($result);

    mysql_free_result($result);

    return $row[0];
}

function AddProblem($problem_date, $problem_title, $problem_text,
$problem_solution, $problem_remarks, $problem_complexity, $subject, $tp)
{
    mysql_query("INSERT INTO `problems` (`AuthorID`, `Date`, `Title`, `Text`, `Solution`,
`Remarks`, `Complexity`, `SubjectID`, `tp`) VALUES ('$this->_id', '$problem_date',
'$problem_title', '$problem_text', '$problem_solution', '$problem_remarks',
'$problem_complexity', '$subject', '$tp'", $this->_link);

    return mysql_insert_id($this->_link);
}

```

```

}

function GetProblem($problem_id)
{
    $query_string = "SELECT `problems`.`Title`, `problems`.`Text`, `problems`.`Remarks`,
`problems`.`Solution`, `problems`.`Complexity`, `subjects`.`ID`, `subjects`.`Name`,
`persons`.`ID`, `persons`.`Name`, `problems`.`tp`";

    $query_string .= " FROM `problems` LEFT JOIN `subjects` ON `problems`.`SubjectID`
= `subjects`.`ID` LEFT JOIN `persons` ON `problems`.`AuthorID` = `persons`.`ID`";

    $query_string .= " WHERE `problems`.`ID` = '$problem_id'";

    $result = mysql_query($query_string, $this->_link);

    $row = mysql_fetch_row($result);

    mysql_free_result($result);

    $problem = array(

        'title' => $row[0],

        'text' => $row[1],

        'remarks' => $row[2],

        'solution' => $row[3],

        'complexity' => $row[4],

        'subject' => array('id' => $row[5], 'name' => $row[6]),

        'author' => array('id' => $row[7], 'name' => $row[8]),

        'tp'=>$row[9]

    );

    return $problem;
}

function GetQuestion($id)

```

```

{
    $query_string="select a.*, b.Name as Subject from `problems` a, `subjects` b
where a.ID='".$id.'" and a.SubjectID=b.ID and b.chair_id='".$this->_chair_id.'";

    $result = mysql_query($query_string, $this->_link);

    if ($row = mysql_fetch_array($result))
    {
        return $row;
    } else {
        return false;
    }
}

function GetProblems()
{
    $result=mysql_query("select a.* from `problems` a, `subjects` b where
a.SubjectID=b.ID and b.chair_id='".$this->_chair_id.'"', $this->_link);

    $questions = array();

    $row = mysql_fetch_array($result);

    while ($row) {
        $questions[] = $row;

        $row = mysql_fetch_array($result);
    };

    mysql_free_result($result);

    return $questions;
}

```

```

function DeleteProblem($problem_id)
{
    mysql_query("DELETE FROM `problems` WHERE `ID` = '$problem_id' AND
`AuthorID` = '$this->_id'", $this->_link);
}

function AssignProblem($problem_id, $scope, $aceid, $assign_date, $revoke_date =
NULL)
{
    if ($revoke_date == NULL)

        mysql_query("INSERT INTO `problemsaccess` (`ProblemID`, `AccessType`,
`ACEID`, `AssignedBy`, `AssignDate`) VALUES ('$problem_id', '$scope', '$aceid', '$this-
>_id', '$assign_date')", $this->_link);

    else

        mysql_query("INSERT INTO `problemsaccess` (`ProblemID`, `AccessType`,
`ACEID`, `AssignedBy`, `AssignDate`, `RevokeDate`) VALUES ('$problem_id', '$scope',
'$aceid', '$this->_id', '$assign_date', '$revoke_date')", $this->_link);
}

function AddMaterial($parent, $name, $text)
{
    mysql_query("INSERT INTO `subjects` (`Name`, `ParentSubjectID`, `Text`,
`TeacherID`, `chair_id`) VALUES ('".addslashes($name)."', '$parent',
'".addslashes($text)."', '$this->_id', '$this->_chair_id')", $this->_link);

    return mysql_insert_id($this->_link);
}

```

```

function DeleteMaterial($id)
{
    mysql_query("Delete from `subjects` where id='".$id.'" or ParentSubjectID='".$id.'"',
$this->_link);

    return mysql_insert_id($this->_link);
}

```

```

function GetMaterials($parent= '0000')
{
    $query_string="select * from subjects where ParentSubjectID='".$parent.'" and
chair_id='".$this->_chair_id.'" order by ID";

    $result = mysql_query($query_string, $this->_link);

    $materials = array();

    $row = mysql_fetch_array($result);

    while ($row) {

        $materials[] = array(

            'ID' => $row["ID"],

            'Name' => str_replace("\'", "", $row["Name"]),

            'Parent' => $row["ParentSubjectID"],

            'Text' => $row["text"],

            'TeacherID' => $row["TeacherID"],

            'ChairID' => $row["chair_id"]

        );

        $row = mysql_fetch_array($result);
    }
}

```

```

};

mysql_free_result($result);

return $materials;

}

function GetMaterial($id)

{

    $query_string="select * from subjects where ID='".$id.'" and chair_id='".$this->_chair_id.'" order by name";

    $result = mysql_query($query_string, $this->_link);

    $materials = array();

    if ($row = mysql_fetch_array($result))

        {

            $str=str_replace("\'", "'", $row);

            return $str;

        } else {

            return false;

        }

}

function GetExams()

{

    $query_string="select * from exam where TeacherID='".$this->_id.'" order by StartDate desc";

    $result = mysql_query($query_string, $this->_link);

    $exams = array();

```

```

$row = mysql_fetch_array($result);

while ($row) {

    $groups=array();

    $query_string="select AccessID from examaccess where
AccessType='G' and examID='".$row["examID"]."'";

    $result1 = mysql_query($query_string, $this->_link);

    while ($row1=mysql_fetch_array($result1))

    {

        $groups[]=$row1["AccessID"];

    }

    $persons=array();

    $query_string="select AccessID from examaccess where
AccessType='P' and examID='".$row["examID"]."'";

    $result1 = mysql_query($query_string, $this->_link);

    while ($row1=mysql_fetch_array($result1))

    {

        $persons[]=$row1["AccessID"];

    }

    $materials=array();

    $query_string="select materialID from exammaterials where
examID='".$row["examID"]."'";

    $result1 = mysql_query($query_string, $this->_link);

    while ($row1=mysql_fetch_array($result1))

    {

        $materials[]=$row1["materialID"];

```

```

    }

    $exams[] = array(
        'ID' => $row["examID"],
        'Title' => $row["ExamTitle"],
        'Duration' => $row["duration"],
        'StartDate' => $row["StartDate"],
        'EndDate' => $row["EndDate"],
        'Groups' => $groups,
        'Persons' => $persons,
        'Materials' => $materials
    );

    $row = mysql_fetch_array($result);
};

mysql_free_result($result);

return $exams;
}

function AddExam($examid, $title, $difficulty, $start, $end, $dur, $materials,
$groups, $students, $c1,$c2,$c3, $k1, $k2, $k3)
{
    if ($examid=="0")
    {
        $query_string="insert into exam (TeacherID, ExamTitle, Difficulty,
StartDate, EndDate, c1, c2, c3, k1, k2, k3, duration) values (".$this->_id.", ".$title.",
".$difficulty.", ".$start.", ".$end.", ".$c1.", ".$c2.", ".$c3.", ".$k1.", ".$k2.", ".$k3.",
".$dur.)";
    }
}

```



```

$result = mysql_query($query_string, $this->_link);

    $id=mysql_insert_id($this->_link);

    } else {

        $query_string="update exam set TeacherID='".$this->_id."',
ExamTitle='".$title."', Difficulty='".$difficulty."', StartDate='".$start."', EndDate='".$end."',
c1='".$c1."', c2='".$c2."', c3='".$c3."', k1='".$k1."', k2='".$k2."', k3='".$k3."',
duration='".$dur.'" where examID='".$examid;

        $result = mysql_query($query_string, $this->_link);

        $id=$examid;

    }

    $query_string="delete from exammaterials where examID='".$id.'";

    $sql=mysql_query($query_string);

    $query_string="delete from examaccess where examID='".$id.'";

    $sql=mysql_query($query_string);

    for ($i=0; $i<sizeof($materials); $i++)

    {

        $query_string="insert into exammaterials (examID, materialID)
values ('".$id."', '".$materials[$i].')";

        $result = mysql_query($query_string, $this->_link);

    }

    for ($i=0; $i<sizeof($groups); $i++)

    {

        $query_string="insert into examaccess (examID, AccessType,
AccessID) values ('".$id."', 'G', '".$groups[$i].')";

```

```

$result = mysql_query($query_string, $this->_link);

}

for ($i=0; $i<sizeof($students); $i++)

{

    $query_string="insert into examaccess (examID, AccessType,
AccessID) values ('".$id."', 'P', '".$students[$i]."')";

    $result = mysql_query($query_string, $this->_link);

}

}

function DeleteExam($examid)

{

    $query_string="delete from exam where examID='".$examid.'";

    $sql=mysql_query($query_string);

    $query_string="delete from examaccess where examID='".$examid.'";

    $sql=mysql_query($query_string);

    $query_string="delete from exammaterials where examID='".$examid.'";

    $sql=mysql_query($query_string);

    $query_string="delete from examresults where examID='".$examid.'";

    $sql=mysql_query($query_string);

}

}

?>

```

<script type="text/javascript">

```
function confirm_delete(id) {

    var message = "Are you sure? ";

    var return_value = confirm(message);

    if ( return_value === true ) {

        //window.navigate("index.php?link=materials&act=del&id=<?=$materials[$i][\"ID\"]?
>");

        //document.write(id);

        //document.write('index.php?link=students&act=del&id='+id);

        window.navigate(id);

    }

}

</script>
```

### **Student.class.php**

```
<?php

class Student

{

    //private
```

```

var $_id;

var $_name;

var $_dept_id;

var $_dept_name;

var $_group_id;

var $_group_name;

var $_link;

//public

function Student($id, $link)

{

    /* set private fields */

    $this->_id = $id;

    $this->_link = $link;

    /* get name */

    $result = mysql_query("SELECT `Name` FROM `persons` WHERE `ID` = '$this->_id'",
$this->_link);

    $row = mysql_fetch_row($result);

    mysql_free_result($result);

    $this->_name = $row[0];

    /* get attributes */

    $result = mysql_query("SELECT `groups`.`ID`, `groups`.`Name` FROM `studentsinfo`
LEFT JOIN `groups` ON `studentsinfo`.`GroupID` = `groups`.`ID` WHERE
`studentsinfo`.`PersonID` = '$this->_id' AND `studentsinfo`.`EndDate` IS NULL", $this-
>_link);

    $row = mysql_fetch_row($result);

```

```
mysql_free_result($result);  
  
$this->_group_id = $row[0];  
  
$this->_group_name = $row[1];  
  
}
```

```
function GetID()  
  
{  
  
    return $this->_id;  
  
}
```

```
function GetName()  
  
{  
  
    return $this->_name;  
  
}
```

```
function GetDeptID()  
  
{  
  
    return $this->_dept_id;  
  
}
```

```
function GetDeptName()  
  
{  
  
    return $this->_dept_name;  
  
}
```

```
function GetGroupID()  
  
{  
  
    return $this->_group_id;  
  
}
```

```
}
```

```
function GetGroupName()
```

```
{
```

```
    return $this->_group_name;
```

```
}
```

```
function GetChairName($chair_id)
```

```
{
```

```
    $result = mysql_query("SELECT `Name` FROM `chairs` WHERE `ID` = '$chair_id'",  
$this->_link);
```

```
    $row = mysql_fetch_row($result);
```

```
    mysql_free_result($result);
```

```
    return $row[0];
```

```
}
```

```
function GetPersonName($person_id)
```

```
{
```

```
    $result = mysql_query("SELECT `Name` FROM `persons` WHERE `ID` =  
'$person_id'", $this->_link);
```

```
    $row = mysql_fetch_row($result);
```

```
    mysql_free_result($result);
```

```
    return $row[0];
```

```
}
```

```

function GetMaterials($parent= '0000')
{
    $query_string="select distinct a.* from subjects a, teachersaccess b where
a.TeacherID=b.TeacherID and ((b.ACEID=".$this->_id." and
AccessType='P')or(AccessType='G' and b.ACEID=(select GroupID from studentsinfo
where PersonID=".$this->_id."))) and ParentSubjectID=".$parent." order by a.TeacherID,
a.ID";

    $result = mysql_query($query_string, $this->_link);

    $materials = array();

    $row = mysql_fetch_array($result);

    while ($row) {

        $materials[] = array(

            'ID' => $row["ID"],

            'Name' => str_replace("\'", "", $row["Name"]),

            'Parent' => $row["ParentSubjectID"],

            'Text' => $row["text"],

            'TeacherID' => $row["TeacherID"],

            'ChairID' => $row["chair_id"]

        );

        $row = mysql_fetch_array($result);

    };

    mysql_free_result($result);

    return $materials;

}

```

```
function GetMaterial($id)
```

```
{
```

```
    $query_string="select * from subjects where ID='".$id.'" order by name";
```

```
    $result = mysql_query($query_string, $this->_link);
```

```
    $materials = array();
```

```
    if ($row = mysql_fetch_array($result))
```

```
        { $str=str_replace("\",\"", $row);
```

```
            return $str;
```

```
        } else {
```

```
            return false;
```

```
        }
```

```
}
```

```
function GetExams()
```

```
{
```

```
    $query_string="select * from exam where EndDate>=NOW() order by  
StartDate desc";
```

```
    $result = mysql_query($query_string, $this->_link);
```

```
    $exams = array();
```

```
    $row = mysql_fetch_array($result);
```

```
    while ($row) {
```

```
        $groups=array();
```

```
        $query_string="select AccessID from examaccess where  
AccessType='G' and examID='".$row["examID"]."'";
```



```

$result1 = mysql_query($query_string, $this->_link);

    while ($row1=mysql_fetch_array($result1))

    {

        $groups[]=$row1["AccessID"];

    }

    $persons=array();

    $query_string="select AccessID from examaccess where
AccessType='P' and examID='".$row["examID"]."'";

    $result1 = mysql_query($query_string, $this->_link);

    while ($row1=mysql_fetch_array($result1))

    {

        $persons[]=$row1["AccessID"];

    }

    $materials=array();

    $query_string="select materialID from exammaterials where
examID='".$row["examID"]."'";

    $result1 = mysql_query($query_string, $this->_link);

    while ($row1=mysql_fetch_array($result1))

    {

        $materials[]=$row1["materialID"];

    }

    $exams[] = array(

    'ID' => $row["examID"],

        'Title' => $row["ExamTitle"],

```

```
'Duration' => $row["duration"],  
    'StartDate' => $row["StartDate"],  
    'EndDate' => $row["EndDate"],  
    'Groups' => $groups,  
    'Persons' => $persons,  
    'Materials' => $materials  
  
    );  
  
    $row = mysql_fetch_array($result);  
  
};  
  
mysql_free_result($result);  
  
return $exams;  
  
}  
  
}>
```