

ҚАЗАҚСТАН РЕСПУБЛИКАСЫ БІЛІМ ЖӘНЕ ҒЫЛЫМ МИНИСТРЛІГІ
Коммерциялық емес акционерлік қоғамы
АЛМАТЫ ЭНЕРГЕТИКА ЖӘНЕ БАЙЛАНЫС УНИВЕРСИТЕТІ
«Компьютерлік технологиялар» кафедрасы

«Қорғауға жіберілді»
Кафедра меңгерушісі
ф.-м.ғ.д., проф. З.Қ. Құралбаев

_____ « _____ » _____ 2014 ж.
(қолы)

ДИПЛОМДЫҚ ЖОБА

Тақырыбы: «Nash Pub кафеісінің желідегі ақпарат жүйесін автоматтандыру
1 бөлім»

5В070400 – Есептеу техникасы және бағдарламалық қамтамасыз ету
мамандығы бойынша

Орындаған: ВТк-10-2 Тусупбеков А.С.

Жетекші: ф.-м.ғ.к., доцент Қоржымбаев Т.Т.

Кеңесшілер :

Экономикалық бөлім бойынша :

_____ доцент Боканова Г.Ш.
_____ « _____ » _____ 2014 ж.
(қолы)

Өмір тіршілігі қауіпсіздігі бойынша:

_____ т.ғ.к., аға оқытушы Муташева Г.С.
_____ « _____ » _____ 2014 ж.
(қолы)

Есептеу техникасын қолдану бойынша :

_____ ф.-м.ғ.к., доцент Қоржымбаев Т.Т.
_____ « _____ » _____ 2014 ж.
(қолы)

Мөлшер бақылаушы:

_____ аға оқытушы Рахимжанова З.М.
_____ « _____ » _____ 2014 ж.
(қолы)

Пікір жазушы :

_____ АТУ, т.ғ.д., проф. Заурбеков Н.С.
_____ « _____ » _____ 2014 ж.
(қолы)

Алматы 2014

ҚАЗАҚСТАН РЕСПУБЛИКАСЫ БІЛІМ ЖӘНЕ ҒЫЛЫМ МИНИСТРЛІГІ
Коммерциялық емес акционерлік қоғамы
АЛМАТЫ ЭНЕРГЕТИКА ЖӘНЕ БАЙЛАНЫС УНИВЕРСИТЕТІ

«Ақпараттық технологиялар» факультеті
«Есептеу техникасы және бағдарламалық қамтамасыз ету» мамандығы
«Компьютерлік технологиялар» кафедрасы

жобаны орындауға берілген

ТАПСЫРМА

Студент Тусупбеков Айдос

Жоба тақырыбы «Nash Pub кафеінің ақпарат жүйесін желідегі
автоматтандыру 1 бөлім»

ректордың « ____ » _____ № ____ бұйрығы бойынша
бекітілген.

Аяқталған жұмысты тапсыру мерзімі: « ____ » _____ 2014 ж.

Жобаға бастапқы деректер (талап етілетін жоба нәтижелерінің параметрлері және нысанның бастапқы деректері):

MySQL дерекқорларды басқару жүйесін және Delphi бағдарламалау ортасын пайдалану арқылы кафе жұмысының дерекқор ақпаратын ұсынатын қосымша құрылды.

Диплом жобасындағы әзірленуі тиіс сұрақтар тізімі немесе диплом жобасының қысқаша мазмұны:

– Автоматтандырылған қызмет көрсету орнының анықтамасы (арналуы, құрылымы, т.б.).

– Ақпараттық жүйелер, дерекқорлар (ДҚ), дерекқорларды басқарудың жүйелері (ДҚБЖ) түсініктері.

– Жүйенің дерекқорын жобалау және жүзеге асыру, жобалау құралдарын таңдау, ER моделін жобалау, дерекқорды құру, пайдаланушы интерфейсін жобалау, дерекқормен байланысын орнату.

– Техникалық-экономикалық негізделуі.

– Өндірістегі еңбек қорғау мен қауіпсіздік шараларын ұйымдастыру.

Сызба материалдарының (міндетті түрде дайындалатын сызуларды көрсету) тізімі:

– Дерекқордың ER диаграммасы.

Негізгі ұсынылатын әдебиеттер

- 1 Васвани. MySQL: использование и администрирование = MySQL Database Usage & Administration. — М.: «Питер», 2011.
- 2 Стив Суэринг, Тим Конверс, Джойс Парк. PHP и MySQL. Библия программиста, 2-е издание = PHP 6 and MySQL 6 Bible. — М.: «Диалектика», 2010.
- 3 Кузнецов Максим, Симдянов Игорь. MySQL на примерах. — Спб.: «БХВ-Петербург», 2008.
- 4 Хавьер Пашеку. Программирование в Borland Delphi 2006 Delphi for .NET Developer's Guide. — М. Вильямс.
- 5 Д.Гудман JavaScript и DHTML, сборник рецептов. Для профессионалов – М.: «Питер», 2004.
- 6 David Flanagan JavaScript. Подробное руководство – М.: «Питер», 2008.

Жоба бойынша бөлімшелерге қатысты белгіленген кеңесшілер

Бөлім	Кеңесші	Мерзімі	Қолы
Негізгі бөлім	Құралбаев З.Қ.		
Тіршілік қауіпсіздігі	Муташева Г.С.		
Экономикалық бөлім	Боканова Г.Ш.		
Норма бақылаушы	Рахимжанова З.М.		
Есептеу техникасын қолдану	Құралбаев З.Қ.		

ДИПЛОМ ЖОБАСЫН ДАЙЫНДАУ

К Е С Т Е С І

№ р/с	Тарау аттары, әзірленетін сұрақтардың тізімі	Жетекшіге ұсыну мерзімдері	Ескерту
1	Автоматтандырылған қызмет көрсету орнының анықтамасы (арналуы, құрылымы, т.б.).		
2	Ақпараттық жүйелер, дерекқорлар (ДҚ), дерекқорларды басқарудың жүйелері (ДҚБЖ) түсініктері.		
3	Жүйенің дерекқорын жобалау және жүзеге асыру, жобалау құралдарын таңдау, ER моделін жобалау.		
4	Дерекқорды құру, пайдаланушы интерфейсін жобалау, дерекқормен байланысын орнату.		
5	Жүйенің дерекқорын шифрлау түрін таңдау, жүзеге асыру		

Тапсырманың берілген уақыты « _____ » _____ 2014ж.

Кафедра меңгерушісі _____ ф.-м.ғ.д., проф. Құралбаев З.Қ.
(қолы)

Жоба жетекшісі _____ ф.-м.ғ.к., доцент Қоржымбаев Т.Т.
(қолы)

Орындалатын тапсырманы қабылдаған студент _____ Тусупбеков А.
(қолы)

Аңдатпа

Бұл дипломдық жобада «Nash Pub» кафесінің ақпарат жүйесі желідегі автоматты жұмыс істеуі қамтылды. Кафе қызметкерлері автоматтандырылған бағдарламаның негізгі және қосымша бөліктерін пайдалана отырып, өз жұмыстарын күнделікті орындау мүмкіншіліктері бар. Ақпараттық жүйеде деректер қорының архитектуралық құрылымы және дерекқорының қауіпсіздігі, қызметкерлер қолдану аумағының ойдағыдай жылдам жұмыс істеу мәселелері қарастырылған.

Өміртіршілік қауіпсіздігін қамтамасыз ету бөлімінде жобаның қолданылатын кафе аумағының жарықтандыру, желдету сияқты түрлі мәселелері шешімін тапты.

Экономикалық бөлімде, жобаның экономикалық тиімділігін дәлелдеп бизнес-жоспарын жасалды.

Аннотация

В этой дипломной работе охвачена автоматическая работа информационной системы кафе "Nash Pub" в сети. Работники кафе имеют возможность ежедневно исполнять свои обязанности, используя основные и дополнительные части автоматизированной программы. В информационной системе рассмотрены архитектурная структура информационного архива, его безопасность и проблемы его быстрого и удобного использования сотрудниками в их сфере. .

В разделе об обеспечении безопасности жизни нашли решения такие проблемы, как освещение и вентиляция площади кафе, участвовавшего в данном проекте.

В экономическом разделе доказана экономическая выгода данного проекта и разработан бизнес-план.

Annotation

This thesis enfolds the automatic work of informational system of the cafe "Nash Pub" in the network. Workers of the cafe have the opportunities daily to fulfill their duties using the main and additional parts of an automatic program. In the informational system the architectural structure of the informational achieve, its security and the problems of its fast and convenient usage by the workers, in their sphere, are considered.

In the column about providing the life security such problems, as lighting and ventilation of the cafe, which took part in this project, are solved.

In economic column the economic benefits of the given project is proved and business-plan is developed.

Мазмұны

Кіріспе	8
1 Автоматтандырылған ақпараттық жүйе (ААЖ)	9
1.1 Негізгі түсініктер	9
1.2 Жұмыс оқу жоспарын өңдейтін автоматтандырылған ақпараттық жүйенің сипаттамасы	9
1.3 ААЖ-ге қойылатын жалпы талаптар	10
2. Мәліметтер базасы(МБ). Мәліметтер базасын басқару жүйелері (МББЖ)	12
2.1 Мәліметтер базасы туралы жалпы ақпарат	12
2.1.1 Мәліметтер базасының кең түрде қолданылуының алғы шарттар	13
2.1.2 Мәліметтер базасын пайдаланушылар мен олардың деңгейлері	13
2.1.3 Мәліметтер базасы әкімшілігінің функциялары	14
2.1.4 Мәліметтер базасы администраторының байланыстары	15
2.2 Мәліметтер базасын басқару теориясы	16
3. Мәліметтер қорын жобалау және оны іске асыру	18
3.1 MySQL мәліметтер базасы	18
3.2 ER – диаграмма құру	18
3.3 Реляциялық модельді құру	19
3.4 MySQL-де мәліметтер базасын құру мен мәліметтер базасын толтыру	20
3.5 МҚ диаграммасы	24
4 Физикалық жобалау	26
4.1 Delphi программалау ортасы	26
4.1.1 Delphi туралы жалпы түсінік	26
4.2 Delphi программалау ортасында мәліметтерге қатынаудың негізгі құрауыштары	27
4.3 TADOTable құрауышының қасиеттері	33
4.4 TADOTable құрауыштарының әдістері	35
5 Қосымша интерфейсін және оның функционалдығын сипаттау	37
6 Техико – экономикалық негізделуі	41

6.1 Жұмысты сипаттау және қажеттілігін негіздеу	41
6.2 Нарықты саралау және орнын анықтау	41
6.3 Бағдарламаның SWOT-анализі	41
6.4 Маркетинг-микс элементі	42
6.5 Бағдарламалық қамтаманың өңделуінің еңбек өнімділігінің есептелуі	42
6.5.2 Еңбекақыға жұмсалатын шығындар	44
6.5.3 Қосымша шығындар статьясы	45
6.5.4 Өзіндік құн нәтижесінің кестесі мен диаграммасы	47
6.6 Бағдарламаның бағасына есептеу	49
6.6.3 Бағдарламалық өнімді иемденудегі кәсіпорындардың бірмезгілдегі шығындарын есептеу	49
6.6.4 Қолдану саласындағы жылдық шығындарды есептеу Ақпараттық өнімді енгізуден алынған үнемдеу мөлшері мен табысты есептеу	50
6.7 Ақшалай құралдардың қозғалысы	52
6.8 Экономикалық тиімділікті есептеу	52
6.8.1 Таза ағымдағы құндылықты есептеу (Net present value, NPV)	52
6.8.2 Пайда индексін есептеу (Profitability index, PI)	53
6.8.3 Табыстың ішкі нормасын есептеу (Internal rate of return, IRR)	53
6.8.4 Өтімділік периодын есептеу (Payback period, PBP)	54
Қорытынды	65
Әдебиеттер тізімі	66
Қосымша А	67
Қосымша Ә	80

Кіріспе

Ақпараттық технологиялардың дамуы мен кең қолданылуы әлемдік бірлестіктің даму үрдісінде маңызды роль атқарады. Қазіргі заманғы технологияларды ақпаратты өңдеу үшін пайдалану бәсекеге қабілеттілікті арттырады, әлемдік жүйеге ықпалдауды кеңейтуге әсерін тигізеді, басқару процессінің тиімділігі мен сапасын арттыруға маңыздылығы көп.

Дипломдық жобаның өзектілігі: офис-тіркеушісінің жұмысын жеңілдетіп, жұмыс уақытын үнемдететін магистратура мамандықтарына оқу жоспарын өңдейтін автоматтандырылған ақпараттық жүйе құру болып табылады.

Бұл дипломдық жобаның мақсаты: «Тұран» университетінің магистратура, докторантура және халықаралық бірлестік департаментіне оқу кестесін құруға көмекші және ақпаратты автоматты түрде өңдейтін ақпараттық жүйе құру болып табылады. Ол модель кафедрадан берілген әр мамандық бойынша жұмыс оқу жоспарын негізгі ақпараттық ағым талдауына негізделген форма түрінде құралып, нәтижесінде магистранттардың сабақтарының оқу кестесін құруға көмектесетін инфологиялық модель шығады. Ол ақпаратқа оперативті және қолжетімді, жұмыс оқу жоспарында өзгеріс болған жағдайда, білім берудің кредиттік жүйесіне сәйкес мәліметтер базасына өзгерістер енгізуге ыңғайлы болуы тиіс. Ал мәліметтер базасында университет, факультет, кафедра, мамандық, оқытушылар, аудитория және пән (кредиттер саны, дәрістер мен семинарлар) жайында мағлұматтар сақталады. Мультимедиялық ақпаратты стандартты құжаттарды формаландыру HTML (HyperText Markup Language) мен PHP тілдері негізінде жазылады.

Алдыға қойған мақсатқа жету үшін орындалатын тапсырмалар:

- пәндік аумақтың талдауы;
- мағлұмат жинау мен оны талдау;
- бағдарламаларға шолу;
- мәліметтер базасының құрылымын жобалау;
- ақпараттық жүйенің құрылымын жобалау;
- ақпараттық жүйенің кодталынуы;
- ақпараттық жүйе жұмысын тестілеу, нәтижелерін талдау;
- автоматтандырылған ақпараттық жүйе түсінігі, кіріспе.

1 Автоматтандырылған ақпараттық жүйе (ААЖ)

1.1 Негізгі түсініктер

Автоматизация — өзін реттейтін техникалық мүмкіндік, экономика-математикалық әдістер мен бақылау жүйесі, энергияны жіберу мен қолдануда, қабылдау процесстерінде, өзгертуде, мәліметтерді немесе ақпараттарды, адамдардың көмегісіз-ақ орындалатын операциялардың күрделілігін кемітетін ғылыми-техникалық прогресстің бағыттарының бірі.

Қосымша датчиктерді (сенсорлар), енгізу құрылғыларын, басқару құрылғыларын (бақылаушылар), шығару құрылғыларын, электронды техниканы және есептеу әдістерін, кейде адамның ойлау мен жүйке жүйелерін көшіріп алатын функцияларды қолдануды талап етеді. Автоматты терминмен қатар, автоматтандырылған термині де жиі қолданады және онда процесс барысында адамның қатысуын керек етеді.

Автоматтандырылады:

- өндірістік процесстер;
- жобалау;
- ұйымдастыру, жоспарлау мен басқару;
- ғылыми ізденістер;
- бизнес-процестер.

Автоматизациялаудың мақсаты — еңбек өнімділігін арттыру, өнімнің сапасын жақсарту, басқаруды ықшамдау, өндірістен адамдардың денсаулығына зиян еңбек-жұмыстарын жою. Автоматизация тапсырманы шешуді комплекстік, жүйелік шешу жолдарын таңдайды, сондықтан да автоматизация алдында тұрған мәселелерді көбіне жүйелік деп атайды.

Автоматтандырылған ақпараттық жүйе (ААЖ) адамды және ақпаратты өңдеу процесі кезінде қатысады, техникалық мүмкіндіктер, яғни басты роль компьютерге назар аударылады. Қазіргі кезде «АЖ» терминінің талдауы кезінде автоматизациялау жүйесі ұғымы да міндетті енгізіледі.

Автоматтандырылған АЖ, басқару процесін ұйымдастыру кезінде кең қолданыс тапқан, оның түрлі модификациясы бар және талдануы мүмкін, мысалы, ақпаратты қолдану мен қолданыс ортасына байланысты мінездемесіне қарай.

Автоматтандырылған АЖ. Автоматтандырылған ақпараттық жүйе адамның қатысуынсыз-ақ ақпаратты өңдеуге арналған бүкіл операцияларды орындай алады.

1.2 Жұмыс оқу жоспарын өңдейтін автоматтандырылған ақпараттық жүйенің сипаттамасы

Жұмыс оқу жоспарын өңдейтін ақпараттық жүйе келесі қасиеттерге ие:

- икемділік – офис-тіркеушінің қалаған уақытында іске асыра алу

мүмкіндігі;

– модульдылық – бір-біріне тәуелсіз оқу курс-модульдерінен оқу жоспарын құрып алу мүмкіндігі;

– қамту – түрлі ақпарат көздеріне бір уақытта үндеу жасай алу мүмкіндігі (кафедраларға, мамандықтарға, факультеттерге, аудиторияларға, оқытушыларға және т.б.);

– үнемділік – оқу кестесін құруға уақыттың үнемділігі және уақытты тиімді қолданылуы, ыңғайлы дайын формаларға мәліметтерді енгізу мүмкіндігі;

– технологиялық – білім беру процессінде ақпараттық және телекоммуникациялық технологияларды қолдану адамды постиндустриалды ақпараттық кеңістікке жылжытуға мүмкіндік береді.

1.3 ААЖ-ге қойылатын жалпы талаптар

ААЖ-ға қойылатын жалпы талаптар:

– жүйелілігі;

– икемділігі;

– тұрақтылығы;

– тиімділігі.

Жүйелілік. ААЖ жұмыс орнының функционалды арналуымен анықталатын құрылымы бар жүйе деп қарастырғаны жөн.

Икемділік. Элементтердің стандартталуы мен құрылудың модульдігінің арқасында жүйе мүмкін болатын қайта құруларға икемді болу керек.

Тұрақтылық. Ішкі мен сыртқы қоздыратын факторлардың әсеріне қарамастан жүйе негізгі функцияларды орындауы керек.

Эффективтілік. ААЖ-ны жүйенің құрылуы мен қолдануына кеткен шығындарына апаратын жоғарыда келтірілген қағидалардың жүзеге асырылу деңгейінің интегралды көрсеткіші де қарастырғаны жөн.

ААЖ бұл әрдайым арнаулы сипатты жүйе. Нақты бір маманға әкімші, экономист, инженер, конструктор, жобалаушы, сәулетші, дизайнер, дәрігер, ұйымдастырушы, зерттеуші, кітапханашы, мұражай қызметкері және т.б. арналған техникалық құралдар мен программалық қамтамасыз етудің жиынтығы.

ААЖ-ны ендіру компьютер ұйытқы болатын ақпаратты өңдеудің машиналық құралдар мен адам арасындағы функциялар мен жүктеменің дұрыс бөліп тарату жағдайында қалаған эффектті беруі мүмкін.

ААЖ бұл тек еңбектің өнімділігін мен басқарудың эффективтілігін жоғарлатуының құралы ғана емес, сонымен қатар мамандардың қоғамдық қолайшылығының құралы.

ААЖ жеке (дербес) , топтық, коллективті болуы мүмкін. Топтық пен коллективті ААЖ-на қатысты олардың ДК-дің жүйе ретінде эффективті функционалдау мақсатында мамандарға әкімшілдеудің нақты функцияларын анықтау қажет. ААЖ адам машиналық жүйе ретінде ашық, икемді, үнемі даму мен жетілдірілуіне бейімді болу қажет.

Мұндай жүйеде келесі қамтамасыз етілу керек:

- мамандардың ақпаратты өңдеу машина құралдарына максималды жақындығы;
- диалогты тәртіпте жұмыс жасау;
- эргономика талаптарына сәйкестілік;
- компьютердің жоғары икемділігі;
- ескілік процестердің максималды автоматтандырылуы;
- жекелік жағдайда мамандардың өнерпаздық белсенділігін арттыратын, ал кейін жүйенің дамуына әкелетін мамандардың еңбек жағдайларына моральды түрде қанағат болуы;
- мамандардың өзін-өзі оқыту мүмкіндігі.

2 Мәліметтер базасы(МБ). Мәліметтер базасын басқару жүйелері (МББЖ)

2.1 Мәліметтер базасы туралы жалпы ақпарат

Мәліметтер базасы (МБ) – белгілі бір арнаулы аумаққа қатысты мәліметтерді баяндау, сақтау мен амал-тәсілдермен ұқсата білудің жалпы принциптерін қарастыратын белгілі бір ережелер бойынша ұйымдастырылған мәліметтердің жиынтығы. Адамның қатысуы ықтимал, автоматтық құралдармен өңдеуге жарамды түрде ұсынылған ақпарат мәліметтер ретінде түсініледі. Арқаулы ауқым ретінде нақты зерттеуші үшін қызықты нақты әлемнің бөлігі түсініледі.

Мәліметтер базасы бұл ақпараттарды сақтау үшін арналған ұйымдастырылған құрылым. Алғашқы кезде, яғни, мәліметтер базасы түсінігі қалыптаса бастаған жылдары, бұл базаларда тек қана мәліметтер сақталып жүрді. Бірақ бүгінгі күнгі мәліметтер базасын басқару жүйелері (МББЖ) өз құрылымында тек қана мәліметтерді ғана емес, сонымен қатар тұтынушылармен немесе басқа да программалық-ақпараттық кешендермен қарым-қатынас жүзеге асырылатын әдістерді (яғни, программалық код) де орналастыруға мүмкіндік береді. Осылайша біз қазіргі заманғы мәліметтер базасында тек қана мәліметтер ғана емес, сонымен бірге ақпарат та сақталатындығын айта аламыз.

Егер, мысалы, оқу процессін алатын болсақ, бұл анықтаманы оңай түсіндіруге болады, мұнда оқылатын пәндер, олардың неше кредиттен тұратындығы, сол пәндерге қатысты сабақ беретін оқытушылар, ол пән таңдау пәндер қатарына жатады ма және т.б. көптеген мәліметтер сақталады. Бұл базаға оқу процессіне қатысты көптеген қызметкерлер кіре алады. Бірақ олардың ішінде бұл базаға толықтай ие болып, оған өзі жеке өзгерістер енгізе алатын адам жоқ. Мәліметтерден басқа базада әр түрлі әдістер мен жабдықтар бар, ол әрбір қызметкерге оның компетенциясына кіретін мәліметтермен ғана жұмыс жасауға мүмкіндік береді. Базадағы мәліметтердің нақты қызметкерлерге тиісті әдістермен өзара байланысы нәтижесінде қызметкерлер пайдаланатын және өз компетенциясы шеңберінде мәліметтерді енгізу мен редактрлеуді жүргізетін ақпарат қалыптасады.

Мәліметтер базасы түсінігімен мәліметтер базасын басқару жүйесі түсінігі тығыз байланысты. Бұл жаңа базаның құрылымын құру, оны мәліметтермен толтыру, мәліметтерді редактрлеу мен ақпараттарды визуалдау үшін арналған программалық жабдықтар кешені. Базаның ақпаратын визуалдау деп берілген критерилерге байланысты көріністегі мәліметтерді іріктеуді, оларды тәртіптеуді, толықтыру және шығару құрылғысына беру немесе байланыс каналдары бойынша жіберуді түсінуге болады.

Әлемде көптеген мәліметтер базасын басқару жүйелері бар. Олардың әр түрлі объектілермен әр түрлі жұмыс істеу мүмкіндігі мен пайдаланушыларға әр

түрлі функциялар мен жабдықтар ұсынатындығына қарамастан МББЖ-нің көбі ортақ негізгі түсініктердің тұрақты кешеніне негізделеді. Бұл бізге бір жүйені қарастырып, оның түсініктерін, әдістерін және тәсілдерін МББЖ-нің барлық классына жалпылауға мүмкіндік береді.

2.1.1 Мәліметтер базасының кең түрде қолданылуының алғы шарттары

МБ-ның айқын басымдылықтары мен объективті алғы шарттары оның кең түрдегі қолданысына әкелді. Оны қолданудың алғы шарттарына келесілер жатады:

- нақты өмірдің объектілері өзара күрделі қарым-қатынаста болады. Бұның бәрі олардың ақпараттық көрінісі өзара байланысқан бір бүтінді қалыптастыруды талап етеді;

- әртүрлі пайдаланушылардың ақпараттық қажеттіліктері өзара қиылысады, ал бұл бір мәліметтер базасын қолдануды және әртүрлі пайдаланушылардың оған кіру мүмкіндігін бір мақсатты қылады;

- ақпараттық қорды құру мен жүргізу функциялары және қажетті мәліметтерді ұсыну әртүрлі тапсырмаларды шешуде универсалды және жалпы болып табылады. Мәліметтерді басқару үшін мамандандырылған программалық жабдықтарды құру – бұл функциялардың орындалу деңгейінің өсуі мен ақпараттық жүйелерді құрудың еңбек сыйымдылығының қысқаруына әкеледі;

- техникалық және программалық қамтымасыз етудің дамуының қазіргі заманғы деңгейі, ақпараттық жүйелерді құру теориялары мен практикасы тиімді МБ-н құруға мүмкіндік береді.

2.1.2 Мәліметтер базасын пайдаланушылар мен олардың деңгейлері

Мәліметтер базасымен оның құру мен функционалдау процесінде әр түрлі категориядағы пайдаланушылар бірге іс-әрекет жасайды.

Пайдаланушылардың негізгі категориясы болып түпкілікті пайдаланушылар, яғни, олардың қажеттіліктері үшін мәліметтер базасы құрылатын пайдаланушылар табылады. Құрылушы мәліметтер базасының ерекшеліктеріне байланысты оның түпкілікті пайдаланушылар әр түрлі бола алады. Бұл мәліметтер базасына мезгіл өткен сайын көңіл аударатын кездейсоқ пайдаланушылар болуы мүмкін немесе тұрақты, яғни, әрдайым пайдаланатын пайдаланушылар болуы мүмкін. Түпкілікті пайдаланушылар бір-бірінен есептеуіш техниканы меңгеру дәрежесімен де ерекшеленуі мүмкін. Түпкілікті пайдаланушылардан есептеуіш техника мен тіл жабдықтары саласында арнайы білім талап етілмеуі керек.

Мәліметтер базасын функционалдау мәліметтер базасын құруды, функционалдауды және дамытуды қамтымасыз етуші мамандардың қатысуынсыз мүмкін емес. Мамандардың мұндай тобы мәліметтер базасының

администраторлары (МБА) деп аталады. Мамандардың бұл тобы мәліметтер базасының құраушы бөлігі болып есептелінеді.

2.1.3 Мәліметтер базасы әкімшілігінің функциялары

Мәліметтер базасының администраторларының тобының құрамында олардың атқаратын функцияларына байланысты әртүрлі топшаларды бөліп көрсетуге болады. Әкімшілік топтың құрамының саны, олардың атқаратын қызметі мәліметтер базасының масштабына, онда сақталатын ақпараттың спецификасына, мәліметтер базасының түріне, қолданыстағы бағдарламалық жабдықтар ерекшеліктеріне және кейбір басқа факторларға белгілі-бір дәрежеде тәуелді болады.

Мәліметтер базасының әкімшілігінің құрамында жүйелік аналитиктер, мәліметтер базасына қатысы бойынша сыртқы ақпараттық қамсыздандыруды жобалаушылар, мәліметтерді өңдеудің технологиялық процесстерінің жобалаушылары, жүйелік және қолданбалы программистер, операторлар, техникалық қамсыздандыру бойынша мамандар болуы керек. Егер әңгіме коммерциялық мәліметтер базасы жөнінде болса, онда бұл жерде маңызды рольді маркетинг бойынша мамандар ойнайды.

Мәліметтер базасының администраторлары жан-жақты функциялардың үлкен шеңберін орындайды. Әрі қарай олардың кейбіреуін қарастырып өтуге болады. Бұл функциялардың негізгілері келесідей:

1) пәндік облыс талдауы: пәндік облысты жазбаша суреттеу, толықтылықты шектеуді табу, ақпарат статусын анықтау, пайдаланушылар қажеттіліктерін анықтау, пайдаланушылар статусын анықтау, "мәліметтер – пайдаланушы" сәйкестігін анықтау, мәліметтерді өңдеудің уақыттық - көлемдік мінездемесін анықтау.

2) мәліметтер базасының құрылымын жобалау: мәліметтер базасының файлдарының құрамы мен құрылымын анықтау, олардың арасындағы байланыстарды анықтау, мәліметтерді тәртіптеу әдістері мен ақпаратқа енуі әдістерін таңдау, мәліметтер базасы мен мәліметтерді жазбаша суреттеу тілдерінің құрылымын жазбаша суреттеу.

3) мәліметтер базасы мен МБ-н өңдеу шараларының құрылымын жазбаша суреттеу кезінде толықтылықты шектеуді беру: пәндік аумаққа жататын толықтылықты шектеуді беру, мәліметтер базасының құрылымымен шақырылған толықтылықты шектеуді анықтау, мәліметтерді енгізу мен түзетулерде МБ-ң толықтылығын қамтамасыз ететін шараларды құру, көп пайдаланушылар режимінде пайдаланушылардың параллельді жұмысында толықтылықты шектеуді қамтамасыз ету.

4) мәліметтер базасын алғашқы жүктеу мен жүргізу: мәліметтер базасын алғашқы жүктеу мен жүргізу технологиясын жасау, енгізу формаларын жобалау, мәліметтерді дайындау, енгізу және енгізуді бақылау.

5) мәліметтерді қорғау:

– жүйеге пароль арқылы кіруді қамтымасыз ету: пайдаланушыларды тіркеу, паролдарды тағайындау және өзгерту.

– Нақты мәліметтерді қорғауды қамтымасыз ету: пайдаланушылар топтарының және жеке пайдаланушылардың ену құқықтарын анықтау, жеке пайдаланушылар үшін мәліметтерге жасалатын рұқсат етілген операцияларды анықтау, мәліметтерді қорғаудың программалық - технологиялық жабдықтарын таңдау/құру.

– мәліметтерді қорғау жабдықтарын тестілеу.

– ақпаратқа енудің қарастырылмаған жолдарын анықтау/тіркеу.

– мәліметтерді қорғауды бұзудың пайда болған жағдайларын зерттеу және оларды жою мен алдын-алу бойынша іс-шаралар жүргізу.

6) мб-н қайта қалыптастыруды қамтымасыз ету: МБ-н қайта қалыптастырудың программалық-технологиялық жабдықтарын құру, жүйелік журналдарды жүргізуді ұйымдастыру.

7) мб-на пайдаланушылардың жолдауының талдауы: пайдаланушылардың МБ-на жолдауының статистикасын жинау, оны сақтау және талдау.

8) мб-ның жұмыс жасау тиімділігінің талдауы мен жүйенің дамуы: жүйенің жұмыс жасау көрсеткіштерінің талдауы (өңдеу уақыты, жадының көлемі, құндық көрсеткіштер), мәліметтер базасын қайта ұйымдастыру мен қайта құрылымдау, мәліметтер базасының құрамын өзгерту, программалық және техникалық жабдықтарды дамыту.

9) пайдаланушылармен жұмыс жасау: пәндік облыстағы өзгерістер туралы ақпарат жинау, пайдаланушылардың МБ-ң жұмысын бағалау туралы ақпарат жинау, пайдаланушыларды оқыту, пайдаланушыларға кеңес беру.

10) жүйелік программалық жабдықтарды дайындау және қолдау: МББЖ және ҚПП туралы ақпарат жинау және талдау, бағдарламалық жабдықтарды иемдену, оларды орналастыру, жұмысқа жарамдылығын тексеру, жүйелік кітапханаларды қолдау, программалық жабдықтарды дамыту.

11) ұйымдастырушылық-әдістемелік жұмыс: МБ-н жобалау әдістемесін таңдау немесе құру, жүйенің даму бағыттарын және мақсаттарын анықтау, МБ-ң даму кезеңдерін жоспарлау, ұйымдастырушылық-әдістемелік материалдарды жасау және шығару.

Әдебиеттерде жиі-жиі пайдаланушылардың тәуелсіз категориясы ретінде қолданбалы программистерді бөліп көрсетеді. Мамандардың бұл тобын ерікті категорияға қоспау керектігі ойластырылуда. Одан да оларды мәліметтер базасының администраторларының құрамына қосу керек.

2.1.4 Мәліметтер базасы администраторының байланыстары

МБ-ң администраторы өз қызметінің барысында мәліметтер базасының басқа пайдаланушыларымен, сонымен қатар МБ-ң пайдаланушылары болып табылмайтын "сыртқы" мамандармен де қарым-қатынасқа түседі.

Ең алдымен егер мәліметтер базасы қандай да бір кәсіпорынды немесе ұйымды ақпараттық қамтымасыз ету үшін құрылса, онда осы ұйымның

әкімшілігімен қарым-қатынас қажет. Жоғарыда көрсетілгендей МБ-сын енгізу ақпаратты өңдеу жүйесінде ғана емес, сонымен бірге ұйымды басқарудың бүтіндей жүйесіне де өзгеріс енгізеді. Бұндай үлкен жобалардың ұйым басшылығының белсенді түрде араласуынсыз және қолдауынсыз орындалмайтыны анық. Ұйым басшылығы МБ-мен ұсынылатын мүмкіндіктермен танысуы керек, оның басымдылықтары мен жеткіліксіздіктері туралы хабарлануы тиіс, сонымен қатар МБ-ның құрылуы мен функционалдануы барысында пайда болатын мәселелер туралы ақпарат алуы тиіс.

Мәліметтер базасы пәндік облыстың динамикалық ақпараттық көрінісі болып табылатындықтан МБ-ның администраторлары өз кезегінде ақпараттық жүйе құрылатын объектінің даму болашағы туралы уақытында хабарлануы тиіс.

Ұйым басшылығы мен МБ-ның администраторы МБ-н құрудың негізгі бағыттары мен уақыты және оның дамуы, пайдаланушыларды қосу кезектілігі туралы келісуі қажет.

МБА-ның МБ-ның бүкіл өмірлік циклінің кезеңдеріндегі өте тығыз байланыс түпкілікті пайдаланушылармен байқалады. Бұл қарым-қатынас пайдаланушылардың қажеттіліктері зерттелген кездегі, пәндік облыстың ерекшеліктері анықталған кездегі жүйені жобалаудың алғашқы кезеңдерінде басталады және жобалау процесі мен жүйенің функционалдануы кезінде де қолдау табады.

Соңғы уақыттарда мәліметтер базасының түпкілікті пайдаланушылары мен администраторлары арасында функцияларды қайта бөлу жүріп жатқандығын айта кеткен жөн. Бұл ең алдымен түпкілікті пайдаланушыларға бағытталған тілдік және программалық жабдықтардың дамуымен байланысты. Бұған қарапайым және соған қоса күшті сұрау салу тілдері, сонымен қатар жобалауды автоматтандыру жабдықтары жатады.

Егер мәліметтер базасы қандай-да бір оны қосушы автоматтандырылған ақпараттық жүйенің (мысалы, АБЖ-да) құрамында функционалданса, онда МБА-а бұл жүйеде мәліметтерді өңдеу бойынша мамандармен қарым-қатынаста жұмыс істеуі қажет.

МБ-ның администраторлары оған қатысты мамандардың сыртқы топтарымен де және ең алдымен МББЖ және ҚПП жеткізушілерімен, басқа МБ администраторларымен қарым-қатынасқа түседі.

МБ-сын мамандандырылған жобалық ұжымдармен бүтіндей немесе МБ жобалаудың өзіндік нысаны ретінде жиі құрылады. Бұл жағдайда МБ-ның әкімшілік қызметі ұйымдық-өндіруші ретінде де, ұйымдық-тапсырыс беруші ретінде де құрылуы керек.

2.2 Мәліметтер базасын басқару теориясы

Мәліметтер базасын басқару теориясы дербес пән ретінде шамамен алғанда ХХ ғасырдың 50-жылдарының басынан бері дами бастады. Осы уақыт

ішінде белгілі бір іргелі ұғымдар жүйесі калыптасты. Олардың бірқатарын келтірейік:

- объект деп мағлұматтары мәліметтер базасында сақталатын ақпараттық жүйенің элементтерін айтамыз.

- атрибут – объектiнiң қасиеттерiн ақпараттық бейнелеу. Әрбір объект атрибуттардың кейбір жиынтығымен сипатталады.

- мәліметтердің басты элементі – деп мәліметтердің басқа элементтерінің мәндерін анықтауға мүмкіндік беретін осындай атрибут аталады.

- алғашқы кілт – объектiнiң (жазбаның) әрбір данасын бірегей түрде жалғастыратын атрибут (немесе атрибуттар тобы).

- екiншiреттегi кiлт - бiрнеше жазбалар (объект даналары) үшiн мәнi қайталанатын атрибут (немесе атрибуттар тобы). Ең алдымен, екiншi реттегi кiлттер жазбаларды iздестiру операцияларында қолданылады.

МББЖ – жаңа база құрылымын құруға арналған программалық құралдар кешені, оның құрамын толықтыру, құрамында барды редакциялау және аппаратпен көрсетушілік (визуализация). База ақпаратын көрсетушілік ретінде берілген критериге сәйкес бейнеленетін мәліметтерді іріктеу, оларды реттеу, сосын шығаратын құрылғыға беру немесе байланыс арналары бойынша беру.

Кез келген МББЖ мәліметтермен төрт қарапайым операция орындауға мүмкіндік береді:

- кестеге бір немесе бірнеше жазбаны қосу;
- кестеден бір немесе бірнеше жазбаны жою;
- кейбір өрістердің бір немесе бірнеше жазбаларындағы мәндерді жаңарту;

- берілген шартты қанағаттандыратын бір немесе бірнеше жазбаларды табу.

МББЖ-ның тағы бір функциясы – мәліметтерді басқару. Мәліметтерді басқару ретінде, әдетте, мәліметтерді рұқсат етілмей қол жетуден қорғау, мәліметтермен жұмыс режимін көп мәрте пайдалануды қолдау және мәліметтердің тұтастығы мен үйлесімділігін қамтамасыз ету түсініледі.

3 Мәліметтер қорын жобалау және оны іске асыру

3.1 MySQL мәліметтер базасы

MySQL – SQL мәліметтер базасының сервері, Интернет сайттарының SQL-қозғалтқышы ретінде қолданушылар қасиеттеріне сәйкестенуі арқасында және техникалық жақсы қолдауымен әуелгі кодына байланысты көп таралған. Сервер тегін және әуелгі кодтарға қойыла береді. Әуелгі кодтар, көп платформалардың астына компилирленген модульдер және түрлі утилиттерін мына сайттан табуға болады <http://www.mysql.com/>.

MySQL – реляциялық мәліметтер базасын басқаратын жүйе. Реляциялық мәліметтер базасында мәліметтер үйіліп жиналмайды, олар жекелеген кестелерде сақталады, соған байланысты жылдам әрі ыңғайлылығымен тиімді болып табылады. Кестелер бір-бірімен қатынастар арқылы байланысады, соған байланысты сұраныс кезінде мәліметтерді бірнеше кестелерге біріктіре алады. SQL MySQL жүйесінің бір бөлігі ретінде сұраныстарға құрылымдандырылған тіл ретінде және мәліметтер базасына қол жеткізу үшін қолданылатын стандартты таралған тіл ретінде де сипаттауға болады.

Алғашында MySQL сервері үлкен көлемді мәліметтер базасын басқару үшін жұмыс жылдамдығын жоғарлату мақсатымен құрылған болатын. Бірнеше жылдар ішінде бұл сервер өндірістік эксплуатация жоғарғы қарқындылығымен дамуда. MySQL үнемі дамуына қарамастан, бүгінгі таңда кең пайдалы функциялар спектрін қамтымасыз етуде. Қол жетімділігіне байланысты, жылдамдығы мен қауіпсіздігі MySQL Интернеттегі мәліметтер базасына қолдану өте ыңғайлы болып табылады.

MySQL жүйесі дамыған артықшылыққа ие, ол мәліметтер базасын, кестелерін және кестелердің бөлек өрістерін қолдану құқығын береді. Сонымен қатар қолданушыларды аты-жөні бойынша ғана емес, сонымен қатар желілік адрестері бойынша да дифференциалдауға болады. Түрлі қолданушыларға түрлі артықшылық жиынтығы қол жетімді, ол артықшылықтың 14 типі бар. Бөлек өрістерге артықшылықтар мәліметтерді түрлі қызметтер үшін нақтырақ басқаруға мүмкіншілік береді.

Жылдамдығы бойынша MySQL SQL серверлерінің ішінен көшбасшы болып табылады. MySQL кодтарының ашықтығына байланысты қазіргі кездегі барлық платформаларды қолдайды. PHP тілі үшін MySQL негізгі стандарт ретінде орын алады.

3.2 ER – диаграмма құру

Мәліметтер базасының оптималды құрылымы ақпараттың қайталануына жол бермейді, мәліметтерді өңдеуді жылдамдатады. Мәліметтердің реляциялық құрылымына «мән-байланыс» («сущность-связь») модельін қолдануға болады. Көптеген құрастырушылар мен аналитиктер бұл модельді

мәліметтерді физикалық іске асу тұрғысынан логикалық тұрғыға келтіруде қайталанбас әдіс деп көрсетеді. ER-диаграмма жүйенің ақпараттық аспектісін сипаттайды (жүйедегі мәліметтердің құрылымын анықтайды).

Мәліметтер базасының логикалық моделін құрғаннан кейін (ER-диаграмманы) оның физикалық іске асырылуын жүзеге асырамыз. Ол үшін мәліметтің логикалық моделін дұрыстауымыз керек. Оқу жоспарын өңдегенге арналған интерфейске сәйкес мәліметтер базасы үшін мән-байланыс модельін құрамыз. Бұл модельдегі ортақ объектілердің бірі ретінде пәнді алатын болсақ, әр пән, қарапайым атрибуттардан басқа, кафедра, оқытушы, мамандық объектілерімен байланысады. Байланыстың түрі – «көптің – біреуге» қатынасы. Әр кестелер үшін қысқаша жазғанда «көптің – біреуге», «біреудің-біреуге» қатынасытары арқылы байланысады. Нәтижесінде төменде көрсетілген мәліметтер базасындағы байланысқан кестелердің схемасын құраймыз.

3.3 Реляциялық модельді құру

Реляциялық мәліметтер базасының құрылымын схемадағыдай құру керек, яғни ER-модельді реляциялық модельге келтіру керек:

- алынған қатынастарға сәйкес, мәліметтер базасының кестелерінің құрылымын жобалаймыз. Кестелердің құрылымында атауы, типтері, өрістің өлшемі, кілттік өрістерді белгілеу төмендегі кестелерде келтірілген.

- университет (уникалды университет кодынан, университеттің толық атауынан, адресі және телефонынан тұрады).

- факультет (факультеттің уникалды кодынан, факультет атауынан, қай жерінде орналасқан адресінен, телефон және университеттің кодынан құралған).

- кафедра (кафедраның коды, атауы, орналасқан орны, телефон, қай факультетке қатысты екендігі болатын факультет кодынан тұрады).

- мамандық (мамандықтың уникалды коды, атауы, бөлімше және қай кафедраға қатысты екендігі бар кафедр кодынан тұрады).

- оқытушылар (әр оқытушыға уникалды кодынан, аты-жөні, телефонынан, қызметінен (немесе дәрежесінен) құралған).

- аудитория (аудиторияның уникалды коды, атауы, аудитория типінің коды, көлемі, қай кафедра көбіне сабақ жүргізетіндігі болатын кафедраның коды көрсетілуі керек).

- пән (пәннің уникалды коды, толық атауы, қай кафедра оқитындығы көрсететін кафедраның кодынан құрылады).

- пән блогы (пән блогының уникалды коды, блогтың атауы).

- группа (группаның кодынан, атынан, қай бөлімнен екендігі, нешінші курста екендігі, эдвйзерінің кодынан, студенттер санынан және кафедраның кодынан тұрады).

- аудитория түрлері (аудитория түрінің коды және аталынуы).

– жұмыс оқу жоспары (ЖОЖ) (ЖОЖ-дың уникалды коды, ЖОЖ-дың аталынуы, ЖОЖ-да құрылған кестелердің аталынуы).

– жаңа кесте (пән блогының уникалды коды, пәннің уникалды коды, кредиттер саны, барлық сағат, дәріс сағаты, лабораториялық сабақ сағаты, семинар сағаты, магистранттың оқытушымен өзіндік жұмыс сағаты, магистранттың өзіндік жұмыс сағаты, бақылау формасы, пәнді беретін оқытушының аты-жөні, қай кафедраға қатысты екені).

3.4 MySQL-де мәліметтер базасын құру мен мәліметтер базасын толтыру

Клиенттік бөлікте MySQL-файлын C:\MySQL\BIN\mysql.exe қосып, MySQL командасын жазуға болатын консольды қосымша ашылады. Мәліметтер базасын құру үшін төмендегі қадамдарды орындау қажет:

1) `create database gur_mag;` – командасын теру арқылы «gur_mag» атты мәліметтер базасын құру;

2) `use gur_mag;` – командасы арқылы мәліметтер базасын таңдап алу;

3) браузерді қосып, localhost – локальды желісін таңдап, оның ішінде phpMyAdmin папкасын таңдау арқылы, phpMyAdmin қосымшасына ену;

4) phpMyAdmin қосымшасынан «gur_mag» МБ-ны таңдап, арнайы меню арқылы кесте құру.

Құрайтын мәліметтер жүйесінде 11 кесте болатындығын жоспарланған. МБ-ның кестелерінің көрінісі төмендегі суретте келтірілген:

К е с т е 3.1 – koreya_2bludo қатынасы

Өріс аты	Атрибут аты	Мәліметтер типі	Ескерту
Тағам коды	id	int	Алғашқы кілт
Тағам аты	name	varchar(50)	Қажетті өріс
Тағам бағасы	sena	int	Қажетті өріс
Калькуляция коды	kal_id	int	Қажетті өріс

К е с т е 3.2 – koreya_zakuski қатынасы

Өріс аты	Атрибут аты	Мәліметтер типі	Ескерту
Тағам коды	id	int	Алғашқы кілт
Тағам аты	name	varchar(50)	Қажетті өріс
Тағам бағасы	sena	int	Қажетті өріс
Калькуляция коды	kal_id	int	Қажетті өріс

К е с т е 3.3 – supу қатынасы

Әріс аты	Атрибут аты	Мәліметтер типі	Ескерту
Тағам коды	id	int	Алғашқы кілт
Тағам аты	name	varchar(50)	Қажетті өріс
Тағам бағасы	sena	int	Қажетті өріс
Калькуляция коды	kal_id	int	Қажетті өріс

К е с т е 3.4 – salaty қатынасы

Әріс аты	Атрибут аты	Мәліметтер типі	Ескерту
Тағам коды	id	int	Алғашқы кілт
Тағам аты	name	varchar(50)	Қажетті өріс
Тағам бағасы	sena	int	Қажетті өріс
Калькуляция коды	kal_id	int	Қажетті өріс

К е с т е 3.5 – zakuski қатынасы

Әріс аты	Атрибут аты	Мәліметтер типі	Ескерту
Тағам коды	id	int	Алғашқы кілт
Тағам аты	name	varchar(50)	Қажетті өріс
Тағам бағасы	sena	int	Қажетті өріс
Калькуляция коды	kal_id	int	Қажетті өріс

К е с т е 3.6 – govjadina қатынасы

Әріс аты	Атрибут аты	Мәліметтер типі	Ескерту
Тағам коды	id	int	Алғашқы кілт
Тағам аты	name	varchar(50)	Қажетті өріс
Тағам бағасы	sena	int	Қажетті өріс
Калькуляция коды	kal_id	int	Қажетті өріс

К е с т е 3.7 – kurica қатынасы

Әріс аты	Атрибут аты	Мәліметтер типі	Ескерту
Тағам коды	id	int	Алғашқы кілт
Тағам аты	name	varchar(50)	Қажетті өріс
Тағам бағасы	sena	int	Қажетті өріс
Калькуляция коды	kal_id	int	Қажетті өріс

К е с т е 3.8 – garnir қатынасы

Әріс аты	Атрибут аты	Мәліметтер типі	Ескерту
Тағам коды	id	int	Алғашқы кілт
Тағам аты	name	varchar(50)	Қажетті өріс
Тағам бағасы	sena	int	Қажетті өріс
Калькуляция коды	kal_id	int	Қажетті өріс

К е с т е 3.9 – pub_system қатынасы

Өріс аты	Атрибут аты	Мәліметтер типі	Ескерту
Код	id	int	Алғашқы кілт
Кухня коды	kuhnya_id	int	Қажетті өріс
Бар коды	bar_id	int	Қажетті өріс
Акциялар коды	akcii_id	int	Қажетті өріс

К е с т е 3.10 – paritki қатынасы

Өріс аты	Атрибут аты	Мәліметтер типі	Ескерту
Сусын коды	id	int	Алғашқы кілт
Сусын аты	name	varchar(50)	Қажетті өріс
Сусын бағасы	sena	int	Қажетті өріс
Калькуляция коды	kal_id	int	Қажетті өріс

К е с т е 3.11 – viski қатынасы

Өріс аты	Атрибут аты	Мәліметтер типі	Ескерту
Сусын коды	id	int	Алғашқы кілт
Сусын аты	name	varchar(50)	Қажетті өріс
Сусын бағасы	sena	int	Қажетті өріс
Калькуляция коды	kal_id	int	Қажетті өріс

К е с т е 3.12 – alcohol қатынасы

Өріс аты	Атрибут аты	Мәліметтер типі	Ескерту
Сусын коды	id	int	Алғашқы кілт
Сусын аты	name	varchar(50)	Қажетті өріс
Сусын бағасы	sena	int	Қажетті өріс
Калькуляция коды	kal_id	int	Қажетті өріс

К е с т е 3.13– zakaznoi_list қатынасы

Өріс аты	Атрибут аты	Мәліметтер типі	Ескерту
Тапсырыс коды	id	int	Алғашқы кілт
Тапсырыс аты	name	varchar(50)	Қажетті өріс
Тапсырыс саны	kolichestvo	int	Қажетті өріс

К е с т е 3.14– zakazy_na_den қатынасы

Өріс аты	Атрибут аты	Мәліметтер типі	Ескерту
Тапсырыс коды	id	int	Алғашқы кілт
Тапсырыс аты	name	varchar(50)	Қажетті өріс
Тапсырыс саны	kolichestvo	int	Қажетті өріс

К е с т е 3.15– kalkulyaciya қатынасы

Өріс аты	Атрибут аты	Мәліметтер типі	Ескерту
Калькуляция коды	id	int	Алғашқы кілт
Тағам аты	name	varchar(50)	Қажетті өріс
Тапсырыс саны	obiem	int	Қажетті өріс

К е с т е 3.16– ostatok қатынасы

Өріс аты	Атрибут аты	Мәліметтер типі	Ескерту
Тапсырыс коды	id	int	Алғашқы кілт
Тапсырыс аты	name	varchar(50)	Қажетті өріс
Тапсырыс саны	kolichestvo	int	Қажетті өріс

К е с т е 3.17– efes қатынасы

Өріс аты	Атрибут аты	Мәліметтер типі	Ескерту
Сусын коды	id	int	Алғашқы кілт
Калькуляция	kal_id	varchar(50)	Қажетті өріс
Тапсырыс саны	kolichestvo	int	Қажетті өріс

К е с т е 3.18– grecheskii қатынасы

Өріс аты	Атрибут аты	Мәліметтер типі	Ескерту
Тағам коды	id	int	Алғашқы кілт
Калькуляция	kal_id	varchar(50)	Қажетті өріс
Тапсырыс саны	kolichestvo	int	Қажетті өріс

К е с т е 3.19– svejii_salat қатынасы

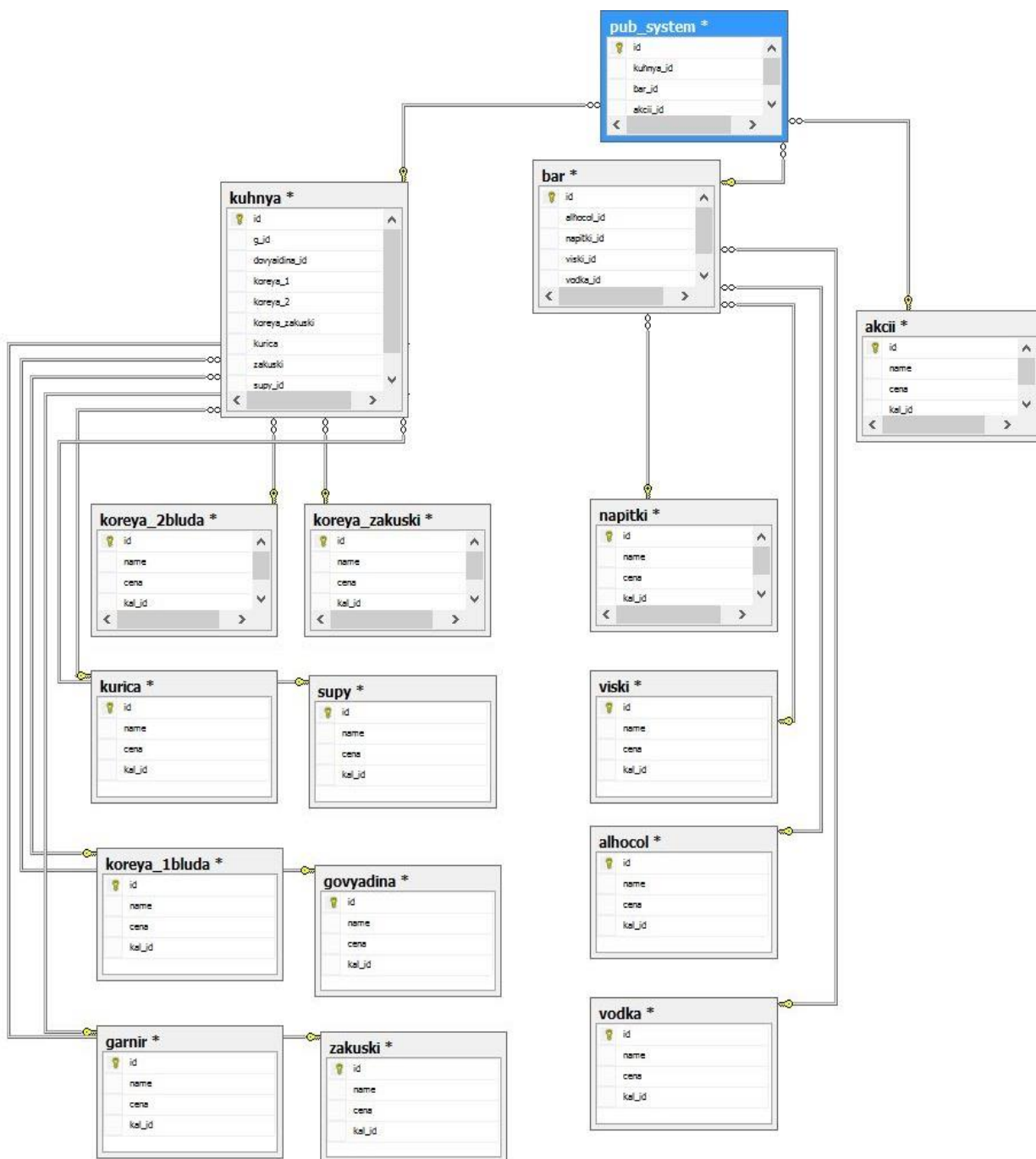
Өріс аты	Атрибут аты	Мәліметтер типі	Ескерту
Тағам коды	id	int	Алғашқы кілт
Калькуляция	kal_id	varchar(50)	Қажетті өріс
Тапсырыс саны	kolichestvo	int	Қажетті өріс

К е с т е 3.20– vodk қатынасы

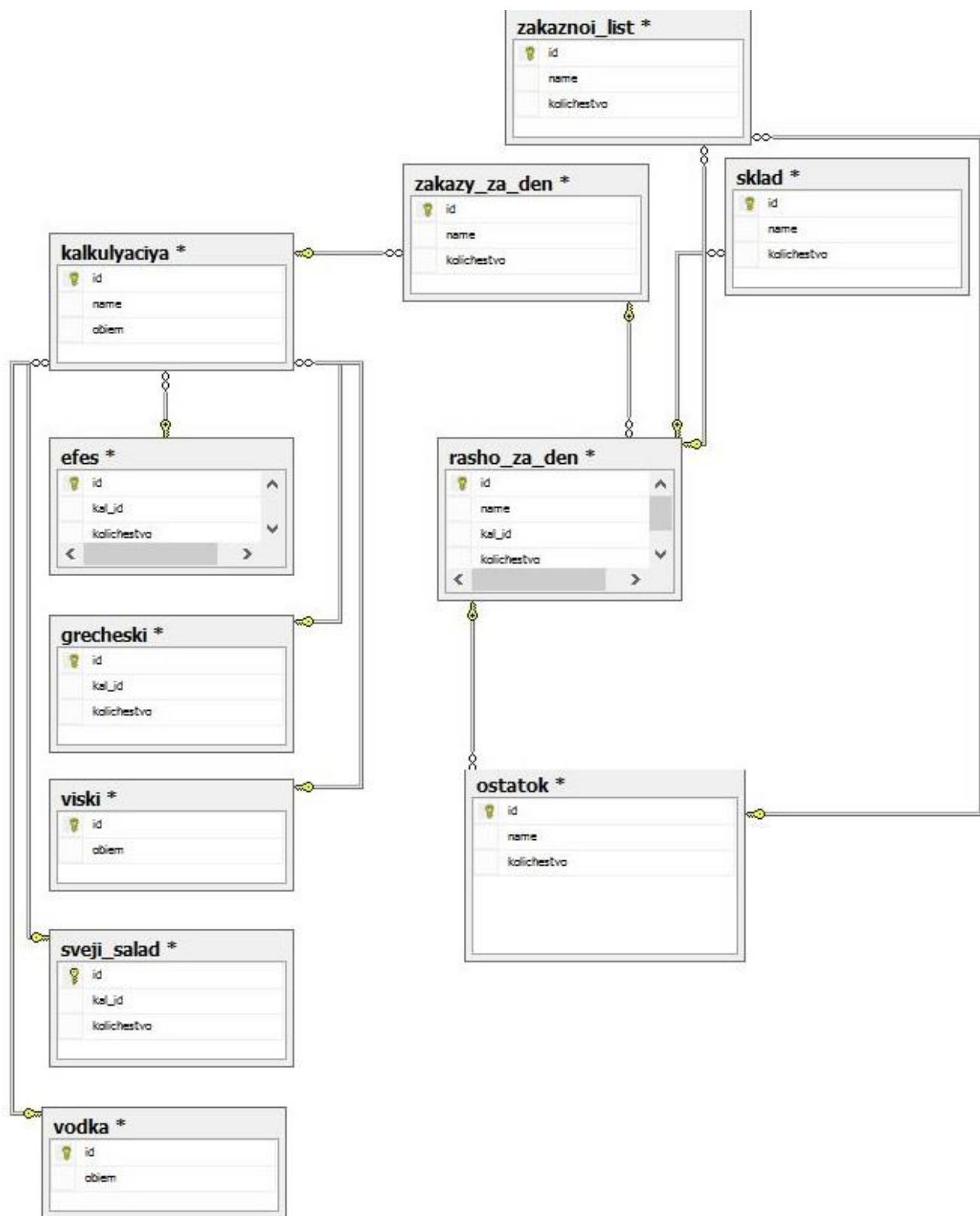
Өріс аты	Атрибут аты	Мәліметтер типі	Ескерту
Сусын коды	id	int	Алғашқы кілт
Саны	Obiem	int	Қажетті өріс

3.5 МҚ диаграммасы

Алдыңғы кестелердегі мәліметтер МҚ-да қалай байланысқандығы төмендегі суреттерде келтірілген:



3.1-сурет – Бірінші МҚ диаграммасы



3.2-сурет – Екінші МҚ диаграммасы

4 Физикалық жобалау

4.1 Delphi программалау ортасы

4.1.1 Delphi туралы жалпы түсінік

Жергілікті мәліметтер қорының көпшілігі тек бір ғана кестені сақтауы мүмкін (dBase, Paradox, XML). Бірақ та бір файлда бірнеше кестелер (мысалы, осы тарауда қарастыратын Access) бекітілген жергілікті қорлар өкілдері де кездеседі.

Жергілікті мәліметтер қорынан біз кең таралған реляциялық мәліметтер қорларын ғана қарастырамыз. Реляциялық мәліметтер қоры дегеніміз не? Ол бағандары ретінде ондағы сақталатын мәліметтер атауы, ал жолдары мәліметтердің өзін сақталатын кесте. Мәліметтер қоры кестесі Excel электрондық кестесіне ұқсас (дәлірек айтсақ, Excel өз мәліметтерін мәліметтер қоры технологиясы негізінде құрылған өзіндік пішімде сақтайды). Мәліметтер қорының жергілікті қатты дискіде немесе орталықтандырылған файлдық сервердің желілік дискісіне сақталынуы мүмкін. Бұл файлдарды кез келген басқа файл сияқты стандартты тәсілдердің бірін қолдана отырып көшірмесін алуға болады, өйткені мәліметтер қорының арнайы орынға байланысқан емес. Ең бастысы, бағдарлама өз кестесін тапса болғаны.

Әрбір кестеде жолды бірімәнді идентификациялайтын бір ерекше өріс болуы тиіс. Ол өріс кілттік өріс деп аталады. Бұл өрістер бірнеше кестелерді байланыстыру үшін қажет болады. Егер де біздің кестеміз байланыспаса да кілттік өріс міндетті түрде қажет. Айталық, біз телефон туралы мәліметтер қорын жазып отырмыз делік. Бізде «Айжандар»-дың бірнешеуі кездеседі. Оларды қалай ажыратамыз? Міне бұл жерде бізге кілт көмектеседі. Кілт негізінде сандық типті және егер мәліметтер қоры мүмкіндігі болса "autoincrement" (автоматты үлкейіп/кішереyetін сан немесе санағыш) қолданған дұрыс.

Мәліметтер қоры кестесіндегі бағандар атаулары да сондай-ақ ерекше болуы керек, бірақ бұл жерде міндетті түрде сандық емес. Әрбір баған (мәліметтер қоры өрісі) міндетті түрде анықталған типті болуы керек. Типтердің саны және олардың түрлері мәліметтер қорының типіне байланысты, мысалы dBase пішімі (dBF кеңейткішті файлдар) 6 типті ғана қолдайды, ал Paradox 15-ке дейін.

Мәліметтер қоры бір файлда (Access) немесе бірнешеуінде сақталынуы мүмкін (Paradox, dBase). Дәлірек айтсақ, кестелер мәліметтері әрдайым бір файлда сақталынады, ал қосымша ақпарат жекелеген файлдарда орналасуы мүмкін. Қосымша ақпарат ретінде индекстер, шектеулер немесе үнсіз келісім бойынша нақты өрістер үшін мәндер тізімі. Егер де тым болмаса бір файл

бұзылса немесе жойылып кетсе мәліметтер редакциялауға жарамсыз болуы мүмкін.

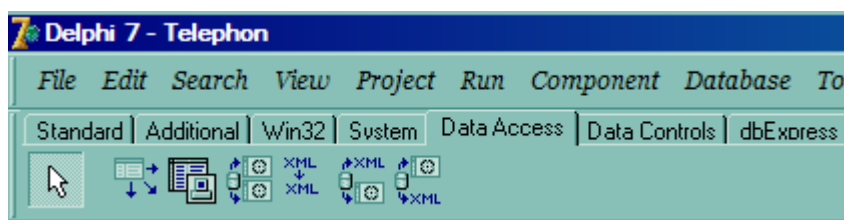
Кестелердің мәліметтері қандай да бір өзгертулерге өте жиі әкелуі мүмкін, өйткені қайсыбір жолды түзетпес бұрын оны тауып алу қажет. Анықтамалықтар негізінде қолданылатын статистикалық кестелерде сұралатын мәліметтерді шығару алдында оны іздеу амалдарына әкеліп соқтырады. Егер кесте өте көп жолдардан тұрса іздеу операциясы қыруар жұмысты алады. Индекстерді бұл процедураны тездетуге арналған, сондай-ақ сұрыптау кезінде жіберу нүктесі ретінде де қолданылуы мүмкін. Аталған кезеңде индекстелмеген өрісті реттеу мүмкін емес екендігін білу жеткілікті.

Егер бізге қандай да бір кестені «Фамилия» өрісі бойынша реттеу керек болса, оны алдымен индекстеп алу қажет. Содан кейін ғана көрсету кесте осындай индекспен істейтінін және автоматты түрде сұрыпталатынын көрсету керек.

4.2 Delphi программалау ортасында мәліметтерге қатынаудың негізгі құрауыштары

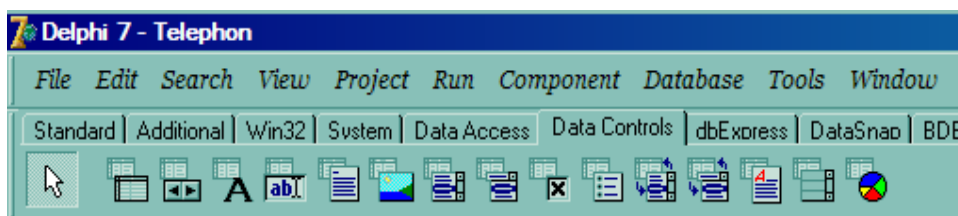
Өз өнімдеріне ADO технологиясын бейімдеген Microsoft – қа қарағанда Borland фирмасы бізге әр түрлі технологиялар арқылы жұмыс істейтін түрлі құралдарды берді және тек қана онымен шектеліп қалған жоқ. Мұндай жағдай басқа программистер алдында үлкен абырой әкеледі. Бұған қоса кез келген уақытта қолданылуы мүмкін топтар да бар, бұл жерде мен бізге жеткілікті құралдарға шолу жасау керек.

Data Access бетбелгісінде мәліметтерге қатынаудың негізгі құрауыштары орналасқан. Бұл құрауыштар барлығына ортақ және басқа да құрауыштар топтарымен бірлесіп қолданылуы мүмкін.



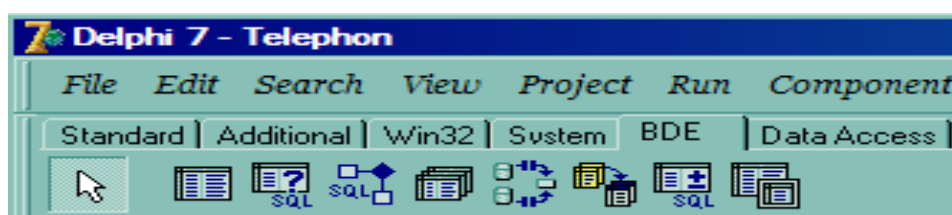
4.1-сурет – Құрауыштар палитрасының Data Access бетбелгісі

Data Controls бетбелгісінде кестелердегі мәліметтерді шығару және редакциялау үшін қажетті құрауыштар орналасқан. Бұл құрауыштар да сол сияқты мәліметтерге қатынаудың қолданылатын технологиясынан тәуелсіз қолданылуы мүмкін.



4.2-сурет – Құрауыштар палитрасының Data Controls бетбелгісі

BDE бетбелгісі мәліметтер қорына қатынауға мүмкіндік туғызатын Borland фирмасымен құрастырылған Borland Database Engine атауы бар технологияға құрауыштардан тұрады. Бұл технология ескіріп кетті, бұрынғы нұсқалары үшін үйлесімділікке қойылады. Осыған қарамастан ол мәліметтер қорының Paradox, dBase сияқты ескі типтерімен жақсы жұмыс істейді.

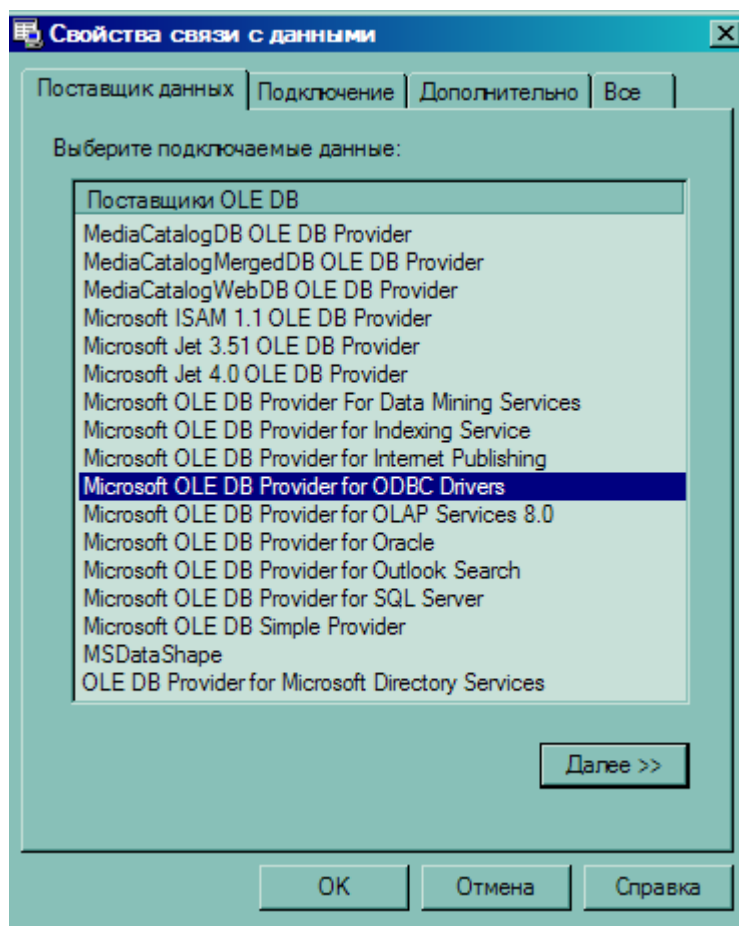


4.3-сурет – Құрауыштар палитрасының BDE бетбелгісі

DBExpress – Borland фирмасының мәліметтерге қатынаудың жаңа технологиясы. Ол ерекшеленеді және мәліметтер қоры қолданатын клиент серверлік қосымшаларды бағдарламалауға арналған. Бір бетбелгілі құрауыштарды серверлік технология бойынша құрылған Oracle, DB2 немесе MySQL сияқты мәліметтер қорына қолдану керек.

ActiveX Data Objects (ADO) технологиясы. Delphi – де мәліметтер қорымен жұмыс істеудің альтернативті мүмкіндігі енгізілген. Бұл Microsoft фирмасында құрылған Activex Data Objects (ADO) технологиясы. ADO- бұл реляциялық және реляциялық емес МҚ электрондық пошта, жүйелі, мәтінді, графикалық файлдардан тұратын мәліметтерден кез келген типіне сәйкес келетін қолданушы интерфейсі мәліметтермен байланысу OLEDB технологиясының қатысуымен жүзеге асырылады.

ADO – ны қолдану мәліметтермен жұмыс істеуге қамтамасыз етеді. ADO – мен жұмыс істеу үшін компьютерде ADO 2.1 және одан жоғары нұсқалы жүйесі орнатылуы қажет.



4.6-сурет – Қосылу жолын құру терезесі

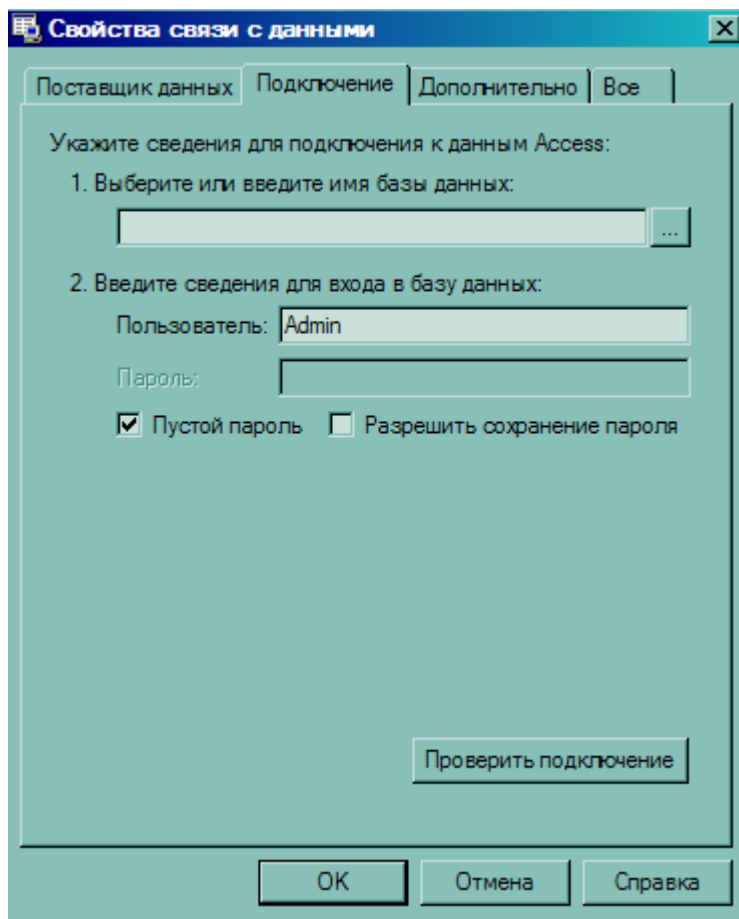
Provider бетбелгісінде мәліметтер қорына қатынау мүмкіндігі бар бар ADO драйверлер көрсетілген. Егер қандай да бір драйвер жоқ болса, онда үнсіздікте ерекшеленген «Microsoft OLE DB Provider for ODBC Drivers»-ті қолдану болады. Бұл драйвер ODBC драйвер арқылы мәліметтер қорына қатынас жасауға мүмкіндік береді. Біздің жағдайымызда MS Access мәліметтер қорына қатынауға «Microsoft Jet OLE DB Provider» қолданылады. Мұндай драйвер машинаға міндетті түрде MS Office –мен бірге орнатылады, ал Windows-тың соңғы нұсқаларында ол үнсіздіктен орнатылады.

Жалпы компьютерлерде бұл драйвердің екі нұсқасы да орнатылған, сондықтан біз оның жаңасы - «Microsoft Jet 4.0 OLE DB Provider» -ді таңдаймыз. Бұдан кейін Next батырмасына басамыз немесе «Connection» бетбелгісіне өтеміз .

«Connection» бетбелгісінің түрі таңдалған драйверге байланысты болады. Біздің жағдайымызда 4.7-суретінде көрсетілгендей болады.

Ең алдымен бұл терезеде мәліметтер қорының атауын (қажет болса жолды да) «Select or enter a database name» жолына енгізу қажет. Егер мәліметтер қоры жүктелетін файлмен бір директорияда орналасатын болса жолды көрсетіп қажеті жоқ. Біз мәліметтер қорын жүктелетін файлмен бір директорияда орналастыру туралы кеңес береміз. Егер біз файлдаарды

жүктелетін файлдан бөлек сақтасақ, онда толық жолды көрсету қажет, ал ол бағдарламаны басқа компьютерге көшірген кезде қиындық туғызады. Өйткені бағдарлама мәліметтер қорын көрсетілген жол бойынша іздейді. Егер файлдарды басқа директорияда сақтағымыз келсе, онда ағымдағы директорияға қатысты жолды көрсетеміз.



4.7-сурет – «Connection» бетбелгісі

Мәліметтер қоры файлының жеңіл таңдау үшін енгізу жолының оң жағында орналасқан нүктелерге шерту қажет. Сонымен қатар бізге келесі өрістерді толтыру қажет:

1) қолданушы есімі (User name) үнсіздіктен қоюға болады, егер MS Access –ті құрғанда басқа есім берілмесе;

2) пароль (Password) – егер мәліметтер қорында пароль болса, онда оны көрсету қажет;

3) бос пароль (Blank password) – егер пароль қажет болмас, онда бұл жерде қанатбелгіні орнату орынды;

4) паролды сақтау мүмкіндігін беру (Allow saving password). Егер бұл жерде қанатбелгіні орнатсақ, онда пароль сақталынуы мүмкін.

Мәліметтер қорын таңдағаннан кейін байланысты тестілеу үшін батырмасын бас. Егер барлығы дұрыс көрсетілсе, біз «Test Connection succeeded» хабарламасын көреміз. Қосылу жолын құру терезесін жабу үшін ОК

батырмасын басу керек және қосылу жолының редактор терезесін жабу үшін тағы да бір рет ОК батырмасын басу керек. Енді ADOConnection құрауышының қасиеттерінде LoginPrompt қасиетін алып тастап, false деп орнатайық. Бұл мәліметтер қорына енуде қайта-қайта пароль енгізбес үшін қажет. Ал енді мәліметтер қорымен байланыс орнатылуы үшін Connected қасиетін True деп берелік.

Сонымен, байланысты аяқталған деп есептеуге болады. Енді бізге өзіміз құрған «Справочник» кестесіне қатынау алу керек. Ол үшін формаға құрауыштар палитрасынан ADO бетбелгісінен ADOTable құрауышын қоямыз. Бірден оның Name қасиетін BookName деп өзгертейік.

TADOTable.

Бұл құрауышта да ConnectionString қасиеті бар, оны да тура солай баптауға болады. Неге олай болады? Өйткені, олай жасамас үшін біз формаға ADOConnection құрауышын қойдық. Енді біз BookName құрауышындағы Connection қасиетінде өзіміз құрастырған мәліметтер қорымен байланысу құрауышын көрсете аламыз.

Connection қасиетіндегі төмен созылған тізімді шертіп жалғыз ADOConnection1 пунктін таңда. Енді бізге ConnectionString қасиетін толтырмай-ақ қоюға да болады.

TableName қасиетінде біздің кестеміздің атауын (Справочник) таңдау керек. Кестемен байланыс көрсетілді, енді қосылуға болады. Ол үшін Active қасиетін true деп орнат.

TdataSource.

Кестедегі мәліметтерді бейнелеу үшін формаға құрауыштар палитрасынан Data Access бетбелгісінен DataSource құрауышын орнату керек. Енді бұл құрауышқа қандай кестені бейнелейтінін көрсету керек. Ол үшін DataSet қасиетінде төмен созылған тізімнен біздің BookTable кестесін таңдау қажет. Дайындықтар бітті, мәліметтерді шынайы бейнелеуге кірісуге болады. Кестені бейнелеудің ең қарапайым тәсілі – DBGrid құрауышын орнату.

DBGrid.

Бұл мәліметтерді кесте түрінде бейнелеуге арналған құрауыш-тор. Бұл құрауышта біздің кестеміздің жолдарын қосуға, жоюға және редакциялауға болады.

Және біздің қосымшаны құрудың соңғы кезеңі – құрауыш торды кестені бейнелеу құрауышымен байланыстыру. Ол үшін DBGrid құрауышының DataSource қасиетінде өзіміз құрастырған DataSource құрауышын көрсету керек.

Кату	Фамилиясы	Аты	Телефон	e-mail
1	Асқар	Айжан	83331457887	
2	Әділбек	Айболат	83331245789	
3	Бұқарзада	Боғатөз	83331245856	
4	Наби	Бакытұр	87054178122	
5	Нұрсалқан	Тумен	83331456855	

4.8-сурет – Соңғы нәтижені беретін Delphi ортасындағы кесте

Енді міне біздің қосымшамыз дайын болды. Біз байқағанымыздай бірде-бір жол код жазған жоқпыз. Міне Delphi қандай дәрежеге дейін мәліметтер қорын бағдарламалау үрдісін ықшамдайды, ештеңені бағдарламалаудың қажеті де жоқ. Бұл мысалды жүктеп көрейік және бірнеше жолдар қосып, бұрыннан бар жолдарды редакциялап, жойып көрейік. Жолды қосу үшін Ins пернесін, ал жою үшін Ctrl+Del пернелерінің комбинациясын қолданамыз.

4.3 TADOTable құрауышының қасиеттері

TADOTable құрауышы пайдалы қасиеттер жиынынан тұрады. Олардың көпшілігі қолданыста қарапайым, сондықтан оларды қолдануға мысалдар жиынтығын жазбас үшін бұл жерде мен олардың ең негізгілерін сипаттап кетейік. Келешекте іс жүзінде біз олардың кейбіреуімен танысамыз, ал кейбірі біз үшін ойлануға қосымша ақпарат сияқты болады.

MasterSource – бұл қасиетте ағымдағы кестеге қарағанда негізгісі көрсетіледі. Біз бұл қасиетті байланысқан кестелерді қарастырған кезде толығырақ іс жүзінде де қарастырамыз.

ReadOnly – егер бұл қасиет true-ге тең болса, онда кестені редакциялау мүмкін емес. Бұл жағдайда мәліметтер тек бейнеленеді. Міндетті түрде бұл қасиетті мәліметтер өзгермеуге тиіс және қолданушы оларға ешқандай өзгертулер енгізбейтін кестелерге орнату керек.

TableDirect – бұл қасиет кестеге қатынау қалай жүргізілетінін бейнелейді. Егер бұл параметр true-ге тең болса, кестеге атауы бойынша тікелей қатынау жүргізіледі. Егер false болса онда бізге білінбей мәліметтер қорына арнайы SQL сұранысы (SQL сұраныстар туралы төмменнен оқуға болады) жүргізіледі. Мәліметтер қорының барлығы тікелей байланыс арқылы жұмыс істеуге мүмкіндік бермейді, сондықтан бұл қасиет үнсіздікте false-ге тең.

TableName – біз өндегіміз келетін кесте атауы.

CacheSize – кэш жадының өлшемі. Егер 50 санын орнатса, кестеге ең алғаш қосылғанда құрауыш алғашқы 50 жолды таңдап оны жергілікті жадқа орналастырады, оларға қатынауды тездетеді.

CanModify – ReadOnly қасиетіне ұқсас және кестенің мәліметтерін редакциялауға мүмкіндікті көрсетеді.

CommandTimeout – команданың орындалуын күту уақыты. Құрауыш команданы мәліметтер қорына бағыттаған кезде ол күту таймерін жүктейді, ол тоқтағаннан кейін (команда орындалмаса) қате туралы хабарлама шығады.

Connection – бұл жерде қосылу орындалатын TADOConnection құрауышы көрсетіледі.

ConnectionString – мәліметтер қорына қосылу жолы.

CursorLocation – мәліметтерді есептеп кестедегі ағымдық позицияны көрсететін курсордың орналасуы. Курсор серверде немесе клинет машинасында орналасуы мүмкін.

CursorType – курсор типі. Бұл жерде келесі нұсқалардың бірі болуы мүмкін:

- ctUnspecified курсордың орналасуы көрсетілмеген;
- ctOpenForwardOnly – курсор тек алға ғана жылжи алады;
- ctKeyset мұндай курсорда бір қолданушымен енгізілген өзгертулер бұл кестеге қосылған басқа қолданушыларға көрінбейді. Егер бір кестемен бір уақытта бірнеше қолданушы жұмыс істесе, онда мұндай курсорда басқа қолданушылардың енгізген өзгерістерін бейнелеу үшін мәліметтер қорынан шығып кетіп оған қайтадан қосылу керек;

- ctDynamic динамикалық курсор, мұнда бір қолданушың өзгерістерін басқалары да көріп отырады;

- ctStatic статикалық курсор. Бір қолданушының өзгерістері басқаларына көрінбейді.

Егер курсор клиентте орналасса, онда статикалық курсорды ғана қолдануға болады. Курсорлардың барлық типтері белгілі бір мәліметтер қорымен жұмыс істей алмайды. Бір мәліметтер қоры бір типті қолданады, ал басқасы барлығын да қолдайды.

Filter – фильтр жолы.

Filtered – кесте фильтрленген бе? Егер мұнда false деп орнатсақ онда фильтр жолы ескерілмейді.

IndexFieldNames – индекстелген бағана атауы. Индекстер мәліметтерді сұрыптауға және кестелер арасындағы байланыстар үшін қажет.

RecNo – ағымдағы ерекшеленген жол нөмірі.

RecordCount – кестедегі жолдар саны.

Sort – сұрыптау көрсетілетін жол. Мысалы, «Телефон» өрісі бойынша сұрыптауға мына жолды жазу керек: ADOQuery1.Sort:='Телефон ASC'.

ASC операторы өсу реті бойынша сұрыптау керектігін айтады. DESC операторы кему реті бойынша сұрыптау керектігін айтады.

Active – егер бұл қасиет true-ге тең болса, онда кесте ашық.

AggFields – бұл жерде барлық агрегаттық өрістер сақталады.

AutoCalcFields – егер true болса, онда өрістерді автоматты түрде есептеу керек.

`EOF` – бұл қасиетке әсер етуге болмайды, бірақ `true` болса, онда біз файлдың басында отырмыз.

`Bookmark` – бұл жерде ағымдағы бетбелгі орналасқан.

`EOF` – бұл қасиетке әсер етуге болмайды, бірақ `true` болса, онда біз файлдың соңында отырмыз.

`FieldCount` – кестедегі өрістер саны сақталады.

`Fields` – бұл өріс арқылы өрістердің мәндеріне қатынауға болады. Айталық, біз 4-ші өрісте қандай мән сақталғанын білгіміз келді делік. Ол үшін `Table.Fields.Fields[4].AsString`. `AsString` әдісі бізге жол түріндегі мәнді алу керектігін көрсетеді. Бұл қарапайым тәсіл, бірақ оны қолданудың керегі шамалы.

`FieldValues` – бұл қасиеттің көмегімен көрсетілген өрістің кез келген мәніне қатынауға болады. Өріс атауын квадраттық жақшаға алу қажет. Мысалы, `Table1.FieldValues[‘Телефон’]=’3346598’`;

`FilterOption` – фильтрдің баптауы. Бұл жерде келесі параметрлерді көрсету керек:

- `foCaseInsensitive` фильтр регистрге тиіссіз болады;
- `foNoPartialCompare` егер бұл параметр тұрса, онда ұқсастықтар фильтрдегі көрсетілген мәнің дәл көшірмесімен жүргізіледі. Егер параметр көрсетілмесе, онда фильтр оның тура көшірмесі болып табылмайтын фильтрдегі мән құралатын жолдарға түседі;
- `Modified` – егер қасиет `true` –ге тең болса, онда кестеде өзгерістер енгізілді.

4.4 TADOTable құрауыштарының әдістері

Көріп отырғанымыздай қасиеттер өте көп және олардың көпшілігі өте пайдалы. Бұл тарауда біз олармен практикада танысамыз. Ал енді көптеген әдістердің сипаттамасын алуға дайын болыңыз.

`BookmarkValid` – бұл әдіс бетбелгінің дұрыстығын тексереді. Жалғыз параметр негізінде `Bookmark` сияқты бетбелгі типін көрсетуге болады және ол нақты болса, онда оның нәтижесі `true` болады.

`CancelUpdates` – кэш жадыда сақталған жаңартуларды болдырмау.

`CompareBookmark` – екі бетбелгіні салыстыру. Бұл әдісте екі параметр бар.

`Bookmark`. Бұл екі бетбелгілер салыстырылады. Егер бетбелгілер тең болса, онда нәтиже нөлге тең. Егер біріншісі екіншісінен кіші болса, нәтиже -1. Егер біріншісі екіншісінен үлкен болса, онда нәтиже бірге тең.

`DeleteRecords` – жазбаларды жою. Бұл әдісте бір ғана параметр – қандай жазбаларды жою. Біз параметрдің орнына келесі мәндерді көрсете аламыз:

- `arCurrent` тек ағымдағы жазбаны жою;
- `arFiltered` орнатылған фильтрді қанағаттандыратын жазбаларды жою;
- `arAll` - барлық жазбалар;

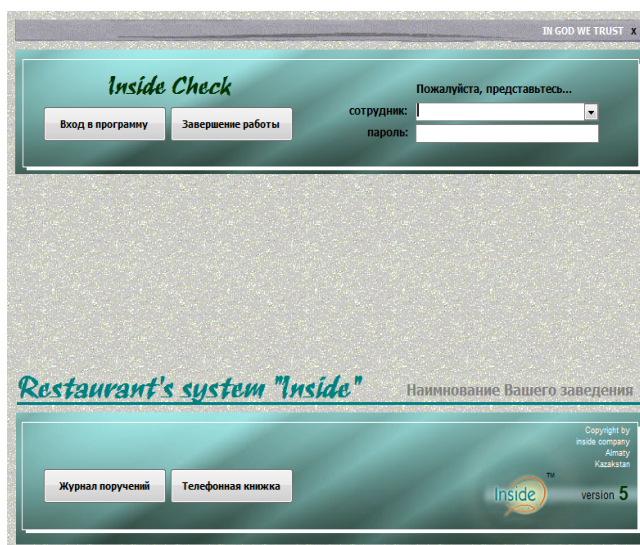
- arAllChapters ADO барлық бөлімдеріндегі жазбаларды жою;
- Append - кесте соңына жаңа жазбаны кірістіру;
- Cancel – ағымдағы жолды өзгертуді болдырмау, егер өзгертулер Post әдісінің көмегімен сақталмаған болса;
- Close - кестені жабу;
- Delete - ағымдағы жолды жою;
- Edit - редактілеу режиміне өту. Бұдан кейін өрістердің мәндерін өзгертуге болады;
- FieldByName - Атауы бойынша өрісті табу. Жалғыз параметр ретінде жолда өрістің атауын көрсетіп және нәтижесінде TField объектісі түріндегі өріске сілтеме аламыз;
- First – кестедегі бірінші жолға өту;
- Insert - кестеге жаңа жол қосу;
- IsEmpty - егер әдіс онда кестеде жазбалар жоқ;
- Last - кестедегі соңғы жазбаға өту;
- Next - келесі жазбаға өту;
- Post - барлық өзгертулерді қабылдау;
- Prior - кестедегі алдыңғы жазбаға жылжу;
- Refresh - мәліметтер туралы ақпаратты жаңарту;
- Update Record - ағымдағы жазбаны жаңарту.

5 Қосымша интерфейсін және оның функционалдығын сипаттау.

Қосымшамен жұмыс істеу

Жобаның интерфейсі Delphi бағдарламау тілінде жазылды. Интерфейс бағдарлама листингі Ә қосымшасында көрсетілген.

Қызметкер бағдарламаны ашып басты терезеге өтеді. Басты терезеде әр қызметкер өзінің жеке құпия сөзін енгізеді. Енгізілген құпия сөз деректер қорының қызметкерлер туралы ақпарат сақталған арнайы кестедегі ақпаратпен тексеріледі. Енгізілген құпия сөздің дұрыстығы расталғаннан кейін бағдарламаның келесі сәйкес терезесіне ауысады. Құпия сөздің бұрыстығы анықталса соған сәйкес ескертпе сөз көрсетіледі. Бағдарламаның басты терезесі келесі суретте көрсетілген.



5.1-сурет – Бағдарламаның басты терезесі

Бағдарламаны қолданатын қызметкерлер екі дәрежелі пайдаланушылардан тұрады :

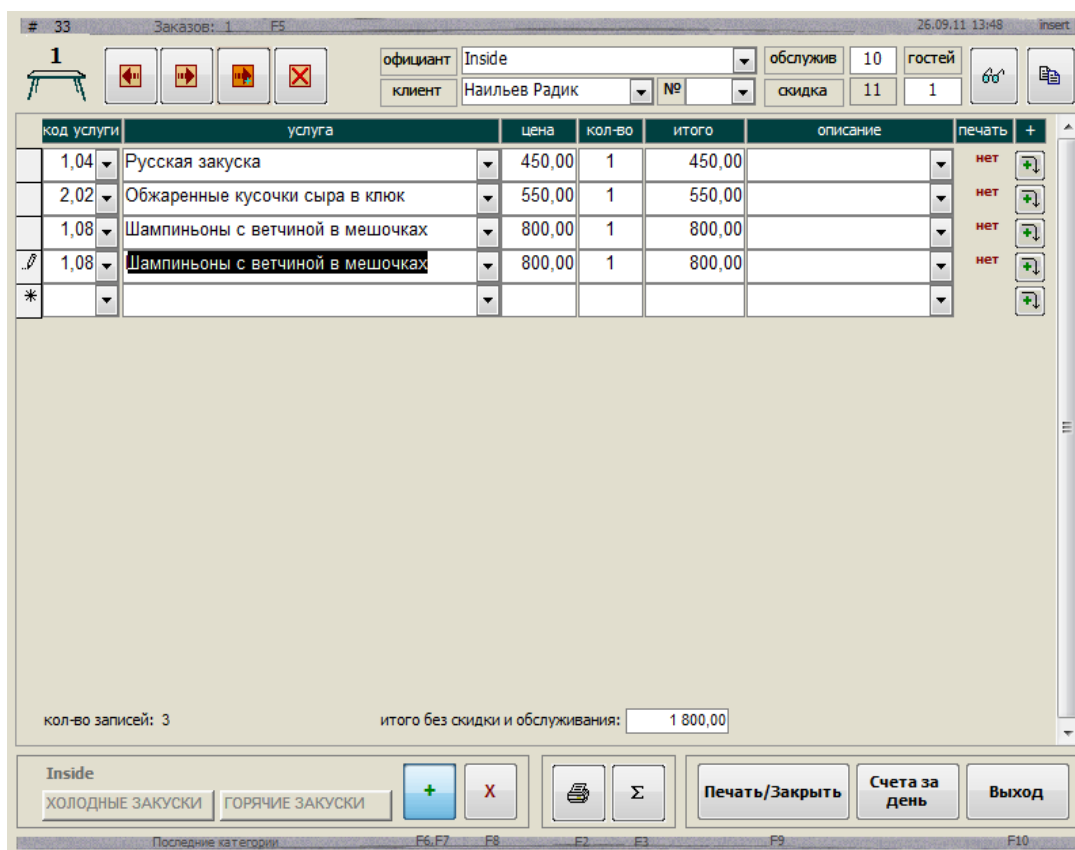
- администратор терезесі;
- официант терезесі;

Официант терезесінде бағдарламаны қолдану мүмкіндіктері администратор қолдану мүмкіндіктерінен әлдеқайда аз. Бұл пайдаланушы суретте көріп тұрғаныңыздай тек жаңа үстелді ашу, жаңа тағам қосу және оны басып шығару, ашылған үстелді жабу сияқты шектеулі мүмкіншіліктері бар.



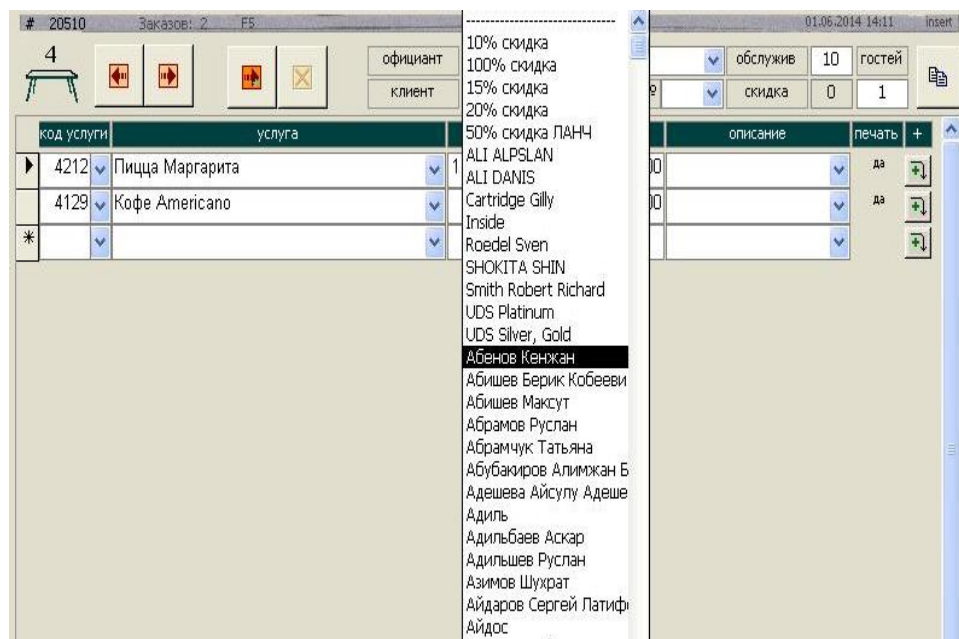
5.2-сурет – Официант жұмыс терезесі

Официанттың құпия сөзі арқылы кіріп ашық тұрған үстел ішіне кіргенде шығатын терезе мысалы келесі суретте көрсетілген.



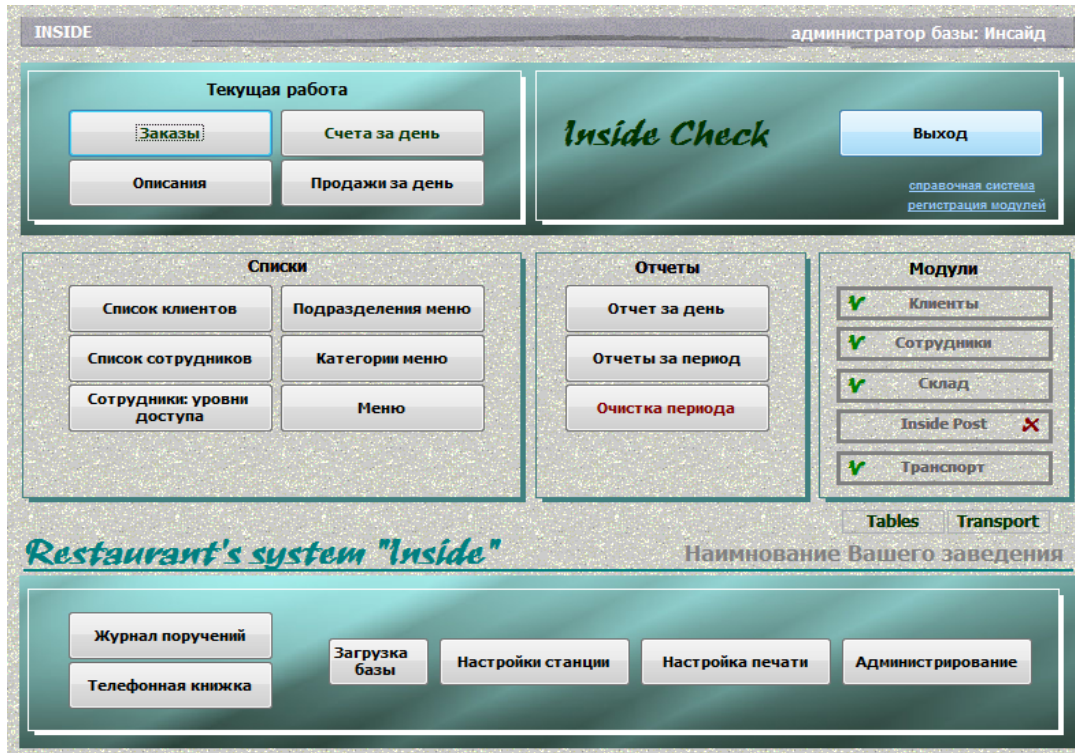
5.3-сурет – Ашылып тұрған үстел

Төменде көрсетілген суреттен байқап отырғаныңыздай официант әр ашылған үстелді жабу мезетінде жеңілдіктер типін таңдау мүмкіндігі бар.



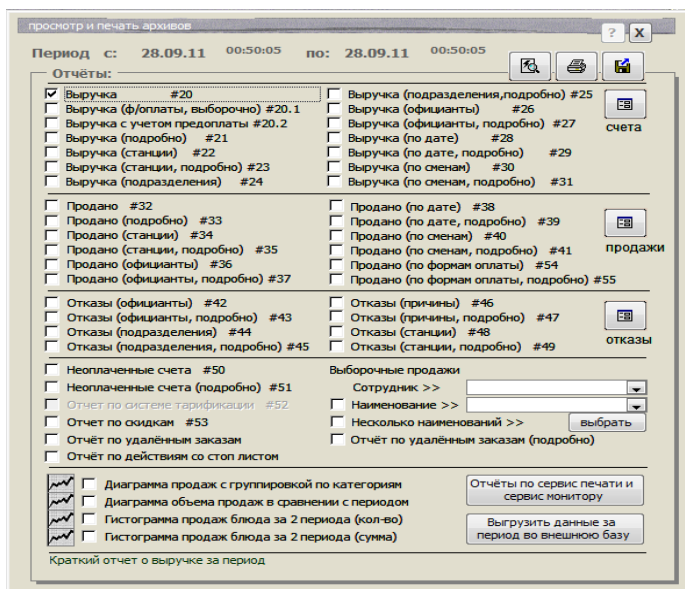
5.4-сурет – Жеңілдіктер таңдау батырмасы

Администратор құпия сөзі арқылы кіру бағдарламадағы болып жатқан барлық көріну мүмкіндігі бар операцияларды көру алады. Барлық бір күндік тапсырыстарды және саттыған тауарлар тізімін, әр күндік есе беру қағазын басып шығару немесе көру мүмкіндіктері қарастырылған.



5.5-сурет – Администратор жұмыс істеу терезесі

Келесі суретте администратор басып шығара алатын есеп беру қағаздарының әртүрлілігін бақылауға болады.



5.6-сурет – Есеп беру қағазын шығару

6 Технико – экономикалық негізделуі

6.1 Жұмысты сипаттау және қажеттілігін негіздеу

Берілген бітіру жұмысының тақырыбы – "Nash Pub" кафесінің желідегі жұмысын автоматтандыру.

Берілген өнімді әзірлеудің мақсаты болып ресторан саласында жұмыс істейтін мекемелердің жұмысын автоматтандыру, қойма жұмысын автоматтандыру көмегі арқылы мекемелердің шығынын азайту, қызметкерлер санын азайту.

Берілген бөлімде осы жобаның жүзеге асырылуының экономикалық құрамының қарастырылуы келтіріледі. Ол уақыттық, еңбектік және қаржылай шығындарды көрсетеді. Бұл бағдарлама :

- 1) мекеме шығынын азайтады;
- 2) барлығы адамның қатысынсыз автоматты түрде жүргізіледі;
- 3) қызметкерлер санын азайтуға мүмкіндік береді.

6.2 Нарықты саралау және орнын анықтау

Бағдарламалық қамтаманың басты мақсаты мекемеде орнатылған деректер қорының административтік және қауіпсіздік жағынан істеу барысын басқару.

Біздің өнімнің экономикалық көрсеткіштерін нарықта үш категорияға бөлуге болады:

- 1) өнім (Product) – бұл адам еңбегін азайтатын, жұмыс өнімділігін жүйені автоматтандыру арқасында жүзеге асыратын бағдарлама;
- 2) құн (Price)– бұл бағдарламалық қамтаманы пайдаланатын мекемеге қойылған ақша сомасы;
- 3) дистрибуция - бағдарламалық қамтама жеткізілетін ұсыныс берген мекеме;
- 4) қозғалу, даму (Promotion) — мекеменің алдымен тәжірибе жүзінде бір мекемеге бағдарламалық қамтаманы қоюы, кейіннен өзге мекемелерге орнату мүмкіндігі зор бағдарлама.

6.3 Бағдарламаның SWOT-анализі

SWOT анализі бағдарламаның маркетингтік жоспарын жасау үшін қажет. Келесі кестеде SWOT анализі көрсетілген.

К е с т е 6.1 – Бағдарламаның SWOT-анализі кестесі

Артықшылықтары	Кемшіліктері
Мекеменің барлық жұмысы автоматтандырылған сондай-ақ қойма жұмысы, соның арқасында кафе	Қымбат баға

шығындарын азайтады және адам тарапынан кететін қателіктер мөлшерін азайтады.	
Мүмкіншілігі	Қауіпі
Заманауи талаптарға сай барлық мекемелерінде қолданылу мүмкіндігі	Бағдарлама орнатылған компьютер жұмыс жасамаған жағдайында мекеме жұмысына зияны тиеді.

6.4 Маркетинг-микс элементі

Кесте 6.4.1 – Бағдарламаның маркетинг-микс кестесі

Маркетинг-микс элементтері	Қажетті істер мен талдаулар
Product	
Мекеменің жұмысын автоматтандыру	Жоспар бойынша бірнеше компоненттер пайдаланылады: 1) Техникалық өнім; 2) Бағдарлама интерфейсі C# тілдерінде орындалмақ
Price	
Базалық бағасы	Техникалық құралдың түріне қарай бағдарламалық қамтаманың бағасы анықталады.
Promotion	
	Жарнама жоқ, сатылам тек тапсырыс бойынша
Place	
Фирма 1, фирма 2, фирма 3	Автоматты жүйені қолданатын барлық ресторан кафе саласында жұмыс жасайтын мекемелер

6.5 Бағдарламалық қамтаманың өңделуінің еңбек өнімділігінің есептелуі

Еңбекке кеткен шығынның базалық көрсеткіштері мына формула бойынша есептеледі:

$$Q=q*c \quad (6.1)$$

мұндағы q- бағдарламалық өнімдегі (бастапқы команда) операторлар саны, менің жұмысымда операторлар саны 2500-ге тең, c-бағдарлама күрделілігінің коэффициенті, ол менде 1.26 ге тең [16]. Осы әдіспен базалық көрсеткішті табамыз:

$$Q=2500*1.26=3150$$

Ары қарай бағдарламалық өнімді әзірлеуге кететін уақытты есептеу керек. Уақыт адам-сағатпен есептеледі, ал T_d нақты істелген уақытпен алынады,

ал қалған кезеңнің уақыты Q командасының шартты санына байланысты есептік жолмен анықталады.

Бағдарламалық өнімін дайындауға кеткен әр кезеңнің уақытын анықтаймыз:

1) $T_{ПО}$ (мақсат сипатын дайындау уақыты), нақтылы деректер бойынша алынады және келесі мәнге тең деп алынады (3-тен 5 күнге дейін, 8 сағаттан):

$$T_{ПО} = 24 \text{ адам} / \text{сағ}$$

2) T_O (мақсат сипаттамасының уақыты) келесі формуламен анықталады:

$$T_O = Q \times B / (50 \times K), \quad (6.2)$$

$$T_O = 3150 \times 1,5 / 50 \times 0,8 = 118,125 \text{ адам/сағ}$$

мұндағы B – мақсат есебі өзгерісінің коэффициенті, B коэффициенті мақсат күрделігіне және өзгеріс санына тәуелді – 1,2-ден 1,5-ке дейін ;

K – бағдарлама жасаушы біліктілігін ескеретін коэффициент, менің берілген салада жұмыс істеу өтілім 2 жылға дейінгі уақытты құрағандықтан, K коэффициенті мәні – 0,8-ге тең болады.

3) T_A (алгоритм құруға кеткен уақыт) мына формуламен есептейміз:

$$T_A = Q / (50 \times K). \quad (6.3)$$

$$T_A = 3150 / (50 \times 0,8) = 78,75 \text{ адам/сағ}$$

4) T_{BC} (блок – сұлба құруға кеткен уақыт) T_A сияқты 3-формуламен есептеледі.

5) T_H (бағдарламаның тілінде жазуға кеткен уақыт) келесі формуламен анықталады:

$$T_H = Q \times 1,5 / (50 \times K). \quad (6.4)$$

$$T_H = 3150 \times 1,5 / (50 \times 0,8) = 118,125 \text{ адам/сағ}$$

6) T_{II} (бағдарлама теру уақыты) келесі формуламен анықталады:

$$T_{II} = Q / 50 = 63 \text{ адам/сағ} \quad (6.5)$$

7) T_{OT} (бағдарламаны реттеу және тестілеу уақыты) келесі формуламен анықталады:

$$T_{OT} = Q \times 4,2 / 50 \times K \quad (6.6)$$

$$T_{OT} = 3150 \times 4,2 / 50 \times 0,8 = 330,75 \text{ адам/сағ}$$

8) T_D (құжаттарды рәсімдеу уақыты), нақтылы деректер бойынша алынады және құрылады (3-тен 5 күнге дейін, күніне 8 сағат)

$$T_D = 24 \text{ адам / сағ.}$$

Еңбек шығындарының сомасы еңбек шығынының құрама сомасы ретінде 7 формуламен есептеледі:

$$T = T_{\text{ПО}} + T_{\text{ТО}} + T_A + T_{\text{БС}} + T_H + T_{\text{П}} + T_{\text{ОТ}} + T_D \quad (6.7)$$

$$T = 24 + 118,125 + 78,75 + 78,75 + 118,125 + 63 + 330,75 + 24 = 835,5 \text{ адам/сағ}$$

6.5.2 Еңбекақыға жұмсалатын шығындар

Еңбек құны екі құрамнан құралады: негізгі еңбек құны және қосымша еңбек құны.

Негізгі еңбек құны:

$$Z_{\text{осн}} = t_{\Sigma} * TC / (t_{\text{cp}} * 8), \text{ теңге} \quad (6.8)$$

мұндағы t_{Σ} – (7) формуласы бойынша есептелетін жалпы еңбек шығыны;

t_{cp} – айдағы орташа күн саны, 21 күнге тең, жұмыс күніндегі 8 сағатқа көбейтіледі.

Тарифтік мөлшерлеме минималды өлшемдегі еңбек ақы 01.01.2014ж бастап ҚР-да 19966 теңгені құрады.

TC – тарифтік қойылым = 19966 -ке тең болды.

t_{Σ} – (6.7) формуласы бойынша 473 –ге тең.

Осылайша негізгі еңбек ақы:

$$Z_{\text{осн}} = 835,5 * 2 * 19966 / (21 * 8) = 198530,97 \text{ теңге.}$$

Қосымша еңбек ақы негізгі еңбек ақының 20%-ын құрайды. Мына формула бойынша анықталады:

$$Z_{\text{доп}} = 0,2 * Z_{\text{осн}} \quad (6.9)$$

$$Z_{\text{доп}} = 0,2 * 198530,97 = 39706,2 \text{ теңге.}$$

Жалпы еңбек ақысы (еңбекті төлеу фонды) негізгі және қосымша еңбек ақының қосындысы ретінде анықталады (6.10):

$$\text{ФОТ} = Z_{\text{доп}} + Z_{\text{осн}} \quad (6.10)$$

$$\text{ФОТ} = 198530,97 + 39706,2 = 238237,16 \text{ теңге.}$$

6.5.3 Қосымша шығындар статьясы

Бағдарламалық қамтамасыз етудегі қосымша шығындар: материалдар мен құрамаға кеткен шығындар(құрылғының өз құны, яғни компьютерлер есепке алынбайды), социалды сақтандыруға аударым, жүккүжатка кеткен шығындар, амортизациялық аударым, техникалық қондырғылар қызметіне кеткен шығын, компьютермен жұмыс істеу кезінде кеткен электроэнергиялық шығындар).

Қондырғының құны бағдарламалық қамтама өңделуінің өзіндік құнына кірмегенімен, қосымша шығындар статьясындағы есептеулерде қолданылады. ЭЕМ - на бағдарлама жазу кезінде құрылғы ретінде дербес компьютер қолданылады. Құны $C_{обор} = 300000$ теңге (дербес компьютердің қазіргі уақыттағы нарықтағы теңгемен берілген құны алынды)

Материалдар мен көмекші бөлшектер шығыны, бағдарламалық өнімді жазу барысында қолданылды ($C_{МжК}$), сонымен қатар техникалық қызмет көрсету шығыны ($C_{ТО}$), жабдықтың құнынан 1.5% және 2.5% [16] құрайды және мына формулалар мен есептеледі:

$$C_{МжК} = 0,015 \times C_{обор} \quad (6.11)$$

$$C_{МжК} = 0,015 * 300000 = 4500 \text{ теңге.}$$

Амортизациялық аударым өнімнің физикалық және моральдық жарамсыздығына байланысты сол өнімді толық қалпына келтіру үшін жиналатын ақша қоры болып есептеледі. Амортизациялық аударым амортизация нормасы бойынша орнатылады, (6.12)-формуласы бойынша есептеледі:

$$A = \frac{C_{обор} \times H_A \times N}{100 \times 12 \times t} \quad (6.12)$$

мұндағы $C_{обор}$ – компьютер құны;

H_A – амортизация нормасы;

N – жұмыс орындалуына кеткен күннің саны;

t – дербес компьютерді қолдануға кеткен жалпы уақыт.

$$N = 835,5 / 24 = 34,812 \text{ күн}$$

бұл жерде T еңбек шығынына тең, оның мәні – 473 адам/сағ тең болған.

Дербес компьютерде жалпы жұмыс істеу уақыты мына формуламен есептеледі:

$$t = 78,75 + 78,75 + 118,125 + 63 + 330,75 = 669,375 \text{ адам/ сағ}$$

Амортизация нормасы (6.13)-формуласы бойынша анықталады:

$$H_A = \frac{C_{обор} - C_{ликв}}{T_{норм} \cdot C_{обор}} \cdot 100\%, \quad (6.13)$$

мұндағы $C_{ликв}$ – таратылым құны, жабдықтың құнынан 5% құрайды:

$$C_{ликв} = 0,05 \cdot C_{обор} \quad (6.14)$$

$$C_{ликв} = 0,05 \cdot 300000 = 15000 \text{ теңге.}$$

$T_{норм}$ – нормативті қызмет ету өтілі (дербескомпьютер үшін ҚР да $T_{норм} = 4$ жыл). Осыдан:

$$H_A = \frac{300000 - 15000}{4 \cdot 300000} \cdot 100\% = 23,75\%,$$

$$A = 300000 \cdot 23,75 \cdot 669,379 / 100 \cdot 34,812 \cdot 12 = 57084,304 \text{ теңге.}$$

Бағдарламалық қамтаманы жазу кезіндегі электроэнергияның шығынын қоса есептеу керек. Электроэнергияның құны мына формула бойынша есептеледі (6.15):

$$C_{ЭЭ} = M \cdot k_3 \cdot F_{эф} \cdot C_{кВт-ч} \quad (6.15)$$

мұндағы M – ЭВМ қуаты (450 Вт);

k_3 – жүктеу коэффициенті (0.8);

$C_{кВт.ч}$ – 1 кВт-сағ электроэнергияның құны, оның мәні ҚР – да 14,36 теңгеге тең;

$F_{эф}$ – жұмыс уақытының эффективті фонды, мына формула бойынша есептеледі (6.16):

$$F_{эф} = D_{ном} \cdot d \cdot (1 - f / 100) \quad (6.16)$$

$$F_{эф} = 258 \cdot 8 \cdot (1 - 2,2 / 100) = 2019 \text{ сағ.}$$

мұндағы $D_{ном} = 258$ – бір жылдағы жұмыс күнінің номиналды саны;

$d = 8$ – жұмыс күнінің ұзақтығы [сағ];

$f = 2.2\%$ – ЭВМ –ды жөндеуге кеткен жоспарланған уақыт.

Берілген коэффициенттер мен параметрлер бойынша электроэнергияның құны:

$$C_{ЭЭ} = 0,45 \cdot 0,8 \cdot 2019 \cdot 14,36 = 10437 \text{ теңге.}$$

Дегенмен, алынған амортизациялық аударым мен электроэнергияға шығын – жылдық шығынның мәні, оны жалпы жылдық эксплуатациондық шығынмен анықталатын уақыт коэффициентімен дұрыстау керек(6.17):

$$\mathcal{E}_3 = t_{\Sigma'} * C_{\mathcal{E}} / F_{\mathcal{E}\phi}, \text{ теңге} \quad (6.17)$$

мұндағы $C_{\mathcal{E}} = C_{\mathcal{E}\mathcal{E}} + C_{TO} + A_{\text{жыл}}$ – ЭЕМ-ның жалпы жылдық эксплуатационды шығын құны:

$$C_{\mathcal{E}} = 10437 + 3750 + 57084,307 = 71271,307 \text{ теңге.}$$

мұндағы $F_{\mathcal{E}\phi}$ – жұмыс уақытының эффективті фонды, (6.16) формуласы бойынша анықталды, $t_{\Sigma'}$ - ЭЕМ-нің (6.7) формуласы бойынша анықталған есепті шешу үшін жалпы қолданылған уақыты, бірақ бұл жерде тек компьютердегі жұмыс уақытын ескере отырып анықталған:

$$t_{\Sigma'} = 78,75 + 330,75 + 24 = 433,5 \text{ сағ.}$$

Сәйкесінше, ЭЕМ-ның жалпы жылдық эксплуатационды шығын құны:

$$\mathcal{E}_3 = 433,5 * 71271,307 / 2019 = 15302,68 \text{ теңге.}$$

Әлеуметтік салыққа кететін аударымдар бүкіл жалақының 11% құрайды [11%, ҚР ның Салық Кодексі], (6.19) формуласы бойынша анықталады:

$$CC = (\text{ФОТ} - 0,1 * \text{ФОТ}) * 0,11 \quad (6.19)$$

$$CC = 238237,16 - 0,1 * 238237,16) * 0,11 = 23585,5 \text{ теңге.}$$

Жүктеме шығындар қызмет көрсету мен басқаруға қатысты, құрылғының эксплуатациясы мен басқа да өндіріс процесін қамтамасыз ететін қосымша шығындар болып табылады, еңбек жалақысы фондының 50% құрайды, (6.20) формуласы бойынша анықталады:

$$C_{\text{накл}} = 0,5 * \text{ФОТ} \quad (6.20)$$

$$C_{\text{накл}} = 0,5 * 238237,16 = 11792,74 \text{ теңге .}$$

6.5.4 Өзіндік құн нәтижесінің кестесі мен диаграммасы

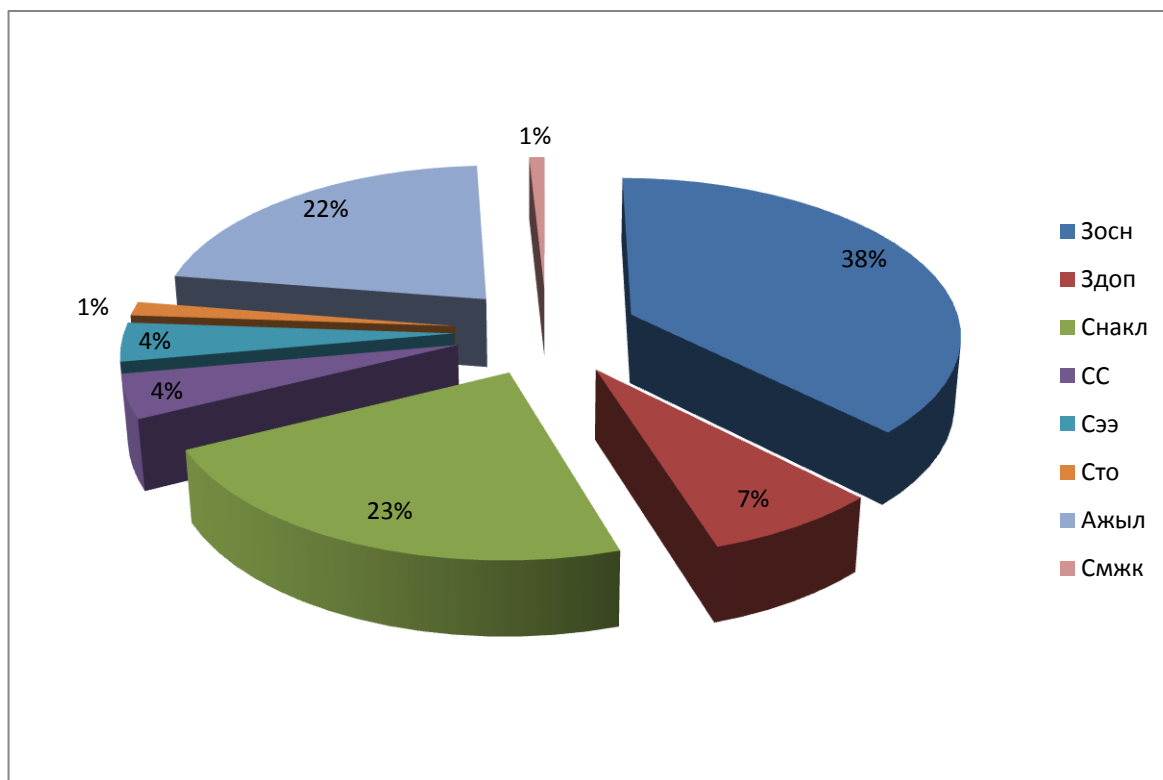
Бағдарламалық қамтаманы өңдеуге кеткен жалпы шығын еңбек жалақысы фондының, эксплуатационды шығынның, социалды сақтандыру, жүктеме шығындар мен материал және құрамалардың қосындысына тең болады.

Бағдарламалық қамтаманы өңдеудің қорытынды құны 6.3-кестеде көрсетілген.

К е с т е 6.3 – Өзіндік құн нәтижесінің кестесі

Шығын статьялары		Сомма, теңге	Жалпы соммадан, %
ФЗП	$Z_{осн}$	198530,97	48,4
	$Z_{доп}$	39706,2	9,67
Жүктеме шығындар, $C_{накл}$		11792,74	12,8
Әлеуметтік сақтандыру, CC		23585,5	5,74
Эксплуатационды Шығындар	$C_{ээ}^*$	10437	2,5
	$C_{то}$	7500	1,8
	$A_{жыл}^*$	114157,81	27,8
Материалдар мен комплектілер, $C_{мик}$		4500	1,09
Барлығы:		410208,51	100

Өзіндік құнының диаграммасы:



6.1-сурет – Өнімнің өзіндік құнының диаграммасы

6.6 Бағдарламаның бағасына есептеу

Өнімнің өзіндік құны 410208,51 теңге (жоғарыда кесте 6.3) болды. Өнімнің минималды бағасы келесі формуламен есептеледі:

$$Ц_{\min} = (C_{\text{пс}} + Ц_{\text{мд}} + Ц_3)(1 + r/100) \quad (6.21)$$

$$Ц_{\min} = 410208,51 * 1,21 = 496352,3 \text{ теңге.}$$

мұндағы $Ц_{\text{мд}}$ - магнитті диск бағасы, қазіргі нарықтық бағасын қарастырып, 200 теңге деп алынды;

$Ц_3$ – жазба бағасы;

r - Өзіндік құнға қатынасымен алынған тиімділіктің жоспарланған пайызы=21%;

НДС ті есептегендегі минималды баға келесі формуламен есептеледі (ҚР да НДС 12%):

$$Ц_{\min_{\text{НДС}}} = Ц_{\min} + Н_{\text{ндс}} Ц_{\min} \quad (6.22)$$

$$Ц_{\min_{\text{НДС}}} = 496352,3 + 0,12 * 496352,3 = 555914,57 \text{ теңге.}$$

Сонда бағдарламаның құны 555914,57 теңге.

6.6.3 Бағдарламалық өнімді иемденудегі кәсіпорындардың бірмезгілдегі шығындарын есептеу

Кәсіпорындардың бірмезгілдегі шығындары келесі бөлімдерден тұрады:

- жүйе құны;
- көлік шығындары;
- оқуға кететін төлемдер;

Жүйе құны 555914,57 теңге (6.22-) формула.

Көлік шығындары жүйе құнының 20% құрайды:

$$C_{\text{тр}} = 555914,57 * 0,2 = 111183 \text{ теңге.}$$

Жұмысшыларды жүйемен жұмыс істеуге оқыту шамамен бес сағатта консалтингті фирманың маманы көмегімен сағатына 1000 теңгедей болып табылады [16]. Соған сәйкес оқу бағасы:

$$C_{\text{обуч}} = 1000 * 5 = 5000 \text{ теңге.}$$

Кәсіпорынның бірмезгілдегі шығындарын есептеулерінің нәтижесі 6.4-кестеде көрсетілген.

К е с т е 6.4 – Ақпаратты жүйені енгізуге арналған бірмезгілдегі шығындардың есептелу нәтижесі

Шығындар түрлері	Сомма, теңге
Жүйе құны	555914,57
Көлік шығындары	111183
Жұмысшыларды оқыту бағасы	5000
Қорытынды:	672097

6.6.4 Қолдану саласындағы жылдық шығындарды есептеу

Негізгі шығындар түрлері:

- жұмысшы еңбегіне төленетін ақша қоры;
- әлеуметтік салық;
- басқада шығындар.

Дұрыстау ПК операторымен 30000 теңге жалақы көлемінде жөндеу жұмыстарын жүргізіп отырады.

$$C_{\text{опер}}=30000*12=360000 \text{ теңге.}$$

Әлеуметтік салыққа жұмсалатын аударымдар (ҚР да әлеуметтік салық мөлшері 11% жұмысшы еңбегіне төленетін ақша қорынан, ҚР Салық Кодексі)

$$CC=(360000-360000*0,1)*0,11=35640 \text{ теңге.}$$

К е с т е 6.5 – Кәсіпорынның жылдық бірмезгілдегі шығындарын есептеу нәтижесі

Шығындар түрлері	Сомма, теңге
Еңбекке төленетін қор	360000
Әлеуметтік салыққа аударымдар	35640
Қорытынды:	395640

6.6.5 Ақпараттық өнімді енгізуден алынған үнемдеу мөлшері мен табысты есептеу

Ақпараттық өнімді құрастыратын фирма үшін табыс көзі ретінде тапсырыс берушіге бағдарламалық өнімді сату болып табылады. Фирманың шығындары – ол құрастыруға кететін шығындар және жүйенің көбейтілуіне кететін шығындар. Қаржыландыру көзі ретінде құрастырушы-фирманың өзінің ақша қоры алынады.

Тапсырыс беруші-кәсіпорын үшін үнемдеу көзі есептеу уақытын біршама қысқартатын, қол еңбегін алмастыратын машина болып табылады. Кәсіпорын шығындары ақпараттық жүйені иемденудегі бірмезгілдік шығындардан, ақпараттық жүйені тасымалдау және енгізу шығындарынан, және де жүйеге талдау жүргізу мен бақылауға кететін шығындардан тұрады.

Жүйені енгізу қолмен есептеу шараларын автоматтандырылған шараға ауыстыру нәтижесінде өндірістің қорын біраз үнемдеуге алып келеді деп жоспарланады.

Үнемдеу мөлшерін есептейік. Ол үшін өткізу қабілетін есептеуге кететін кәсіпорынның шығындарын қолмен есептеу керек.

Қол еңбегімен келесі шығындар түрлері есептеледі:

- жұмысшы еңбегіне төленетін ақша қоры;
- әлеуметтік салық;
- басқа да шығындар.

Қолмен есептеу барысында еңбек өнімділігін талдау үшін 67000 тенге көлеміндегі жалақы мөлшерінде жұмыс істейтін мамандандырылған қойма қызметкері қажет. Қолмен есептеу барысында еңбектің жылдық төлем қоры:

$$EA=67000*12=804000 \text{ теңге.}$$

Әлеуметтік салыққа жұмсалатын аударымдар (**ҚР да әлеуметтік салық мөлшері 11% жұмысшы еңбегіне төленетін ақша қорынан,ҚР Салық Кодексі**) :

$$CC=(804000-804000*0,1)*0,11=71640 \text{ теңге.}$$

К е с т е 6.6 – Ақпараттық жүйені қолданбай еңбек өнімділігін талдауға кететін шығындарды есептеу

Шығындар түрлері	Сомма, теңге
Еңбекке төленетін қор	804000
Әлеуметтік салыққа аударымдар	71640
Қорытынды:	875640

Ақпараттық жүйені қолданғандағы жылдық шығындар жоғарыда есептелді және 21980 теңге болды.

Осыған сәйкес, шығындардың шартты үнемделуі мынаған тең болады:

$$C_{\text{экон}}=875640-395640=480000 \text{ теңге.}$$

Шығынның шартты үнемделуі бағдарлама енгізілгенге дейін кететін шығын мен бағдарлама енгізілгеннен кейін қызметкерлерді қысқарту арқасында үнемделетін ақша қорының азайтылуы арқасында шығатын үнемделу. Жүйені енгізу қолмен есептеу шараларын автоматтандырылған шараға ауыстыру нәтижесінде өндірістің қорын біраз үнемдеуге алып келеді деп жоспарланады.

Сонымен шартты үнемделуіміз 480000 теңгені құрайды.

6.7 Ақшалай құралдардың қозғалысы

Мекеменің әр мезгілде шығындарын және пайдасын келесі кестеден көре аласыз.

К е с т е 6.7 – Ақшалай тәсілдердің қозғалысы, теңге

Аты				Барлығы
	2014	2015	2016	
Бірмезгілдік шығындар	672097			672097
Операциялық кәсіпкерліктен пайда		480000	480000	960000
Дисконттау коэффициенті (21% мөлшерінде)	1	0,826	0,683	
Таза дисконтталған табыс (ТДТ)	-672097	396694	327846	
ТДТ өспелі нәтижесімен	-672097	-275402	52443	

Таза дисконтталған табыс (ТДТ) = Таза табыс* Дисконттау коэффициенті (6.23)

$$\text{ТДТ}_1 = 480000 * 0.826 = 396694 \text{ теңге};$$

$$\text{ТДТ}_2 = 480000 * 0.683 = 327846 \text{ теңге}.$$

6.8 Экономикалық тиімділікті есептеу

6.8.1 Таза ағымдағы құндылықты есептеу (Net present value, NPV)

Шығындары бірмезгілде тек жобаның басында күрделі салымдар (C_0) ретінде іске асатын жобалар үшін NPV (6.24) формуламен есептеледі:

$$NPV = \sum_{i=1}^n \frac{B_i}{(1+r)^i} - C_0. \quad (6.24)$$

мұндағы B_i – i -ші жылдағы жобадан алынатын пайда;
 r – дисконттеу мөлшері.

$$NPV(21\%) = \left(\frac{480000}{(1+0.21)^1} + \frac{480000}{(1+0.21)^2} \right) - 672097 = 52443 \text{ тг} > 0$$

сондықтан ұсынылып отырған жоба табысты. Сол себептен жобаны қажет және жобаны талдау мен оған баға беруді жалғастыру керек.

6.8.2 Пайда индексі есептеу (Profitability index, PI)

PI жобаның салыстырмалы пайдасын көрсетеді. Бұл көрсеткіш келесі (6.25) формуламен есептеледі:

$$PI = \frac{NPV}{C_0} \quad (6.25)$$

$$PI = (724541/672097) = 1,07\%$$

6.8.3 Табыстың ішкі нормасын есептеу (Internal rate of return, IRR)

Табыстың ішкі нормасы (ТИН немесе IRR) r пайыздық мөлшерінің мәні болып табылады, мұндағы $NPV = 0$. Бұл нүктеде r^* қосылған шығындардың дисконтталған ағыны қосылған дисконтталған табыс ағындарына тең.

$$IRR = r_1 + \frac{f(r_1)}{f(r_1) - f(r_2)} * (r_2 - r_1) \quad (6.26)$$

$r_b = 30\%$:

$$PV_1 = 480000 / (1 + 0,3) = 369231 \text{ теңге,}$$

$$PV_2 = 480000 / (1 + 0,3)^2 = 284024 \text{ теңге,}$$

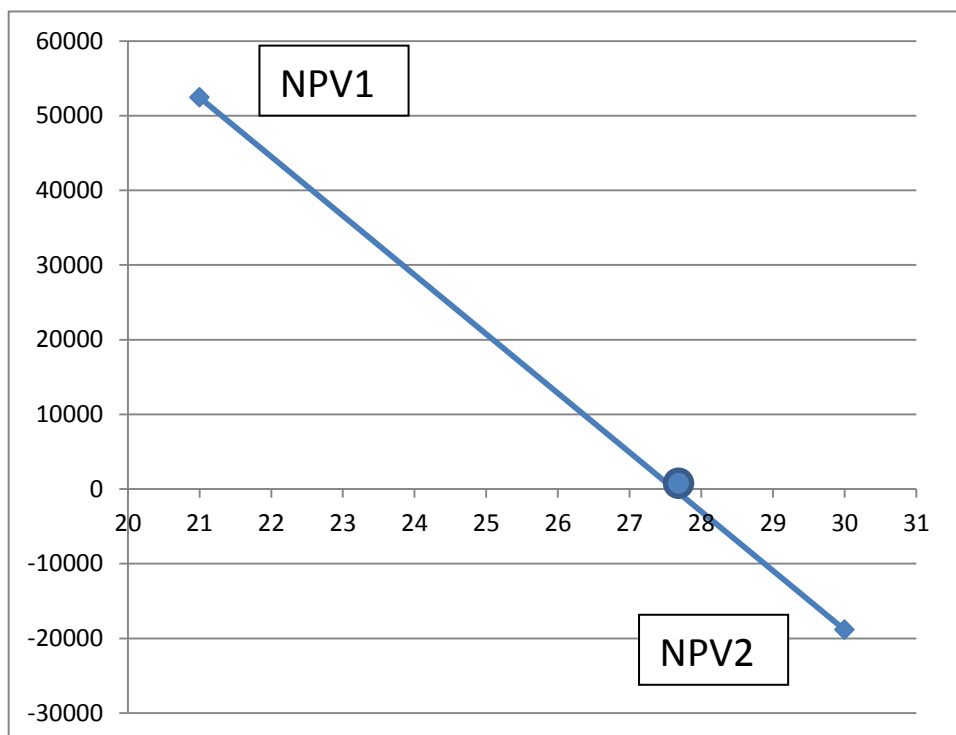
$$NPV(30\%) = (369231 + 284024) - 672097 = -18843 \text{ теңге.}$$

$$NPV(21\%) = 52443 \text{ теңге,}$$

$$NPV(30\%) = -18843 \text{ теңге,}$$

$$IRR = 21 + \frac{52443}{52443 - (-18843)} * (30 - 21) = 27,62\%$$

Табыстың ішкі нормасы сурет 2-де көрсетілген:



6.2-сурет – Іріктеу әдісі арқылы IRR табу

6.8.4 Өтімділік периодын есептеу (Payback period, PBP)

Өтімділік периоды мына формуламен есептеледі:

$$T_{ок} = t + \frac{C_0 - \sum_{i=1}^n B_i}{B_{i+1}} \quad (6.27)$$

мұндағы C_0 – жоба басындағы салым;

B_i – i -ші жылдағы жобадан алынатын пайда.

Сонда біздің өтімділік периодымыз:

$$T_{ок} = 1 + \frac{672097 - 396694}{327846} = 1,84 = 1 \text{ жыл } 10 \text{ ай.}$$

Бұл жобаның өтімділік периоды 1 жыл 10 айға тең, яғни жобамыз осы уақытта өз-өзін ақтайды.

7 Тіршілік қауіпсіздігі

7.1 Инженердің жұмыс жағдайын талдау

Дипломдық жұмыста МАИ мекемесінің Автоматтандырылған Жұмыс Орнын (АЖО) жобалау қарастырылған. АЖО құру және жобалау үшін MS SQL дерекқор басқару жүйесі (ДҚБЖ) және Visual Studio бағдарламалық ортасы таңдалған. Қолданыстағы дерекқор негізінде жаналау көзделеді.

Офис 5.1-суретте көрсетілген. Бөлменің ұзындығы 6 м, ені 6 м және биіктігі 3 м, ұзындығы 2 м екі терезе бар. Бөлмеде үш адам жұмыс істейді, жұмыс графигі - аптасына бес күн, күніне сегіз сағат. ДК операторының жұмысы ұзақ көру жұмысымен байланысты болғандықтан, офистегі жарықтануды есепке алу қажет. Жарықтану деңгейі психикалық функциялардың күйіне және ағзадағы физиологиялық үрдістерге әсер етеді. Бөлме компьютерлік құрылғылар мен оргтехникамен жабдықталған, сол себептен мекеменің персоналы артық жылулық сәулеленуге шалдығады. Сондықтан персоналдың қолайлы еңбек ету шарттарын қамтамасыз ету үшін микроклимат параметрлерін нормалау қажет. Микроклиматтың бөлек параметрлерінің ұсынылған мәндерінен ауытқуы жұмысшының еңбекке қабілеттілігін төмендетеді, көңіл күйін нашарлатады және кәсіби ауруларға әкелуі мүмкін. 7.1-кестеде ГОСТ 12.0.003-88. ССБТ сәйкес категориясы I а жеңіл физикалық жұмыс үшін қалыпты микроклиматтық шарттар келтірілген. Офисбөлмесіндегі жаз уақыт кезіндегі температура +26°C-ге дейін көтеріледі, ал қыс кезіндегі температура +18-ден +20°C-ге дейін. Қажетті микроклиматтық шарттарды сақтау үшін бөлме кондиционермен жабдықталған. Бөлменің терезелер арқылы түсетін табиғи жарықтануы, және тәуліктің қараңғы уақытында жұмыс істеу мүмкіндігін беретін жасанды жарықтануы бар. Жасанды жарықтану жоғары дәлдікті көру жұмысының III, а разрядының талаптарына сәйкес келеді. Жасанды жарықталу люминесцентті шамдар арқылы жүзеге асырылады.

Бұл дипломдық жобада энергетикалық кәсіпорында басқару және ақпарат желісін пайдаланудағы сұрақтар шешіледі. Желімен жұмыс істегенде компьютер негізгі рөл атқарады. Сондықтан, жұмысшылардың компьютермен жұмыс істеген кездегі еңбектің қауіпсіздігімен жұмысқа қабілеттілігін сақтауға арналған сұрақтарды талқылауымыз қажет.

Адамның денсаулығына әсер ететін негізгі әсерлерге мыналар жатады:

- жұмыс орнын күндізгі және түнгі уақыттарда жарықтандыру;
- жазғы жыл уақытындағы бөлменің артық жылулығы;
- электрлік тоқпен адамның зақымдануы;
- өрттің нәтижесінде адамның күйік алуы мен улануы.

Бұл бөлімде қауіпті және зиянды факторлардың әсері, ЕҚЖӨҚ бойынша ұйымдық шаралар, жұмыс орнын орналастыру, микроклимат параметрлері, электр қауіпсіздігі, өрт қауіпсіздігі туралы қарастырамыз.

Техника қауіпсіздігі бойынша нұсқауды өткен азаматтар ғана жұмысқа жіберіледі. Жобаланатын учаскеде келесі нұсқаулар қаралады:

– кіріспе нұсқаулар қызметкерлерді техника қауіпсіздігімен, өндірістік санитарлықпен, ішкі еңбек тәртіп ережелерімен және негізгі заңдылықпен таныстыру үшін өткізіледі. Бұл нұсқауды жұмысқа қабылданған адамдарға мамандығына, квалификациясына, біліміне, лауазымына қарамастан еңбек қорғау инженері 2 сағат көлемінде жүргізеді;

– арнайы мамандық үшін ЕҚ нұсқауы жұмыс орнында арнайы өткізіледі. Ол әр қызметкермен жеке өткізіледі;

– ағымдағы нұсқау жоғары қауіпті өндіріс жұмысында өткізеді, кіру рұқсат наряды тіркеледі;

– қайталама нұсқау барлық қызметкерлермен квалификациясына байланыссыз өткізіледі.

Бұл нұсқау жарты жылда бір рет өткізіледі және топтық болуы мүмкін немесе техника қауіпсіздігі бұзылған кезде жеке әңгіме ретінде өткізіледі; жоспарсыз нұсқау, техника қауіпсіздігінің нұсқауы өзгерген кезде өткізіледі.

Еңбек қорғау заңдылығын қадағалау Госнадзордың техникалық инспекторы өткізеді.

Бөлмеде келесі құрал-жабдық қолданылады:

1) дербес компьютер-3 дана.

Құрылғының техникалық сипаттамалары:

– toshiba dx2300 IntelCore i8 4200/8Gb/120Gb/Trio/Win8 дербес компьютері;

– 25" LG LI720 0.3dpi монитормы;

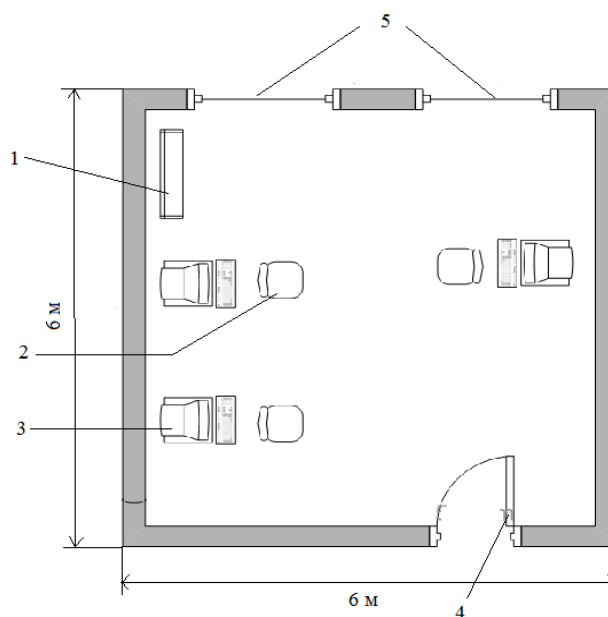
– мөлшерлер 1200x750x1150 мм (дербес компьютер+үстел);

– электрлік қоректену көзі: айнымалы кернеу 220-250 В, 50 Гц жиілігі, қуаты 400 Вт.

2) сплит-жүйе плазма LG S 12 LHK – кондиционер, қуаты 3.5 кВт.

К е с т е 7.1 – Микроклимат параметрлерінің қалыпты нормалары

Жыл мезгілі	Жұмыс категориясы	Температура, °С	Ауа қозғалысының жылдамдығы, м/с
Салқын	I a	18-26	0,1
Жылы	I a	20-30	0,2



7.1-сурет – Офис жоспары

- 1 - кондиционер;
- 2 - орындық;
- 3 - үстел және ДК;
- 4 - есік;
- 5 - терезе.

Операторлар бөлмесінің микроклиматтық шамалары: жыл мезгілінің суық кездерінде ауа қозғалысының жылдамдығы және салыстырмалы ылғалдылығы 0,1 м/с, 60%, ауа температурасы 18–26°C шамасында болады.

Ал жыл мезгілінің жылы кездерінде ауа қозғалысының жылдамдығы және салыстырмалы ылғалдылығы 0,2 м/с, 60–70%, . Келтірілген шамалар адам организміне ыңғайлы нормаларға сай келмейді. Сондықтан операторлар бөлмесінде ауаны кондиционерлеу мәселесі қарастырылған.

Адамның электр тогынан зақымдану ықтималдығына әсер ететін біздің бөлмеміздің класын анықтайық:

- едендер бір қабатты поливинилхлоридті антистатикалық линолеуммен қапталған, сондықтан ол ток өткізбейтін болып табылады;
- ауаның салыстырмалы ылғалдылығы 60%-дан аспайды, сондықтан бөлме құрғақ;
- ауа температурасы Цельсий бойынша плюс 30 градустан аспайды;
- адамның бір уақытта бір жақтан жермен байланысы бар технологиялық жабдықтардың корпустарымен және басқа жерлендірілген бөліктермен, екінші жақтан электр жабдықтарының металл корпустарымен немесе ток өткізуші бөліктермен жанасу мүмкіндіктерінің болмауы (кернеу 1000В мәнінен аспағандықтан сымдардың өте жақсы изоляциясында);
- химиялық белсенді заттар жоқ.

ГОСТ 12.1.013-78.ССБТ сәйкес осы бөлмені маңызды қауіпсіздік жоқ бөлме ретінде классификациялауға болады.

Біздің жағдайымызда электр қауіпсіздігін қамтамасыз ету үшін ГОСТ 12.1.030-81 бойынша жерлендіру мүмкіндігін қарастыру қажет. Біздің жағдайымыздағы кернеу - 220В, сондықтан жерлендіру мен нөлдеу міндеттелмейді, бірақ ұсынылады.

Құрылыс конструкцияларын дайындау үшін кірпіш, темір бетон, әйнек, металл және басқа жанбайтын материалдар қолданылады. Сонымен қатар жанбайтын материалдардан жасалған қоршаулар түріндегі өртке қарсы өткелдерді ескеру қажет, олар біздің офистің бөлмелері арасында орнатылады. Ғимараттарда өрт крандары дәлістерде, баспалдақ торларында және кіре беріс аумақтарында орнатылады. Дербес электрондық есептеуіш машинаны қолданушылар бөлмелерінде, архивте және қосымша, қызметтік бөлмелердегі өртті өшіру үшін су қолданылады. Дербес электрондық есептеуіш машина бар бөлмелерде, ақпаратты тасушыларды сақтау бөлмелерінде, қымбат құрылғыларды бұзу немесе толықтай істен шығару қаупінен бақылау-өлшеуіш жабдықтары бар бөлмелерде суды қолдану тек кейбір жағдайларда ғана рұқсат етіледі, мысалы өрт қауіпті ірі көлмеде болғанда. Бірақ судың мөлшері минималды болуы және дербес электрондық есептеуіш машинаны, дыбыстық құрылғыларды презентпен немесе матамен жауып судан қорғау керек.

Барлық бөлмелерді стационарлы автоматты өрт өшіргіш қондырғылармен жабдықтау қажет. Ауа құрамындағы оттегіні тез азайтатын от өшіргіш газбен бөлмені бірден толтыруға негізделген өртті газбен өшіру қондырғыларын қолданған тиімдірек болып табылады.

Зиян химиялық заттардың деңгейін нормалау. Бөлмені ластану көздері сыртқы ортаның және ғимараттың құрылыс материалдарынан, жиһаздардан, киімнен, аяқ-киімнен бөлінетін жүздеген әрекеттесулердің зиян заттары және адамның биоактивті әрекеттесулері (антропотоксиндер) болып табылады.

Бөлменің сыртқы ортаның зиян заттарымен ластануын қарастыра отырып, ең алдымен ғимараттың орналасқан орнын ескеру қажет, біздің жағдайымызда ол автострадаға жақын орналасқан. Бөлмеге сыртқы ортадан келетін жиі ластағыштар көміртек оксиді, азот диоксиді, күкірт диоксиді, қорғасын, шаң және тағы басқалары болып табылады.

Құрылыс конструкциялары бөлменің радон және торонмен ластануын көзі болып келеді, сонымен қатар ең көбірек концентрация нашар желдетуі бар бетоннан жасалған үйлерде кездеседі.

Жиһаз, киім және аяқ-киімдер минералды талшықты, көмір сутегісі, полиэфир қара майы және тағы басқа зиянды заттары бар шаңды бөледі. Биоактивті әрекеттесулердің ең маңыздысы көміртек диоксиді, күкірт сутегісі және тағы басқалары болып табылады.

Дербес электрондық есептеуіш машина қолданушысының, оператордың, жұмыс орнындағы шу көздері – сөйлесіп тұрған адамдар, сыртқы ортаның – компьютердің, принтердің, желдеткіш қондырғының шуы болып табылады.

Олар болмашы мәнде шуды тудырады, сондықтан бөлмеде дыбысты жұтқыштарды қолдану жеткілікті.

Ең жақсы дыбысты жұту қасиеті талшықты-ауа көлемді материалдарда: фибролитті плиталарда, жарықталшықтарында, минералды мақтада, полиуретанды поропластта, ауа көлемді поливинилхлоридте және басқаларында болады. Дыбыс жұтқыш материалдарға дыбыс жұту коэффициенті 0,2 мәнінен төмен емес материалдар жатады.

Жеткіліксіз жарықтандыру көздің, адамның тез шаршауына, жақыннан көргіштікке, жұмыс сапасының төмендеуіне, ақаудың көбеюіне соқтырады. Тым жарық көз қабықшасын тітіркендіреді, шағылыстырады, көз тез шаршайды, өндірістік травматизм көбейеді.

Қаралып жатқан бөлмеде жұмыс істеуге қажетті жарықтандыру қалып бойынша $E_n=300$ лк, осылайша бөлмедегі қажетті жарықтандыруды қамтамасыз етеміз.

Жұмыс бөлмесінде белсенді кондиционерлеу және вентиляция жүйесі жоқ. Операторлар залын талдау барысында жұмыс аумағының қалыпты микроклиматтық шарттарын міндетті түрде қарастыру қажет. Мұндай әмбебап жүйе ретінде автономдық кондиционерлер болып табылады.

ГОСТ 12.1.005-88 ССБТ "Жұмыс істеу аймағының ауасы, жалпы санитарлы-гигиеналық талаптар" сәйкес, компьютерлермен жабдықталған бөлмедегі адамдардың жұмысы жеңіл физикалық жұмысқа жатады. Ағзаның энергия жұмсау жұмыстарының категориялары кесте 7.2-де келтірілген.

Кесте 7.2 – Адам ағзасының энергия жұмсау жұмыстарының категориялары

Жұмыс	Категория	Ағзаның энергия жұмсауы, Ккал/сағ, Дж/с	Жұмыс сипаттамасы
Жеңіл	I а	<138	Жұмыс отырып жүргізіледі

7.2 Кондиционерлеу және ауаны жаңарту жүйелерін есептеу

Аумағы 15-тен 140 м² дейінгі тұрғын және қоғамдық бөлмелерде сплит-жүйе кондиционерлері кең таралуда. Олар сыртқы блоктан (компрессорлы-конденсаторлы) және ішкі блоктан (буландырғыш) тұрады. Сыртқы блок ғимарат қабырғасына, шатырға немесе шатыр астына, қосалқы бөлмеге, балконға, яғни ыстық конденсатор төменірек температурасы бар атмосфера ауасымен салқындатылатын жерге орнатылуы тиіс. Ішкі блок тікелей кондиционерленетін бөлмеге орнатылады және ауаны салқындату немесе жылыту, сүзу және бөлмедегі ауаның қажетті қозғалысын жасау үшін

тағайындалады. Ішкі блоктар берілген температураны ұстайды және бөлмедегі ауаның тең бөлінуін қамтамасыз етеді және шусыз жұмыс істейді (шу деңгейі 35-38 дБ). Кондиционерленген ғимараттың жылулық және ылғал теңгерімін белгілі әдістермен орындалады. Мұнда ғимараттың ауа ортасының қалпы өзгеруіне әкеп соғатын, барлық факторлар есепке алынуы керек.

Кондиционерді таңдау үшін алдымен артық жылудың қосындысын, сонымен қатар оған күннің радиациясынан бөлінетін жылу кіреді, өндірістік жарықтануды, жұмыс істейтін адамдар санын, оргтехникаларды және т.б. есептеу қажет. Салқын өндіргіштік бойынша қосындысы сондай немесе шамалы үлкен мәнді, сонымен қатар қажетті ауа алмасу қамтамасыз ететін кондиционер моделі таңданылады.

Бөлмедегі жылулық баланс мына формуламен есептелінеді

$$Q_{\text{жылу.б}} = Q_{\text{қоршау}} + Q_p + Q_a^a + Q_{\text{жарықтану}} + Q_{\text{құрал}} \text{ Вт.} \quad (7.1)$$

мұндағы $Q_{\text{қоршау}}$ – температура айырымы нәтижесінде алынатын жылу және жылу жоғалту;

Q_p – шынылау арқылы күннің сәулеленуінен келетін жылу;

Q_a^a – адамдардан келетін жылу түсу;

$Q_{\text{жарықтану}}$ – жарықтандыру аспаптарынан келетін жылу;

$Q_{\text{құрал}}$ – оргтехника және құрылғылардан келетін жылу.

Температура айырымы нәтижесінде алынатын жылу және жылу жоғалту. Терезе арқылы күннің радиациясынан түсетін жылу

Күннен бөлінетін жылу әйнектің түріне байланысты 90%-ға дейін бөлме ортасымен жұтылады, қалған бөлігі шағылысады. Ең үлкен жылу жүктемесі тура және шашырай түсетін күн сәулесінің ең үлкен деңгейінде алынады. Сәуле түсу қарқыны жергілікті кеңдікке, жыл мезгіліне және тәулік уақытына байланысты.

Салқын мезгіл үшін есептік сыртқы температура ($t_{\text{сырт.есеп}}$) ең салқын айдың 13 сағатындағы орташа температурасына, жылы мезгіл үшін – ең ыстық айдың 13 сағатындағы орташа температурасына сәйкес келеді. Ал ішкі ($t_{\text{іш.есеп}}$) жайлылық шартын және өндірістік процесстерде көрсетілетін технологиялық талаптарын ескере отырып таңдалады

$$Q_{\text{қоршау}} = V_{\text{бөлме}} X_0 (t_{\text{шыққан}} - t_{\text{келген}}) \text{ Вт.} \quad (7.2)$$

мұндағы $V_{\text{бөлме}}$ – бөлменің көлемі, м^3 . $V_{\text{бөлме}} = 6 \times 6 \times 3 = 108 \text{ м}^3$;

X_0 – меншікті жылулық сипаттама, $\text{Вт}/\text{м}^3 \text{ } ^\circ\text{C}$. $X_0 = 0,42 \text{ Вт}/\text{м}^3 \text{ } ^\circ\text{C}$;

$t_{\text{сырт.есеп}} = 27,6^\circ$ - жылдың жылы мезгіліне арналған сыртқы есептік температурасы;

$t_{\text{сырт.есеп}} = -25^\circ$ - жылдың суық мезгіліне арналған сыртқы есептік температурасы;

$t_{\text{іш.есеп}} = 24^\circ$ - жылдың жылы мезгіліне арналған ішкі есептік температурасы;

$t_{\text{ш.есеп}} = 20^{\circ}$ - жылдың суық мезгіліне арналған ішкі есептік температурасы.

Жылы мезгіл үшін: $t_{\text{сырт.есеп}} = 27,6^{\circ}\text{C}$ [12, кесте 3], $t_{\text{ш.есеп}} = 24^{\circ}\text{C}$

$Q_{\text{қоршау}} = 108 \times (27,6 - 24) \times 0,42 = 163,3 \text{ Вт}$,

Салқын мезгіл үшін: $t_{\text{сырт.есеп}} = -25^{\circ}\text{C}$ [12, кесте 3], $t_{\text{ш.есеп}} = 20^{\circ}\text{C}$

$Q_{\text{қоршау}} = 108 \times 0,42 \times (-25 - 20) = -2041,2 \text{ Вт}$.

Шынылау арқылы күннің сәулеленуінен келетін жылу

Күннің сәулеленуінен (радиация) келетін жылу терезе арқылы сәуле бөлмеге кіріп, күннен шынылау сәулелену мезгілі үшін

$$Q_p = (q_{\text{тура}} + q_{\text{шашыр.}}) K_1^c K_2 \beta_{\text{ж.ө.}} n H_0 B_0 \text{ Вт.} \quad (7.3)$$

Күннің сәулелері терезеден кірмейтін көлеңке мезгілі үшін (шашыраңқы радиация)

$$Q_{p.} = q_{\text{шашыр.}} K_1^T K_2 \beta_{\text{ж.ө.}} n H_0 B_0 \text{ Вт.} \quad (7.4)$$

мұндағы $q_{\text{тура}}$; $q_{\text{шашыр.}}$ – тура және шашыраңқы радиациядан келетін жылулық ағындар, Вт/м^2 ;

$F_0 = n H_0 B_0$ – жарықтық ойықтың ауданы, м^2 (n – терезелердің саны, биіктігі H_0 және ені B_0);

K_1 – қапсырмамен шынылаудың көлеңкелену коэффициенті (K_1^c – сәулеленген ойықтар үшін; K_1^T – көлеңкедегі ойықтар үшін);

K_2 – шынылаудың ластану коэффициенті;

$\beta_{\text{ж.ө.}}$ – жылу өткізу коэффициенті.

1) мекеменің офис бөлмесіндегі шынылаудың ауданы, 44°СШ

$F_0 = 2 \times 2,5 \times 2 = 10 \text{ м}^2$;

2) шынылаудың бағыты: оңтүстік-шығыс (ОШ);

3) ішінде жарық перделері бар. $\beta_{\text{ж.ө.}} = 0,4$ деп қабылдаймыз.

Түске дейін ОШ үшін, яғни сағат 9-дан 12-ге дейін 44°СШ ендікте тура радиацияның мәні (П) $q_{\text{тура}} = 387 \text{ Вт/м}^2$ және шашыраңқы радиацияның мәні (Р) $q_{\text{шашыр.}} = 101 \text{ Вт/м}^2$ тең. 44 - 68°СШ ендік диапазонында металды қапсырмалы екі қабатты шынылау үшін: $K_1 = K_1^c = 0,72$, егер ойық күнмен сәулеленген болса, яғни 9-10 және 13-14 сағат аралығындағы мезгіл үшін. $K_1 = K_1^T = 1,15$, 14-15 және 19-20 сағат аралығындағы мезгіл үшін. Әйнектің бірқалыпты ластануы коэффициенті $K_2 = 0,9$ қабылданады.

Тура сәулелену мезгілінде 9 бен 14 сағат аралығында есептелу мына формула арқылы жүреді (5.3).

$$Q_p = (387 + 101) \times 0,72 \times 0,9 \times 10 \times 0,4 = 1265 \text{ Вт.}$$

ал көлеңкелену мезгілінде 14 пен 20 сағат аралығында мына формуламен есептелінеді (5.4)

$$Q_p = 22 \times 1,15 \times 0,9 \times 10 \times 0,4 = 91 \text{ Вт.}$$

Максималды есептелу уақыты: 9-10 сағат, жылу түсу 1265 Вт.

Адамдардан келетін жылу.

Бөлмеде 3 әйел адам-операторлар отырады. $t = 24 \text{ }^\circ\text{C}$ температурада отырған күйде бір ер адам 67 Вт анық жылу, ал жалпы – 102 Вт жылу бөледі. Әйел адам ересек ер адамның жылу бөлу нормасының 85 %-ын, ал кішкентай бала– 75 %-ын бөледі деп саналады. Бөлмедегі адамдардың бөлетін анық жылуы

$$Q_a^a = 67 \times 3 \times 0,85 = 171 \text{ Вт.}$$

Ал жалпы жылу

$$Q_a^ж = 102 \times 3 \times 0,85 = 260 \text{ Вт.}$$

$t = 20 \text{ }^\circ\text{C}$ температурада бір ер кісі 82 Вт анық жылу және 103 Вт жалпы жылу бөледі . Бөлмедегі адамдардың бөлетін анық жылуы

$$Q_a^a = 82 \times 3 \times 0,85 = 209 \text{ Вт.}$$

Ал жалпы жылуы

$$Q_a^ж = 103 \times 3 \times 0,85 = 263 \text{ Вт.}$$

$t = 24 \text{ }^\circ\text{C}$ үшін ылғалдылық және көміртегі қышқылының мәндерін 9-кестеден интерполяция жолымен табамыз: бір адамнан 50 г/сағ ылғалдылық, 45 г/сағ көміртегі қышқылы бөлінеді. Ал 3 адамның ылғалдылығы $3 \times 50 = 150$ г/сағ, көміртегі қышқылы мөлшері $3 \times 45 = 135$ г/сағ құрайды .

$t = 20 \text{ }^\circ\text{C}$ үшін: 1 адамнан бөлінетін ылғалдылық – 40 г/сағ, көміртегі қышқылы – 45 г/сағ. 3 адамнан бөлінетін ылғалдылық: $3 \times 40 = 120$ г/сағ. 3 адамнан бөлінетін көміртегі қышқылы мөлшері: $3 \times 45 = 135$ г/сағ.

К е с т е 7.3 – Бөлмедегі адамдардан бөлінетін зиянды заттардың есептелуінің нәтижелері

Жыл мезгілі	Температура $^\circ\text{C}$	Жылу, Вт		Ылғалдылық, W г/сағ	CO ₂ г/сағ
		Q _a ^a	Q _a ^ж		
Жылы	24	288	434	250	225
Салқын	20	349	438	200	225

Жарықтану аспаптарынан, оргтехникадан және құрылғылардан келетін жылу

Шамдардан келетін жылу мына формуламен есептеледі

$$Q_{\text{жарықтану}} = \eta N_{\text{жарықтану}} \text{ Вт}, \quad (7.5)$$

мұндағы η - электр энергиясының жылулыққа ауысу коэффициенті. Люминесцентті шамдарды қолдану кезінде $\eta = 0,5-0,6$;

$N_{\text{жарықтану}}$ – шамдардың орнатылған қуаты 65 Вт/м^2 .

Бөлменің еденінің ауданы $F_{\text{еден}} = 6 \times 6 = 36 \text{ м}^2$.

$$Q_{\text{жарықтану}} = 0,6 \times 65 \times 36 = 1170 \text{ Вт}.$$

Оргтехниканың әсерінен пайда болатын жылу ағыны бір компьютерге орташа есеппен 300 Вт алады. Бөлмеде 3 ДК болғандықтан

$$Q_{\text{құрал}} = 3 \times 300 = 900 \text{ Вт}.$$

Орындалған есептеулерден (4.10) формуласы бойынша офисбөлмесіне келетін жылу балансын құрамыз. Жылдың жылы мезгілінде: температура айырымы нәтижесінде келетін жылу $Q_{\text{қоршау}} = 163 \text{ Вт}$; күн радиациясынан $Q_p = 1265 \text{ Вт}$; адамдардан $Q_a^a = 171 \text{ Вт}$; жарықтану аспаптарынан $Q_{\text{жарықтану}} = 1170 \text{ Вт}$; оргтехника мен құрылғылардан $Q_{\text{құрал}} = 900 \text{ Вт}$. Бөлменің жылулық балансы жазда

$$Q_{\text{жылу.б}} = Q_{\text{қоршау}} + Q_p + Q_a^a + Q_{\text{жарықтану}} + Q_{\text{құрал}}$$

$$Q_{\text{жылу.б}} = 1265 + 171 + 163 + 1170 + 900 = 3669 \text{ Вт},$$

$$Q_{\text{жылу.б}} = 3,669 \times 3600 = 13208 \text{ кДж/сағ}.$$

Жылдың салқын мезгілінде: температура айырымы нәтижесінде жоғалатын жылу $Q_{\text{қоршау}} = -2041 \text{ Вт}$; күн радиациясынан келетін жылу $Q_p = 1265 \text{ Вт}$; адамдардан $Q_a^a = 209 \text{ Вт}$; жарықтану аспаптарынан $Q_{\text{жарықтану}} = 1170 \text{ Вт}$; оргтехника және құрылғылардан $Q_{\text{құрал}} = 900 \text{ Вт}$. Бөлменің жылулық балансы қыста

$$Q_{\text{жылу.б}} = Q_{\text{қоршау}} + Q_p + Q_a^a + Q_{\text{жарықтану}} + Q_{\text{құрал}}$$

$$Q_{\text{жылу.б}} = 1265 + 209 - 2041 + 1170 + 900 = 1503 \text{ Вт},$$

$$Q_{\text{жылу.б}} = 1,503 \times 3600 = 5411 \text{ кДж/сағ}.$$

Ауа алмасуды есептеу

$Q_{\text{жылу.б жазда}} > Q_{\text{жылу.б қыста}}$ болғандықтан, $Q_{\text{жылу.б жазда}}$ мәнiмен ауаның жылу кернеулігін мына формуламен есептейміз

$$Q_k = \frac{Q_{\text{жылу.б}} \cdot 860}{V_{\text{бөлме}}} \quad (7.6)$$

$$Q_k = \frac{3,669 \cdot 860}{6 \cdot 6 \cdot 3} = 29,22 \frac{\text{ккал}}{\text{м}^3}$$

$Q_k > 20$ ккал/м³ болғанда $\Delta t = 8^\circ \text{C}$.

Бөлмеге қажет ауаның мөлшері жылулық баланстан алынып, мына формуламен анықталады

$$L = \frac{Q_{\text{жылу.б}} \cdot 860}{C \cdot \Delta t \cdot \gamma} \quad (7.7)$$

$$L = \frac{3,669 \cdot 860}{0,24 \cdot 8 \cdot 1,206} = 1363 \text{ м}^3/\text{сағ}.$$

мұнда $C = 0,24$ ккал/кг⁰С - ауаның жылу сыйымдылығы;

$\gamma = 1,206$ кг/м³ - ағынды ауаның сыбағалы массасы.

Барлық артық жылулар 13208 кДж/сағ немесе 13208 : 3600 = 3,6 кВт құрайды. Бөлмеге қажетті ауа мөлшері $L = 1363 \text{ м}^3/\text{ч} = 22,7 \text{ м}^3/\text{мин}$. Өз таңдауымызды LG S 12 ЛНК сплит-жүйесі кондиционеріне тоқтатамыз, салқын өндіргіштігі – 3,5 кВт.

Кондиционердің техникалық сипаттамалары:

- салқын 3,5 кВт; жылу 3,5 кВт;
- салқын кКал/сағ 3 024 (3 517);
- БТЕ/сағ 12 000;
- жылу кКал/сағ 3 150 (3 663);
- БТЕ/сағ 12 500;
- орек кернеуі В, Гц 1,220~240,50;
- салқынның жұмсайтын қуаты Вт 1 250;
- жылудың жұмсайтын қуаты Вт 1 140;
- салқын/жылу жұмыс тогы А 5.6 /5.1;
- EER (БТЕ/сағ)/Вт (Вт/Вт) 9.60 (2.81);
- COP Вт 3.2;
- ауа шығыны, ішкі, м3/мин(CFM) 9.5 (335);
- ауа шығыны, сыртқым3/мин(CFM) 23 (812);
- дегидратация л/сағ 1.5 (3.19);
- шудың деңгейі, ішкі (жоғ/орт/төм), дБ(А) 36/33/29;
- шудың деңгейі, сыртқы, дБ(А) 48;
- габаритті өлшемдері, Ш/В/Г, Ішкі, мм 900x285x156;
- габаритті өлшемдері, Ш/В/Г, сыртқы, мм 717x481x288;
- салмағы, кг 8/25.

Қорытынды

Бұл дипломдық жобада барлық қойылған мақсаттар орындалды. Мәзірдегі барлық тағамдар мен сусындардың мәліметтер қоры құрылды. Дерекқорды құру барысында барлық кемшіліктер ескерілді. Беріліп отырған бағдарламалық «Nash Pub» кафесінің желідегі ақпарат жүйесін автоматтандыруға мүмкіндік береді.

Деректер қоры MS SQL Server 2008 нұсқасы дерекқоры басқару жүйесінде жасалды. Жаңа түскен тапсырыстарды өңдеуді тек қызметкер жасайалады. Мұндай қатынау принципі кафе қызметкерлері үшін ыңғайлы. Сонымен қатар, администратор деректердің тұтастығы мен қауіпсіздігін қамтамасыз етеді.

«Тіршілік қауіпсіздігі» бөлімінде есептеулер нәтижесінде кондиционердің жыл бойына жұмысын қамтамасыз ету үшін таңдауымыз LG фирмасының кондиционеріне тоқтады, себебі бұл фирманың кондиционерлері түрлі жыл мезгілінде «сплит-жүйесі» тәртібінде тиімді жұмыс істейді.

Техника-экономикалық негізделуі бөлімінде жасалған есептеулер нәтижесінде бағдарламалық өнімді жүзеге асыруға кеткен шығын 480000 теңгені құрады. Әсіресе ірі мекемеде кең таралған, сондықтан жүзеге асырылуының жоғары бағасына қарамастан, берілген өнім экономикалық тиімді болып саналады.

Әдебиеттер тізімі

- 1 К.Дж. Дейт Введение в системы баз данных/ Дейт К.Дж. Диалектика,1998. - 784с.
- 2 Д. Петкович Microsoft SQL Server 2012. Руководство для начинающих. –БХВ-Петербург, 2013. – 816с.
- 3 Дарахвелидзе П.Г. Программирование в Delphi 7. / П.Г. Драхвелидзе, Е.П. Марков – СПб.: БХВ-Петербург, 2003. – 784 с.
- 4 Сатимова Е.Г. Проектирование баз данных. Методические указания к выполнению лабораторных работ для студентов специальностей 050704 – Вычислительная техника и программное обеспечение и 050703 – Информационные системы. – Алматы: АИЭС, 2009
- 5 Марк Шпеник, Оррин Следж. Microsoft SQL Server 2000 DBA Survival Guide. – М.: Вильямс, 2001. – 236 с.
- 6 Хомоненко А. Самоучитель Delphi / А. Хомоненко, В. Гофман. – СПб.: БХВ – Петербург, 2003г. – 556 с
- 7 Мамаев Е. MySQL Server. Проектирование и реализация баз данных. Сертификационный экзамен. – СПб.: ВHV, 2004. – 416с.
- 8 Базылов Қ.Б., Алибаева С.А., Нурмагамбетова С.С. Бітіруші жұмысының экономикалық бөлімі үшін әдістемелік нұсқаулар. – Алматы: АИЭС, 2009. – 25 б.
- 9 Иванов И.Н. Экономика промышленного предприятия: Учебник. – М.: ИНФРА–М, 2011. – 395 с.
- 10 Рофе А.И. Экономика труда: Учебник. – М.: КноРус, 2010. – 400 с.
- 11 Остапенко Ю.М. Экономика труда: Учеб. пособие. – М.: ИЦ РИОР, 2010. – 160 с.
- 12 Санатова Т.С., Абдимуратов Ж.С. Өміртіршілігінің қауіпсіздігі. – Алматы: АЭЖБИ, 2008. – 16 б.
- 13 ГОСТ 12.2.032-78 ССБТ. Рабочее место при выполнении работ сидя.
- 14 Баклашов Н.И., Китаева Н.Ж., Терехов Б.Д. Охрана труда на предприятиях связи и охрана окружающей среды. – М.: Радио и связь, 1989. –288 с.

Қосымша А

Бағдарламаның листингі

1.php

```
<html>
<head>
<title>Отдел магистратуры и докторантуры </title>
<frameset cols="250,750">
<frame src="menu.php" >
<frameset rows="200,800">
<frame src="logo.php">
<frame src="conc.php" name="window-1">
</frameset>
</frameset>
</head>
</html>
```

conc.php

```
<html>
<head></HEAD>
<BODY background= "foto knig/5.jpg">
<CENTER>
<BR> <FONT SIZE=6 COLOR="braun" FACE="Times New Roman">
<B> Департамент магистратуры, докторантуры и международного сотрудничества
</B> </FONT>
<BR> <BR><BR>
<a href="DMDiMC.htm" target="window-1">
<BR>
</a>
&nbsp; <FONT SIZE=5 COLOR="blue" FACE="Times New Roman">
<B><i> Для тех кто верит в себя и стремиться к успеху</i></B> </FONT>
</CENTER>
</BODY>
</HTML>
```

menu.php

```
<html>
<head>
</HEAD>
<BODY background= "foto knig/1.jpg">
<FONT SIZE=5 COLOR=braun FACE= "Arial">
<MENU>
<a href="http://www.turan.edu.kz/" target="window-1"> </a>
<BR><BR><BR><BR><BR><BR><BR><BR><BR><BR><BR><BR><BR>
<L1><A HREF= "Table/Univer.php" target="window-1">Университеты</A>
<BR>
<L1><A HREF="Table/facultet.php" target="window-1">Факультеты</A>
<BR>
<L1><A HREF="Table/kafedra.php" target="window-1">Кафедры</A>
<L1><A HREF="Table/cpecialnost.php" target="window-1">Специальности </A>
```

А қосымшаның жалғасы

```
<BR>
<L1><A HREF="Table/prepodovateli.php"target="window-1">Преподаватели </A>
<BR>
<L1><A HREF="Table/auditory.php"target="window-1">Аудитории </A>
<BR>
<L1><A HREF="Table/discipline.php"target="window-1">Дисциплины </A>
<BR>
<L1><A HREF="Table/block_discipline.php"target="window-1">Блок дисциплины</A>
<BR>
<L1><A HREF="Table/gruppa.php"target="window-1">Группа</A>
<BR>
<L1><A HREF="Table/rup.php"target="window-1">РУП</A>
<BR>
<L1><A HREF="Table/typu_auditory.php"target="window-1">Тип аудиторий</A>
</MENU></FONT>
<BR>
</BODY>
</HTML>
logo.php
<html>
<head>
<title>Обработка учебного плана </title>
<body background= "foto knig/1.jpg">
<CENTER> <FONT SIZE=20 COLOR=RoyalBlue FACE= "Arial">
<B> Положение учебного плана </B>
</FONT>
<HR WIDTH=80% SIZE=8 NOSHADE ALIGN =center COLOR=LightBlue3>
</CENTER>
<CENTER><MARQUEE HEIGHT=20% WIDTH=100% SCROLLAMOUNT=3%
SCROLLDELAY=10 FONT= "Arial" BGCOLOR= "SkyBlue">
<FONT COLOR=Braun SIZE=5>
<B>Департамент магистратуры, докторантуры и международного
сотрудничества</B>
</FONT>
</CENTER>
</body>
</head>
</html>
config.php
<?php
$link=mysql_connect("localhost", "root","123") or die("Ошибка
подключения:".mysql_error());
mysql_query('SET NAMES utf8');
mysql_select_db("rup_mag") or die("ошибка не подключился в базу");?>
univer.php
<html>
<head></head>
<body>
<?php
```

```
$ath = mysql_query("select * from universitet;");
if($ath)
  {//Определяем таблицу и заголовок
  echo "<table border=1>";
  echo"<tr>
  <td>код</td>
  <td>название</td>
  <td>адрес</td>
  <td>телефон</td>
  </tr>";
  //Так как запрос возвращает несколько строк, применяем цикл
  while($author = mysql_fetch_array($ath))
  {echo "<tr>
  <td>".$author['Kod_univera']. "&nbsp;</td>
  <td>".$author['Name']. " &nbsp;</td>
  <td>".$author['Adres_univera']. " &nbsp;</td>
  <td>".$author['telephone']. " &nbsp;</td>
  </tr>";}
  echo "</table>";}
else
  {echo "<p><b>Error: " .mysql_error () . "</b><p>";
  exit () ;}?>
  <br><br><br><br><br><br>
  <table border=1>
  <form action = "formma_dob_univer.php" method = "post">
  <input name="submit" type="submit" value="Добавить запись">
  </form>
  <form action = "form_upd_univ.php" method = "post">
  <input name="submit" type="submit" value="Изменить">
  </form>
  <form action = "deluniver.php" method = "post">
  <input name="submit" type="submit" value="Удалить">
  </form>
  <form action = "Pech_Univer.php" method = "post">
  <input name="button1" type="button" value="Версия для печати" onClick="myWin=
  open('Pech_Univer.php','displayWindow','width=400,height=300,status=no,toolbar=no,menubar=yes');">
  </form>
  </table>
  </body>
  </ht
  <b>facultet.php
  <html>
  <body>
  <?php
  include "config.php";
  $ath = mysql_query("SELECT facultet.ID_Facul, facultet.Name_Facul,
  facultet.Adres_Facul, facultet.Phone_Facul, universitet.Name AS universitet FROM facultet,
  universitet WHERE universitet.Kod_univera=facultet.FK_Kod_univera;");
```

```
{echo "<table border=1>";
echo"<tr>
<td>Код</td>
<td>Название</td>
<td>Место</td>
<td>Телефон</td>
<td>Университет</td>
</tr>";
while($author = mysql_fetch_array($ath))
{echo "<tr>
<td>".$author['ID_Facul']."&nbsp;</td>
<td>".$author['Name_Facul']." &nbsp;</td>
<td>".$author['Adres_Facul']." &nbsp;</td>
<td>".$author['Phone_Facul']." &nbsp;</td>
<td>".$author['universitet']." &nbsp;</td>
</tr>";}
echo "</table>";}
else
{echo "<p><b>Error: " .mysql_error () . "</b><p>";
exit () ;}
?>
<br><br><br><br><br><br>
<table border=1>
<form action = "Form_dob_Facul.php" method = "post">
<input name="submit" type="submit" value="Добавить запись">
</form>
<form action = "form_upd_facult.php" method = "post">
<input name="submit" type="submit" value="Изменить">
</form>
<form action = "delfacul.php" method = "post">
<input name="submit" type="submit" value="Удалить">
</form>
<form action = "Pech_facul.php" method = "post">
<input name="button1" type="button" value="Версия для печати" onClick="myWin=
open('Pech_facul.php','displayWindow','width=400,height=300,status=no,toolbar=no,menubar=yes'
);">
</form>
</table>
</body>
</html>
kafedra.php
<html>
<body>
<?php
include "config.php";
$ath = mysql_query("SELECT kafedra.Kod_Kafedri, kafedra.Name_kafedri,
kafedra.Adres_kafedri, kafedra.telephone_kafedri, facultet.Name_Facul AS facultet FROM kafedra,
facultet WHERE facultet.ID_Facul=kafedra.FK_facul;");
if($ath)
```

```
echo"<tr>
<td>Код</td>
<td>Название</td>
<td>Адрес</td>
<td>Телефон</td>
<td>Факультет</td>
</tr>";
while($author = mysql_fetch_array($ath))
{echo "<tr>
<td>".$author['Kod_Kafedri']."&nbsp;</td>
<td>".$author['Name_kafedri']."&nbsp;</td>
<td>".$author['Adres_kafedri']."&nbsp;</td>
<td>".$author['telephone_kafedri']."&nbsp;</td>
<td>".$author['facultet']."&nbsp;</td>
</tr>";}
echo "</table>";}
else
{echo "<p><b>Error: ".mysql_error () . "</b><p>";
exit () ;}
?>
<br><br><br><br><br><br>
<table border=1>
<form action = "formma_dob_kafedra.php" method = "post">
<input name="submit" type="submit" value="Добавить запись">
</form>
<form action = "update_kafedra.php" method = "post">
<input name="submit" type="submit" value="Изменить">
</form>
<form action = "delkafedra.php" method = "post">
<input name="submit" type="submit" value="Удалить">
</form>
<form action = "Pech_kaf.php" method = "post">
<input name="button1" type="button" value="Версия для печати" onClick="myWin=
open('Pech_kaf.php','displayWindow','width=400,height=300,status=no,toolbar=no,menubar=yes');
">
</form>
</table>
</body>
</html>
prepodovateli.php
<html>
<body>
<?php
include "config.php";
$ath = mysql_query("SELECT prepodovateli.id_prepod, prepodovateli.FIO,
prepodovateli.Telephone, prepodovateli.doljnost_prepod, kafedra.Name_kafedri AS kafedra FROM
prepodovateli, kafedra WHERE kafedra.Kod_Kafedri=prepodovateli.FK_Kod_Kafedri;");
if($ath)
{
```

```
echo"<tr>
<td>Код</td>
<td>ФИО</td>
<td>Телефон</td>
<td>Должность</td>
<td>Кафедра</td>
</tr>";
while($author = mysql_fetch_array($ath))
{echo "<tr>
<td>".$author['id_prepod']."&nbsp;  </td>
<td>".$author['FIO']." &nbsp;  </td>
<td>".$author['Telephone']." &nbsp;  </td>
<td>".$author['doljnost_prepod']." &nbsp;  </td>
<td>".$author['kafedra']." &nbsp;  </td>
</tr>";}
echo "</table>";}
else
{echo "<p><b>Error: ".mysql_error () . "</b><p>";
exit () ;}
?>
<br><br><br><br><br><br>
<table border=1>
<form action = "Form_dob_prepod.php" method = "post">
<input name="submit" type="submit" value="Добавить запись">
</form>
<form action = "update_prepodovateli.php" method = "post">
<input name="submit" type="submit" value="ИЗМЕНИТЬ">
</form>
<form action = "delprepodovateli.php" method = "post">
<input name="submit" type="submit" value="УДАЛИТЬ">
</form>
<form action = "Pech_prepod.php" method = "post">
<input name="button1" type="button" value="Версия для печати" onClick="myWin=
open('Pech_prepod.php','displayWindow','width=400,height=300,status=no,toolbar=no,menubar=ye
s');">
</form>
</table>
</body>
</html>
cpesialnost.php
<html>
<body>
<?php
include "config.php";
$ath = mysql_query("SELECT cpesialnost.Kod_cpes, cpesialnost.Name_cpes,
cpesialnost.Otdelenia, kafedra.Name_kafedri AS kafedra FROM cpesialnost, kafedra WHERE
kafedra.Kod_Kafedri=cpesialnost.FK_Kod_Kafedri;");
if($ath)
{echo "<table border=1>";
```



```
<td>Код</td>
<td>Название</td>
<td>Отделения</td>
<td>Кафедра</td>
</tr>";
while($author = mysql_fetch_array($ath))
{echo "<tr>
<td>".$author['Kod_cpes']."&nbsp;</td>
<td>".$author['Name_cpes']." &nbsp;</td>
<td>".$author['Otdelenia']." &nbsp;</td>
<td>".$author['kafedra']." &nbsp;</td>
</tr>";}
echo "</table>";}
else
{echo "<p><b>Error: " .mysql_error () . "</b><p>";
exit () ;}
?>
<br><br><br><br><br><br>
<table border=1>
<form action = "form_dob_cpes.php" method = "post">
<input name="submit" type="submit" value="Добавить запись">
</form>
<form action = "update_cpesialnost.php" method = "post">
<input name="submit" type="submit" value="Изменить">
</form>
<form action = "delcpesialnost.php" method = "post">
<input name="submit" type="submit" value="Удалить">
</form>
<form action = "Pech_cpes.php" method = "post">
<input name="button1" type="button" value="Версия для печати" onClick="myWin=
open('Pech_cpes.php','displayWindow','width=400,height=300,status=no,toolbar=no,menubar=yes')
;">
</form>
</table>
</body>
</html>
discipline.php
<html>
<body>
<?php
include "config.php";
$ath = mysql_query("SELECT discipline.Kod_discipline, discipline.Name_discipline,
kafedra.Name_kafedri AS kafedra FROM discipline, kafedra WHERE
kafedra.Kod_Kafedri=discipline.FK_Kod_Kafedri;");
if($ath)
{echo "<table border=1>";
echo"<tr>
<td>Код</td>
<td>Название</td>
```

```
</tr>";
while($author = mysql_fetch_array($ath))
{echo "<tr>
<td>".$author['Kod_discipline']."&nbsp;</td>
<td>".$author['Name_discipline']." &nbsp;</td>
<td>".$author['kafedra']." &nbsp;</td>
</tr>";}
echo "</table>";}
else
{echo "<p><b>Error: " .mysql_error () . "</b><p>";
exit () ;}
?>
<br><br><br><br><br><br>
<table border=1>
<form action = "form_dob_dis.php" method = "post">
<input name="submit" type="submit" value="Добавить запись">
</form>
<form action = "update_discipline.php" method = "post">
<input name="submit" type="submit" value="Изменить">
</form>
<form action = "deldiscipline.php" method = "post">
<input name="submit" type="submit" value="Удалить">
</form>
<form action = "Pech_dis.php" method = "post">
<input name="button1" type="button" value="Версия для печати" onClick="myWin=
open('Pech_dis.php','displayWindow','width=400,height=300,status=no,toolbar=no,menubar=yes');"
>
</form>
</table>
</body>
</html>
auditory.php
<html>
<body>
<?php
include "config.php";
$ath = mysql_query("SELECT auditory.Id_auditory, auditory.Name_auditory,
auditory.Volium, kafedra.Name_kafedri AS kafedra, typu_auditory.Name_type_aud AS
typu_auditory FROM auditory, kafedra, typu_auditory WHERE
(kafedra.kod_Kafedri=auditory.FK_Kod_Kafedri) and
(typu_auditory.Id_Type_aud=auditory.FK_Type_auditory)");
if($ath)
{echo "<table border=1>";
echo"<tr>
<td>код</td>
<td>название</td>
<td>Объем</td>
<td>Кафедра</td>
<td>Тип аудиторий</td>
```

```
while($author = mysql_fetch_array($ath))
{echo "<tr>
<td>".$author['Id_auditory']."&nbsp;</td>
<td>".$author['Name_auditory']." &nbsp;</td>
<td>".$author['Volium']." &nbsp;</td>
<td>".$author['kafedra']." &nbsp;</td>
<td>".$author['typu_auditory']." &nbsp;</td>
</tr>";}
echo "</table>";}
else
{echo "<p><b>Error: " .mysql_error () . "</b><p>";
exit () ;}
?>
<br><br><br><br><br><br>
<table border=1>
<form action = "form_dob_aud.php" method = "post">
<input name="submit" type="submit" value="Добавить запись">
</form>
<form action = "update_auditory.php" method = "post">
<input name="submit" type="submit" value="Изменить">
</form>
<form action = "delauditory.php" method = "post">
<input name="submit" type="submit" value="Удалить">
</form>
<form action = "Pech_aud.php" method = "post">
<input name="button1" type="button" value="Версия для печати" onClick="myWin=
open('Pech_aud.php','displayWindow','width=400,height=300,status=no,toolbar=no,menubar=yes');
">
</form>
</table>
</body>
</html>
block_discipline.php
<html>
<body>
<?php
include "config.php";//Подключение к серверу и выбор базы данных
$ath = mysql_query("select * from block_discipline;");
if($ath)
{echo "<table border=1>";
echo"<tr>
<td>код</td>
<td>название</td>
</tr>";
while($author = mysql_fetch_array($ath))
{echo "<tr>
<td>".$author['Id_block_dis']."&nbsp;</td>
<td>".$author['Name_block']." &nbsp;</td>
</tr>";}
}
```

```
echo "</table>";}
else
{echo "<p><b>Error: " .mysql_error () . "</b><p>";
exit () ;}
?>
<br><br><br><br><br><br>
<table border=1>
<form action = "formma_dob_block_discipline.php" method = "post">
<input name="submit" type="submit" value="Добавить запись">
</form>
<form action = "update_block_discipline.php" method = "post">
<input name="submit" type="submit" value="Изменить">
</form>
<form action = "delblock_discipline.php" method = "post">
<input name="submit" type="submit" value="Удалить">
</form>
<form action = "Pech_block.php" method = "post">
<input name="button1" type="button" value="Версия для печати" onClick="myWin=
open('Pech_block.php','displayWindow','width=400,height=300,status=no,toolbar=no,menubar=yes'
);">
</form>
</table>
</body>
</html>
```

gruppa.php

```
<html>
<body>
<?php
include "config.php";
$ath = mysql_query("SELECT gruppa.ID_Grup, gruppa.Name_grup, gruppa.Otdelenia,
gruppa.Kurs, prepodovateli.FIO AS prepodovateli,gruppa.Kol_stud,kafedra.Name_kafedri AS
kafedra, cpesialnost.Name_cpes AS cpesialnost FROM gruppa, prepodovateli, kafedra, cpesialnost
WHERE (prepodovateli.id_prepod=gruppa.FK_ID_Edvaizer) and
(kafedra.Kod_Kafedri=gruppa.FK_Kod_Kafedri) and
(cpesialnost.Kod_cpes=gruppa.FK_ID_Cpes)");
if($ath)
{
echo "<table border=1>";
echo"<tr>
<td>Код</td>
<td>Название</td>
<td>Отделения</td>
<td>Курс</td>
<td>Эдвайзер</td>
<td>Количество студентов</td>
<td>Кафедра</td>
<td>Специальность</td>
</tr>";
while($author = mysql_fetch_array($ath))
```

```
{echo "<tr>
<td>".$author['ID_Grup']. "&nbsp;</td>
<td>".$author['Name_grup']. " &nbsp;</td>
<td>".$author['Otdelenia']. " &nbsp;</td>
<td>".$author['Kurs']. " &nbsp;</td>
<td>".$author['prepodovateli']. " &nbsp;</td>
<td>".$author['Kol_stud']. " &nbsp;</td>
<td>".$author['kafedra']. " &nbsp;</td>
<td>".$author['cpesialnost']. " &nbsp;</td>
</tr>";
}
echo "</table>";
}
else
{
echo "<p><b>Error: " .mysql_error () . "</b><p>";
exit () ;
}
?>
<br><br><br><br><br><br>
<table border=1>
<form action = "form_dob_grup.php" method = "post">
<input name="submit" type="submit" value="Добавить запись">
</form>
<form action = "update_gruppa.php" method = "post">
<input name="submit" type="submit" value="Изменить">
</form>
<form action = "delgruppa.php" method = "post">
<input name="submit" type="submit" value="Удалить">
</form>
<form action = "Pech_grup.php" method = "post">
<input name="button1" type="button" value="Версия для печати" onClick="myWin=
open('Pech_grup.php','displayWindow','width=400,height=300,status=no,toolbar=no,menubar=yes')
;">
</form>
</table>
</body>
</html>
typu_auditory.php
<html>
<body>
<?php
include "config.php";
$ath = mysql_query("select * from typu_auditory;");
if($ath)
{
echo "<table border=1>";
echo"<tr>
<td>Код типа аудиторий</td>
```

```
<td>Название типа аудиторий</td>
</tr>";
while($author = mysql_fetch_array($ath))
{
echo "<tr>
<td>".$author['Id_Type_aud']."&nbsp;</td>
<td>".$author['Name_type_aud']." &nbsp;</td>
</tr>";
}
echo "</table>";
}
else
{
echo "<p><b>Error: " .mysql_error () . "</b><p>";
exit () ;
}
?>
<br><br><br><br><br><br>
<table border=1>
<form action = "formma_dob_tpyu_auditory.php" method = "post">
<input name="submit" type="submit" value="Добавить запись">
</form>
<form action = "update_tpyu_auditory.php" method = "post">
<input name="submit" type="submit" value="Изменить">
</form>
<form action = "del_tpyu_auditory.php" method = "post">
<input name="submit" type="submit" value="Удалить">
</form>
<form action = "Pech_tpyuaud.php" method = "post">
<input name="button1" type="button" value="Версия для печати" onClick="myWin=
open('Pech_tpyuaud.php','displayWindow','width=400,height=300,status=no,toolbar=no,menubar=y
es');">
</form>
</table>
</body>
</html>
rup.php
<html>
<body>
<?php
include "config.php";//Подключение к серверу и выбор базы данных
$ath = mysql_query("SELECT Id_RUP, Name_rup, table_rup, kafedra.Name_kafedri AS
kafedra FROM rup, kafedra WHERE kafedra.Kod_Kafedri=rup.FK_Kod_kafedri");
if($ath)
{
echo "<table border=1>";
echo"<tr>
<td>Код</td>
<td>Наименование РУПа</td>
```


Қосымша Ә

Бағдарламаның Delphi-дегі листингі

Form_dob_Facul.php

```
<form action="dob_Facul.php" method="post">
Наименование Факультета:
<br><input name="Name_Facul" type="text"><br>
Адрес Факультета:
<br><input name="Adres_Facul" type="text"><br>
Телефон Факультета:
<br><input name="Phone_Facul" type="text"><br>
<br><br>
<?php
include "config.php";
$result = mysql_query("SELECT * from universitet ORDER BY Name",$link);
$myrow = mysql_fetch_array($result);
print ("Университет: ");
print '<select name="universitet"><label>';
do
{
echo '<option value="'. $myrow['Kod_univera']. "'>' . $myrow['Name']. '</option>';
}
while($myrow=mysql_fetch_array($result));
echo '</SELECT>';
?>
<br><br><br>
<input name="submit" type="submit" value="Добавить запись">
</form>
<form action="facultet.php" method="post">
<input name="submit" type="submit" value="Вернуться назад">
</form>
```

dob_Facul.php

```
<html>
<head>
<title>Добавление записи в таблицу "Факультет"</title>
</head>
<body>
<?php
include "config.php"; //Подключение к серверу и выбор базы данных
$ath=mysql_query("SELECT * from facultet;") or die("Ошибка подключения:
".mysql_error());
$Name_Facul = $_POST['Name_Facul'];
$Adres_Facul = $_POST['Adres_Facul'];
$Phone_Facul = $_POST['Phone_Facul'];
$FK_Kod_univera=$_POST['universitet'];
$result = mysql_query("INSERT INTO facultet (Name_Facul, Adres_Facul, Phone_Facul,
FK_Kod_univera)
VALUES ('$Name_Facul', '$Adres_Facul', '$Phone_Facul', '$FK_Kod_univera')");
if($result == 'true')
{echo "Ваши данные успешно добавлены";}
else{echo "Ваши данные не добавлены";}
}
```


Ә қосымшаның жалғасы

```
mysql_close ( $link );
?>
<form action="facultet.php" method="post">
<input name="submit" type="submit" value="Обновить">
</form>
</body>
</html>
```

Қосымша – F

form_upd_facult.php

```
<form name="forma_upd_univ" action="editfacultet.php" method="post">
<br><br>
<?php
include "config.php";
$result = mysql_query("SELECT * from facultet ORDER BY Name_Facul",$link);
$myrow = mysql_fetch_array($result);
print ("Наименование факультета который вы бы хотели изменить: ");
print '<select name="facultet"><label>';
do
{
echo '<option value="'. $myrow['ID_Facul']. "'>'. $myrow['Name_Facul']. '</option>';
}
while($myrow=mysql_fetch_array($result));
echo '</SELECT>';
?>
<br><br><br>
<input name="submit" type="submit" value="Изменить запись">
</form>
<form action="facultet.php" method="post">
<input name="submit" type="submit" value="Вернуться назад">
</form>
```

editfacultet.php

```
<?php
include ("config.php");
$result1 = mysql_query("SELECT * from universitet ORDER BY Name",$link);
$myrow1 = mysql_fetch_array($result1);
echo "<form name='editfacultet' action='editfacultet.php' method='post'>";
print ("Выберите университет: ");
print '<select name="universitet"><label>';
do
{
echo '<option value="'. $myrow1['Kod_univera']. "'>'. $myrow1['Name']. '</option>';
}
while($myrow1=mysql_fetch_array($result1));
echo '</SELECT>';
?>
<?php
```

Ә қосымшаның жалғасы

```
$result2 = mysql_query("SELECT * from facultet where ID_Facul ='$_POST[facultet]",
$link);
$myrow2 = mysql_fetch_array($result2);
echo " <p>Наименование факультета: <input type=text name='Name'
value=$myrow2[Name_Facul]'></p>
<p>Адрес: <input type=text name='Adres_faculteta' value='$myrow2[Adres_Facul]'></p>
<p>Телефон: <input type=text name='telephone' value='$myrow2[Phone_Facul]'></p>
<input type='hidden' name='facultet' value='$_POST[facultet]'>
<input type='submit' id='submit' value='Изменить'>
</form>
";
$Name = $_POST['Name'];
$Adres_faculteta = $_POST['Adres_faculteta'];
$telephone = $_POST['telephone'];
$Univer = $_POST['universitet'];
if (isset($_POST['Name']))
{
$result2 = mysql_query("UPDATE facultet SET Name_Facul = '$Name', Adres_Facul =
'$Adres_faculteta', Phone_Facul = '$telephone', FK_Kod_univera = '$Univer' WHERE
ID_Facul='$_POST[facultet]'");
if($result2 == 'true')
echo "Данные успешно изменены!!!";
else{
echo "Данные не были изменены";
echo mysql_error();
}
}
?>
del_facul.php
<form action="delfacul.php" method="post">
<br><br>
<?php
include "config.php";
$result = mysql_query("SELECT * from facultet ORDER BY Name_Facul",$link);
$myrow = mysql_fetch_array($result);
print ("Наименование факультета которую вы бы хотели удалить: ");
print '<select name="facultet"><label>';
do
{
echo '<option value="'. $myrow['ID_Facul']. "'>'. $myrow['Name_Facul']. '</option>';
}
while($myrow=mysql_fetch_array($result));
echo '</SELECT>';
?>
<br><br><br>
<input name="submit" type="submit" value="Удалить">
</form>
<form action="facultet.php" method="post">
<input name="submit" type="submit" value="Вернуться назад">
```

Ә қосымшаның жалғасы

```
</form>
<?php
if(isset($_POST['facultet'])){
$result2 = mysql_query ("DELETE FROM facultet WHERE ID_Facul=$_POST[facultet]");
if ($result2)
echo "Данные успешно удалены из базы.<br>";
else
echo "Произошла ошибка. Данные не были удалены из базы.";
}
?>
Pech_facul.php
<html>
<head></head>
<body>
<?php
include "config.php";//Подключение к серверу и выбор базы данных
$ath = mysql_query("SELECT facultet.ID_Facul, facultet.Name_Facul,
facultet.Adres_Facul, facultet.Phone_Facul, universitet.Name AS universitet FROM facultet,
universitet WHERE universitet.Kod_univera=facultet.FK_Kod_univera;");
if($ath)
{
//Определяем таблицу и заголовок
echo "<table border=1>";
echo"<tr>
<td>Код</td>
<td>Название</td>
<td>Место</td>
<td>Телефон</td>
<td>Университет</td>
</tr>";
//Так как запрос возвращает несколько строк, применяем цикл
while($author = mysql_fetch_array($ath))
{
echo "<tr>
<td>".$author['ID_Facul']."&nbsp;</td>
<td>".$author['Name_Facul']." &nbsp;</td>
<td>".$author['Adres_Facul']." &nbsp;</td>
<td>".$author['Phone_Facul']." &nbsp;</td>
<td>".$author['universitet']." &nbsp;</td>
</tr>";
}
echo "</table>";
}
else
{
echo "<p><b>Error: " .mysql_error () . "</b><p>";
exit () ;
}
?>
```

```
<br><br><br><br><br><br>
</body>
</html>
Create_rups.php
<form name="create_rups.php" action="create_rups.php" method="post">
<p>Создать новую таблицу</p>
<p>Введите имя таблицы:</p>
<input type="text" name="Name_rup"><br>
<p>Введите название таблицы:</p>
<input type="text" name="table_rup">
<br>
<?php
include "config.php";
$result1 = mysql_query("SELECT * from kafedra ORDER BY Name_kafedri",$link);
$myrow = mysql_fetch_array($result1);
print ("Кафедра: ");
print '<select name="kafedra"><label>';
do
{ echo '<option value="'. $myrow['Kod_Kafedri']. "'>'. $myrow['Name_kafedri']. '</option>';
}
while($myrow=mysql_fetch_array($result1));
echo '</SELECT>';
?>
<input type="submit" value="Создать">
</form>
<?PHP
if (isset($_POST['Name_rup']))
{
$Name_rup = $_POST['Name_rup'];
include "config.php";
$result = mysql_query("CREATE TABLE $Name_rup (id mediumint NOT NULL
PRIMARY KEY AUTO_INCREMENT, Fk_Block_dis
mediumint, Kod_discipline mediumint, Kol_Creditov varchar(3), Total_hours varchar(3), Lecture
varchar(2), Laboratory
varchar(2), Seminar varchar(2), CRMP varchar(2), CRM varchar(2), Form_kontr varchar(10),
FIO_prepod mediumint,
FK_Kod_kafedri mediumint)", $link);
$result = mysql_query("CREATE INDEX Fk_Block_dis ON $Name_rup (Fk_Block_dis)");
$result1 = mysql_query("CREATE INDEX Kod_discipline ON $Name_rup
(Kod_discipline)");
$result2 = mysql_query("CREATE INDEX FIO_prepod ON $Name_rup (FIO_prepod)");
$result3 = mysql_query("CREATE INDEX FK_Kod_kafedri ON $Name_rup
(FK_Kod_kafedri)");
if($result)
echo "Таблица $Name_rup была успешно создана <form action='form_dob_table.php'
method='post'>
<input name='submit' type='submit' value='Добавить запись в созданную таблицу'>
</form> ";
else
```

Ә қосымшаның соңы

```
    echo mysql_error();
}
    $ath=mysql_query("SELECT * from rup;") or die("Ошибка подключения:
".mysql_error());
    $Name_rup=$_POST['Name_rup'];
    $table_rup = $_POST['table_rup'];
    $FK_Kod_kafedri = $_POST['kafedra'];
    $result1 = mysql_query("INSERT INTO rup (Name_rup, table_rup, FK_Kod_kafedri)
        VALUES ('$Name_rup', '$table_rup', '$FK_Kod_kafedri')",$link);

mysql_close ( $link );
?>
```