STATE COMMITTEE OF COMMUNICATION, INFORMATIZATION AND TELECOMMUNICATION TECHNOLOGIES OF THE REPUBLIC OF UZBEKISTAN
TASHKENT UNIVERSITY OF INFORMATION TECHNOLOGIES

Allow the protection
Head of the department
_____
_____2014y.

FINAL QUALIFYING WORK

On theme:Development of information storage systematization and administration subsystem about awarding academic degrees and titles «Olim».

Graduate	_____	Kogay A. A.
	sign	Fullname
Scientific adviser	_____	Kogay V. N.
	sign	Fullname
Life safe consultant	_____	Abdullaeva S.M.
	sign	Fullname
Reviewer	_____	_____
	sign	Fullname

Tashkent – 2014

THE STATE COMMITTEE FOR COMMUNICATION, INFORMATIZATION AND TELECOMMUNICATION TECHNOLOGIES OF THE REPUBLIC OF UZBEKISTAN
TASHKENT UNIVERSITY OF INFORMATION TECHNOLOGIES

Faculty:Computer engineering, department:Multimedia technologies, direction (specialty): 5521900 – «Informatics and information technologies»

A P P R O V E D

Head of the department _____

« ___ »_____ 2014.

ASSIGNMENT

ofKogay Artyom Arkadevichfinal qualifying work
(full name)

1. Theme: Development of information storage systematization and administration subsystem about awarding academic degrees and titles «Olim».
2. Theme approved in the University decree № 254-15 on April 19,2014.
3. Work completion date:31.05.2014
4. Resources: technical data, HAC materials, training manuals, internet resources, guides
5. Contents of the explanatory notes (listofaddressed issues during development): Analysis of the subject area, requirements analysis for the subsystem development, database structuredesign, subsystem architecturedesign, subsystem realization via a programming language.
6. List of graphic material: Tables, diagrams, user interfaces, presentation.
7. Assignmentissuedate:03.03.2014

Scientific adviser _____
sign

Assignment received _____
sign

8. Individual sections consultantsof the graduation work

Section	Full name of scientific adviser	Signature, date	
		Task issued	Task received
Main part	Kogay V.N.		
Life safety	Abdullaeva S.M.		

9. Progress chart

№	Name of the section	Deadline	Completion mark
1	Introduction	05.03.2014 – 17.03.2014	
2	Analysis of the subject area	18.03.2014 – 31.03.2014	
3	Subsystem development	01.04.2014 – 30.04.2014	
4	Life safety	01.05.2014 – 15.05.2014	
5	Conclusion	15.05.2014 – 19.05.2014	
6	Preparation of presentation	20.05.2014 – 31.05.2014	

Graduate_____ «____»_____ 2014 year.

sign

Scientific adviser _____ «____»_____ 2014 year.

sign

This final qualification work is devoted to develop the information system «Olim» to store and process information about awarding academic degree and titles for science teachers in areas of science, technology, education and culture of Republic of Uzbekistan. In this qualification work was realizedtools forsystematization and administration of storage information, were worked out the statement of the problem and practical implementation of the system are given.ASP.NET, programming language C #, and also DBMS Microsoft SQL Server 2008R2 were used in developing the platform.

Данная выпускная квалификационная работа посвящена разработке информационной системе «Olim» для хранения и обработки информации о присужденных научных степеней и ученых званий научным, научно-педагогическим работникам по направлениям науки, техники, образования и культурыРеспублики Узбекистан. В данной работе был реализован инструментарий для систематизации и администрировании хранимой информации, приведена постановка задачи и практическая реализация системы. При разработке были использованы платформа ASP.NET, язык программирования C#, а также СУБД Microsoft SQL Server 2008 R2.

Ушбу битирув малакавий иши Ўзбекистон Республикаси олимларига, илмий ва илмий педагогик олимларига берилган илмий даража ва илмий унвонлар хакидаги маълумотларни саклаш ва ишлов бериш учун мўлжалланган «Олим» тизимини яратишга багишланган. Ушбу битирув малакавий ишида сакланаётган маълумотларни системалаштириш ва бошкариш амалга оширилган, масаланинг кўйилиши ва амалга кўйилиши кўрсатилган. Тизимни яратишда ASP.NET, C# дастурлаш тили, МОБТ Microsoft SQL Server 2008 R2 дан фойдаланилди.

CONTENT

INTRODUCTION

With the development of modern society information technology deeply into people's lives. They very quickly became a vital stimulus for the development not only of the global economy, but also in other spheres of human activity. Now it is difficult to find an area which is not the use of information technology.

At present, Uzbekistan is experiencing major changes in information technology. Is being developedNational Information System, which is defined as the totality of information systems of state organs, sectoral and regional information systems and information systems of businesses and individuals.

Purpose of formation the National Information System is to create a unified technical policy to ensure the integration of information systems, resources, and databases that are used in government agencies, as well as common technological solution to provide online public services for the implementation of "e-government".

This system also includes an information system created in order to ensure uniform state policy in the field of state attestation of scientific and scientific-pedagogical personnel to ensure fast and efficient work, as well as to organize and administer the stored information.

The actuality of the topic and the implementation of the tasks can reference to the Resolution of the President of the Republic of Uzbekistan dated 21 March 2012 "On measures for further implementation and development of modern information and communication technologies", determine targets of development of informatization processes in the country in the near future.

Starting from 2013in the country begun work on the formation and funding of the state order for projects to implement interagency integrated information systems of public authorities, the creation of interagency data networks and creating a system of "electronic government" and the national segment of the Internet.

As is generally known, during the activity of any organization accumulates a lot of different information.Today in our country, such information in many organizations are stored in paper form and during processing information is necessary to get all the documents from the archives and repositories. It takes a lot of time and labor, it also happens that some documents were lost. Implementation of information systems can significantly reduce the volume of documents in paper form, as well as significantly decreases the required time. Their further development in the future will help to create interagency information systems, which will be available at once to several organizations.The development of such systems will significantly reduce the time required to send and receive information between the organizations, will save the amount of paper, will give an opportunity to receive government services electronically to citizens and legal entities.

The **purpose** of the final qualification work is development of information storage systematization and administration subsystem about awarding academic degrees and titles "Olim".

Objective is to automate the management of information resources including planning database and therefore improves the speed and quality of work, and the systematization of stored information that allows you to quickly and obtain the necessary information.

Graduate work consists of an introduction, three chapters and a conclusion.

In the introduction, given the relevance of the work, as well as the goals and tasks necessary for implementation.

In the first chapter includes the description of the object of automation, as well as basic concepts of systematization and administration information.

The second chapter shows the algorithm subsystem design, and steps to design and implement methods.

The third chapter provides labor protection requirements and technical safety.

Conclusion contains the basic theoretical and practical conclusions to the final qualification work.

CHAPTER 1. THEORETICAL BASES

1.1. Description of the object of the subject area

Object of the subject area is the Higher Attestation Commission (HAC) - a government body for awarding degrees and awarding academic titles for science teachers in areas of science, technology, education and culture.

Higher Attestation Commission has rights of Cabinet of Ministers Office and is responsible for the condition and improvement of the public attestation of scientific and scientific-pedagogical personnel in the country.

The main tasks of the Higher Attestation Commission are:

- management of certification of scientific and pedagogical personnel and ensuring uniformity of requirements to academic degrees and academic titles;

- quality control of special councils and dissertations, their scientific and practical value;

- reduction the qualification requirements for academic titles and degrees in line with international standards.

HAC in accordance with its tasks:

- creates on the motions of ministries and agencies, academies of sciences specialized councils on protection of dissertations for the degree of doctor and candidate of sciences at leading universities and research institutions;

- approves chairmen and membership of the specialized councils and establishes a list of occupations for which the council are entitled to admission thesis defense;

- together with the concerned ministries and departments considering in order to control themes planned doctoral and master's theses to the requirements of scientific and technical and social progress;

- awards at the request of the specialized council Doctorate degree, PhD and assigns academic rank of professor, associate professor and senior researcher;

- provides nostrification diplomas and certificates of academic degrees and academic ranks of other states, draws diplomas doctors and candidates of sciences and certificates professor, associate professor and senior researcher;

- analyzes protected dissertations and systematically sends information to ministries, departments of the Republic on the research performed, the results of which are outlined in the thesis, with appropriate recommendations on the possibility of their implementation in practice.

HAC publishes bulletin on attestation of scientific and scientific-pedagogical personnel. The bulletin also publishes information of thesis defense and conferring degrees.

HAC has the right to:

- perform state control over the activities of the specialized councils and scientific councils on certification of scientific personnel;

- listen at the meetings of the Scientific Council and the Presidium of the Higher Attestation Commission reports chairpersons of the specialized councils for the purpose of studying and summarizing work experience;

- take action in respect of specialized councils until the closing in case of substandard work or flagrant violation of the established procedure for the protection of theses;

- in necessary cases appoint opponents and referees, attract highly qualified specialists of higher educational institutions, academies of sciences, enterprises, institutions and organizations regardless of their departmental subordination to inspect the activities of the specialized councils control examination of individual theses. During their participation in the inspection activities of the specialized councils serve as an opponent and the referee attracted scientists and specialists are exempt from fulfilling their duties with pay at the place of permanent employment. The participation of scientists in the system attestation of scientific and scientific-pedagogical personnel is considered and taken into account as an important aspect of their social and scientific activities;

- exercise control over organization and the level of candidate examinations;

- get in the prescribed manner the annual accounts of organizations and agencies, where the Council for awarding degrees and conferring titles, regardless of their affiliation and necessary information from ministries and agencies to develop proposals and undertake activities to improve the training and certification of scientific and teachers in the country;

- convene in the prescribed manner interagency meeting to address issues related to improving the system of attestation of scientific and scientific-pedagogical personnel, improve the scientific level and the practical value of theses;

- participate in the work of international organizations, meetings, conferences on training and certification of scientific and scientific-pedagogical personnel of higher qualification, the exchange of experience with similar organizations in other countries;

- direct if necessary to specialized councils dissertations for additional opinion on the importance of science and the national economy in accordance with the requirements imposed on the theses;

- deprive the established procedure of scientific and pedagogical workers degrees and academic titles.

HAC is guided by the laws of the Republic of Uzbekistan, other decisions of the Supreme Council of the Republic of Uzbekistan, decrees of the President of the Republic of Uzbekistan, decrees, orders of the Cabinet of the President of the Republic of Uzbekistan, these Regulations and other regulations relating to the scope of its activities.

HAC headed by a chairman appointed by Presidential Decree. HAC has one first deputy and a deputy chairman and chief scientist Secretary, appointed by Cabinet of Ministers at the President of the Republic of Uzbekistan. Allocation of responsibilities between the Vice chairman and Chief Scientific Secretary by the chairman of HAC.

Higher Attestation Commission chairman, his deputies, Chief Scientific Secretary are responsible for implementation of the HAC functions, direct and

control the activities of expert councils, departments and other units of the Higher Attestation Commission.

Chairman of the Higher Attestation Commission publishes on the basis and in pursuance of the laws, decrees of the President of the Republic of Uzbekistan, as well as the decisions and orders of the Cabinet of Ministers of the orders and instructions on the organization of attestation of scientific and scientific-pedagogical personnel, obligatory for execution by enterprises, organizations, institutions, regardless of their departmental subordination, as well as checks of their execution.[1]

Higher Attestation Commission was established by Presidential Decree of 31 March 1992 "On the formation of the Higher Attestation Commission under the Cabinet of Ministers of the Republic of Uzbekistan."

By government decisions HAC provided the following rights:

1. Higher Attestation Commission under the Cabinet of Ministers (WAC of Republic of Uzbekistan) coordinates the activities of the Academy of Sciences, branch academies, ministries, departments, corporations, associations, research organizations and institutions of the country regardless of their departmental subordination and forms of ownership in the implementation of state policy on training and certification scientific and pedagogical personnel, international cooperation in the certification of scientific personnel.

2.

InaccordancewithitstasksandactivitiesprovidetheHigherAttestationCommissionundertheCabinetofMinisterstherightto:

-develop and approve a nomenclature of specialties of scientific and pedagogical workers in areas of science, technology, education and culture;

- create specialized councils and approve their personal structure for doctoral and master's theses, expert advice on the directions of science, consider and approve the proposal of the scientific councils for awarding titles;

- implement a systematic and effective control over the work of the specialized councils, to make changes in their composition, to issue permits for single defense of theses, take the decision to suspend operations or close the tips;

- conduct state examination defended dissertations involving in necessary cases leading scientists and specialists of the republic, as well as foreign experts;

- receive from the ministries and departments of the Academy of Sciences, branch academies, corporations, associations, organizations and institutions materials necessary to perform assigned tasks;

- publish the journal "Bulletin of HAC."[2]

1.2. Systematization of information

An important feature of effective work is quick access to the necessary resources. If the job requires information equipment, it is necessary to provide easy, quick search of information, as well as the systematization of new information.

The primary and most important stage of many business processes in any organization is to systematize information. Thanks to meticulous systematization of information can achieve good results in the optimization of workflow, as well as savings company and employee time. Without prior systematization information is impossible to imagine such critical business processes such as document management, records management, creation of material and electronic archive various databases.

Systematization of information include:

- search methods and accumulation of information;

- classification and indexing of information;

- ways to access to information;

- ways of presenting information;

- processing of requests for information search.[3]

The systematization of information means a kind of classification of all documents of the organization for the various groups. Each company selects the most convenient method of organizing information, one or the other type of classification (or a combination of these types).

Systematization of information involves the processing of information in order to bring it to a certain type of information and interpretation, allowing the individual to react in a certain way on the information received.

Exactly systematization of information is a preliminary stage for important work in proceedings as compiling nomenclature of affairs, document indexing, cataloging. Without high quality and meticulous systematization impossible to create an archive of the company or a paper or electronic, it is impossible to full functioning. Qualitative systematization of documents - this is the order and reliability, the search is completed at the time of the case and the lack of problems in relations with the inspection bodies and inspection. Systematization of information is the key to the success of every organization.

1.3. Administration of data

Administration of data - management of information resources, including planning database, development and implementation of standards, determination of restrictions of and procedures, as well as conceptual and logical database design.[4]

Database - a collection of records organized in such a way as to facilitate the search for a specific record or set of related records, or certain information contained in these records.

There are numerous varieties of databases that differ by various criteria.

But because the basis of any database is the information structure database is divided into three types: tabular (relational), network, hierarchical.

Hierarchical database - each object in such information storage is represented as a specific entity, that is, this entity can have child elements, the

parent items, and those child may be more child elements, but there is one object from which it starts.An example of hierarchical database can be an XML document or your file system, the file system example I gave the computer when considering the structure of an XML document, under the heading Notes on XML.

This type of database optimized for reading information, i.e., a database with a hierarchical structure are able to quickly choose and give the requested information to its users. But such a structure does not allow as quickly sort through the information, then can give an example of life, the computer can easily work with a particular file or folder (which are essentially hierarchical structure of objects) but the scan of your computer is carried out for very long. Second example - the registry Windows.

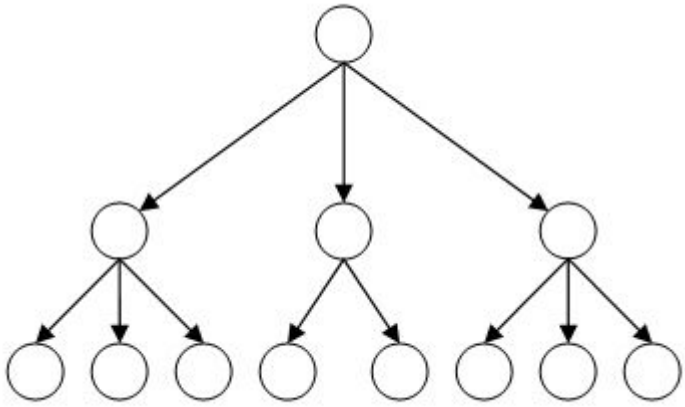

Figure 1.1. The hierarchical structure of the database.

On figure 1.1. shows the structure of the hierarchical database. At the top there is a parent or root element, below are child elements the elements that are at the same level are called brothers, well or adjacent elements. Accordingly, the lower the level of an element, the nesting of this element more.

The network databases are kind of modification of hierarchical databases. In hierarchical databases in each child can have only one descendant. The network databases are different from hierarchical order that the child element can be ancestors, that is, elements of standing above him. On figure 1.2 shows the network structure of the database.

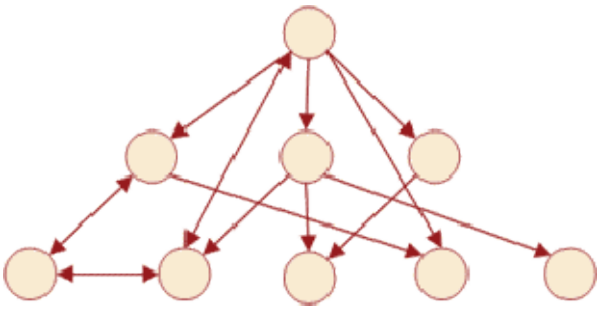

Figure 1.2. The network structure of the database.

Arelational databaseis a set of related tables, each of which contains information about the objects of a certain kind. Each row contains information about a single object (such as a car, computer, client), and the columns of the table contain different characteristics of these objects - attributes (e.g., engine number, brand processor, phone companies or customers).

Table rows are called records. All table entries have the same structure - they consist of fields (data elements) are stored in the object's attributes (figure 1.3.). Each field contains one characteristic of an object and represents a specified data type (for example, a text string, number, date). To identify records using the primary key. The primary key is a set of fields in a table whose values uniquely identify each record in a table.

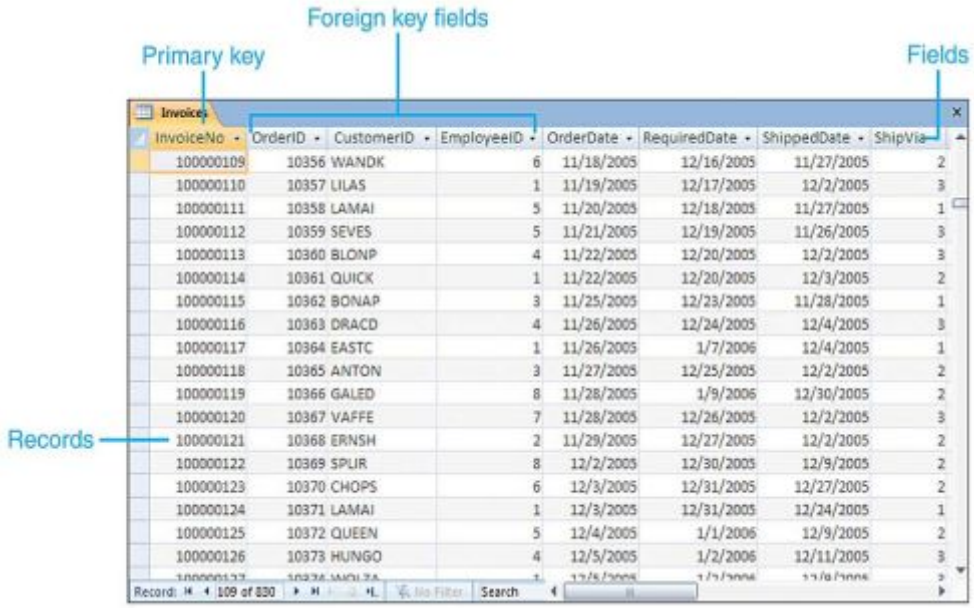

Figure 1.3. Structure of relational table.

The dignity of the relational data model is in its simplicity, clarity and ease of physical implementation on a computer. The simplicity and clarity for the user were the main reason for their widespread use. In designing a relational database apply strict rules based on mathematical apparatus. Complete data independence. When the structure of relational changes that require your applications to produce minimal.[5]

Database Design - the process of creating a database schema and identify constraints.

The development of databases can be divided into several steps:

1.Conceptual design - the collection, analysis and editing of the data requirements. To do this, the following activities:

- domain analysis, the study of its information structure
- identification of all fragments, each of which is characterized by user submission, information objects and relationships between them, the processes of information objects
- modeling and integration of all representations

2.Logical design - creating a database schema based on a specific data model, such as the relational data model. For relational data model logical model - a set of schemes relations usually with a primary key, and the "connections" between relationships representing foreign keys.

- Convert the conceptual model into a logical model is usually carried out by formal rules. This step can be largely automated.
- At the logical design takes into account the specifics of the particular data model, but can not take into account the specifics of a particular DBMS.

3**.** Physical database design- describe the process of preparing the implementation of the database on secondary storage devices; at this stage are considered basic relationships, business files and indexes intended to provide efficient access to data, as well as all related constraints and remedies., атакжевсесвязанныесэтимограниченияцелостностиисредстваzащиты.

1.4. Statement of the problem

Is required to create a system of «Olim» systematization and administration of stored information, in accordance with the requirements established in this system.

«Olim» is designed to collect, process, store and systematization of information about the approved scientific degrees and academic titles, diplomas of nostrification to award degrees.

This information system is formed and maintained on the basis of information resources HAC.

Requisites of the «Olim» includes the following data:

1. Surname, first name
2. Academic degree
3. Name of the doctoral dissertation
4. Code
5. Title for doctoral dissertation

Information reference book scientists of Uzbekistan Academy of Sciences, must satisfy the following functional requirements

- store classifier scientific of directions and specialties of Uzbek Academy of Sciences;
- store and provide access to information and scientific degrees;
- store and provide access to a list of scientists who have academic degrees;
- store and provide access to a list of scientists with academic titles;
- store and provide access to the list of institutions where scientists are working;
- store and provide access to information to list of scientific papers;
- store and provide access to information about dissertations and thesis.
- generate reports according to established forms;
- ensure convenient input, searching and processing information.

Information reference book of scientific publications Uzbek Academy of Sciences, and HAC, Coordination Council of Scientific and Technological

Development under the Cabinet of Ministers must satisfy the following functional requirements:

- store and provide access to information about the responsible institutions, organizations and agencies, ministries;

- store and provide access to a list of scientific publications Uzbek Academy of Sciences, Coordination Council of Scientific and Technological Development under the Cabinet of Ministers and HAC;

- store and provide access to the list of international publications of scientific and educational institutions of Uzbekistan;

First chapter summary

In the first chapter was analyzed object of automation – HAC, were analyzed reasons for its creation, considered its functions and powers.

Were considered the theoretical basis organizing and storing information and analyzed existing types of database design.

Were given formulation of the problem and identified modules of the system, requiring design and development.

CHAPTER 2. DESIGNING OF SYSTEM

2.1. The modeling

Domain modeling tool during the conceptual design is a model of "entity-relationship". It is often called ER-model.

Model "entity-relationship" (ER-model) - s a data model for describing the data or information aspects of a business domain or its process requirements, in an abstract way that lends itself to ultimately being implemented in a database such as a relational database. The main components of ER models are entities (things) and the relationships that can exist among them.

Entity–relationship modeling was developed by Peter Chen and published in a 1976 paper.However, variants of the idea existed previously, and have been devised subsequently such as supertype and subtype data entities and commonality relationships.

This is the highest level ER model in that it contains the least granular detail but establishes the overall scope of what is to be included within the model set. The conceptual ER model normally defines master reference data entities that are commonly used by the organization. Developing an enterprise-wide conceptual ER model is useful to support documenting the data architecture for an organization.

A conceptual ER model may be used as the foundation for one or more logical data models. The purpose of the conceptual ER model is then to establish structural metadata commonality for the master data entities between the set of logical ER models.

The main advantages of ER-models:

• clarity;

• models allow to design a database with a large number of objects and attributes;

• ER-model implemented in many computer-aided design databases (e.g., ERWin).

Figure 2.1 shows the model of "entity-relationship"of subsystem «Olim».

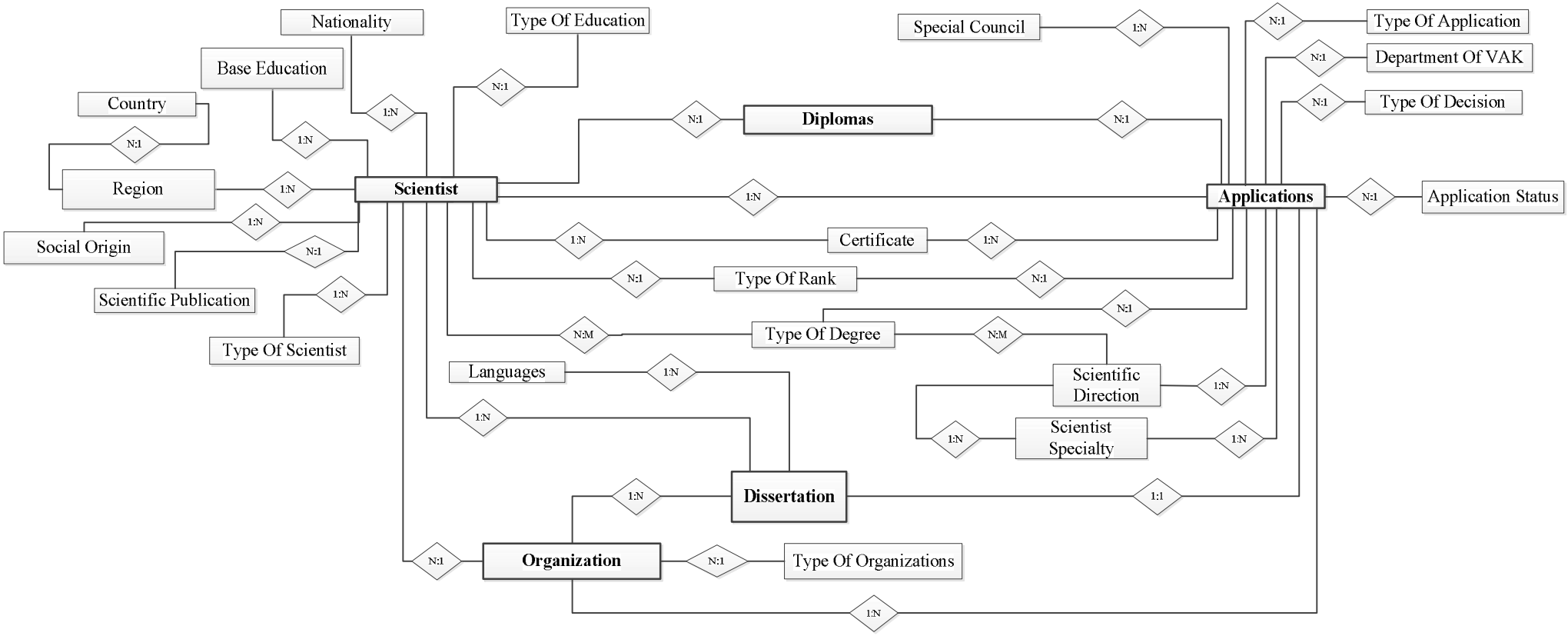

Figure 2.1. The model of "entity-relationship" of subsystem «Olim».

Relational datamodel represents the objects and the relationships between them in the form of tables, andrelationship also considered as objects. All the lines that make up a table in a relational database must have a primary key.

The purpose of the relational model is to provide a declarative method for specifying data and queries: users directly state what information the database contains and what information they want from it, and let the database management system software take care of describing data structures for storing the data and retrieval procedures for answering queries.

Most relational databases use the SQL data definition and query language; these systems implement what can be regarded as an engineering approximation to the relational model. A table in an SQL database schema corresponds to a predicate variable; the contents of a table to a relation; key constraints, other constraints, and SQL queries correspond to predicates.

This model is characterized by the simplicity of the data structure, user-friendly tabular presentation and the possibility of using the formal apparatus of the algebra of relations and relational calculus for data processing.

Each relational table is a two-dimensional array that has the following properties:

1. Each element of table corresponds to a single data element.

2. All columns in the table are homogeneous, i.e. all elements in a column are the same type and length.

3. Each column has a unique name.

4. Duplicate rows in the table are not available;

5. Order of the rows and columns can be arbitrary.

Based on the relational model are built relational databases.

Figures 2.2.2.3., 2.4.и 2.5.shows the relational model of the database of the subsystem «Olim».

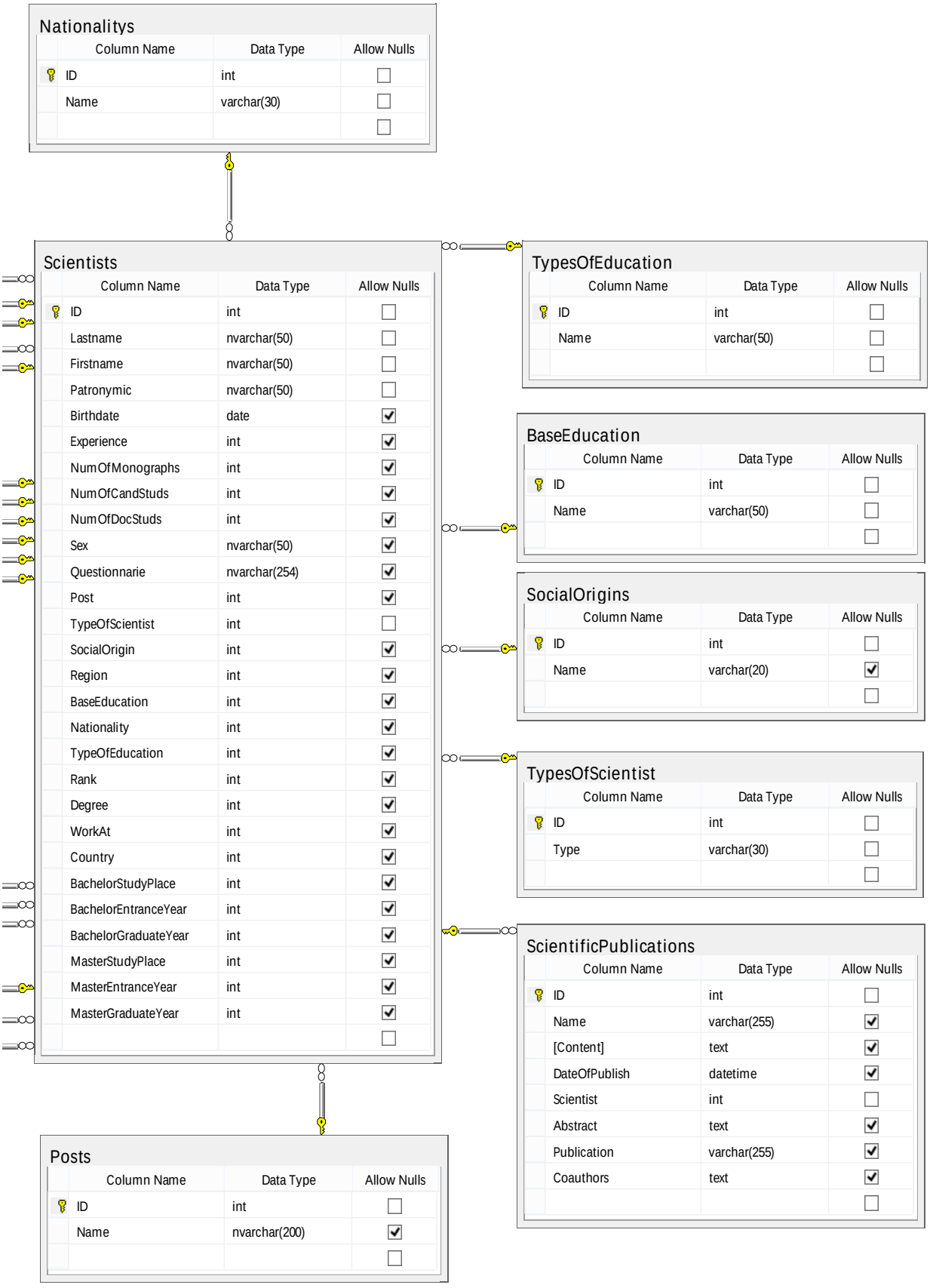

Figure2.3. Schema of relational database of subsystem «Olim».

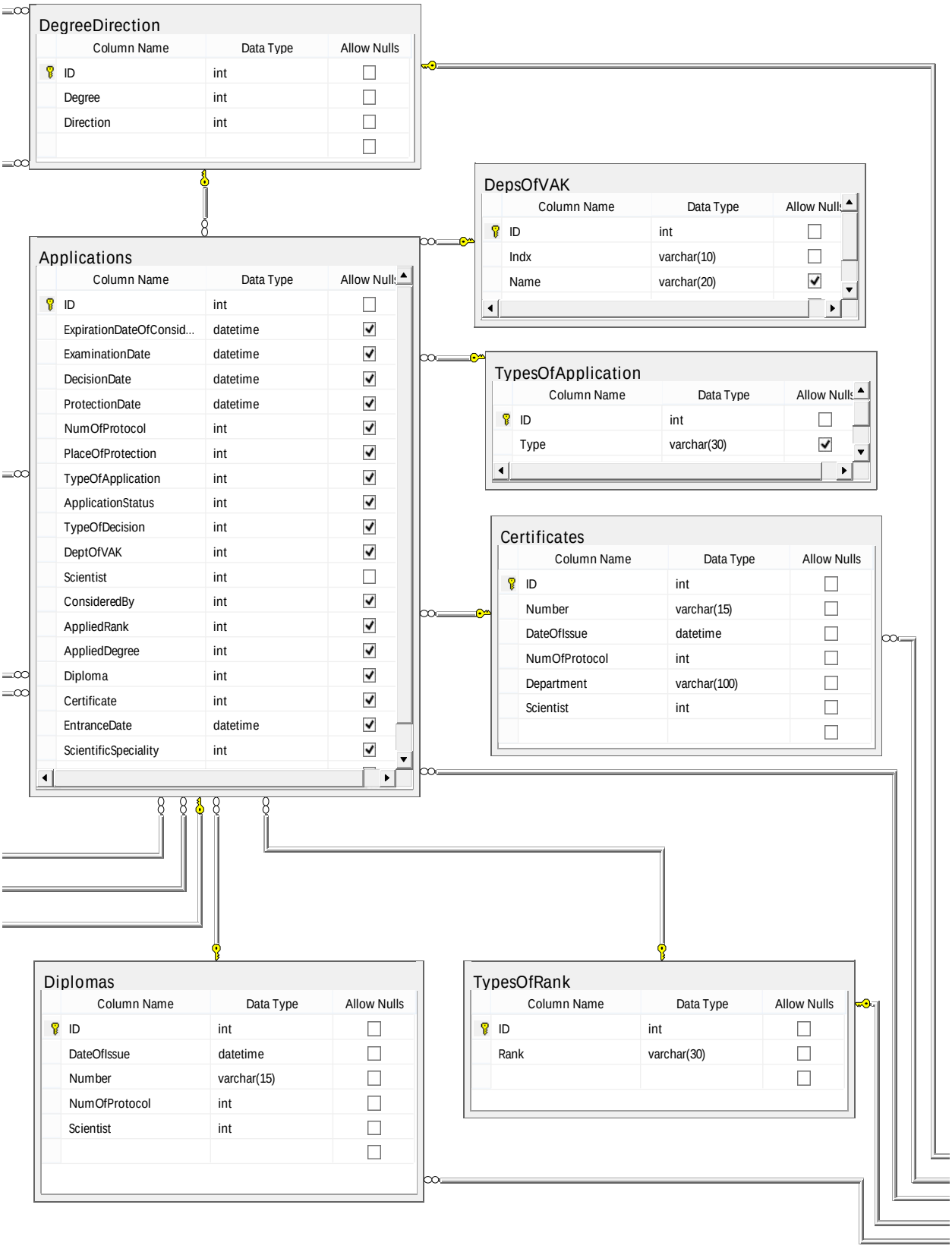

Figure 2.4. Schema of relational database of subsystem «Olim».

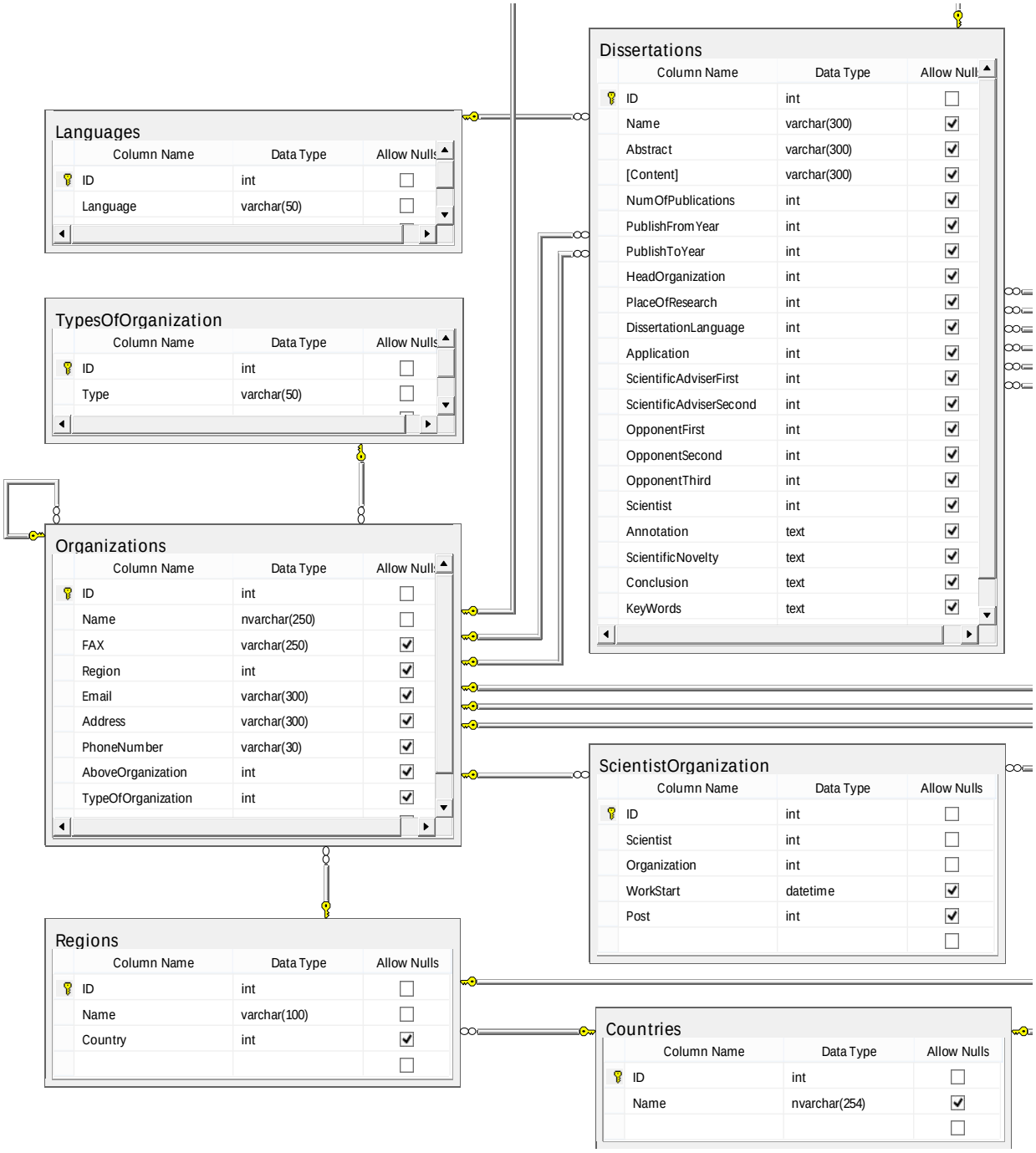

Figure 2.5. Schema of relational database of subsystem «Olim».

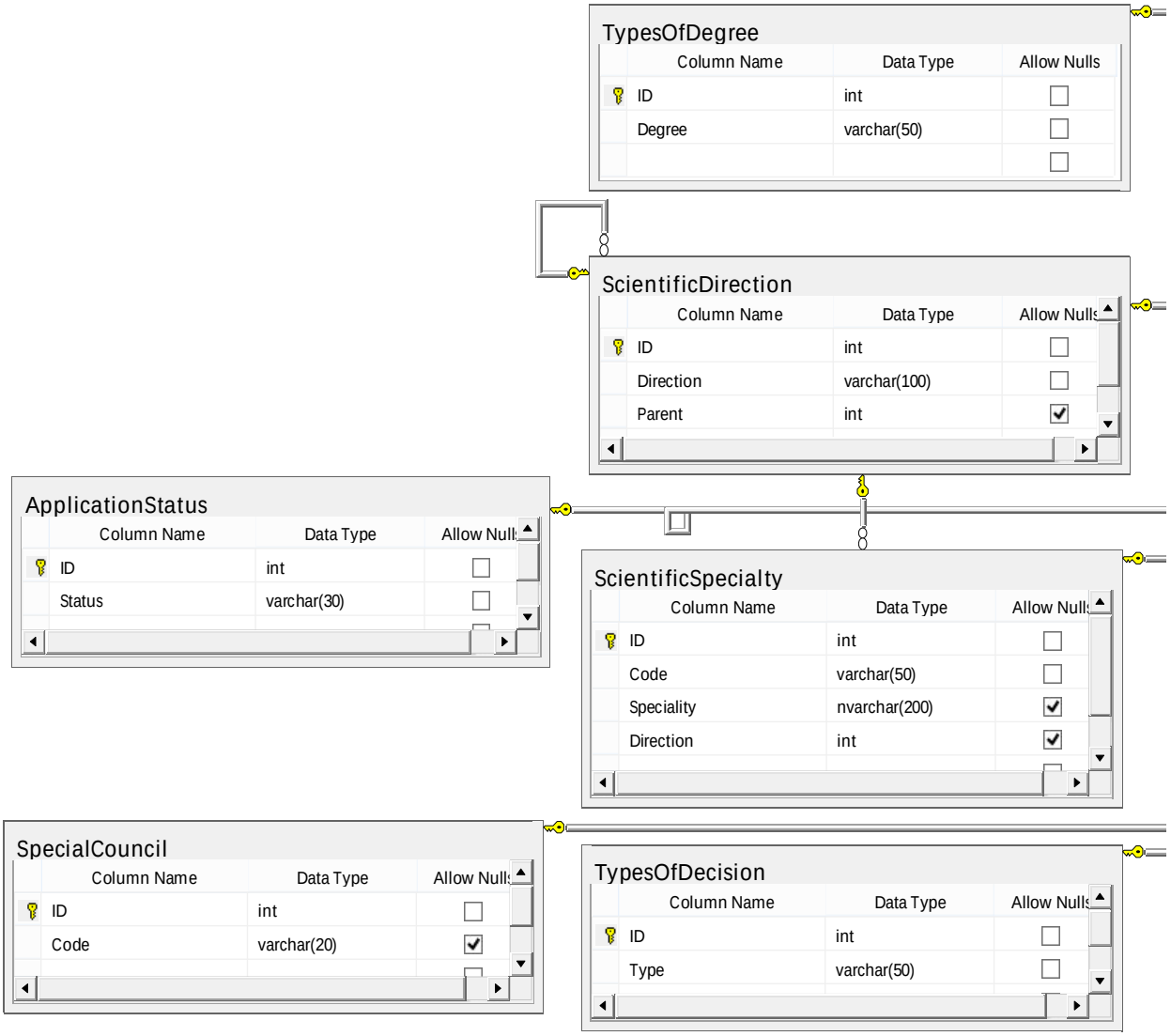

Figure 2.6. Schema of relational database of subsystem «Olim».

Further described structure of each table and views.

Applications – this table contains the application

Table 2.1

Field name (physical name)	**Data type**	**Restrictions on field**	**Field description**
ID	INTEGER	PK	ID identifies Applications
ExpirationDateOfCosiderationDedline	DateTime		ExpirationDateOfCosideratio nDedlinerelate toApplications
ExaminationDate	DateTime		ExaminationDaterelate to Applications
DecisionDate	DateTime		DecisionDaterelate to Applications
ProtectionDate	DateTime		ProtectionDate relate to Applications
NumOfProtocol	INTEGER		NumOfProtocol relate to Applications
PlaceOfProtection	INTEGER	FK1	PlaceOfProtection relate to Applications
TypeOfApplication	INTEGER	FK2	TypeOfApplication relate to Applications
ApplicationStatus	INTEGER	FK3	ApplicationStatus relate to Applications
TypeOfDecision	INTEGER	FK4	TypeOfDecision relate to Applications
DeptOfVAK	INTEGER	FK5	DeptOfVAK relate to Applications
Scientist	INTEGER	FK6	Scientist relate to Applications

ConsideredBy	INTEGER	FK7	ConsideredBy relate to Applications
AppliedRank	INTEGER	FK8	AppliedRank relate to Applications
AppliedDegree	INTEGER	FK9	AppliedDegree relate to Applications
Diploma	INTEGER	FK10	Diploma relate to Applications
Certificate	INTEGER	FK11	Certificate relate to Applications
EntranceDate	DateTime		EntranceDate relate to Applications
ScientificSpecialty	INTEGER	FK12	ScientificSpecialtyrelate to Applications

ApplicationStatus – this table contains all the possible status of the application

Table 2.2

Field name (physical name)	**Data type**	**Restrictions on field**	**Field description**
ID	INTEGER	PK	ID identifies ApplicationStatus
Status	VARCHAR(30)		Status relate to ApplicationStatus

BaseEducation– this table contains the basic education

Table 2.3

Field name (physical name)	**Data type**	**Restrictions on field**	**Field description**
ID	INTEGER	PK	ID identifies BaseEducation
Name	VARCHAR(50)		Name relate to BaseEducation

Certificates– this table contains certificates of scientists

Table 2.4

Field name (physical name)	**Data type**	**Restrictions on field**	**Field description**
ID	INTEGER	PK	ID identifies Certificates
Number	VARCHAR(15)		Number relate to Certificates
DateOfIssue	DATETIME		DateOfIssue relate to Certificates
NumOfProtocol	INTEGER		NumOfProtocol relate to Certificates
Department	VARCHAR(100)		Department relate to Certificates
Scientist	INTEGER	FK1	Scientist relate to Certificates

Countries – this table contains the names of countries

Table 2.5

Field name (physical name)	Data type	Restrictions on field	Field description
ID	INTEGER	PK	ID identifies Countries
Name	NVARCHAR(254)		Name relate to Countries

DegreeDirection– this table links the tableof academic degrees with the table of scientific direction and creates a table academic degrees in scientific direction.

Table 2.6

Field name (physical name)	Data type	Restrictions on field	Field description
ID	INTEGER	PK	ID identifies DegreeDirection
Degree	INTEGER	FK1	Degree relate to DegreeDirection
Direction	INTEGER	FK2	Direction relate to DegreeDirection

DepsOfVAK – this table contains the departments of HAC

Table 2.7

Field name (physical name)	Data type	Restrictions on field	Field description
ID	INTEGER	PK	ID identifies DepsOfVAK

Indx	VARCHAR(10)		Indx relate to DepsOfVAK
Name	VARCHAR(20)		Name relate to DepsOfVAK

Diplomas – this table contains diplomas of scientists

Table 2.8

Field name (physical name)	**Data type**	**Restrictions on field**	**Field description**
ID	INTEGER	PK	ID identifies Diplomas
DateOfIssue	DATETIME		DateOfIssue relate to Diplomas
Number	VARCHAR(15)		Number relate to Diplomas
NumOfProtocol	INTEGER		NumOfProtocol relate to Diplomas
Scientist	INTEGER	FK1	Scientist relate to Diplomas

Dissertations – this table contains dissertations of scientists

Table 2.9

Field name (physical name)	**Data type**	**Restrictions on field**	**Field description**
ID	INTEGER	PK	ID identifies Dissertations
Name	NVARCHAR(300)		Name relate to Dissertations
Abstract	NVARCHAR(300)		Abstract relate to

			Dissertations
Content	NVARCHAR(300)		Content relate to Dissertations
NumOfPublications	INTEGER		NumOfPublications relate to Dissertations
PublishFromYear	INTEGER		PublishFromYear relate to Dissertations
PublishToYear	INTEGER		PublishToYear relate to Dissertations
HeadOrganization	INTEGER	FK9	HeadOrganization relate to Dissertations
PlaceOfResearch	INTEGER	FK3	PlaceOfResearch relate to κ Dissertations
DissertationLanguage	INTEGER	FK1	DissertationLanguage relate to Dissertations
Application	INTEGER	FK2, U1	Application relate to Dissertations
ScientificAdviserFirst	INTEGER	FK4	ScientificAdviser relate to Dissertations
ScientificAdviserSecond	INTEGER	FK5	ScientificAdviserSecond relate to Dissertations

OpponentFirst	INTEGER	FK6	OpponentFirst relate to Dissertations
OpponentSecond	INTEGER	FK7	OpponentSecond relate to Dissertations
OpponentThird	INTEGER	FK8	OpponentThird relate to Dissertations
Scientist	INTEGER	FK10	Scientist relate to Dissertations
Annotation	TEXT		Annotation relate to Dissertations
ScintificNovelty	TEXT		ScintificNovelty relate to Dissertations
Conclusion	TEXT		Conclusion relate to Dissertations
KeyWords	TEXT		KeyWords relate to Dissertations

Languages – this table contains the languages for dissertations

Table 2.10

Field name (physical name)	**Data type**	**Restrictions on field**	**Field description**
ID	INTEGER	PK	ID identifies Languages
Language	VARCHAR(50)		Language relate to Languages

Nationalities– this table contains the nationality

Table 2.11

Field name (physical name)	Data type	Restrictions on field	Field description
ID	INTEGER	PK	ID identifies Nationalities
Name	VARCHAR(30)		Name relate to Nationalities

Organizations – this table contains the organization

Table 2.12

Field name (physical name)	Data type	Restrictions on field	Field description
ID	INTEGER	PK	ID identifies Organizations
Name	NVARCHAR(250)		Name relate to Organizations
Address	NVARCHAR(250)		Address relate to Organizations
PhoneNumber	NVARCHAR(30)		PhoneNumber relate to Organizations
AboveOrganization	INTEGER	FK1	AboveOrganization relate to Organizations
TypeOfOrganization	INTEGER	FK2	TypeOfOrganization relate to Organizations
Email	NVARCHAR(50)		Email relate to Organizations

Fax	NVARCHAR(50)		Fax relate to Organizations
Region	INTEGER	FK3	Region relate to Organizations

Posts– this table contains the posts

Table 2.13

Field name (physical name)	**Data type**	**Restrictions on field**	**Field description**
ID	INTEGER	PK	ID identifies Posts
Name	NVARCHAR(200)		Name relate to Posts

Regions – this table contains the regions

Table 2.14

Field name (physical name)	**Data type**	**Restrictions on field**	**Field description**
ID	INTEGER	PK	ID identifies Regions
Name	VARCHAR(100)		Name relate to Regions
Country	INTEGER	FK	Country relate to Regions

ScientificDirection – this table contains scientific directions

Table2.15

Field name (physical name)	**Data type**	**Restrictions on field**	**Field description**
ID	INTEGER	PK	IDidentifies ScientificDirection
Direction	NVARCHAR(100)		Direction частично идентифицируетScientificDirection
Parent	INTEGER	FK1	Parent частично идентифицируетScientificDirection

ScientificPublications– this table contains scientific publications of scientists

Table 2.16

Field name (physical name)	**Data type**	**Restrictions on field**	**Field description**
ID	INTEGER	PK	ID identifies ScientificPublications
Name	VARCHAR(255)		Name relate to ScientificPublications
Content	TEXT		Content relate to ScientificPublications
DateOfPublish	DATETIME		DateOfPublish relate to ScientificPublications
Scientist	INTEGER	FK1	Scientist relate to ScientificPublications
Abstract	TEXT		Abstract relate to ScientificPublications

Publication	VARCHAR(255)		Publication relate to ScientificPublications
Coauthors	TEXT		Coauthors relate to ScientificPublications

ScientificSpecialty– this table contains scientific specialty

Table 2.17

Field name (physical name)	**Data type**	**Restrictions on field**	**Field description**
ID	INTEGER	PK	ID identifies ScientificSpecialty
Code	NVARCHAR(50)		Code relate to ScientificSpecialty
Specialty	NVARCHAR(200)		Specialtyrelate to ScientificSpecialty
Direction	INTEGER	FK1	Directionrelate to ScientificSpecialty

ScientistOrganization – this table links the table of scientist and table of organizations and creates a table of history of scientist work

Table 2.18

Field name (physical name)	**Data type**	**Restrictions on field**	**Field description**
ID	INTEGER	PK	ID identifies ScientistOrganization
Scientist	INTEGER	FK1	Scientist relate to ScientistOrganization
Orgnization	INTEGER	FK2	Orgnization relate to

			ScientistOrganization
WorkStart	DATETIME		WorkStart relate to ScientistOrganization
Post	INTEGER	FK3	Post relate to ScientistOrganization

Scientist – this table contains of scientists

Table 2.19

Field name (physical name)	**Data type**	**Restrictions on field**	**Field description**
ID	INTEGER	PK	ID identifies Scientist
Lastname	NVARCHAR(50)		Lastname relate to Scientist
Firstname	NVARCHAR(50)		Firstname relate to Scientist
Patronymic	NVARCHAR(50)		Patronymic relate to Scientist
Birthday	DATE	FK1	Birthday relate to Scientist
Experience	INTEGER		Experience relate to Scientist
NumOfMonogrphs	INTEGER		NumOfMonogrphs relate to кScientist
NumOfCandStuds	INTEGER		NumOfCandStuds relate to Scientist
NumOfDocStuds	INTEGER		Lastname relate to Scientist

Sex	NVARCHAR(50)		Sex relate to κScientist
Questionnaire	NVARCHAR(254)		Questionnaire relate to Scientist
Post	INTEGER	FK2	Post relate to Scientist
TypeOfScientist	INTEGER	FK3	TypeOfScientist relate to Scientist
SocialOrigin	INTEGER	FK4	SocialOrigin relate to κScientist
Region	INTEGER	FK5	Regionrelate to Scientist
BaseEducation	INTEGER	FK6	BaseEducation relate to Scientist
Nationality	INTEGER	FK7	Nationalityrelate to Scientist
TypeOfEducation	INTEGER	FK8	TypeOfEducation relate to Scientist
Rank	INTEGER	FK9	Rank relate toScientist
Degree	INTEGER	FK10	Degreerelate to Scientist
WorkAt	INTEGER	FK11	WorkAt relate to Scientist
Country	INTEGER	FK12	Country relate to Scientist
BachelorStudyPlace	INTEGER	FK13	BachelorStudyPlace relate to Scientist

MasterStudyPlace	INTEGER	FK14	MasterStudyPlace relate to Scientist
BachelorEntranceYear	INTEGER		BachelorEntranceYear relate to Scientist
BachelorGraduateYear	INTEGER		BachelorGraduateYear relate to Scientist
MasterEntranceYear	INTEGER		MasterEntranceYear relate to Scientist
MasterGraduateYear	INTEGER		MasterGraduateYear relate to Scientist

SocialOrigins – this table contains the social origins

Table 2.20

Field name (physical name)	**Data type**	**Restrictions on field**	**Field description**
ID	INTEGER	PK	ID identifies SocialOrigins
Name	VARCHAR(20)		Name relate to SocialOrigins

SpecialCouncil – this table contains special councils

Table 2.21

Field name (physical name)	**Data type**	**Restrictions on field**	**Field description**
ID	INTEGER	PK	ID identifies SpecialCouncil

Code	VARCHAR(20)		Code relate to SpecialCouncil

TypesOfApplication – this table contains the types of applications

Table 2.22

Field name (physical name)	Data type	Restrictions on field	Field description
ID	INTEGER	PK	ID identifies TypesOfApplication
Type	VARCHAR(30)		Type относится к TypesOfApplication

TypesOfDecision–this table contains the types of decisions

Table 2.23

Field name (physical name)	Data type	Restrictions on field	Field description
ID	INTEGER	PK	ID identifies DecisionType
Type	VARCHAR(50)		Status относится к DecisionType

TypesOfDegree – this table contains the types of degrees

Table 2.24

Field name (physical name)	Data type	Restrictions on field	Field description
ID	INTEGER	PK	ID identifies TypesOfDegree
Degree	NVARCHAR(5		Degree

	0)		относитьсяTypesOfDegree

TypeOfEducation – this table contains the types of education

Table 2.25

Field name (physical name)	**Data type**	**Restrictions on field**	**Field description**
ID	INTEGER	PK	ID identifies TypeOfEducation
Name	VARCHAR(50)		Name относится к TypeOfEducation

TypesOfOrganization – this table contains the types of organizations

Table 2.26

Field name (physical name)	**Data type**	**Restrictions on field**	**Field description**
ID	INTEGER	PK	ID identifies TypesOfOrganization
Type	VARCHAR(50)		Type относится к TypesOfOrganization

TypesOfRank – this table contains the types of academic titles

Table 2.27

Field name (physical name)	**Data type**	**Restrictions on field**	**Field description**
ID	INTEGER	PK	ID identifies TypesOfRank
Rank	NVARCHAR(30)		Rankотносится к TypesOfRank

TypesOfScientist – this table contains the types of scientists

Table 2.28

Field name (physical name)	Data type	Restrictions on field	Field description
ID	INTEGER	PK	ID identifies TypesOfScientists
Type	VARCHAR(30)		Type идентифицирует TypesOfScientists

2.2. Designing

Information subsystem «Olim» is based on a three-tier application architecture, which includes the following levels:

- The presentation layer
- The level of business logic
- DataAccessLayer

Presentation layeris usually some variation of a «thin» client, which only calls the server displays the data and accepts input from the user.

Business logic layerdoes all the processing. This level is an intermediate between the database and the client application and can work as, for example, transaction processing monitor or object request broker.

The data access layer is typically a functional processing data stored in the database.[6]

Three-tier architecture of application of the subsystem «Olim» is displayed on figure2.7.

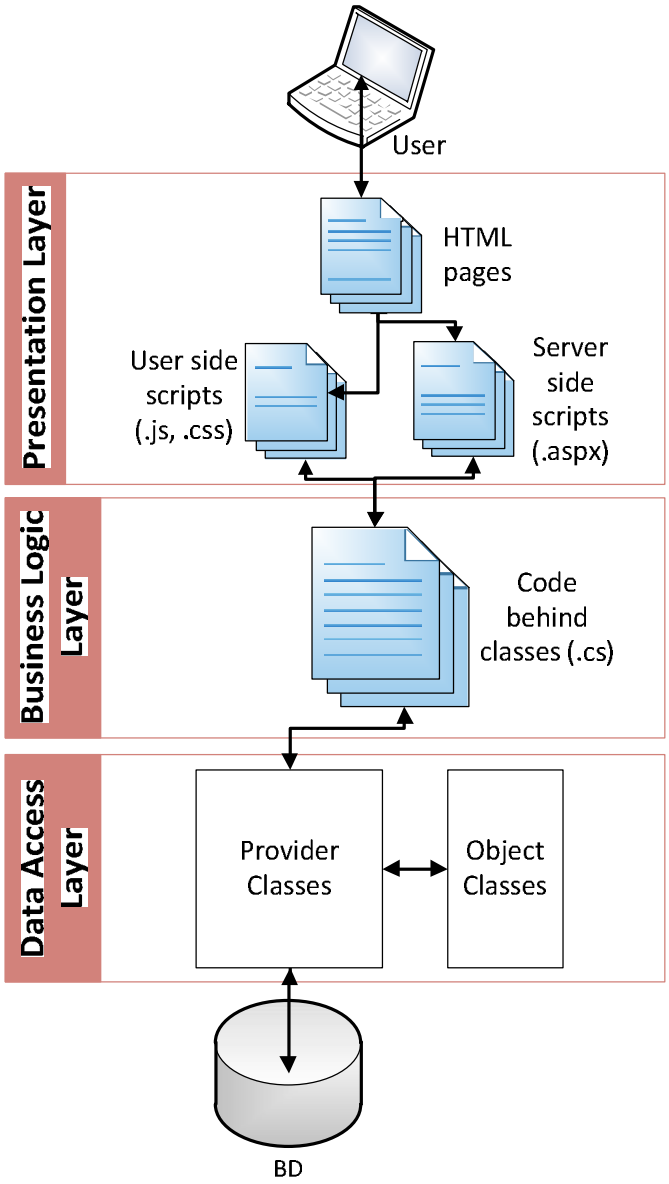

Figure 2.7 Architecture of application.

Information system «Olim» is written mainly using object-oriented programming, and data stored in relational databases. In this case the problem is to transform the object in such a form in which they may be stored in files or databases, and which can easily be removed in a subsequent, with preservation of objects and relations between them. Of this there is a need to display (projection) of objects in relational structures and vice versa. Thereby, the subsystem «Olim» classes have been developed that connect object classes and a relational database. Structural scheme of these classes is shown in figure 2.8

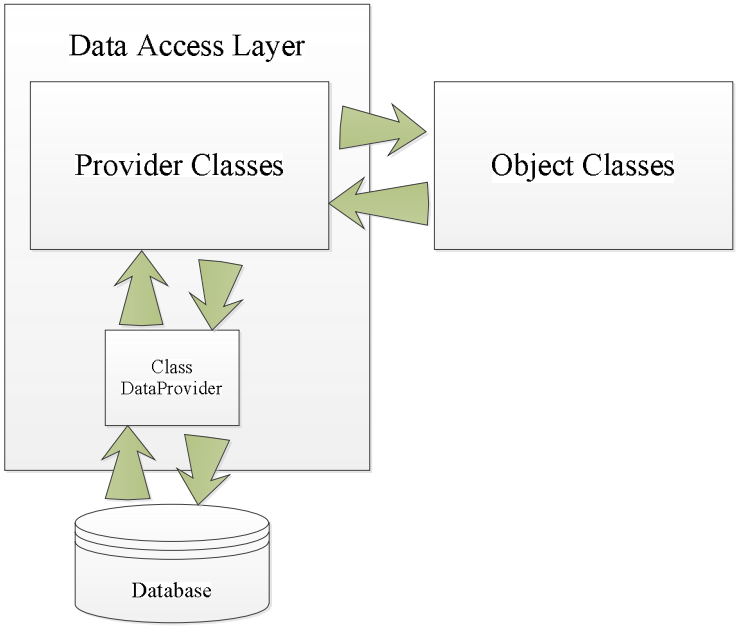

Figure 2.8. Structural scheme of the Data Access Layer.

For each object class there exists a specific class Provider which keeps all data processing methods and communicates with the DataProvider class which in turn creates a sql database queries. Figure 2.9, 2.10 and 2.11showing Provider classes and class DataProvider.

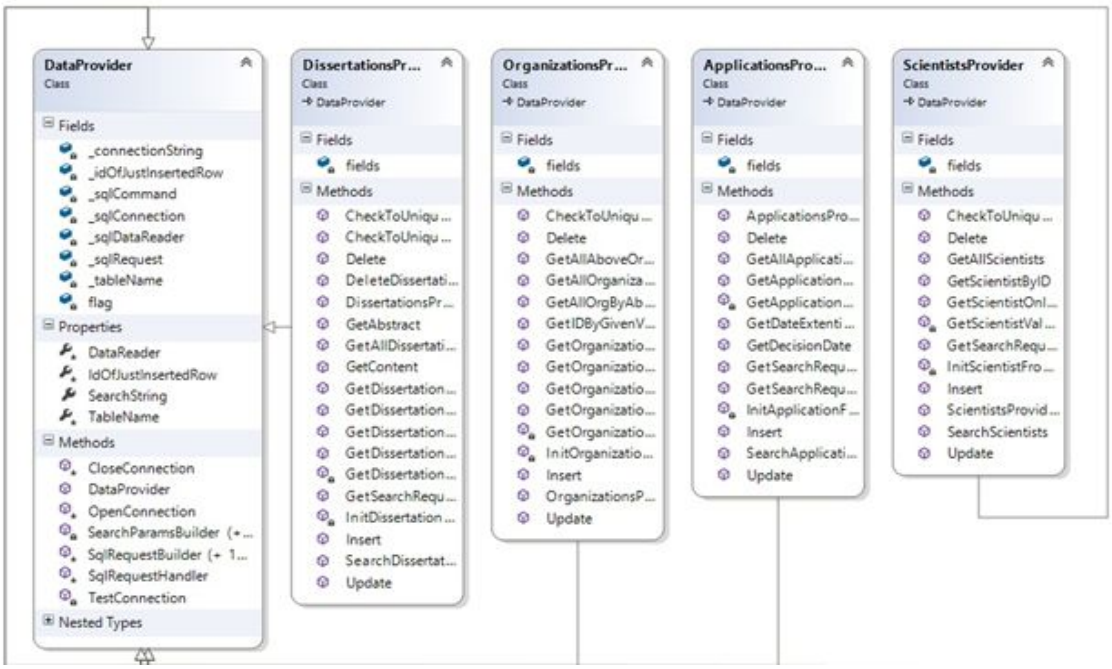

Figure 2.9. Provider Class and Class DataProvider.

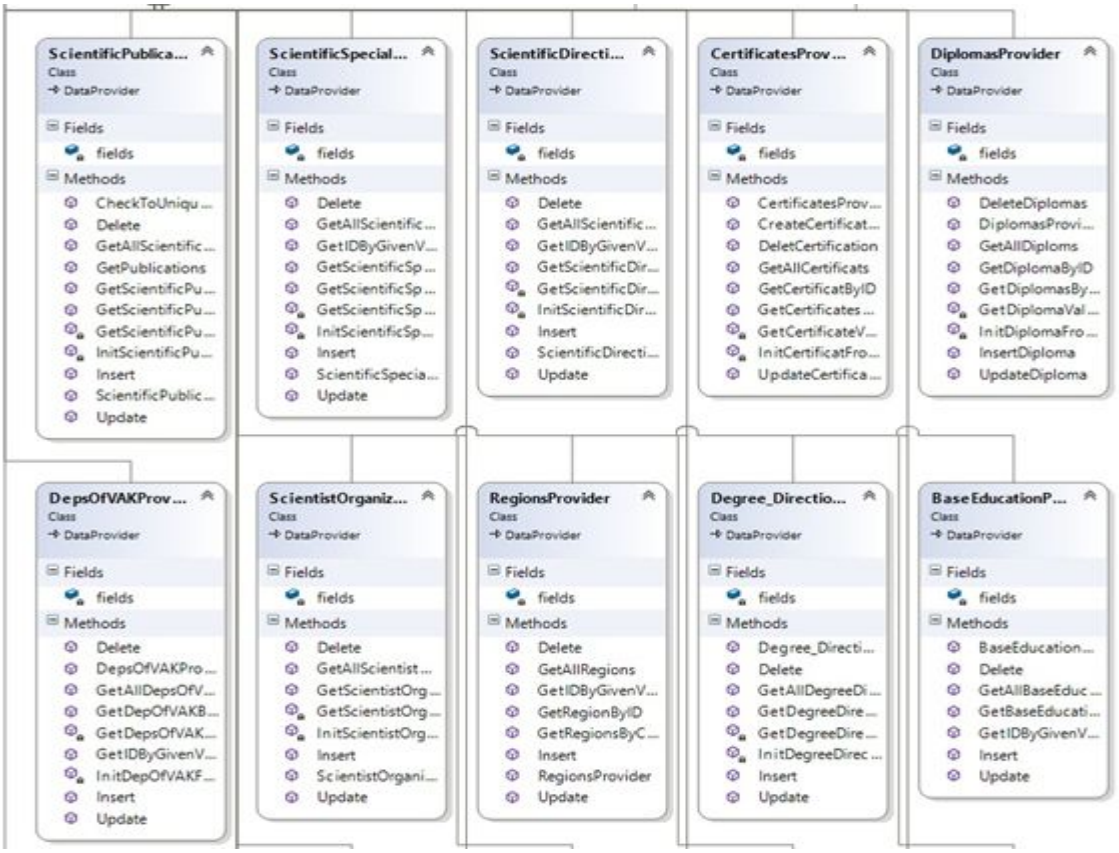

Figure 2.10. Provider Class and Class DataProvider.

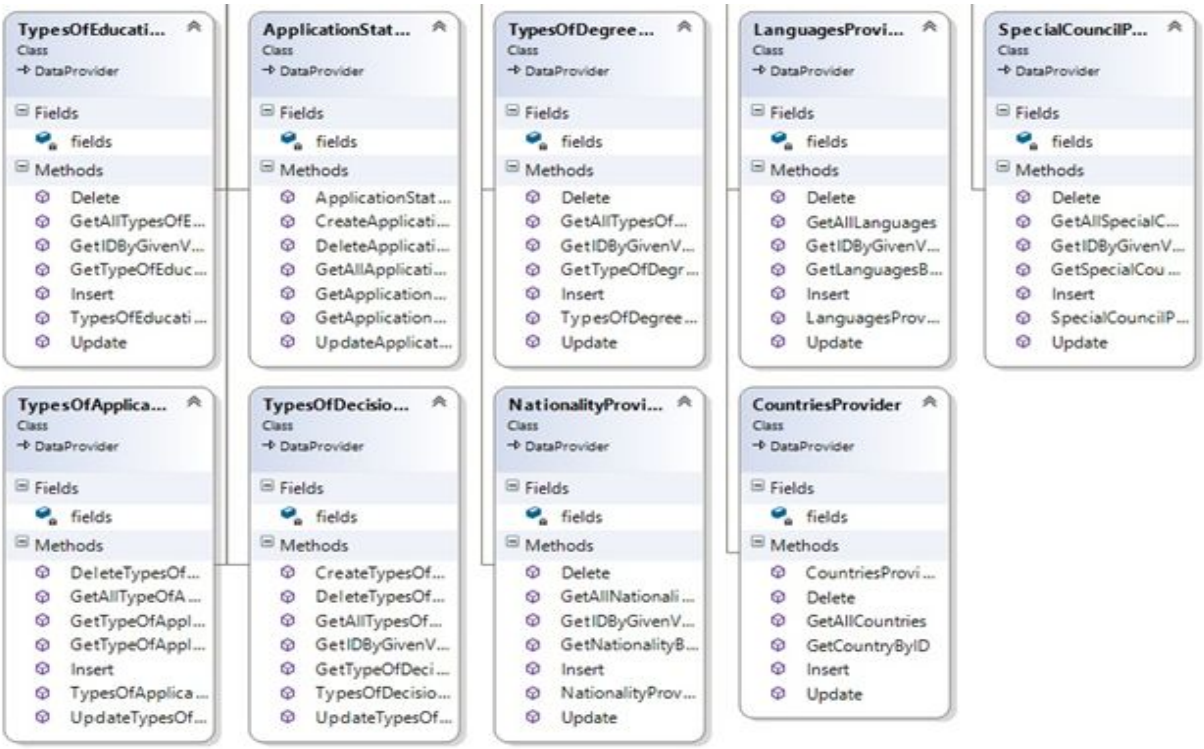

Figure 2.11. Provider Class and Class DataProvider.

Data access layer creates, edits, deletes, and retrieves data stored in databases by object-oriented classes. Figure 2.12 shows the form of adding an application for the degree.

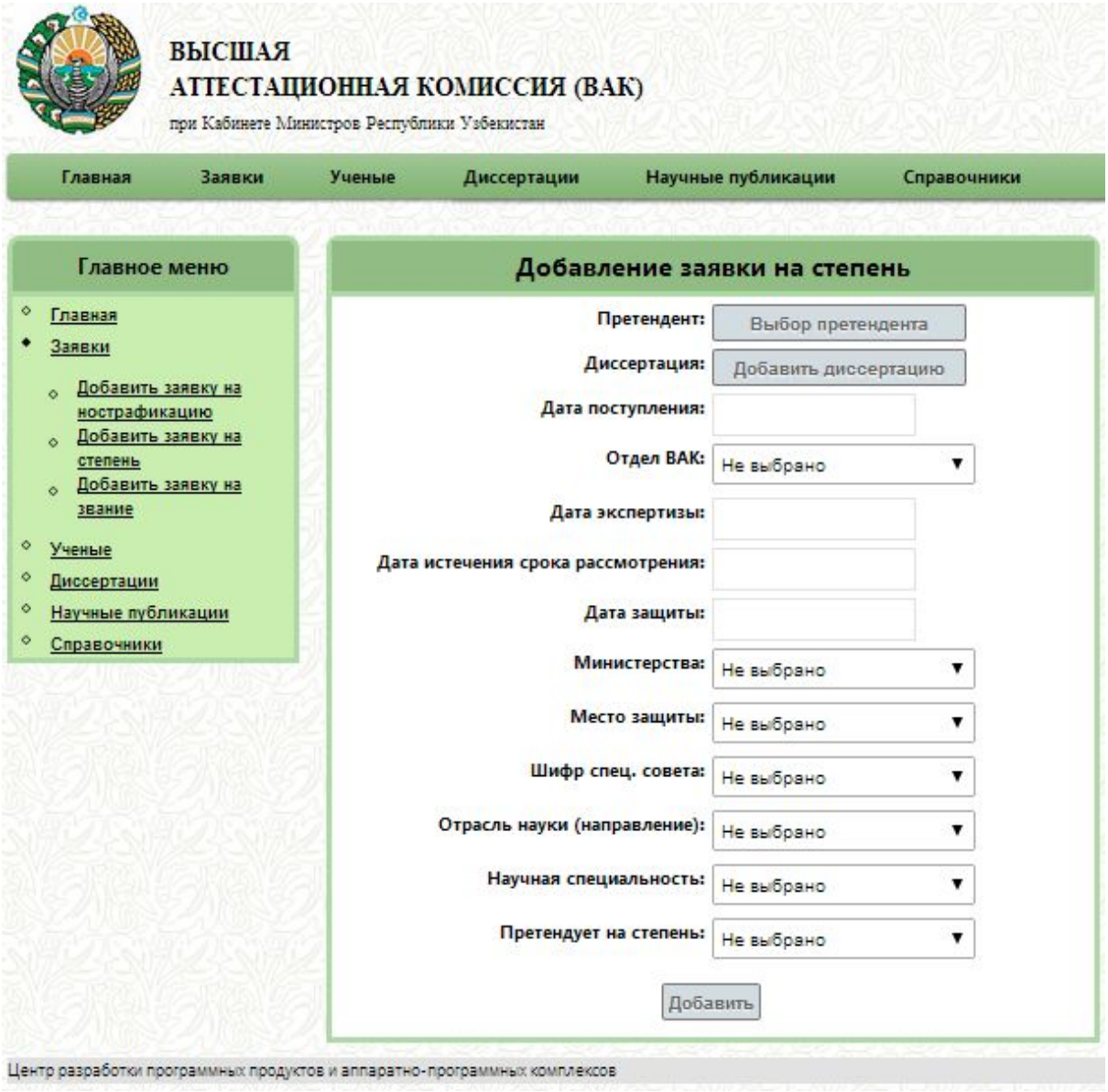

Figure 2.12 The form of adding an application for the degree.

After completing all the required fields, the Application object passed to ProviderClasses, to be exact in ApplicationsProvider which in turn builds a list of values for entry into the database and sends the list to the method SqlRequestHandler that handles the data and creates an entry in the database.

The list of values in ApplicationsProvider and method to connect to the DataProvider:

```
privateList<Object>GetApplicationValues(Application application)
        {
List<Object> values = newList<Object>();
if (application.ExpirationDateOfConsiderationDeadline == null)
values.Add(DBNull.Value);
else
values.Add(application.ExpirationDateOfConsiderationDeadline);
if (application.ExaminationDate == null)
values.Add(DBNull.Value);
else
values.Add(application.ExaminationDate);
if (application.DecisionDate == null)
values.Add(DBNull.Value);
else
values.Add(application.DecisionDate);
if (application.ProtectionDate == null)
values.Add(DBNull.Value);
else
values.Add(application.ProtectionDate);
if (application.NumOfProtocol == 0)
values.Add(DBNull.Value);
else
values.Add(application.NumOfProtocol);
if (((Organizations)application.PlaceOfProtection).ID == 0)
values.Add(DBNull.Value);
else
values.Add(((Organizations)application.PlaceOfProtection).ID);
if (((TypesOfApplication)application.TypeOfApplication).ID == 0)
values.Add(DBNull.Value);
else
values.Add(((TypesOfApplication)application.TypeOfApplication).ID);
if (((ApplicationStatus)application.ApplicationStatus).ID == 0)
values.Add(DBNull.Value);
else
values.Add(((ApplicationStatus)application.ApplicationStatus).ID);
if (((TypesOfDecision)application.TypeOfDecision).ID == 0)
values.Add(DBNull.Value);
else
values.Add(((TypesOfDecision)application.TypeOfDecision).ID);
if (((DepsOfVAK)application.DeptOfVAK).ID == 0)
```

```
values.Add(DBNull.Value);
else
values.Add(((DepsOfVAK)application.DeptOfVAK).ID);
if (((Scientist)application.Scientist).ID == 0)
values.Add(DBNull.Value);
else
values.Add(((Scientist)application.Scientist).ID);
if (((SpecialCouncil)application.ConsideredBy).ID == 0)
values.Add(DBNull.Value);
else
values.Add(((SpecialCouncil)application.ConsideredBy).ID);
if (((TypeOfRank)application.AppliedRank).ID == 0)
values.Add(DBNull.Value);
else
values.Add(((TypeOfRank)application.AppliedRank).ID);
if (((Degree_Direction)application.AppliedDegree).ID == 0)
values.Add(DBNull.Value);
else
values.Add(((Degree_Direction)application.AppliedDegree).ID);
if (((Diploma)application.Diploma).ID == 0)
values.Add(DBNull.Value);
else
values.Add(((Diploma)application.Diploma).ID);
if (((Certificates)application.Certificate).ID == 0)
values.Add(DBNull.Value);
else
values.Add(((Certificates)application.Certificate).ID);
if (application.EntranceDate == null)
values.Add(DBNull.Value);
else
values.Add(application.EntranceDate);
if (((ScientificSpecialty)application.ScientificSpeciality).ID == 0)
values.Add(DBNull.Value);
else
values.Add(((ScientificSpecialty)application.ScientificSpeciality).ID);
returnvalues;
        }
publicint Insert(Application application)
        {
List<object> values = newList<object>();
values = GetApplicationValues(application);
base.SqlRequestHandler(SqlRequestType.Create, fields, values);
base.CloseConnection();
```

```
returnbase.IdOfJustInsertedRow;
      }
```

Method SqlRequestHandler:

```
protectedvoidSqlRequestHandler(SqlRequestTypesqlRequestType,
List<string> fields = null,
List<object> values = null,
stringnameOrederingBy = null,
List<FilterValue>filterVals = null)
       {
if (!flag)
this._sqlRequest     =     this.SqlRequestBuilder(sqlRequestType,     fields,     values,
nameOrederingBy);
switch (sqlRequestType)
           {
caseSqlRequestType.Max:
this._sqlCommand.CommandText = _sqlRequest;
this._sqlCommand.Parameters.Clear();
break;
caseSqlRequestType.Create:
this._sqlCommand.CommandText = _sqlRequest;
this._sqlCommand.Parameters.Clear();
for (int counter = 0; counter <values.Count; counter++)
                   {
if (values[counter] != null)
this._sqlCommand.Parameters.Add(newSqlParameter(string.Format("@{0}",    fields[counter]),
values[counter]));
else
this._sqlCommand.Parameters.Add(newSqlParameter(string.Format("@{0}",    fields[counter]),
DBNull.Value));
                   }
break;
caseSqlRequestType.Update:
this._sqlCommand.CommandText = _sqlRequest;
this._sqlCommand.Parameters.Clear();
for (int counter = 0; counter <fields.Count; counter++)
                   {
if (values[counter] != null)
this._sqlCommand.Parameters.Add(newSqlParameter(string.Format("@{0}",    fields[counter]),
values[counter]));
else
this._sqlCommand.Parameters.Add(newSqlParameter(string.Format("@{0}",    fields[counter]),
DBNull.Value));
```

```
                    }
break;
caseSqlRequestType.Delete:
this._sqlCommand.CommandText = _sqlRequest;
this._sqlCommand.Parameters.Clear();
this._sqlCommand.Parameters.Add(newSqlParameter(string.Format("@{0}",  fields[fields.Count
- 1]), values[values.Count - 1]));
break;
caseSqlRequestType.Read:
if (values != null)
                    {
this._sqlCommand.CommandText = _sqlRequest;
this._sqlCommand.Parameters.Clear();
this._sqlCommand.Parameters.Add(newSqlParameter(string.Format("@{0}",  fields[fields.Count
- 1]), values[values.Count - 1]));
                    }
else
this._sqlCommand.CommandText = this._sqlRequest;
break;
caseSqlRequestType.Search:
this._sqlCommand.CommandText = this._sqlRequest;
this._sqlCommand.Parameters.Clear();
break;
            }
try
            {
OpenConnection();
switch (sqlRequestType)
                {
caseSqlRequestType.Read:
caseSqlRequestType.Max:
caseSqlRequestType.Search:
this._sqlDataReader = _sqlCommand.ExecuteReader();
break;
caseSqlRequestType.Create:
this._idOfJustInsertedRow = (int)this._sqlCommand.ExecuteScalar();
break;
caseSqlRequestType.Update:
caseSqlRequestType.Delete:
this._sqlCommand.ExecuteNonQuery();
break;
                }
            }
```

```
catch (SqlException e)
            {
thrownewApplicationException("Error occured while handling sql request"
+ "\nExceptionmessage: "+ e.Message);
            }
finally
            {
            }
        }
```

Second chapter summary

Were modeled:

1. The entity-relationship scheme
2. Structural Database Schema

Describes the architecture of the application, and was designed data access layer. Reviewed the work of the Data Access Layer.

In the end, there was realized a data access layer subsystem «Olim» based on the Microsoft. NET.

CHAPTER 3. LIFE SAFETY

3.1. The rational organization of a workplace

Scientific and technological progress has made major changes in the conditions of production activity of knowledge workers. Their work has become more intense, stressful, requiring a significant investment of mental, emotional and physical energy. It required comprehensive solution of problems of ergonomics, hygiene and labor organization, regulatory regimes of work and rest.

Nowadays computer technology is widely used in all areas of human activity. When working with a computer person is exposed to a number of dangerous and harmful factors: electromagnetic fields, infrared and ionizing radiation, noise and vibration, static electricity, etc.

Computer work is characterized by significant mental stress and neuro-emotional stress operators, high intensity visual work and big enough load on the muscles of the hands when using the keyboard. Great importance is the rational design and layout of the workplace, it's important to maintain optimal working posture of the human operator.

During the working with the computer, you must observe proper work and rest. Otherwise, the staff observed considerable tension with the advent of the visual apparatus dissatisfaction complaints, headaches, irritability, sleep disturbance, fatigue and pain in the eyes, in the back, the neck and hands.

Colouring rooms and furniture should help create favorable conditions for visual perception, good mood.

Light sources such as lamps and windows which give a reflection from the surface of the screen, significantly affect the accuracy of signs and entail interference physiological nature, which could result in significant stress, especially for prolonged use. Reflection, including reflections from the secondary light sources should be minimized. To protect against excessive brightness of windows can be applied curtains and screens.

Depending on the orientation of windows recommend the following painting the walls and floor:

windows oriented to the south: - wall teal or light blue; floor - green;

windows oriented to the north: - wall light orange or orange-yellow;

floor - a reddish-orange;

windows oriented to the east: - wall yellow-green; floor green or reddish-orange;

windows oriented to the west: - wall yellow-green or blue-green;

floor green or reddish-orange.

In rooms where there is a computer, must be provide the following values of the reflection coefficient: Ceiling: 60 ... 70%, for the walls: 40 ... 50%, floor: about 30%. For other substrates and working furniture: 30 ... 40%.

Correctly designed and executed industrial lighting improves the conditions of visual work, reduces fatigue, increases productivity, beneficial effects on the production environment, providing a positive psychological impact on the worker, increases safety and reduces injuries.

Lack of light causes eye strain, reduces attention, leads to premature onset of fatigue. Overly bright lighting causes glare, irritation and pain in the eyes. Wrong direction of light in the workplace can create harsh shadows, reflections, disorient worker. All of these factors can lead to accidents or occupational diseases, so much important the right calculation of illumination.

There are three types of lighting - natural, artificial and combined (natural and artificial together).

Natural lighting - lighting of premises with daylight coming through the skylights in the outer space enclosing structures. Natural lighting is characterized in that varies widely depending on time of day, time of year, the character area and other factors.Artificial lighting used for working in the dark and in the afternoon, when you can not provide the normalized values of the coefficient of natural light (cloudy, short daylight hours). Lighting that insufficient standards for natural light supplemented by artificial called combined lighting.

Artificial lighting is divided into operating, emergency, evacuation, security. Work lighting, in turn, can be shared or combined. General - lighting that lights are placed in the upper zone of the room evenly or in relation to the location of the equipment. Combined - lighting that adds to the overall local coverage.

When carrying out the visual category of high accuracy (the smallest size of object discrimination 0.3 ... 0.5 mm), the coefficient of natural lighting of (KEO) should not be below 1.5%, while the average accuracy of visual work (minimum size of the object 0 distinguishing 5 ... 1,0 mm) KEO must not be lower than 1.0%. As artificial light sources commonly used luminescent lamp type LB or DRL, which pairs together in light, which must be placed above the work surfaces evenly.

Requirements for lighting in rooms where computers are set as follows: when the works of visual precision general illumination shall be 300lk and combined - 750lk; similar requirements when working medium accuracy - 200 and 300lk respectively.

In addition, all field of view should be fairly evenly lighted - this basic hygiene requirements. That is to say, the degree of room lighting and brightness of your computer screen should be approximately equal, because bright light in the area of peripheral vision greatly increases eye strain and, consequently, leads to their fatigue.

Parameters of the microclimate can vary widely, while a necessary condition of human life is to maintain a constant body temperature through thermoregulation, i.e. the body's ability to regulate the heat transfer to the environment.

Principle of regulation of microclimate - to create optimal conditions for the heat of the human body with the environment.Computing equipment is a source heat release material, which may lead to an increase in the temperature and decrease in relative humidity. In rooms must be followed certain parameters of the microclimate. In sanitary norms set parameter values microclimate, creating a comfortable environment.These standards are set, depending on the time of year,

the nature of the labor process and the nature of the production premises. They are listed in table 3.1

The volume of space, which posted workers computing centers, must not be less than 19.5 m^3/person with the maximum number of concurrent shift. The supply rate of fresh air in the room where the computers are given in table 3.2

Table 3.1Microclimate parameters for the room where the computers

Period of year	Parameters of microclimate	Value
Cold	Indoor temperature	22…24 °C
	Relative humidity	40…60 %
	The velocity of air	до 0,1 m/s
Warm	Indoor temperature	23…25 °C
	Relative humidity	40…60 %
	The velocity of air	0,1…0,2 m/s

Table 3.2 The supply rate of fresh air in the room where the computers are located

Characteristics of room	Volumetric flow rate supplied into the room of fresh air, m^3 / per person per hour
Volumuntill 20 m^3per person	Not less than 30
20…40 m^3per person	Not less than20
Morethan 40 m^3per person	Natural ventilation

To provide comfortable conditions are used as organizational methods (rational organization of the work, depending on the time of day and year, the alternation of work and rest) and hardware (ventilation, air conditioning, heating system).

Noise worsens working conditions providing a harmful effect on the human body. Working in long-term noise exposure experienced irritability, headaches, dizziness, memory loss, fatigue, loss of appetite, pain in the ears, etc.Such violations in a number of organs and systems of the human body can cause adverse changes in the emotional state of the person until the stress.Such violations in a number of organs and systems of the human body can cause adverse changes in the emotional state of the person until the stress.Under the influence of noise is reduced concentration, disturbed physiological functions appears fatigue due to increased energy costs and mental stress, deteriorating voice switching.All this reduces the capacity of man and its performance, quality and safety.

Table. 3.3 shown limiting sound levels depending on the category of severity and intensity of labor, which are safe with respect to preserving the health and performance.

Table 3.3Limit values, db, in the workplace.

The category of tensions of labor	The category of severity of labor			
	I. Easy	II. Average	III. Hard	IV. Very hard
I. Little stressful	80	80	75	75
II. Moderately stressful	70	70	65	65
III. Stressful	60	60	-	-
IV.Very Stressful	50	50	-	-

The noise level in the workplace mathematicians programmers and operators of video should not exceed 50dBA, and in the halls of information processing on computers - 65dBA.To reduce noise in the walls and ceiling room where the computers may be lined with sound-absorbing materials. Vibration level in areas of computing centers can be reduced by installing special equipment on vibration isolators.

Most scientists believe that both short-term and long-term impact of all types of radiation from the monitor screen is not dangerous for health personnel serving computers. However, comprehensive data about the dangers of exposure to

radiation from the monitor working with computers and there is no research in this area continues.

Valid values of non-ionizing electromagnetic radiation from computer monitor are shown in table3.4

The maximum level of X-ray radiation in the workplace computer operator does not usually exceed 10mkber / h, and the intensity of ultraviolet and infrared radiation from the monitor screen is in the range 10 ... 100mVt/m^2.

Table 3.4Valid values of non-ionizing electromagnetic radiation**.**

The parameter name	Valid values
The tension of the electric component of the electromagnetic field at a distance 50sm from the surface of the monitor	10V / m
The magnetic component of the electromagnetic field at a distance 50sm from the surface of the monitor	0.3 A / m
Electrostatic intensity shall not exceed: for adult users for children of preschool institutions and students secondary and higher educational institutions	20 kV / m 15kV / m

To reduce the impact of these types of radiation are recommended monitors with reduced radiation, install protective screens and observe regulated modes of work and rest.

Designing workplaces equipped with video terminals, is among the most important problems of ergonomic design in computer science.

Workplace and relative position of all its elements must match the anthropometric and physical and psychological requirements. Another important

factor is the nature of work. In particular, the organization of the workplace programmer subject to the following general conditions: optimal placement of equipment, which is part of the workplace and ample workspace, allowing to perform all the necessary movements and movement.

The main elements of the workplace are the programmer table and chair. The main working position is a sitting position.

Engine field - space workplace, which can be carried out human motor actions.

Maximum Reach hands - is part of the workplace motor field bounded by arcs described by the maximum outstretched hands as they move in the shoulder joint.

The optimum region - part of workplace motor field bounded by arcs described by the forearm motion in the elbow joints with a support at the elbow and shoulder with a relatively fixed.

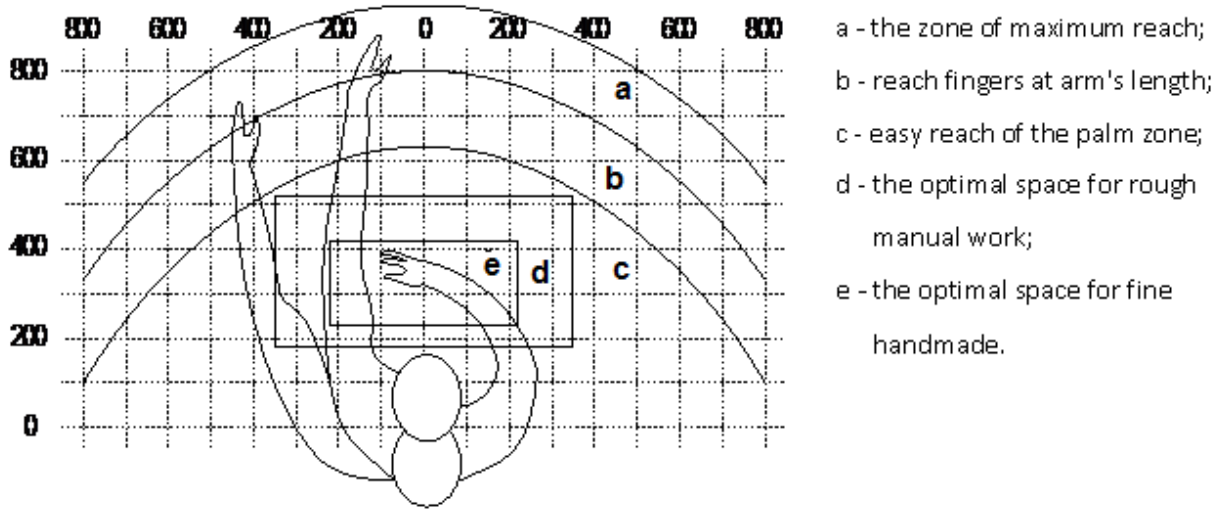

Figure 3.1 The reach of the hands in a horizontal plane

Optimal placement of objects of labor and documentation in the areas of reach:

DISPLAY located in a zone (center);

SYSTEM CONTROL located in the specified niche table;

KEYBOARD - zoned/e**;**

«MOUSE» - in zonecon the right;

SCANNER – in zone a/b (left);

PRINTER – in zone a (right);

DOCUMENTATION: necessary at work - in easy reach of the palm zone – and in drawers and table - literature, unused.

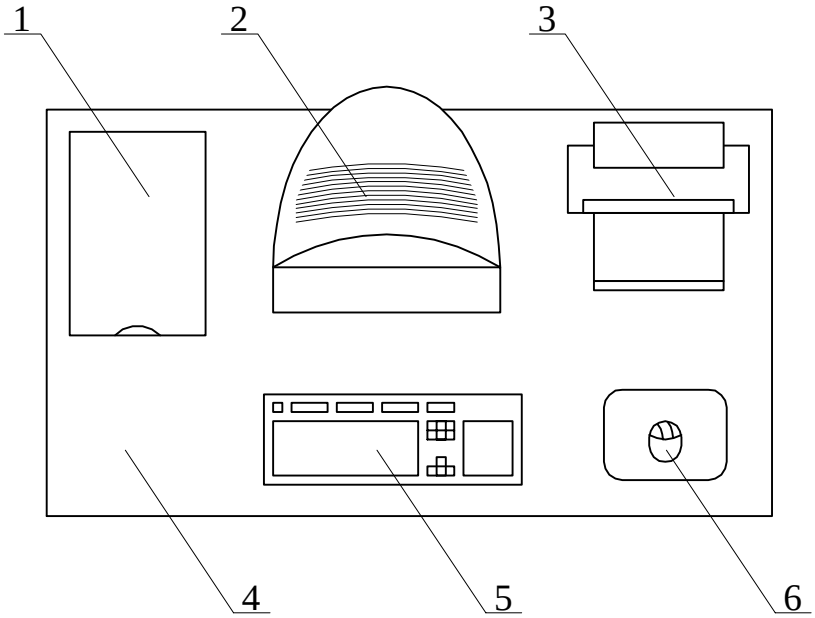

Figure 3.2 Accommodation basic PC components and peripherals.

The figure shows an example of placing the main and peripheral components of the PC desktop programmer.

1 – scanner, 2 – display, 3 – printer, 4 – desktop surface,

5 – keyboard, 6 – mouse

For comfortable work table must satisfy the following conditions:

- table height should be chosen taking into account the opportunity to sit freely in a comfortable position, if necessary, based on the armrests;
- the bottom part of the table must be designed so that a programmer can sit comfortably, was forced to urge the legs;
- surface should have properties to prevent glare in the field of view of the programmer;
- table design should include the presence of drawers;

- height of the work surface is recommended within the 680-760mm. The height of the surface on which the keyboard is installed, should be about 650mm.

Great importance is attached the characteristics of the desk chair.Recommended seat height above floor level is within the 420-550mm. Surface of the seat is soft, rounded front edge and back angle - adjustable.

Is necessary to provide an opportunity for the design of various organize documents: the side of the video terminal, between the monitor and keyboard, etc.In addition, in cases where the video display is of poor quality images, such as visible flashing, the distance from the eye to make the screen larger (about 700 mm) than the distance from the eye to the document (300-450mm).Generally with high image quality on a visual distance from the user's eyes to the screen, keyboard and document may be equal.

Position of the screen is determined by:

- Reading distance (0.6 ... 0.7 m);
- Angle reading, viewing direction 20° below the horizontal center of the screen, and the screen is perpendicular to this direction.
- Should also be possible to control the screen:
- 3 sm in height;
- The slope of -10° to +20°from the vertical;
- Left and right directions.

Great importance is also attached to the correct working posture of the user. When uncomfortable working posture may appear pain in muscles, joints and tendons. Requirements for working posture video terminal user as follows:

- The head should not be tilted more than 20°,
- Shoulders should be relaxed,
- Elbows - at an angle of 80° ... 100°,
- Forearms and hands - in a horizontal position.

The reason caused improper posture users the following factors: no good stand for documents, the keyboard is too high, and documents - low, nowhere to put the arm and hand, not enough legroom.

In order to overcome these disadvantages general recommendations: best mobile keyboard; should provide special tools for adjusting the height of the table, the keyboard and screen, as well as the palm rest.

Essential for productive and quality work on the computer characters have dimensions, the density of their placement, contrast and luminance ratio of characters and background screen. If the distance from the eye to the operator display screen is 60 ... 80 cm, the height of the sign shall be not less than 3 mm, the optimal ratio of the width and height of the sign is 3:4, and the distance between the marks - 15 ... 20% of their height. The ratio of the background brightness of the screen and characters - from 1:2 to 1:15.

During using a computer physicians advise to mount the monitor at a distance of 50-60 cm from the eye. Experts also believe that the upper part of the video should be at eye level or slightly below. When a person looks straight ahead, his eyes opened wider than when he looks down.In this viewing area increases significantly, causing dehydration of the eye. In addition, if the screen is set high, and his eyes wide open, disturbed function of blinking. This means that your eyes do not close completely, washed teardrops do not get enough moisture, which leads to their fatigue.

Creating favorable conditions of work and the right aesthetic appearance manufacturing jobs is of great importance as to facilitate the work and to increase its attractiveness, positive effect on productivity.[7]

3.2. Emergency

Emergency (ES) - the situation in a certain area, established as a result of an accident, natural hazards, disasters, natural or other disasters that may cause or

have caused human casualties, damage to human health or the environment, considerable material losses and disruption living conditions of the people.

Under the emergency source understand of natural hazards, accident or a dangerous man-made accident, common infectious disease in humans, farm animals and plants, as well as the use of modern means of destruction, resulting occurred or may occur in an emergency.

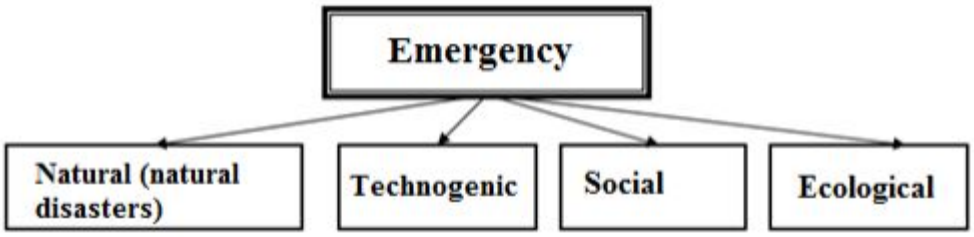

Figure 3.3The causal classification of emergency

Emergencies constantly accompany human rights and threaten his life, bring pain, suffering, injury, death, damage and destroy wealth, cause great damage to the environment, society and civilization.

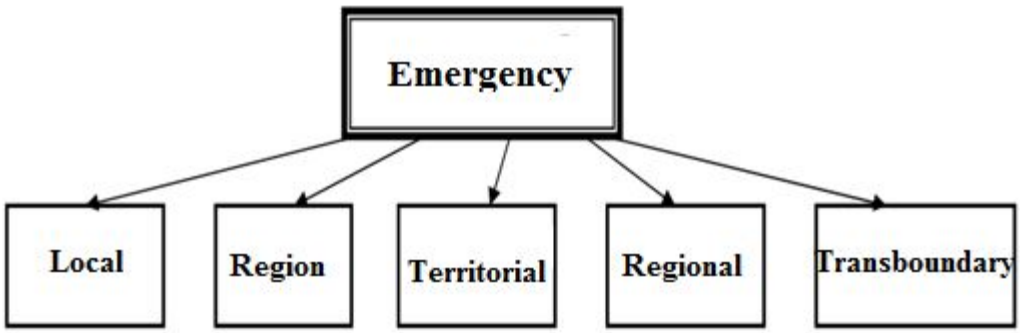

Figure 3.4Territorial emergency classification.

To local include emergency, which affected 10 people, or violated the conditions of life of not more than 100 persons, or property damage is not more than one thousand times the minimum wage on the day of an emergency, emergency zone does not extend beyond the territory production facility or social purpose.

To region include emergency, which affected 10 to 50 people, or violated the conditions of life of 100 to 300 people, or material damage is from 1 thousand to 5000 the minimum wage on the day of an emergency, the emergency area not beyond the village, city or area.

To regional refers emergency, which affected more than 500 people, or violated the conditions of life of 500 and more than 1,000 people.

To transboundary refers emergency affecting factors which are beyond the country.

Bring a lot of misery to mankind natural emergencies (natural disasters). These include emergency associated with the manifestation of the elemental forces of nature such as earthquakes, floods, tsunamis, hurricanes, wildfires, landslides, avalanches, heavy rains, storms, droughts, blizzards, cold, heat, volcanoes, hail, strong snow, thunderstorms, fog, ice, frost. Natural fires

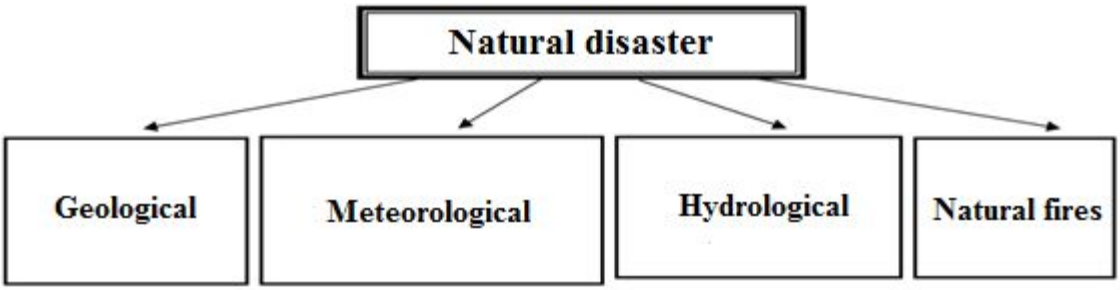

Figure 3.5 Natural disaster classification.

Natural disasters - natural phenomena is significant scale, which result in a threat to life or health, may occur or destruction of property will be harmed the environment.

History of the development of a terrestrial civilization is inseparably linked with the creation of conditions for the occurrence of disaster manmade. ES, which are a consequence of industrial and human activities are called man-made.

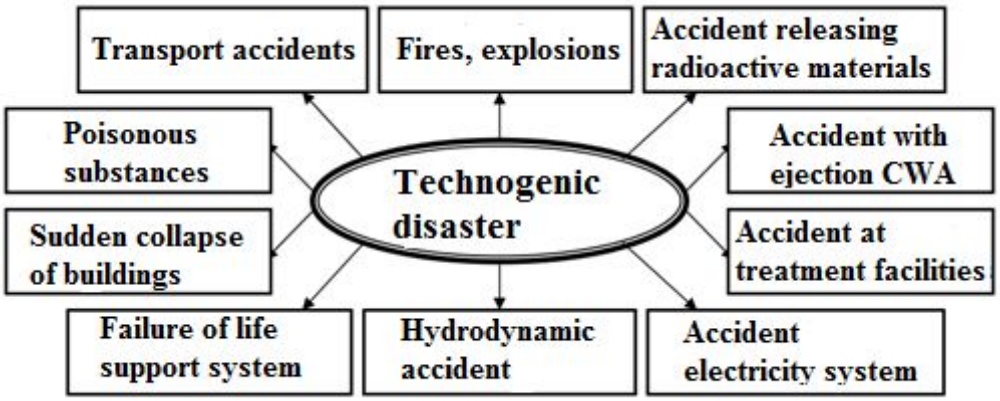

Figure 3.6Classification of technogenic disaster.

Technogenic disaster lead to injury and loss of life, destruction of property, significant economic and environmental losses.

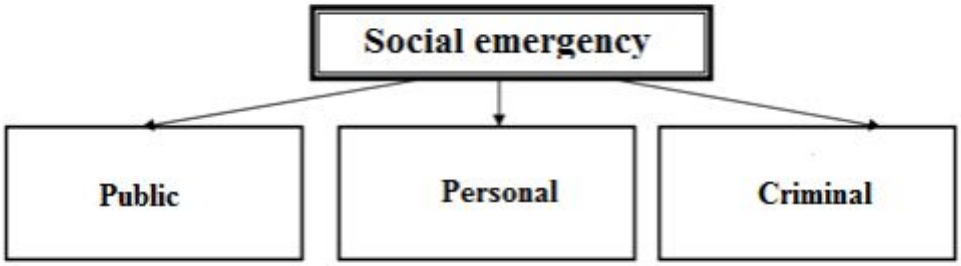

Figure 3.7The classification of social disaster.

An important place in modern society occupy social disaster: war, terrorism, hostage-taking, theft, fraud, gambling, rape, harmful and dangerous habits, mass gatherings, informal associations. Their total number, variety, scale effects are constantly growing. A specific feature of social disaster is that they arise in the relationship between people and depend on the person. Emergency social consciously planned, prepared , carried out by people . These people are trying to help solve social disaster , their political, national, religious , criminal , financial , personal tasks . For this purpose, various brutal methods : threats, blackmail , violence, fraud , hostage-taking , theft, murder. Organizers , customers, artists do not stop what social disaster victims often are civilians .

For a comfortable stay in the world, humanity must pay great attention to the environmental disaster, these include: the extinction of plants and animals, pollution, salinization, waterlogging and soil erosion, air pollution, water pollution of the oceans, environmental pollution, depletion of water resources, etc.

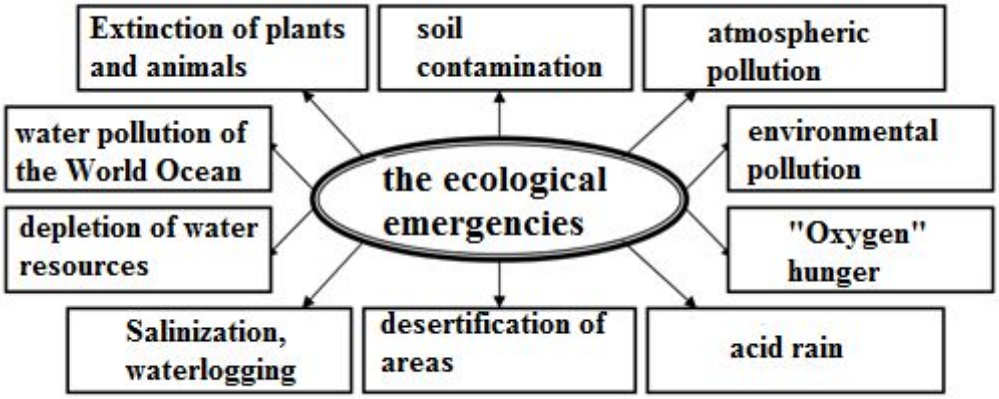

Figure 3.8 Classification of ecological disaster.

Third chapter summary

1. Was considered a rational organization of the workplace.
2. Was examined in detail emergencies, their causes, effects on humans.

CONCLUSION

As a result of this work were obtained the following results:

- The theoretical bases of systematization information and administration of stored data were considered.
- An analysis of the subject area was conducted.
- The relational data model was developed.
- The data access layer was developed.

The result of all the work is the development of tool to systematize information and administration of stored data in the information subsystem «Olim» for awarding degrees and awarding academic titles for science teachers in areas of science, technology, education and culture.

During of writing this final work was confirmed by the facts that modern information technology with their rapidly growing potential and quickly reduce costs open up many possibilities for new forms of work organization and employment both within individual corporations, and society as a whole.

One of the most important stages in many business processes in any organization is to systematize information. Thanks to meticulous systematization of information can achieve good results in the optimization of workflow, as well as savings company and employee time. No pre-ordering information is impossible to imagine such critical business processes such as document management, records management, creation of material and electronic archive various databases.

Developed data access layer allows to process and store information in a convenient way for the user and for the programmer and has great potential in the development, due to the fact that this level allows you to add new features without overwriting the old one.

This level of data access has been implemented on the platform of ASP.NET, used programming language C #, and for the realization of the database was used DBMS Microsoft SQL Server 2008 R2.

REFERENCES

1. http://lex.uz/pages/GetAct.aspx?lact_id=495104 regulations of the Higher Attestation Committee under the Cabinet of Ministers of the Republic of Uzbekistan
2. http://lex.uz/pages/GetAct.aspx?lact_id=495104 Resolution Of Cabinet Under The President Of The Republic Of Uzbekistan «Issues of activity organization Higher Attestation Committeeunder the Cabinet of Ministers»
3. http://www.archives.su/poleznoe/sucsess/?nuin=427 Method Of systematization of information
4. http://ru.wikipedia.org/wiki/Администрирование_данныхAdministration of data
5. http://zametkinapolyah.ru/zametki-o-mysql/bazy-dannyx-vidy-i-tipy-baz-dannyx-struktura-relyacionnyx-baz-dannyx-proektirovanie-baz-dannyx-setevye-i-ierarxicheskie-bazy-dannyx.html types of database
6. http://kavaliou.blogspot.com/2011/04/blog-post_19.html three-tier application architecture
7. http://www.km.ru/59C218F4FB8342919F15D78DA3A3FD98#/ The rational organization of a workplace

APPENDIX

DataProvider:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Configuration;
using System.Data.Sql;
using System.Data.SqlClient;
using System.Data.SqlTypes;
using System.Data;
using FAN.HelpClasses;

namespace FAN.DAL
{
publicclassDataProvider
    {
        #region Fields
privatestring _connectionString;
privatestring _tableName = null;
privateSqlConnection _sqlConnection;
privateSqlCommand _sqlCommand;
privatestring _sqlRequest;
privateSqlDataReader _sqlDataReader;
privateint _idOfJustInsertedRow;
privatebool flag = false; // to search for
        #endregion
        #region Properties
protectedint IdOfJustInsertedRow
        {
get { returnthis._idOfJustInsertedRow; }

        }
protectedstring TableName
        {
get { return _tableName; }
set { _tableName = value; }
        }
protectedSqlDataReader DataReader
        {
get { return _sqlDataReader; }
        }
        #endregion
        #region Methods
privatebool TestConnection(string connectionString)
        {
try
            {
using (var connection = newSqlConnection(connectionString))
                {
                    connection.Open();
returntrue;
                }
            }
catch
            {
returnfalse;
            }
        }
privatestring SearchParamsBuilder(List<FilterValue> filterVals)
        {
```

```
string search = "";
string operation = "";
foreach (FilterValue filterVal in filterVals)
            {
switch (filterVal.filterType)
                {
caseFilterType.Equal:
                        operation = "=";
break;
caseFilterType.Greater:
                        operation = ">";
break;
caseFilterType.Less:
                        operation = "<";
break;
caseFilterType.NotEqual:
                        operation = "<>";
break;
caseFilterType.Like:
                        operation = "%";
break;
                }
if (filterVal.value isstring)
                {
if (((string)filterVal.value).Contains("Select"))
                        search    +=    filterVal.fieldName    +    operation    +'('    +
filterVal.value+')' + " and ";
elseif (filterVal.filterType == FilterType.Like)
                        search += filterVal.fieldName + " like " + "\'" + operation +
filterVal.value + operation + "\'" + " and ";
else
                        search   +=   filterVal.fieldName   +   operation   +   "\'"   +
filterVal.value + "\'" + " and ";
                }
            }
            search = search.Remove(search.Length - 4, 4);
return search;
        }

publicstring SearchParamsBuilder(List<FilterValue> filterVals, bool isVoid)
        {
returnthis.SearchParamsBuilder(filterVals);
        }
publicstring SqlRequestBuilder(string searchString)
        {
returnstring.Format("Select * from {0} where {1}", _tableName, searchString);
        }
publicstring SearchString
        {
set
            {
this.flag = true;
this._sqlRequest = string.Format("Select * from {0} where ", _tableName);
this._sqlRequest += value;
          }
get
            {
returnthis._sqlRequest;
            }
        }
protectedstring SqlRequestBuilder(SqlRequestType sqlRequestType,
List<string> fields = null,
List<object> values = null,
string nameOrederingBy = null,
```

```
List<FilterValue> filterVals = null)
        {
string sqlRequest = "";
switch (sqlRequestType)
            {
caseSqlRequestType.Max:
                    sqlRequest  =  string.Format("Select  max({0})  as  maxValueOf{1}Column
from {2}", values[values.Count - 1], values[values.Count - 1], _tableName);
break;
caseSqlRequestType.Create:
                    sqlRequest = string.Format("Insert into {0}(", _tableName);
for (int counter = 0; counter < values.Count; counter++)
                    {
                        sqlRequest += " " + fields[counter] + ",";
                    }
                    sqlRequest = sqlRequest.Remove(sqlRequest.Length - 1, 1);
                    sqlRequest += ") values( ";
for (int counter = 0; counter < values.Count; counter++)
                    {
                        sqlRequest += " @" + fields[counter] + ",";
                    }
                    sqlRequest = sqlRequest.Remove(sqlRequest.Length - 1, 1);
                    sqlRequest += ");";
                    sqlRequest += " select cast(scope_identity() as int)";
break;
caseSqlRequestType.Update:
                    sqlRequest = string.Format("Update {0} set", _tableName);

for (int counter = 0; counter < fields.Count - 1; counter++)
                    {
                        sqlRequest   +=   string.Format("   {0}=@{1}",   fields[counter],
fields[counter]) + ",";
                    }
                    sqlRequest = sqlRequest.Remove(sqlRequest.Length - 1, 1);
                    sqlRequest += " where ID=@ID";
break;
caseSqlRequestType.Delete:
                    sqlRequest = string.Format("Delete from {0} where ", _tableName);
                    sqlRequest  +=  string.Format("{0}=@{1}",  fields[fields.Count  -  1],
fields[fields.Count - 1]);
break;
/*to the input parameters to add a field list <string> searchParameters
                  it is necessary to retrieve data from several parameters
                  or consider another way to make a selection on several parameters
                  or write separately to select the method*/
caseSqlRequestType.Read:
                    sqlRequest = "Select ";
if (fields == null)
                        sqlRequest += string.Format(" * from {0}", this._tableName);
else
                    {
for (int counter = 0; counter < fields.Count; counter++)
                        {
                            sqlRequest += string.Format(" {0},", fields[counter]);
                        }
                        sqlRequest = sqlRequest.Remove(sqlRequest.Length - 1, 1);
                        sqlRequest += string.Format(" from {0}", _tableName);
                    }
if (values != null)
                    {
                        sqlRequest      +=      string.Format("      where      {0}=@{1}",
fields[fields.Count - 1], fields[fields.Count - 1]);
                    }
if (nameOrederingBy != null)
```

```
                    {
                        sqlRequest += string.Format(" ORDER BY {0}", nameOrederingBy);
                    }
break;
caseSqlRequestType.Search:
if (fields == null)
                        sqlRequest  =  string.Format("Select  *  from  {0}  where  {1}",
_tableName, SearchParamsBuilder(filterVals));
else
                        sqlRequest  =  string.Format("Select  {0}  from  {1}  where  {2}",
fields[0], _tableName, (SearchParamsBuilder(filterVals)));
break;
//need to produce a query string to retrieve all the relevant parameters id request
            }
return sqlRequest;
        }
protectedvoid SqlRequestHandler(SqlRequestType sqlRequestType,
List<string> fields = null,
List<object> values = null,
string nameOrederingBy = null,
List<FilterValue> filterVals = null)
        {
if (!flag)
this._sqlRequest     =     this.SqlRequestBuilder(sqlRequestType,     fields,     values,
nameOrederingBy);
switch (sqlRequestType)
            {
caseSqlRequestType.Max:
this._sqlCommand.CommandText = _sqlRequest;
this._sqlCommand.Parameters.Clear();
break;
caseSqlRequestType.Create:
this._sqlCommand.CommandText = _sqlRequest;
this._sqlCommand.Parameters.Clear();
for (int counter = 0; counter < values.Count; counter++)
                    {
if (values[counter] != null)
this._sqlCommand.Parameters.Add(newSqlParameter(string.Format("@{0}",    fields[counter]),
values[counter]));
else
this._sqlCommand.Parameters.Add(newSqlParameter(string.Format("@{0}",    fields[counter]),
DBNull.Value));
                    }
break;
caseSqlRequestType.Update:
this._sqlCommand.CommandText = _sqlRequest;
this._sqlCommand.Parameters.Clear();
for (int counter = 0; counter < fields.Count; counter++)
                    {
if (values[counter] != null)
this._sqlCommand.Parameters.Add(newSqlParameter(string.Format("@{0}",    fields[counter]),
values[counter]));
else
this._sqlCommand.Parameters.Add(newSqlParameter(string.Format("@{0}",    fields[counter]),
DBNull.Value));
                    }
break;
caseSqlRequestType.Delete:
this._sqlCommand.CommandText = _sqlRequest;
this._sqlCommand.Parameters.Clear();
this._sqlCommand.Parameters.Add(newSqlParameter(string.Format("@{0}",  fields[fields.Count
- 1]), values[values.Count - 1]));
break;
/*to the input parameters to add a field list <string> searchParameters
```

```
                    it is necessary to retrieve data from several parameters
                    or consider another way to make a selection on several parameters
                    or write separately to select the method*/
caseSqlRequestType.Read:
if (values != null)
                      {
this._sqlCommand.CommandText = _sqlRequest;
this._sqlCommand.Parameters.Clear();
this._sqlCommand.Parameters.Add(newSqlParameter(string.Format("@{0}",  fields[fields.Count
- 1]), values[values.Count - 1]));
                      }
else
this._sqlCommand.CommandText = this._sqlRequest;
break;
caseSqlRequestType.Search:
this._sqlCommand.CommandText = this._sqlRequest;
this._sqlCommand.Parameters.Clear();
break;
            }
try
             {
                 OpenConnection();
switch (sqlRequestType)
                    {
caseSqlRequestType.Read:
caseSqlRequestType.Max:
caseSqlRequestType.Search:
this._sqlDataReader = _sqlCommand.ExecuteReader();
break;
caseSqlRequestType.Create:
this._idOfJustInsertedRow = (int)this._sqlCommand.ExecuteScalar();
break;
caseSqlRequestType.Update:
caseSqlRequestType.Delete:
this._sqlCommand.ExecuteNonQuery();
break;
                    }
             }
catch (SqlException e)
             {
thrownewApplicationException("Error occured while handling sql request"
+ "\nException message: "
                                                        + e.Message);
             }
finally
             {
             }
        }
public DataProvider()
        {
this._connectionString                                                                  =
ConfigurationManager.ConnectionStrings["FANConnectionString2"].ConnectionString;
this._sqlConnection = newSqlConnection(_connectionString);
this._sqlCommand = newSqlCommand();
this._sqlCommand.Connection = SingletonConnection.Instance.Connection;
        }

protectedvoid OpenConnection()
        {
if (SingletonConnection.Instance.Connection.State == ConnectionState.Closed) {
SingletonConnection.Instance.Connection.Open();
             }
        }
```

```
protectedvoid CloseConnection()
        {
if (SingletonConnection.Instance.Connection.State == ConnectionState.Open)
            {
SingletonConnection.Instance.Connection.Close();
            }
        }
        #endregion

protectedenumSqlRequestType
        {
            Create,
Read,
            Update,
            Delete,
            Max,
            Search
        };
publicenumFilterType
        {
            Equal,
            NotEqual,
            Greater,
            Less,
            Like
        };
        [Serializable]
publicstructFilterValue
        {
publicstring fieldName;
publicobject value;
publicFilterType filterType;
}

    }
}
```