

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.К. Құралбаев

« _____ » _____ 2014 ж.
(колы)

ДИПЛОМДЫҚ ЖОБА

Тақырыбы: «ҚР мемлекеттік автомобильді инспекциясының деректер қорын құру»

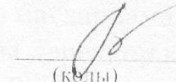
5В070400 – Есептеу техникасы және бағдарламалық камтамасыз ету мамандығы бойынша

Орындаған: ВТк-10-2 Мадғали Д.

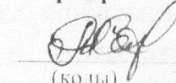
Жетекші: ф.-м.ғ.д., проф. Құралбаев З.К.

Кенесшілер :

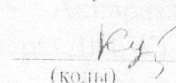
Экономикалық бөлім бойынша :

 « 02 » 06 доцент Боканова Г.Ш.
(колы) 2014 ж.

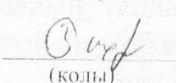
Өмір тіршілігі қауіпсіздігі бойынша:

 « 28 » 05 т.ғ.к., аға оқытушы Муташева Г.С.
(колы) 2014 ж.

Есептеу техникасын қолдану бойынша :

 « 2 » 06 ф.-м.ғ.д., профессор Құралбаев З.К.
(колы) 2014 ж.

Мөлшер бақылаушы:

 « 2 » 06 аға оқытушы Рахимжанова З.М.
(колы) 2014 ж.

Пікір жазушы :

« _____ » _____ ҚазҰТУ, п.ғ.д., проф. Сатыбалдиев О.С.
(колы) 2014 ж.

Алматы 2014

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

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

жобаны орындауға берілген

ТАПСЫРМА

Студент Мағғали Диас

Жоба тақырыбы «ҚР мемлекеттік автомобильді инспекциясының
деректер қорын құру»
ректордың « 24 » қаржылық 2013 № 115 бұйрығы бойынша
бекітілген.

Аяқталған жұмысты тапсыру мерзімі: « 3 » маусым 2014 ж.

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

MS SQL Server 2008 дерекқорларды басқару жүйесін және MS Visual
Studio бағдарламалау ортасын пайдалану арқылы компанияның қызметкерлері
туралы дерекқор көзі шифрланған ақпаратты 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.

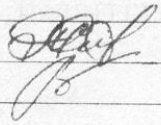
– Жүйенің дерекқорын жобалау және жүзеге асыру, жобалау
құралдарын таңдау, ER моделін жобалау, дерекқорды құру, пайдаланушы
интерфейсін жобалау, дерекқормен байланысын орнату.

– Техникалық-экономикалық негізделуі.

– Өндірістегі еңбек қорғау мен қауіпсіздік шараларын ұйымдастыру.

Сызба материалдарының (міндетті түрде дайындалатын сызуларды
көрсету) тізімі:

– Дерекқордың ER диаграммасы.

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

№ p/c	Тарау аттары, әзірленетін сұрақтардың тізімі	Жетекшіге ұсыну мерзімдері	Ескерту
1	Автоматтандырылған қызмет көрсету орнының анықтамасы (арналуы, құрылымы, т.б.).	11.03.14	
2	Ақпараттық жүйелер, дерекқорлар (ДК), дерекқорларды басқарудың жүйелері (ДҚБЖ) түсініктері.	26.03.14	
3	Жүйенің дерекқорын жобалау және жүзеге асыру, жобалау құралдарын таңдау, ER моделін жобалау.	3.04.14	
4	Дерекқорды құру, пайдаланушы интерфейсін жобалау, дерекқормен байланысын орнату.	17.04.14	
5	Жүйенің дерекқорын шифрлау түрін таңдау, жүзеге асыру	14.05.14	

Тапсырманың берілген уақыты « 25 » ақпан 2014ж.

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

Жоба жетекшісі _____ ф.-м.ғ.д., проф. Құралбаев З.К.
(қолы)

Орындалатын тапсырманы қабылдаған студент Мадғали Д. Мадғали Д.
(қолы)

Аңдатпа

Бұл дипломдық жұмыста ҚР мемлекеттік автомобильдік инспекциясының деректер қорын құру жүзеге асырылды. Деректер қорынан пайдаланушы барлық қызметкерлер жайында қажет ақпаратты алуға мүмкіндігі бар. Дипломдық жобада ақпараттық жүйе архитектурасы, ДҚ құрылымы және қызметкерлер бағдарламасының қосымшасын жасау мәселелері қарастырылған.

Өміртіршілік қауіпсіздігін қамтамасыз ету бөлімінде еңбек ету жағдайына талдау жасалған, жарық беру және желдету есептеуге алынған.

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

Аннотация

В данном дипломном проекте была разработана база данных государственной автомобильной инспекции РК. Пользователь может получить всю нужную информацию о сотрудниках из базы данных. В дипломном проекте рассмотрены такие вопросы, как разработка архитектуры информационной системы, структура БД и создание приложения.

В разделе обеспечения безопасности жизнедеятельности проведен анализ условий труда, сделан расчет освещения и кондиционирования.

В экономической части составлен бизнес план проекта, подтвердивший его экономическую целесообразность.

Annotation

A database was developed for state inspection of vehicular of RK. Information system's architecture development, database structure and creation of application were considered in this project. The user can get all necessary information about employees from this database.

In the section of living safety provision working conditions were analyzed , as well as the calculation of lighting and air conditioning was provided.

In the economic part of the project was compiled a business plan, which confirmed its economic practicability.

Мазмұны

Кіріспе	14
1. ДҚБЖ және оны іске асыру	15
1.1 ДҚБЖ дамуының жаңа тенденциялары.....	15
1.2 Деректер қорын қолданудың жаңа аймақтары	16
1.3 Мәліметтердің модельдік деңгейлері.....	19
1.4 Деректер қорымен жұмыс	20
1.4.1 ДҚ іздеу, сорттау және индекстеу, формаларды және есептерді құру.....	20
1.4.2 ДҚ физикалық ұйымдастыру, хештау және индекстеу файлдары. Журнализация.....	21
1.4.3 ДҚ тұтастығы және қорғау.....	21
2. SQL құрылымдық сұранымдар тілі.....	22
2.1 SQL даму тарихы.....	22
2.2 SQL мүмкіндіктері мен артықшылықтары.....	23
2.3 SQL көмегімен мәліметтер қорын басқару.....	25
2.3.1 Реляциялық мәліметтер қоры.....	25
2.3.2 Ұсыныстармен жұмыс	25
2.3.3 Сақталатын процедуралар, оларды құру, орындау және жою	27
2.3.4 Мәліметтерді басқару, кестедегі берілгендерді өзгерту	29
2.4 ДҚ диаграммасы.....	31
3 VISUAL BASIC программалау тілі	32
3.1 VISUAL BASIC туралы жалпы түсінік.....	32
3.2 VB-те программалау	32
3.3 Қолданушының графиктік интерфейсін құру	33
3.3.1 Графикалық интерфейсін формаларымен жұмыс.....	33
3.3.2 Программаны жүктеу.....	34
3.3.3 Программаға код қосу.....	35
3.3.4 Программаны дискіде сақтау.....	36
4 Қосымша интерфейсін сипаттау. Қосымшамен жұмыс істеу	37
5 Тіршілік қауіпсіздігі.....	41
5.1 Инженердің жұмыс жағдайын талдау	41
5.2 Кондиционерлеу және ауаны жаңарту жүйелерін есептеу.....	45
6 Техника – экономикалық негізделуі	51
6.1 Жұмысты сипаттау және қажеттілігін негіздеу	51
6.2 Нарықты саралау және орнын анықтау	51
6.3 Бағдарламаның SWOT-анализі.....	51
6.4 Маркетинг-микс элементі.....	52
6.5 Бағдарламалық қамтаманың өңделуінің еңбек өнімділігінің есептелуі.....	52
6.5.2 Еңбекақыға жұмсалатын шығындар	54
6.5.3 Қосымша шығындар статьясы	54
6.5.4 Өзіндік құн нәтижесінің кестесі мен диаграммасы	57

6.6 Бағдарламаның бағасына есептеу.....	58
6.6.3 Бағдарламалық өнімді иемденудегі кәсіпорындардың бірмезгілдегі шығындарын есептеу	58
6.6.4 Қолдану саласындағы жылдық шығындарды есептеу Ақпараттық өнімді енгізуден алынған үнемдеу мөлшері мен табысты есептеу.....	59
6.7 Ақшалай құралдардың қозғалысы.....	60
6.8 Экономикалық тиімділікті есептеу	61
6.8.1 Таза ағымдағы құндылықты есептеу (Net present value, NPV).....	61
6.8.2 Пайда индексін есептеу (Profitability index, PI)	61
6.8.3 Табыстың ішкі нормасын есептеу (Internal rate of return, IRR)	62
6.8.4 Өтімділік периодын есептеу (Payback period, PBP).....	62
Қорытынды	64
Әдебиеттер тізімі.....	65
Қосымша А.....	66
Қосымша Ә.....	74

Кіріспе

Берілген дипломдық жұмыста ҚР мемлекеттік автомобильдік инспекциясының жетілдіріген деректер қорын құру жобаланды.

Берілген өнімді әзірлеудің мақсаты деректерді енгізу, сақтау, өңдеу және тасымалдау үшін қолданылатын ақпарат пен қорлардың тұтастығын, қол жеткізулік оңтайлығын қолдау болып табылады. Бұл бағдарлама көмегімен жүргізушілердің өз бетінше айыппұлды қарау мүмкіншілігі бар.

Дерекқор құру үшін алдымен зерттелетін облыстың талдауы жасалады. Инфологиялық модель MS SQL Server жүйесінде жасалады. Қызметкерлердің дерекқоры MS SQL Server дерекқорларды басқару жүйесінде құрылғаннан кейін, пайдаланушылардың барлық сұраныстарын қанағаттандыратын қосымша жобаланады. Бағдарлама MS Visual Studio бағдарламалық ортасында іске асырылады.

Бағдарлама жасау нәтижесінде, жүргізушілер мен көлік және олардың айыппұлдарының деректер қорымен жұмыс жасаудың тиімділігі, жылдамдығы және ыңғайлылығы артады, ақпарат толықтығы қамтамасыз етіледі.

1. ДҚБЖ және оны іске асыру

1.1 ДҚБЖ дамуының жаңа тенденциялары

Ақпаратты технологиялар терминологияға қатысты екі өте қызық тенденциямен сүйемделеді. Бір жағынан біздер бір заттың бірнеше атаумен өзгеретінін байқаймыз (әрине, технология дамып жатыр, бірақ олардың атауы тез қарқынмен өзгеріп жатыр). Былай қарасақ біздер ескі терминдерді мағынасы басқа бұрынғыдай болмасада түсіну үшін қолданамыз. Екінші жағдай нақ ДҚБЖ-ға қатысты.

2002 жылы шыққан есептеу техникасының түсініктеме сөздігінде, деректер қорының басқару жүйесіне мынадай анықтама келтірілген (database management system): «деректер қорында ақпаратты құру, сақтау, жаңарту мен іздеуді қамтамасыз ететін қосымша және де деректердің бүтіндігі мен қауіпсіздігін басқару». Шынына келсе бұл түсінік 30 жыл бұрында расталған, бірақ қазір ДҚБЖ-ның маңызды бөлімінде бұрынғы кездегі мағынасына қарағанда басқа (сегіз жыл бұрын ұғымды анықтау үшін қолданған қосымша сөйлем анықтамада жоқ екенін айта кеткен жөн, — «пайдаланушы мен деректер қорының арасында орналасқан программалық қабықша»).

Соңғы онжылдықта біз ДҚБЖ-ның аса ішкі технологиялық толықтырулардан қолданбалы программалардың айналасында пайдаланушы үшін қосымша құрылып дербес өнімге айналғанын байқап отырмыз; басқаша айтқанда ақпараттық жүйенің бір компонентін осындай жүйені құру үшін платформаға айналдыру.

Бұл жағдайда «Microsoft платформасы» мазмұнының өзгеруіне мән берген жөн. Дәстүрлі түрде бұл терминде Windows операциялық жүйесі туралы айтылады. Бірақ серверлік платформаға сәйкес көбінесе Windows Server + SQL Server байланысын жиі кездестіреміз. Сонымен қатар, Microsoft SQL Server-дің (жұмыстық атауы Yukon) жаңа нұсқасы шығысымен біз Microsoft өнімдерінің Windows емес, ал Yukon деген атаумен аталатын жағдайға соқтығысамыз. Дегенмен үстелдік деректер қоры бар және болады: таңғаларлық жай Microsoft-тың ұсынысы бойынша Microsoft Access – бұл да ДҚБЖ. Тарихи тұрғыдан деректер қорының басқару жүйесі бірінші кезекпен құрылымдық ақпараттың транзакциялық өңдеуімен байланысты есептерді шығаруға дағдыланды. Сөзсіз, бұл жерде ең жақсы, уақытпен тексерілген шешім ДҚБЖ-ның реляционды моделі болып табылады. Бірақ соңғы уақытта деректер қорын қолдану аймақтары өзгеріссіз кеңейді. Бір жағынан, бірлескен ақпараттарды басқарудың жалпы мәселелерін шешуден өтіп, мәліметтер форматын кең терімде басқару қажет. Басқаша айтқанда — нақ ДҚБЖ өзіне мәліметтер интеграциясының негізгі функциялары мен бірлескен жүйе қосымшаларын алады. Нақ осы жағдай сәулеттік принципін талқылаудағы белсенді назар және постреляциондық, объектті-реляционды, XML сияқты әр түрлі моделді деректер қорын жүзеге асыру мүмкіндігімен түсіндіріледі.

1.2 Деректер қорын қолданудың жаңа аймақтары

Егер деректер қорын қолдану аймақтарын жіктеп тырысса, және де қазіргі уақыттағы олардың даму перспективасын бағаласа, онда деректер қорын қолданудың барлық аймақтарын қолдану және таратудағы кең таралым класстарының үлгілі тізімін аламыз. Бұл тізім келесі түрде келтіріледі:

- құжатты-графикалық және құжаттылық барлық басқару мен өкімет органдар қорында қолданылады;

- өнеркәсіп, құрылыс және ауылшаруашылық өнімдері байланысты деректер қоры;

- экономика және конъюктурлы ақпаратқа байланысты деректер қоры (статикалық, несие-финансылық, сыртқы саудалық);

- әлеуметтік орта және халық туралы мәліметтен құралған фактографиялық әлеуметтік деректер қоры;

- көлік жүйесінің деректер қоры;

- мекеме және халық үшін анықтама мәліметтер (энциклопедиялар және анықтамалар, расписания самолет және поезд кестесі, мекеме мен азаматтардың мекен-жайы және телефон нөмірлері);

- табиғи ресурс туралы фактографиялық ақпараттан тұратын факресурсты деректер қоры (жер, су, жер қойнауы, биокорлар, гидрометеорология, екінші қайтара қор және қалдықтар, экологиялық жағдай);

- ғылыми зерттеулер негізі қамтамасыз ететін ғылыми мәліметтердің фактографиялық қоры және ақпараттары ;

- лингвистикалық деректер қоры, яғни әр түрлі аймақтағы және түрдегі машиналы сөздіктер.

Экономикалық есептерді орындау үшін қажет ДҚБЖ программалық қамтамасы өте кең және әр түрлі. Оның негізінде әр түрлі деңгейдегі мекеменің ақпараттық жүйесі құрылады (кішкенесіне ірісіне дейін). Деректер қорын қолдану аймақтарын дәстүрлі түрде үлкен көлемдегі әр түрлі ақпараттармен соқтығысатын адам қызметінің аймақтары алып жатыр. Ең бірінші деректер қоры мынадай ғылым негіздерінде, ядролық физика, химия, космановтика және тағы басқа мәліметтермен жүйелік жұмысты талап ететін ғылымда қолданылды. Алдағы уақыттағы компьютерлік технология мен қоғамның компьютеризациялануының дамуы деректер қорының адам қызметінің барлық сферасында өңделіп және ауылшарушылықтан бастап финансты-экономикалы жүйеге дейін әр түрлі мекемелерде қолдануға болатын мүмкіндікке әкеп соқты. Деректер қорын қолданудың соңғы инновациясы болып өзінің мәні бойынша үлкен деректер қоры болып табылатын дүниежүзілік топ Internet табылады. Соған сәйкесті мұндай деректер қорын тарату олардың жаңа программалық құралдардың басқаруын талап етеді.

Деректер қорының жаңа құралдарын өңдеу қажеттілігі мен олардың қолдану мүмкіндіктерін анықтайтын жаңа ұрпақтың бірнеше қосымша мысалдары келтіріледі. Біз қысқаша осындай бес қосымшаны қарастырамыз.

- 1) Жерді бақылау жүйесінің деректер қоры (EOSDIS).

Жерді бақылау жүйесі (EOS - Earth Observing System) 1998 жылдан бастап NASA жіберген көптеген серіктерді ұсынады. Олардың тағайындауы – атмосфера, мұхит, жер қабатының жағдайының ұзақмерзімді тенденцияларын зерттеумен бос емес зерттеушілерге қажет ақпараттар жиынтығы. Серіктер жылына 1/3 Пбайт (Petabyte - 10¹⁵ байт) көлемінде ақпарат жеткізіп отырады. Бұл деректер бұрыннан бар ақпарат пен басқа қайнардағы мәліметпен (шет ел серіктері, жердегі бақылау станциялары) интегралданады деп болжалды және де EOSDIS (EOS Data and Information System) деректер қорында бұрын соңды болмаған масштабта жинастырылады.

EOSDIS маман және маман еместерге ақпаратты қызмет көрсетуге арналған. Мысалы, оған жанартау құбылысы әсерімен және т.б ауа-райы шартының құрылу үлгісімен танысуға тіпті оқушыларда рұқсат ала алады деп болжанды. Міне, бұл жобамен байланысты туындайтын ең қиын есептер осы.

Өз бетімен немесе регламентті сипатта (мысалы, күнделікті мәліметтердің жаңаруы) болатын сұраныс көлемі мен үлкен қарқындағы ақпараттың көптеген мыңдаған тұтынушыларының демеуі. Нәтижелі қарау мехмнизмі және қажет ақпаратты іздеуді өндіру.

2) Электронды сауда.

Қазіргі уақытта ортақ мақсаты - потенциалды тұтынушыларға келесі түрдегі сатып алудың электронды дайындауының тауар каталогына тез рұқсат беретін бірнеше жобалар бар. Ұқсас жүйелердің мүмкін аралық буыны болып электронды делдал деп болжанады. Делдалдар ақпаратты жинау жолымен көптеген қайнардан мәліметтерді шоғырлайды, мысалы, бірнеше каталогтан киімдік заттар. Соңғы сатып алушыға делдал сатып алудың тез дайындалуы жолын ұсынады.

EOSDIS жобасы сияқты, электронды сауда жүйесі сауда келісіміндегі көптеген қатысушылар санының желілік әрекеттестігін болжайды. Олардың айырмашылығы, EOSDIS-та бір бас жабдықтаушы мен оның көптеген тұтынушылары бар, ал сауда жүйесі көптеген жабдықтаушы мен көптеген тұтынушының барын түсіндіреді. Сонымен қатар, қатысушылар бұл жағдайда өзара анықталған сенімсіздікті сынауы мүмкін және де өзінің жеке жабық ақпараттық жүйесі болуы мүмкін. Осындай түрдегі жобалармен байланысты ең қиын мәселелер келесіде.

Электронды сауда жүйесінде таралған аутентификацияның жоғарысенімді құралдары мен ақшалай сома аударымы болуы қажет.

3) Денсаулық сақтаудың ақпараттық жүйесі.

Дәрігерге жұмыс жасау барысында көптеген ақпарат көздеріне рұқсат қажет. Мысалы, бір пациенттің ауру тарихы әр түрлі емханада, клиникада, сақтандыру мекемесінде болуы мүмкін. Толық мағлұматты алу үшін олардың бәрін жинау керек. Осындай сияқты дәрі-дәрмек, емдеу процедуралары, диагностика құралдары туралы мағлұмат беретін көптеген жүйелер мен деректер қоры бар.

Емдеуші дәрігердің жазбасы, тексерудің шешімі, емдеу төлемақысы туралы ақпарат, медициналық сақтандыру келісім шарты әр пациент үшін

электронды түрде бекітіліп, келесі қолдануға рұқсат етіліп қалуы тиіс. Денсаулық сақтау аймағына қазіргі заманға сай ақпараттық технологияны енгізу баға, сапа, жаппай қолжетерлік сияқты медициналық қызмет көрсетудің сипатына түбегейлі әсер етуі мүмкін. Міне, осындай ұқсас жүйелерді іске асырумен байланысты туындайтын бірнеше мәселелер.

Жиналған ақпараттардың әр текті көздерінің интеграциясы. Қажет жасырын деңгейін қамтамасыз ететін бақылау рұқсатының құралы. Денсаулық сақтау жұмыскерлерінің әр түрлі категориясы үшін ыңғайлы ақпаратқа қол жеткізу интерфейсы.

4) Электронды жариялымдар.

Баспа кәсіпкерлігінде денсаулық сақтау сферасындағыдай алдағы уақытта бірқатар терең қзгерістер күтіп тұр. Мысалы, кітаптар мен мақалаларды электронды түрде сақтау және жоғарыжылдамдықты желілік арналар арқылы оларды тұтынушыларға тез жеткізу мүмкін болып барады. Онан әрі, жариялым ұғымының өзі кеңейіп келеді – құжат графикалық, аудио немесе видео-қосылулар, андатпа, басқа жолдама элементтерден тұруы мүмкін. Бүгінгі күнде қол жетерлік ақпараттың жалпы көлемі EOSDIS деректер қорының өлшемінен асады, ал алдағы уақытта оның өлшемі бірнеше ретке өсуі мүмкін деп күтіледі.

Бұл өзгерудің табиғи салдары баспа мен білім беретін сфераны жақындатуы мүмкін. Аз ғана студенттерге оқылатын «тірі» лекциялар орнын «білім беретін өнімдер» алады – мәтіннен, аудио, видео мен басқа компоненттен тұратын және интерактивтік тренинг элементінен құралған электронды құжаттар. Мұндай өнім көптеген студенттер қажеттілігін қанағаттандыруы мүмкін. Мұндай перспективамен байланысты келесі зерттеу бағыттарын белгілеуге болады.

Үлкен көлемдегі, жоғары жылдамдықты мәліметтердің өңделуі мен жіберілуі. Типтік құжат мегабайттан гигабайтқа дейінгі диапазондағы мәліметтер объектісі өлшемінен құралған және нақты уақыт режимінде жеткізуді талап етеді. Интеллектуальды меншікті қорғау. Оны қайта сатуға тыйым салу, ақпаратты қолданғаны үшін кішігірім ақшалай қаражатты өндіріп алу түсіндіріледі. Үлкен көлемдегі ақпараттың ұйымы және оларға рұқсатты қамтамасыз ету.

5) Ұжымдық жобалау.

Ірі және қиын жобалар, мысалы, ұшаққұру аймағында бүгінгі таңда бірнеше тәуелсіз компаниялардың біріккен жігерімен іске асырылады. Осындай ұқсас жобаларға жататын ақпараттың өмір сүру уақыты онжылдықпен есептелуі мүмкін, себебі ол қолдауға, дамуға және модификацияға қажет. Алдымен физикалық шындық болмас бұрын, құрастырылған шешімдер - жұмыс қасиетін зерттеу, бұйымды құрастыру ыңғайлылығы, жұмыс жасаудың дұрыстығы үшін компьютерді үлгілеу сатысынан өтеді. Құрастыру сұлбасының эволюциясы бірінші бұйым шыққанға дейін басталады да, ұзақ уақытқа дейін жалғасады, содан кейін ол ағымдағы өңдеу жағдайын, тәжірибе жүзіндегі нұсқаны, тарихи дамуды көрсететін ақпараттық кескін үйлесімнің ұлғаюына әкеп соғады. Әр түрлі құрастыру сферасы үшін әр түрлі үлгі және белгілеу

жүйесі неізіндегі әр текті құрастыру аспаптық жүйесін қолдану сипатталады. Және де құрастыру процесі қолданылатын аспап болғанша ұзаққа созылуы мүмкін, яғни сол бір конструкция компоненті аспаптық жүйенің әр түрлі нұсқасының қолданысымен өңделуі мүмкін. Сол себептен, электронды жобалаумен байланысты келесі есептерді қисынға келтіруге болады.

Кейбір бұрын аталған сфера сияқты бұл жерде де бұрынғы жиналған ақпараттың әр текті көзінің интеграция есебі қозғалады.

Ұжымдық жобалау біріккен рұқсат басқаруының деректер қоры мен ақпараттың бөліну механизмінің жаңа үлгісін талап етеді.

Моделдеу мен құрастыру сияқты әр текті процестің бірге орындалуын жөнге салу үшін ұзақмерзімді транзакция арқылы анықталған өзара әрекеттестіктің негізіндегі жұмыс ағынының басқару құралы қажет.

1.3 Мәліметтердің модельдік деңгейлері

Кез келген деректер қорын құру үшін, құрушысы бірнеше модель жасайды. Бірінші модель - инфологиялық мәліметтер моделі. Бұл кезеңде деректер қорын құрушы пәндік аймақ туралы мәліметтер жинақтайды және сол жинақталған мәліметтерді табиғи тіл арқылы, кестелер арқылы, математикалық формулалар арқылы графикалар арқылы сипаттайды. Бұл кезде ДҚ құрушы ешқандай МҚБЖ қолданбайды. Екінші модель - мәліметтердің даталогиялық модулі - қандайда бір МҚБЖ тілінде ДҚ сипаттау. Үшінші модель - мәліметтердің физикалық моделі сипаттайды. Инфологиялық моделінің мақсаты жиналған модель ақпараттық жүйелік оны ДҚ қалай сақтайтындығын бейнелеу. Мысал ретінде мектептің базасын құруға болады.

Инфологиялық моделін иерархиялық ER - диаграммалары немесе (ақпараттық модель) деп те атайды. ER- диаграммаларда қолданатын белгілер: тіктөртбұрыштың ішінде-стержн.домалақта ключ. Домалақта атрибут. Трапецияда- характеристика. Ромбыда – ассоция. Параллелограмда – белгі. Бұнда бұл белгілер үшке бөлінеді. Біреуі – стерженді. Екіншісі – ассоциясы. Үшінші – мінездемелік. Ассоциативті қатынастарды ішкі классқа бөлінуі – белгілеу деп аталады.

МҚ көрсетілген физикалық нұсқаның кестесін немесе МҚ физикалық моделін қарастырайық. Кестені физикалық түрге келтірген кезде құрылымы инвариантты болып табылады, яғни кесте құрылымы өзгеріссіз қалады. Бірақ кесте параметрі нақты мәнді қабылдайды.

МҚ екі кестеден тұрады, оның біріншісі – мастер кесте, екіншісі – деталь. Олардың арасындағы қатынас бірді – көпке.

МҚ келесіден тұрады:

- МҚ кестесі;
- индекс;
- ескертулер.

Бір кестені сақтау үшін жеке файл құрылады. Осындай жеке файлдар кесте индексімен Метод өрісті сақтауға құрылады. Бұл файлдар бірге сақталып

орналасу қажет, әйтпесе МҚБЖ қате көрсетеді. Сондықтан МҚ компоненттері бірге сақталынады.

МҚ орналасуының тапсырмасы. Барлық жағдайда МҚ нақты компьютерде, директорияда сақталу қажет. Егер МҚ сол компьютердің дискіде сақталынса, онда МҚ локальді аталады. Бұны директорияны анықтау арқылы жүргізуге болады.

1.4 Деректер қорымен жұмыс

1.4.1 ДҚ іздеу, сорттау және индекстеу, формаларды және есептерді құру

Кестені қарапайым екі өлшемді кесте ретінде қарауға болады. Кестенің аты болады, ол- идентификатор, сол арқылы онымен байланыс орнатамыз. Бағандар әр-түрлі мінездемелерге сәйкес келеді және әрбір алаң сақталатын мәннің типімен және атымен сипатталады. Алаң аты- ол идентификатор және ол әр түрлі программаларда манипуляция жасау үшін қолданылады. Мұның аты латын әріптерімен жазылады да, алаң типі алаңда сақталатын мәннің типімен сәйкес келеді және ол әріптер, сандар, булөвті мәндер, үлкен тексттер, суреттер т.б болуы мүмкін. Кестенің әрбір жолы бір объектіге сәйкес келеді және оны біз жазба деп айтамыз. Кестені құрғанда ақпарат қарама-қайшылыққа ұшырамас үшін кілт алаңын кіргіземіз, ол алаңның уникальдылығын қамтамасыз етеді. Алаң мәндерін қолданушы енгізеді, ол мәліметтер қорында сақталады және мұндай алаңдардан басқа есептеуші алаңдарды да енгізуге болады. Мұндай алаңдарға қолданушы мән енгізе алмайды, ал оның мәні басқа алаңдардың көмегімен есептеледі. Қолданушы кестемен жұмыс істегенде көрсеткіш жазба бойымен сырғып отырған секілді әсер береді. Әрбір уақыт кезінде бір жазбамен жұмыс істеледі. Жазбалар белгілі бір ретпен орналасуы да мүмкін, реттеу үшін индекс түсінігі енеді. Индекс қолданушыға қандай ретпен орналасу керектігін көрсетеді. Ол қолданушы мен кесте арасындағы делдал ретінде болады. Көрсеткіш индекспен сырғып отырады, ал индекс белгілі бір жазбаны көрсетеді. Қолданушы үшін кесте реттелген болып көрінеді, сонымен қатар ол индексті өзгерту арқылы қарап отырған жазбалардың ретін өзгертуге болады, мұнан индекстер өзгереді, яғни жазбаларға сілтеме реті өзгереді.

Индекстер 1-ші ретті немесе 2-ші ретті болуы мүмкін. Мысалы, 1-ші ретті индекстерге мәліметтер қорының кілт ретінде құрылған алаң болуы мүмкін, ал 2-ші ретті кілттер басқа алаңдардан құруға болады, я болмаса онымен артынан жұмыс істеу барысында құрып алуға болады. Егер бірнеше алаң индекстелген болса, онда реттеу алғашқы құрылған индекс алаңынан басталады. Мәліметтер қоры көптеген кестелерден тұруы мүмкін. Қолданбалы программалар бөлек кестелермен жұмыс жасамайды. Олар кестелер жиынтығымен жұмыс жасайды. Көбінде бір кесте ең басты кесте болады да, ал қалғандары бағынышты болады. Басты және көмекші кестелер кілт арқылы байланысуы мүмкін, кілт ретінде екі кестеде де бар бірдей алаң болады.

1.4.2 ДҚ физикалық ұйымдастыру, хештау және индекстеу файлдары. Журнализация

Кез келген деректер қорының талаптарының біріне - сыртқы жадыға мәліметтерді сақтау сенімділігі болып табылады. Сақтау сенімділігі ол - кез келген ақаудан кейін деректер қорын қайтадан қалыпқа келтіру. Ақаулар аппараттық және программалық болуы мүмкін. Аппараттық ақаулар екі түрге бөлінеді: жұмсақ және қатты. Жұмсаққа компьютердің аяқ астынан өшуі. Ал қатты ақауға: сыртқы жады тасымалдаушысындағы ақпараттың жоғалуы. Осы ақаулардан кейін, деректер қорын қалыпқа келтіру үшін қосымша ақпарат керек. Сондықтан да деректер қорындағы мәліметтерді қосымша сақтау керек. Ол үшін журнализация әдісін қолданады. Журнал – ол мәліметтер қорының қолданушының қолы жетпейтін деректер қорының бір бөлігі. Кейде әртүрлі дискілерде журналдың көшірмелері сақталады. Онда негізгі деректер қорындағы өзгерістердің бәрі тіркеліп отырады. Әртүрлі МҚБЖ деректер қорындағы өзгерістер журналдарда әртүрлі деңгейде жазылады.

1.4.3 ДҚ тұтастығы және қорғау

Тұтастылық (сақталу, өзгертуден сақтау бір қалыпты ұстау) – бұл ұғым мәліметтерді әрқашанда өзгертпей ұстауды білдіртеді. Яғни кестедегі мәліметтерді қандай жағдай болса да, бастапқы қалыпта болу керек. Мысалы: кестедегі мән тоғыз болса, ал оның орнына басқа сан жазылатын болса, тексеру мүмкін емес. Бірақ, кейбір жағдайдарда, шектеулер қоюға болады. Аптаның күндері 1...7 мәліметтер қорының бүтінділігі: мәліметтерді өзгерістерден қорғаудан және өзгертуден қорғауға бағытталған. Тұтастылық ережелері үш топқа бөлінеді:

- 1) кестелер бойынша тұтастылық;
- 2) сілтемелер бойынша тұтастылық;
- 3) қолданаушымен анықталатын тұтастылық.

Барлық реляциялық ДҚ-ның жалпы ережелері:

- 1) бастапқы кілтке қатысты қандай да бір атрибуттың мәні белгісіз болмау керек;
 - 2) кілттер арасындағы мәндер бірдей болуы керек;
 - 3) атрибуттардың ерекше болуы керек мәндерінің, диапазоны болуы керек.
- Қандай да бір мәндер тобына жату-жатпайтынын анықтау керек т.с.с.

2. SQL құрылымдық сұранымдар тілі

2.1 SQL даму тарихы

Реляцияда МҚ басқару жүйелері 80 жылдардың аяғында кеңінен танымал болды. Мұндай мәліметтер қорында ақпарат қарапайым кестеде сақталатындықтан, мәліметтерді беретін басқа модельдерге қарағанда артықшылығы көп. SQL реляциялы МҚ тілі болғандықтан, оның тарихы да мәліметтерді осы тәсілмен берудің дамуымен тығыз байланысты.

Жоғарыда айтылғандай, реляциялы МҚ түсінігі 70 жылдардың басында IBM компаниясының ғылыми қызметкері Э.Ф.Кодпен енгізілді. Осыдан кейін реляциялық мәліметтер қоры одбысында IBM компаниясының System/R деп аталған жобасымен бірге зерттеулер жүргізіле бастады, мұнда реляциялық модельдің жұмыс істеу қабілеті тексеріліп және дәлелденді. Сондай-ақ МҚБЖ-ін өңдеуден басқа жоба көлемінде МҚ-на сұранымдар тілін құру жұмысы жүргізілді, олардың бірі SEQUEL (Structured English Query Language-ағылшын құрылымдық сұранымдар тілі)-деп аталды.

70 ж. соңында System/R жобасының екінші таралымы жүзеге асырылды, нәтижесінде жүйе IBM компаниясының бірнеше тапсырушыларының компьютерлерінде орнатылды. Ол System/R МҚБЖ-мен және оның МҚ тілімен жұмыста алғашқы нақты тәжірибеге әкелді, ал оның аты SQL деп ауыстырылды. Нәтижесінде IBM компаниясы реляциялық МҚ жұмыс істеуге қабілетті және программалық өнімдер құрудың негізі болып қызмет ете алады деген қорытындыға келді.

System/R жобасы және оның шеңберінде құрылған МҚ-мен жұмыс істейтін тіл, дүние жүзінің мамандарының назарын аударғты. Осылайша, 1977 ж SQL-ға негізделген, реляциялық МҚБЖ-ін құру үшін Relational Software Inc(қазір Oracle Corporation) компаниясы ұйымдастырылды. Oracle деп аталған бұл МҚБЖ, бірінші юлып 1979 ж нарыққа шықты.

Беркли қаласының университетінде (Калифорния штаты) IBM компаниясының зерттеушілер тобы реляциялық МҚБЖ-нің прототибін құрды, QUEL сұранымдар тілін қосқан өз жүйелерін Ingres деп атады. 1980 ж бірнеше мамандар Берклиді тастап, Ingres жүйесінің коммерциялық жүйесін құру үшін Relational Technology Inc компаниясының негізін құрды, оны нарыққа шығару 1981 ж басталды. 1989 ж аты Ingres Corporation болып ауысқан Relational Technology компаниясы жүйеге өзгерістер қатарын енгізген реляциялық МҚБЖ жетекші таратушысы болып қалды. Мысалы, 1986 ж алғашқы QUEL сұранымдар тілі табыспен SQL болып ауыстырылды.

1981 ж IBM фирмасы System/R негізінде, SQL/DataSystem(SQL/DS) деген атпен коммерциялық өнім шығарды, ал 1983 ж тағы бір МҚБЖ-Database(DB2) пайда болды. DB2 IBM компаниясының жетекші реляциялық МҚБЖ болды, және осы жүйенің SQL тілі МҚ үшін стандарты болды. DB2-де іске асырылған технология, кейін IBM компаниясының барлық бағыттарының программалық

өнімдерінде, дербес компьютерден бастап, желілік серверлерге және үлкен ЭЕМ пайдаланылды.

80 жылдардың бірінші жартысында реляциялық МҚ таратушылар өз өнімдерін мойындату үшін күрес жүргізді, өйткені дәстүрлі МҚ архитектурасымен салыстырғанда, реляциялық программалық өнімдердің бірқатар кемшіліктері болды. Бірақ реляциялық жүйелердің үлкенартықшылығы болды: олардың реляциялық сұранымлар тілі МҚ-на сұранымлы жазбай орындау және нәтижені бірден жазу мүмкіндігін берді.

80жылдардың екінші жартысында реляциялық МҚ болашақ технологиясы болып санала бастады, алдыңғы версияларына карағанда өнімділігі 2-3 есе артық болған МҚБЖ-нің жаңа версиялары Ingres және Oracle пайда болды. 1986 ж ANSI/ISO-мен қабылданған SQL стандартын жариялау, SQL-ға стандарт статусын ресми түрде жариялады. Сондай-ақ , бұл статус компьютерлік жүйелер үшін UNIX ОЖ-сі негізінде анықталды.

Компьютердің қуатының күшеюі мен олардың жергілікті желіге бірігуінен, дербес компьютерлер үшін күрделі МҚБЖ қажеттігі туды. Мұндай МҚБЖ-нің таратушылары жаңа жүйелерді құруды оларға SQL-ды негіз етіп қойды, ал мини-компьютерлердің МҚБЖ-сін таратушылар жергілікті есептеу желісінің нарығына шықты. 90ж басында SQL таратуын жаңашаландыру және қуатты процессорлардың пайда болуы бұл тілді транзакцияны өңдеу қосымшаларында қолдануға мүмкіндік берді. Енді SQL ақпаратты өңдеу салыстырмалы аз шығымен жүзеге асатын жүйеге желілік сервер ең дербес компьютерді байланыстыратын клиент/сервер архитектурасыныңкілттік бөлігі болды. Осы себепті барлық жаңа МҚБЖ-не SQL-қолдауын енгізе бастады, SQL-ді мойындаудың маңызды қадамы бұл тілге стандарттардың пайда болуы. SQL стандартын ескертуде дәстүр бойынша ANSI/ISO –мен тағайындалған ресми стандарт туралы сөз болады. Бірақ басқа да маңызды SQL стандарттары бар, соның ішінде IBM компаниясының DB2 жүйесі, және UNIX ортасындағы SQL үшін X/OPEN стандарты. 1982 ж реляциялық МҚ тілінің ресми стандартымен ANSI-дің жұмысы басталды, және комитет өз таңдауын SQL-ға тоқтатты. Стандарт негізі ретінде DB/2 жүйесінің SQL алынды 1989 ж стандарт SQL-89 немесе SQL1 деп аталды. Осыдан кейін ANSI жұмысын жалғастырып, жаңа жоба SQL2-ні құрды, ал одан кейінгі стандарт SQL3 үшін басқа өзгерістер ұсынылды. SQL2 стандарты ANSI-де бекітілді және толығымен 1992 ж қазанда қабылданды.

2.2 SQL мүмкіндіктері мен артықшылықтары

1987 жылы SQL маманданған реляциялық МҚБЖ үшін стандартты тіл болып барлық кең таралған жүйелерге ене бастады. Ол келесі момнеттерімен байланысты. Тез әрекеттіліктің өсуі, және энергияны тұтынудың, компьютер өлшемдері мен бағаларының төмендеуі, тұтыну нарығының, қолданушылар ортасының және түрлері мен бағаларының кеңеюіне әкеліп соқты. Сондай-ақ, әртүрлі программалық жабдықтауларға да сұраныс көбейді. Сатып алушы

фирмаларға күресте, программалық жабдықтаушылар нарыққа интеллектуалды, әрі көлемді программалар кешенін шығара бастады. Олар сатып ала отырып, көптеген ұжымдар мен жеке қолданушылар ол программаларды өз компьютерлеріне орната алмады. Ақпаратпен алмасу және оларды тарату үшін ЭЕМ желілері құрылды, онда программалар мен мәліметтерді арнайы файлдық серверлерге орналастырды.

Файлдық серверлермен жұмыс істейтін МҚБЖ, бір-бірінен алыс арақашықтықта орналасқан әртүрлі ЭЕМ қолданушыларына бір МҚ кіру мүмкіндігін берді. Мұндай жағдайда ұжымдарды, оқу кешендерін, ақпараттық және басқа жүйелерді әртүрлі автоматталған басқару жүйелерін өндіру оңайлатылады. Бірақ бұл кезде қарапайым сұраным құру үшін де файлдық серверден немесе серверге файлдарды жазу керек болады, ал ол конфликтілік ситуацияларды туғызуы мүмкін. Мұндай кемшіліктерді бодырмау үшін клиент/сервер технологиясы ұсынылған боатын, бірақ ол үшін сервермен хабарласу тілі қажет-мұндай тіл ретінде SQL таңдалды.

SQL концепцияларында операциялардың мәліметтердің кесте түрінде берілуіне бағытталуы, аз сөйлемді жинақы тілді құруға мүмкіндік берді. SQL сұранымдарды орындау үшін, және де қолданбалы программаларды құру үшін де пайдаланылуы мүмкін. Онда келесі сөйлемдер бар:

- мәліметтерді анықтау тілі-МҚ анықтау, және де кестелер мен индекстерді анықтау және жою;
- мәліметтерді таңдау сұранымы-SELECT сөйлемі;
- мәліметтерді түрлендіру сөйлемі-мәліметтер қосу, жою және өзгерту;
- мәліметтерді басқару сөйлемі-мәліметтерге кіру артықшылығын беру немесе қайтару, транзакцияларды басқару және т.б.

Сондай-ақ SQL бұл сөйлемдерде келесі есептерді орындау мүмкіндігін береді:

- арифметикалық есептеулер, функционалдық түрлендірулер, және арифметикалық өрнектер мен текстерді салыстыру операцияларын орындау;
- кесте мазмұнын экран дисплейіне немесе баспаға шығарар кезде жолдар мен бағаналарды реттеу;
- қолданушыларғы МҚ-да мәліметтердің көлемін өзгертпей интерпретациялау мүмкіндігін беретін ұсыныстар құру;
- сұраным бойынша шығарылатын бір немесе бірнеше кестелер және ұсыныстар мазмұнын басқа кестеде сақтау;
- мәліметтерді топтау, және осы топтарға арифметикалық орта, сумма, максимум, минимум операцияларын қолдану;

SQL тілі көптеген МҚБЖ-ң негізі болып табылад, өйткені ол мәліметтердің физикалық құрылымдануына және дискіге жазылуына сондай-ақ мәліметтерді дискіден физикалық оқуға жауап береді, және МҚБЖ-ң басқа компоненттерінен және қолданушы қосымшаларынан SQL-сұранымдарды қабылдау мүмкіндігін береді. Олай болса, SQL қолданушыларға, программаларға реляциялы МҚ ақпараттарына кіруді қамтамасыз ететін мықты құрал.

2.3 SQL көмегімен мәліметтер қорын басқару

2.3.1 Реляциялық мәліметтер қоры

Реляциялық МҚ – ол жолдар мен бағандардан тұратын екі өлшемді кестеде сақталатын байланысқан ақпарат. МҚБЖ терминінде жазу деп аталатын әрбір жол мәліметтердің қандай да бір ерекшелігіне сәйкес келеді. Өріс деп аталатын кестенің әрбір бағаны мәліметтердің әрбір типінің мәнінен тұрады. Олай болса, реляциялық МҚ негізі ақпараты бар екі өлшемді кесте болып табылады.

Бірақ МҚ бір кестеден тұратын реляциялық МҚ сирек кездеседі. Мәліметтермен күрделі және қуатты операцияларды өрнектеу үшін өзара байланысқан ақпаратты фрагменттері арасындағы байланысқа байланысты. Жүйенің максималды иілгіштігін қолдау үшін кестенің жолдары арнайы ретпен орналаспауы керек. Реляциялық МҚ жүйелерінде ақпаратты реттейтін қуатты функция бар.

Жолдардан айырмашылығы, кестенің өрістері реттеледі және оларға ат беріледі. Яғни, кестенің әрбір бағанының ерекшеленген аты болуы керек.

Әдетте SQL бірден көп қолданушылары бар компьютерлік жүйелерде жұмыс істейді, сәйкесінше оларды өзара ажырату қажеттілігі туады. Әдетте мұнда жүйеде әрбір қолданушының хұқын тексеру коды болады. Компьютермен сеанстың басында қолданушы жүйеге өзінің арнайы ID(идентификатор) айтып кіреді. Кірудің ID-ін пайдаланатын кез-келген адамдар саны, жеке қолданушы болып табылады.

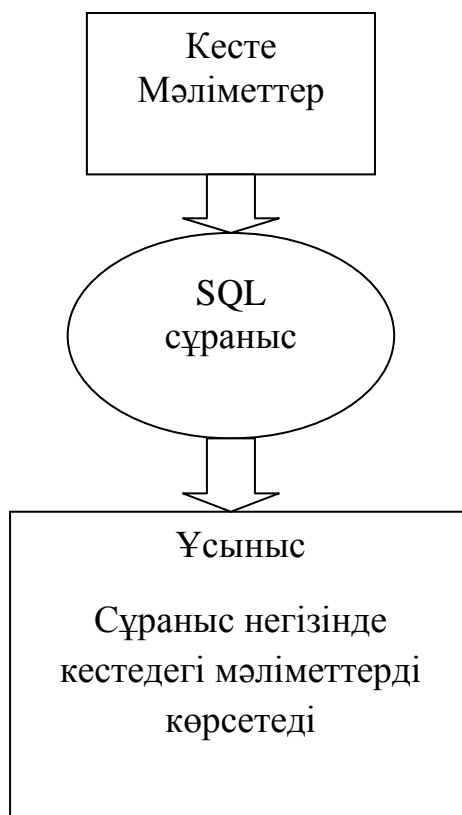
Көптеген SQL орталарындағы әрекеттер арнайы идентификаторлармен сәйкес рұқсат етілген. Кесте немесе басқа объект толық билігі бар қолданушыға тәуелді. Және де қолданушыда объектілермен әрекеттер жүргізу артықшылығы бар. Төменде келтірілетін мысалдарда кез-келген қолданушының әрекеттер жасауға артықшылығы бар деп санаймыз.

2.3.2 Ұсыныстармен жұмыс

Ұсыныс (VIEW) мәліметтер қорының объектісі. Онымен жұмыс қарапайым кестемен жұмыстан өзгешеленбейді. Ұсыныстардың кестелерден айырмашылығы келесіде. Мәліметтер қорының қарапайым кестелері мәліметтерден тұрады. Ұсыныстар мәліметтерден тұрмайды, ал олардың мазмұны басқа кестелерден таңдалынады (немесе басқа ұсыныстардан). Ұсыныстар тұжырымдалатын кестелер (немесе ұсыныстар) базалық кестелер (немесе базалық ұсыныстар) деп атау қабылданған.

Шындығында ұсыныстар әрбір хабарласқан сайын орындалатын сұраныс болып табылады. Әрбір уақыт мезетінде осы сұраныстың орындалу нәтижесі ұсыныс мазмұны болып табылады. Ұсыныстың базалық кестесіндегі мәліметтерді өзгертсе ұсыныс мазмұны да өзгереді.

Ұсыныс мазмұнын өзгертсе, осы ұсынысты құруда негізге алынған кесте мазмұны да өзгереді. Төменде ұсыныстың тұжырымдалу процесінің схемасы 2.1-суретте көрсетілген:



2.1-сурет – Ұсыныстың тұжырымдалу схемасы

Ұсынысты пайдалану кесетені пайдаланудан өзгешеленеді. Ұсыныстардан мәліметтерді таңдау қарапайым кестедегі сияқты орындалады. Сондай-ақ ұсыныстың мәліметтермен басқару операцияларына рұқсат етіледі, бірақ ол мұнда кейбір шектеулер болады.

Ұсыныстар кестелерден айырмашылығы дискілік кеңістіктен орын алмайды (немесе дәлірек айтсақ, ұсыныс алатын дискілік кеңістік өте кішкентай-тек сұранысты сақтауға қажеттілері).

Ұсыныстар көбінде екі жағдайда қолданылады:

- мәліметтерді қорғау мақсатында,
- қорытынды мәліметтерді тұжырымдау үшін.

Бірінші жағдайда ұсыныстар бүкіл кестеден емес, тек кейбір өрістеріндегі ақпараттарды беру үшін қолданылады.

Ұсыныстарды құру.

Ұсыныстарды құру үшін CREATE VIEW операторы пайдаланылады. Ұсыныс бір немесе бірнеше ұсыныстар негізінде құрылуы мүмкін. Ұсынысты құрудың операторының типтік синтаксисінің түрі келесідей:

```
CREATE VIEW ұсыныс_аты AS
```

{таңдау операторы}

Ұсыныс құрылған соң онымен ұсыныс аты ретінде берілген аты бар қарапайым кесте сияқты жұмыс істеуге болады. Қандай да бір өзгешелігі топтау сөйлемі бар ұсыныстар. Мұндай ұсыныстар үшін мәліметтер таңдауда ешқандай шектеу жоқ, бірақ оларға мәліметтерді басқару операторын қолданбайды.

Ұсыныстарды жою.

Ұсыныстарды жою DROP VIEW операторының көмегімен орындалады, оны шақырғанда RESTRICT және CASCADE параметрлері көрсетілуі мүмкін. Бұл параметрлер басқа ұсыныстар және/немесе шектеулер сілтелетін ұсыныстарды жою кезіндегі әрекеттерді анықтайды. RESTRICT вариантын падаланғанда, бұл жағдайда қате туралы хабарлама шығарылады, және жою орындалмайды. Егер CASCADE режимі пайдаланылса, онда DROP VIEW операторының орындалуы, базалық ұсыныстардың және шектеулердің жойылуына әкеледі.

DROP VIEW операторының типтік синтаксисі келесі түрде болады:

DROP VIEW ұсыныс_аты [RESTRICT | CASCADE]

2.3.3 Сақталатын процедуралар, оларды құру, орындау және жою

Сақталатын процедуралар (Stored Procedure) - олар байланысқан SQL-операторларының топтары. Сақталатын процедураларды пайдалану мәліметтер қорымен жұмыстың қосымша икемділігін қамтамасыз етеді, өйткені сақталатын процедураларды орындау SQL жеке операторлар тізбегінен әлдеқайда қарапайымдау.

Сақталатын процедуралар орындалудың жоғары жылдамдығын қамтамасыз ететін мәліметтер қорында компиляцияланған түрде сақталады.

Сақталатын процедураларға кіретін параметрлер ала-алады, қосымшаға мәндерді қайтарады және қосымшадан шақырылады.

Сақталатын процедураларды пайдаланудың негізгі артықшылықтары келесіде қорытындыланады:

- сақталатын процедуралар логика бөлігін мәліметтер қорының серверіне шығаруға мүмкіндік береді. Ол ақпараттық жүйенің мәліметтер қорының клиенттік бөлігінен тәуелділігін әлсіретеді;

- сақталатын процедуралар жобаның модульдігін қамтамасыз етеді. Олар бір мәліметтер қорына хабарланатын клиенттік қосымшалар үшін жалпы болады, ол кодтың қайталануын болдырмайды және қосымша өлшемін кішірейтеді;

- сақталатын процедуралар қосымшалардың бәрге жүруін оңайлатады: процедуралар жаңартқан кезде өзгерулер автоматты түрде барлық қосымшаларда бейнеленеді;

- сақталатын процедуралар ақпараттық жүйенің жұмысының тиімділігін арттырады: олар клиентпен емес, желілік трафикті төмендететін сервермен орындалады;

– сақталатын процедуралардың орындалу жылдамдығы жеке SQL операторлар тізбегінен жоғары. Ол сақталатын процедуралардың серверде компиляцияланған түрде сақталуына байланысты.

Сақталатын процедуралардың екі түрін бөледі:

– таңдау процедуралары қосымшалар оларды мәліметтерді таңдау операторында кесте немесе ұсыныстар орнына пайдалануы мүмкін;

– орындалатын процедуралар олар арнайы операторлар пайдаланылып шақырылады. Орындалатын процедура шақырылатын программаның нәтижелерін қайтармауы мүмкін.

Сақталатын процедураларды құру

Сақталатын процедураларды құру үшін CREATE PROCEDURE операторы пайдаланылады.

CREATE PROCEDURE операторы мәліметтер қорынан жаңа сақталатын процедураны анықтайды. Процедуралар тілі SQL таратуларынан тәуелді, бірақ ереже бойынша SQL барлық инструкцияларын және мыналарды қосады:

– шартты операторлар;

– цикл операторларының әртүрлі түрлерін;

– ерекше жағдайларды өңдеу мүмкіндіктерін.

– сақталатын процедуралар тақырыптан және денесінен тұрады.

Процедураның тақырыбы:

– мәліметтер қорындағы кесте мен процедура аттарының ішінде ерекше болуы тиіс процедура атынан;

– процедура шақырылатын программадан қабылдайтын кіретін параметрлер және олардың мәліметтері типтерінің тізімінен;

– шығатын параметрлер және олардың мәліметтері типтерінің тізімінен тұрады, егер процедура мәндерді шақырылатын программаға қайтарса.

– процедура денесі мыналардан тұрады:

– жергілікті айнымалылар және олардың мәліметтері типтерінің тізімінен (егер олар процедура кодында пайдаланылса);

– BEGIN және END кілттік сөздерінің арасында қорытылған процедуралар және триггерлер тіліндегі инструкция блогынан.

Сақталатын процедураларды орындау

Сақталатын процедураны орындауға жіберетін оператор процедураға тәуелді болады. Таңдау процедурасы оларға SELECT -мәліметтерді таңдау операторының көмегімен хабарласқанда орындалады.

Орындалатын процедураны шақыру үшін арнайы EXECUTE операторын пайдаланған жөн.

Сақталатын процедураларды жою

Сақталатын процедураларды жою үшін DROP PROCEDURE операторы пайдаланылады. Бұл оператор синтаксисі келесі түрде болады:

DROP PROCEDURE имя_хранимой_процедуры

2.3.4 Мәліметтерді басқару, кестедегі берілгендерді өзгерту

Мәліметтер қорында сақталатын мәліметтерді басқару үшін, командалардың жеке типтері ретінде бөлінетін және мәліметтерді басқару тілі (DML-Data Manipulation Language) деп аталатын SQL операторларының типтері пайдаланылады. DML операторларының көмегімен қолданушы кестеге жаңа мәліметтерді жүктеп, бар мәліметтерді түрлендіруге және жоюға болады:

SQL тілінде тек қана 3 негізгі DML операторы анықталған:

- INSERT;
- UPDATE;
- DELETE.

Кестеге жаңа ақпаратты қосу.

Мәліметтер қорының кестесіне жаңа ақпаратты қосу процесі әдетте мәліметтерді жүктеу деп аталады. Мәліметтерді жүктеу үшін INSERT операторы пайдаланылады.

Кестеге жаңа жазу қосу.

Кестеге жаңа жазу қосу үшін INSERT операторының келесі синтаксистік формасы пайдаланылады:

```
INSERT INTO имя_таблицы
```

```
VALUES (значение_1.значение_2...значение_N)
```

INSERT операторының бұл формасын пайдаланғанда VALUES тізімінде мәндер саны кесте өрістерінің санына тең болуы керек. Және де VALUES тізімінде көрсетілген әдбір мәннің типі, өріс мәліметтерінің типімен сәйкес болуы керек.

Символдық типтер мен даталарға қатысты мәндер апостроф ішінде тұруы тиіс. Мәндер тізімінде NULL мәні де пайдаланылуы мүмкін.

Кестенің жеке өрістеріне мәліметтер енгізу.

Кестеге мәліметтер қосқанда барлық өрістерін емес, кейбіреулерін ғана толтыруға болады. Бұл жағдайда INSERT операторының келесі синтаксистік формасы пайдаланылады:

```
INSERT INTO имя_таблицы (имя_поля_1.имя_поля_2...имя_поля_N)
```

```
VALUES (значение_1.значение_2...значение_N)
```

Басқа кестедегі мәліметтерді кестеге енгізу.

Кейде бір кестедегі ақпаратты екінші кестеге тасымалдау қажет болады. Бұл типтегі операцияларды INSERT операторымен SELECT мәліметтерді таңдау операторының комбинациясының көмегімен орындауға болады.

INSERT және SELECT операторын біріктіріп, басқа кестеге сұраныс орындалуы нәтижесінде алынатын мәліметтерді кестеге қосуға болады. Бұл жағдайда INSERT операторының синтаксисі келесі түрде болады:

```
INSERT INTO имя_таблицы(имя_поля_1.имя_поля_2...имя_поля_N)
```

```
FROM имя_таблицы
```

```
WHERE условие.
```

Бұл операторда VALUES сөйлемінің орнына SELECT операторы пайдаланылады. Бұл оператордың синтаксисінің қысқаша түсініктемесі.

SELECT сөзінен кейін мәндері таңдамаға қосылатын өрістер тізімі көрсетіледі (егер SELECT кейін * символын көрсетсе, таңдамаға барлық өрістер қосылады).FROM сөйлемі мәліметтерді таңдау орындалатын кестенің атын көрсету үшін пайдаланылады. WHERE сөйлемі міндетті емес және таңдамаға қосылатын мәліметтерге шектеулерді қосу үшін пайдаланылады.

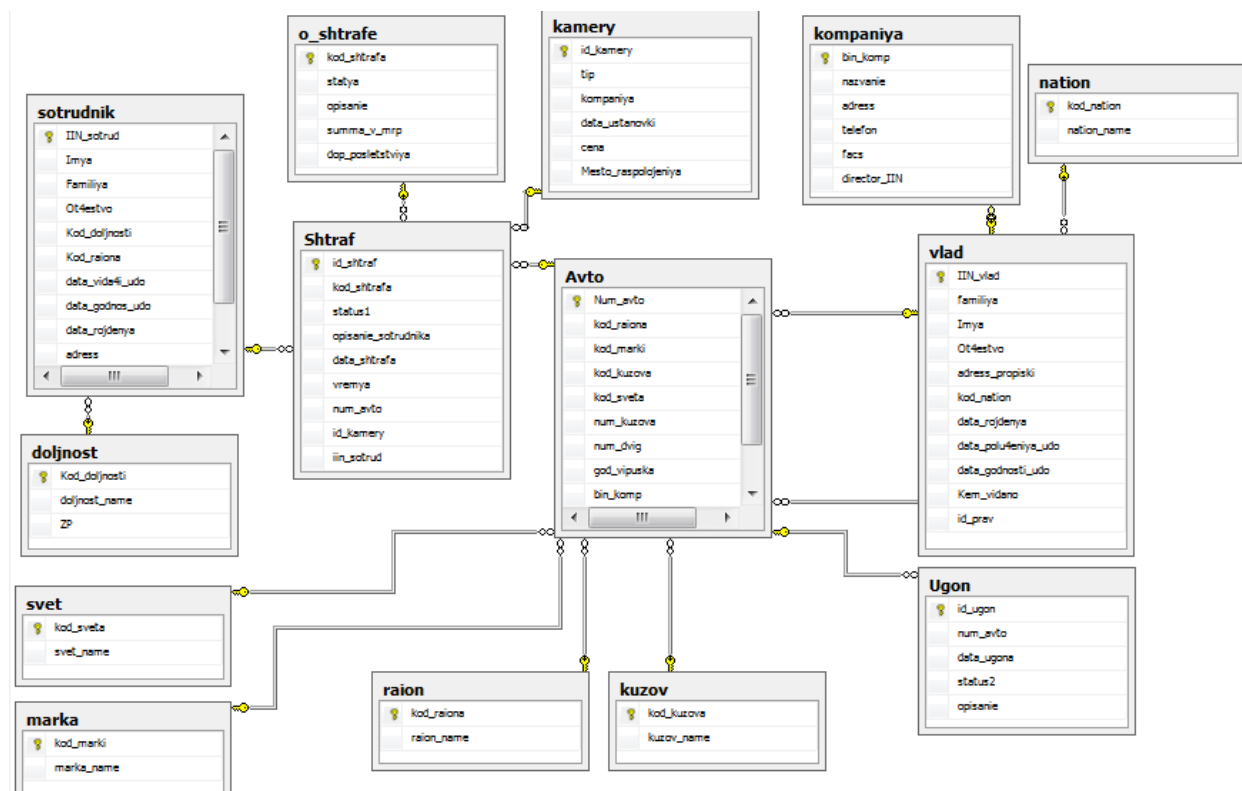
Кесте атынан кейін INSERT операторында жақша ішінде көрсетілетін өрістердің саны таңдамаға қосылатын өріс санына тең болуы керек. Өрістердің сәйкестігі, олардың тізбектелу ретімен анықталады. SELECT операторының тізіміндегі 1-ші өрісі сәйкес келеді және т.с.с.

Кестеде сақталатын мәліметтерді өзгерту.

Кестеге енгізіліп қойылған мәліметтерді өзгерту үшін UPDATE операторы пайдаланылады. Бұл оператор кестеге жаңа жазу қоспайды, бар мәліметтерді жаңаға ауыстырады. UPDATE операторы бір өріске де, бірнеше өрістерге де қолданылуы мүмкін. Өзгертілетін жазулардың саны қолданушы қажеттілігіне тәуелді - UPDATE көмегімен бір немесе бірнеше жазуларды өзгертуге болады.

2.4 ДҚ диаграммасы

Концептуалды моделді дерекқордың сәйкес құрылымына енгізілу тәсілі қандай болса да, ДҚ реляционды сұлбасы жобаға сәйкес болу керек. ДҚ диаграммасы 2.2-суретте көрсетілген.



2.2-сурет – ДҚ диаграммасы

3 VISUAL BASIC программалау тілі

3.1 VISUAL BASIC туралы жалпы түсінік

VISUAL BASIC – тің бүкіл программалауы оқиғалардан құралған. Өңдеу дегеніміз бірінші жолда оқиғаны өңдеушінің атын көрсететін процедура.

Бейсик (Basic, Beginner, All purpose Symbolic Instruction Code жаңадан үйреніп жүргендерге арналған көп мақсаттық символикалық код) оқу мақсатында 60 жылдар барысында Дармут колледжінде дайындалған. өзінің қарапайымдылығының арқасында ол тез халық арасында жайылды және үй компьютерлерінде қолданылды. 70 жылдары Бейсик сол заманның ең қарапайым программалау тілі ретінде Altair дербес компьютерлерінде жүзеге асырылды. Бұл жұмысты Билл Гейтс, Microsoft-тың қазіргі президенті, өзінің серіктесі Пол Алленмен бірге орындады. Гейтстің арқасында Бейсик IBM PC платформасына көшірілді, ал оның ең жетілдірілген QBASIC версиясы (QBASIC, жылдам Бейсик) MS DOS операциялық жүйесінің базалық жабдықтауына енді.

Қазіргі күнге дейін Гейтстің күш салуы және қаржылай қолдауы арқасында Бейсикті дамыту белсенді жалғасуда.

QBASIC тілінің Бейсиктің бұрынғы версияларынан негізгі айырмашылығы құрылымдық программалау тілі Паскаль және Си тіліндеріне ұқсас. Солай дегенмен, жаңадан үйренушілер үшін QBASIC тілі, барлық айнымалыларды жөндеу құралдарының икемділігіне орай Паскаль тілінен әлдеқайда жеңіл.

QBASIC тілінің құндылығы ол бойынан программалаудың диалогтық құралдары ретінде Бейсик тілдерінің ең жақсы қасиеттерін, Паскаль және Си сияқты тілдердің құрылымдық және модульдік программалауының ең жақсы құралдарын жинаған. Бұл тіл көптеген операциялық жүйелер үшін жүзеге асырылған. Жекелеп алғанда, MSDOS операциялық жүйе үшін QBASIC және VISUAL Basic for MSDOS нұсқасы бар. WINDOWS үшін жыл сайын дерлік VISUAL Basic-тің жаңа жаңа нұсқасы шығарылып тұрды. Барлық офистік қолданбаларға Microsoft компаниясы өзгеше бір макротіл ретінде қолданылады.

3.2 VB-те программалау

Программалау – адамның ЭЕМ-мен қарым – қатынас жасау құралдарының бірі. Ол дайын программаларды қолданудан ерекше. Біріншіден ондай программалар қолданушы сұранысын әр уақытта қанағаттандыра алмайды. Сондықтан қандай да бір нақты есепті орындауға арналған программа құру қажеттілігі туындайды.

Осындай мағынада Visual Basic программалау ортасы Windows операциялық жүйесі үшін программалық қамтамасыз етуді құруға арналған

универсал құралдарының бірі болып табылады. Және де, олардың ауырлық деңгейі қарапайым қосымшалардан мәліметтер қорын өңдеуге арналған күрделі программаларға дейін өзгеріп отыруы мүмкін.

Әрі қарай визуальді программалау тілінің көмегімен қалай программа құруға болатындығы туралы сөз болады.

бөлігін қолданушының графикалық интерфейсі деп атайды.

Динамикалық құрылымды мәндер (ДҚМ)

VBA тілінің негізгі бөлігі – Collection класы динамикалық құрылымды мәндермен жұмысты жеңілдетеді, яғни әр түрлі типті мәндерді және тиімді құрылған қасиеттер мен әдістерді жинақтау арқылы жүзеге асырылады.

VBA-дағы коллекция – әр түрлі типті реттелген элементтер жиынтығы. Коллекцияның массивтен айырмашылығы осында. Барлық элементтері индикстелген. Коллекцияның өлшемі алдын-ала анықталмайды және динамикалық түрде өзгеруі мүмкін. Коллекциядағы кез-келген элемент жойылуы мүмкін және элементтер қайта нөмірленеді де индекстер тізбегі үзілмейді. Жаңа элемент коллекцияның кез-келген жеріне қойылуы мүмкін.

Collection класының Count қасиеті элементтерінің санын, ал әдістері:

- add әдісі – коллекцияға элементті қосады;
- remove әдісі – коллекциядан элементті жою;
- item әдісі – берілген мән бойынша элементтің мәнін береді.

Берілген мән ретінде элементтің индексі аталады.

3.3 Қолданушының графикалық интерфейсін құру

3.3.1 Графикалық интерфейсін формаларымен жұмыс

Графикалық интерфейсін құруды бір қосымша негізінде түсіндірейік. Ол саймандар тақтасында орналасқан екі басқару элементтерінің көмегімен құрылады: «батырма» және «тексттік өріс».

Графикалық интерфейсін құру үшін қажетті басқару элементтерін формаға орналастыру қажет. Ол үшін :

– Тышқанның курсор көрсеткішін саймандар тақтасында таңдалатын элемент пиктограммасына орналастыру керек (мысалы «батырма» - оң жақ бағандағы жоғарыдан үшінші компонент);

– Таңдалған пиктограмманы екі рет шерту нәтижесінде форма терезесінде "батырма" элементі пайда болады;

– Тышқан көрсеткішін форма терезесіне апару. Көрсеткіш форма терезесіндегі батырма үстінде болғанда, тышқанның сол жақ батырмасын басып отырып форманың қажетті жеріне жылжытуға болады;

– Формаға тағы бір "батырма" қосу үшін, саймандар тақтасына барып, CommandButton (Командалық батырма) пиктограммасын екі рет шерту керек. Формада тағы бір «батырма» пайда болады. Келесі батырманы орналастырмас бұрын, бірінші батырманы орма центрінен шетіне қарай жылжыту керек. Әйтпесе, екінші батырма мүлдем көрінбей қалады;

– Осылайша формаға "текстік өріс" (TextBox) элементін орналастыруға болады.

Форманың өзі сияқты, онда орналасқан барлық компоненттер бірқатар қасиеттерге ие. Конструкциялау барысында олардың кез келгенін қолдануға болады. Бұл үшін Visual Basic ортасында Properties (қасиеттер) терезесі қарастырылған. Ол экранның оң жақ төменгі бөлігінде орналасады. Оны ашу арқылы форма компоненттерінің аттарын (идентификаторын) көруге болады. Программист олардың кез келгенін таңдау арқылы компоненттің барлық қасиеттерін пайдалана алады. Ортақ қасиеттердің жиі кездесетіндері кесте 3.1-де берілген:

К е с т е 3.1 – Жиі кездесетін компоненттер кестесі

Қасиеті	Тип мәні	Сипаттамасы
Left	Integer(бүтін)	Форманың сол жақ шетіне қатысты басқару элементінің позициясы
Top	Integer(бүтін)	Форманың жоғарғы шетіне қатысты басқару элементінің позициясы
Height	Integer(бүтін)	Басқару элементінің биіктігі
Width	Integer(бүтін)	Басқару элементінің ені
Name	String (жолдық)	Программада басқару элементіне сілтеме жасау үшін қолданылатын атау
Enabled	Boolean (логикалық)	Басқару элементінің алынуы
Visible	Boolean (логикалық)	Программаның орындалуы барысында басқару элементінің экранда көріну күйін анықтайды.

Бұл қасиеттерден басқа әрбір элемент өзіне ғана тән қасиеттерге ие. Конструкциялау барысында оларды өзгерте отырып, программист элементтің жалпы түрін бақылай алады.

Қасиеттерді қолдану Visual Basic-тегі визуальді программалаудың маңызды сатысы болып табылады. Программаның орындалуы барысында басқару элементтерінің қасиеттерін қолдану тәсілдері бесінші пунктте қарастырылады.

Сонымен қолданушының графикалық интерфейсі құрылды. Осыдан кейін программаны орындауға жіберуге болады.

3.3.2 Программаны жүктеу

Программаны жүктеу үшін Run менюінен Start командасын таңдауға немесе <F5> пернесін басуға болады.

Программаның орындалуы кезінде программалау ортасының кейбір элементтері экранда көрінбей қалады. Сонымен қатар форма түрі де өзгереді. Конструкциялау (проектілеу) барысында ол нүктелік тормен қапталған. Ал, орындалу кезінде форма кәдімгі Windows – терезесіне айналады. Онда сондай

типті терезелерге тән атрибуттар орналасады. Бұған формадағы батырмаларды шертіп немесе тексттік курсорды TextBox өрісіне орналастыру арқылы көз жеткізуге болады.

Құрылған программа Windows қосымшасы болғанымен, ол ешқандай қызмет атқармайды. Яғни графикалық интерфейс программаның функционалдылығын қамтамасыз етпейді. Ол үшін оған программа коды қосылуы керек.

3.3.3 Программаға код қосу

Программалық код тек проектилеу режимінде ғана қосылуы мүмкін. Егер программа орындауға жіберіліп қойса, кодты қосу алдында оны тоқтату керек. Бұны Run менюінен End командасын таңдау арқылы немесе орындалудағы программа терезесін басқа тәсілмен(мысалы, оның жоғарғы оң жақ бұрышындағы «х» белгісін шерту) жабу арқылы жүзеге асыруға болады.

Код – бұл Visual Basic программалау тілінде жазылған инструкциялар тізімі. Олар программаның қолданушының қандай да бір әрекетіне жауап беру сипатын анықтайды. Яғни код бұл программаның қандай да бір визуальді компонентінің жағдайына реакциясы болып табылады. Әрбір осындай жағдай белгілі бір атауға ие. Және де жағдайлар тізімі формада орналасқан элементтерге немесе фйорманың өзіне тәуелді. Мысалы, қолданушы тышқанмен Command1 батырмасын шерткенде Command1_Click жағдайы, ал форманың жүктелуі кезінде Form_load жағдайы пайда болады.

Visual Basic – те программалау басқа программалау тілдеріне, мысалы C және C++ қарағанда күрделі емес. Сонымен, программаға кодты кодты қосу үшін форма терезесінен код терезесіне ауысу қажет. Ол үшін проект (Project) терезесіндегі сол жақтан бірінші View Code пиктограммасын шерту жеткілікті. Одан кейінгі (View Object) пиктограмманы шерту керісінше код терезесінен конструкциялау терезесіне көшуге мүмкіндік береді.

Программаның код терезесінің негізгі бөліктерін қарастырайық. Тақырып жолағынан кейін тізімі бар екі өріс орналасқан. Біріншісінде формада орналасқан барлық компоненттердің аттары және форма аты болады. Кез келгенін таңдағаннан кейін оң жақ тізімде мүмкін болатын жағдайлардың аттары бейнеленеді. Қандай да бір жағдайды сипаттау алдында тізімнен оның атын таңдау керек. Осыдан кейін Visual Basic жағдайды өңдеу процедурасын құрады. Мысалы, егер қарастырылып отырған мысалда Command1 батырмасы үшін Click жағдайын таңдасақ, онда программа коды терезесінің жұмысшы бөлігінде төмендегідей дайындық пайда болады:

```
Private Sub Commsnd1_Click()  
End Sub
```

Бірінші жол процедура аты. Ал, соңғы жол End Sub оның соңын білдіреді. Бұл сатыда программистің жұмысы осы екі жолдың арасын.а программа кодын қосу болып табылады. Код Visual Basic-тің бір немесе бірнеше операторынан тұруы мүмкін. Бұл операторлар программаның орындалуы барысында

Command1 атты батырмасының басылу жағдайы жүзеге асырылғанда орындалады. Мысалы, батырманы басқанда Text1 тексттік өрісінде «Бәріне сәлем!» жазуы пайда болуы үшін, қарастырылып отырған жағдайдың өңдеу кодына келесі түрдегі операторды жазу керек:

```
Text1.Text=«Бәріне сәлем!»
```

Осылайша формадағы Command2 батырмасының басылу жағдайының өңделу кодын құруға болады.

```
Private Sub Command1_Click ( )
```

```
Text1.Text= «Сау болыңыз!»
```

```
End Sub
```

Бұл жағдайда программа жүктелгеннен кейін Command2 батырмасын басқанда тексттік өрісте «Сау болыңыз!» жазуы пайда болады.

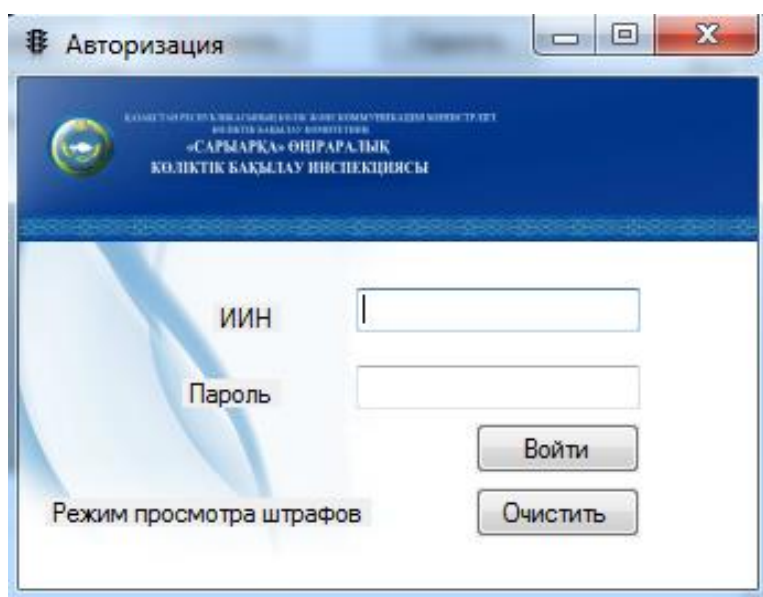
3.3.4 Программаны дискіде сақтау

Кез-келген ақпарат сияқты, Visual Basic тіліндегі қосымша дискіде сақталуын талап етеді. Әрбір проект үшін дербес папка құрған дұрыс. Бұл жағдайда проектке кіретін файлдарға ерекше ат беру қажеттілігі жоқ. Сақтау барысында проектке кіретін форма аттары мен проекттің файл аты ғана сұралады. Оларды Visual Basic ортасы ұсынатын аттармен сақтауға болады: Form1, Form2, ... және Project1.

4 Қосымша интерфейсін сипаттау. Қосымшамен жұмыс істеу

Жобаның интерфейсі Visual Studio 2010 объектілі-нысандалған бағдарламалау ортасында орындалды. Қосымшаның бағдарламалық коды Ә қосымшасында келтірілген.

Қолданушы авторизация формасында өзінің ИИН мен кілттік сөзін енгізеді. Аутентификация арқылы деректер қорында енгізілетін ИИН бар немесе жоқ екендігі тексеріледі. Осындай ИИН бар болған жағдайда, әрі қарай процестің орындалуына қолжетімділік рұқсат етіледі. Терезеде ақпараттық қауіпсіздік негіздері ескерілген. Кілттік сөз дұрыс енгізілбеген жағдайда ескерту терезесі шығады. Аутентификация терезесі 4.1-суретте көрсетілген

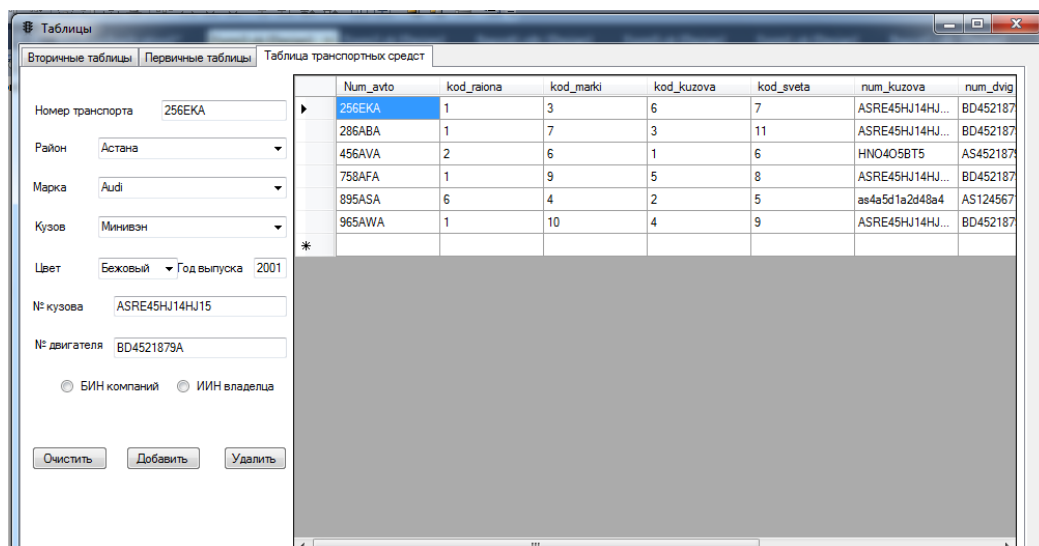


4.1-сурет – Авторизация терезесі

Аутентификация терезесі арқылы өзге үш терезеге өту мүмкіндігі бар:

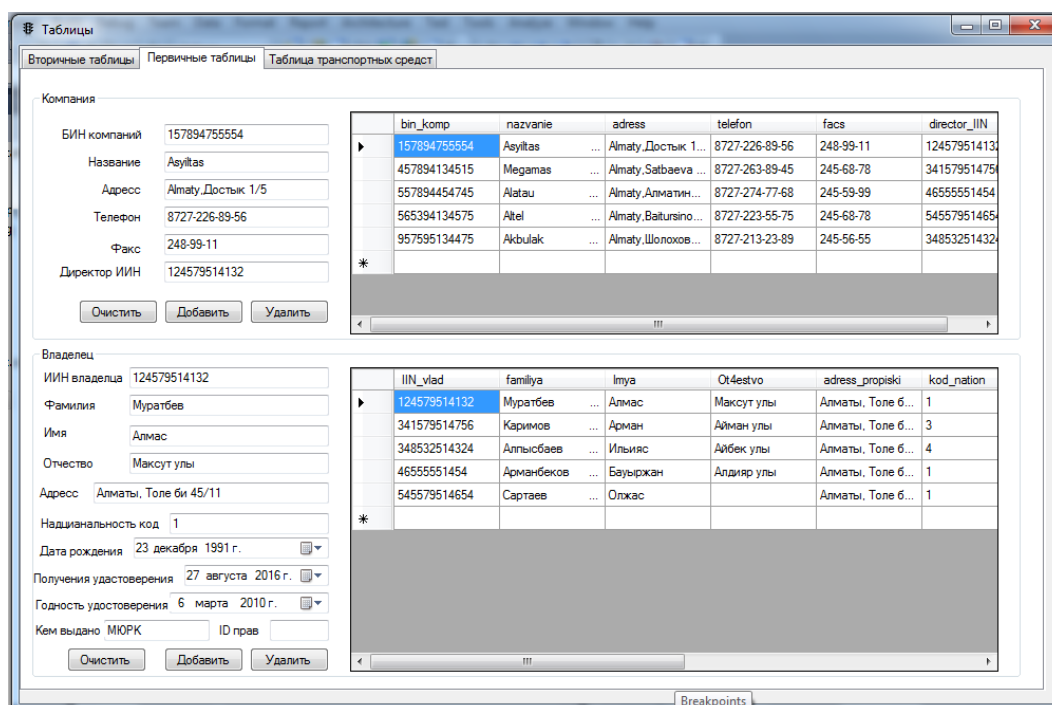
- администратор терезесі;
- қызметкер терезесі;
- жүргізуші терезесі .

Администратор мүмкіншіліктері қалған пайдаланушыларға қарағанда шектелмеген. Администратор деректер қорындағы барлық ақпаратқа толық рұқсаты бар. Ол айыппұлдар кестесін реттей алады. Сурет 4.2-де көрсетілгендей кестелер үш топқа бөлінген. Ең басты кесте жүргізушілер кестесі болып табылады. Ол кестеде жүргізушілер туралы толық мәлімет камтылған.



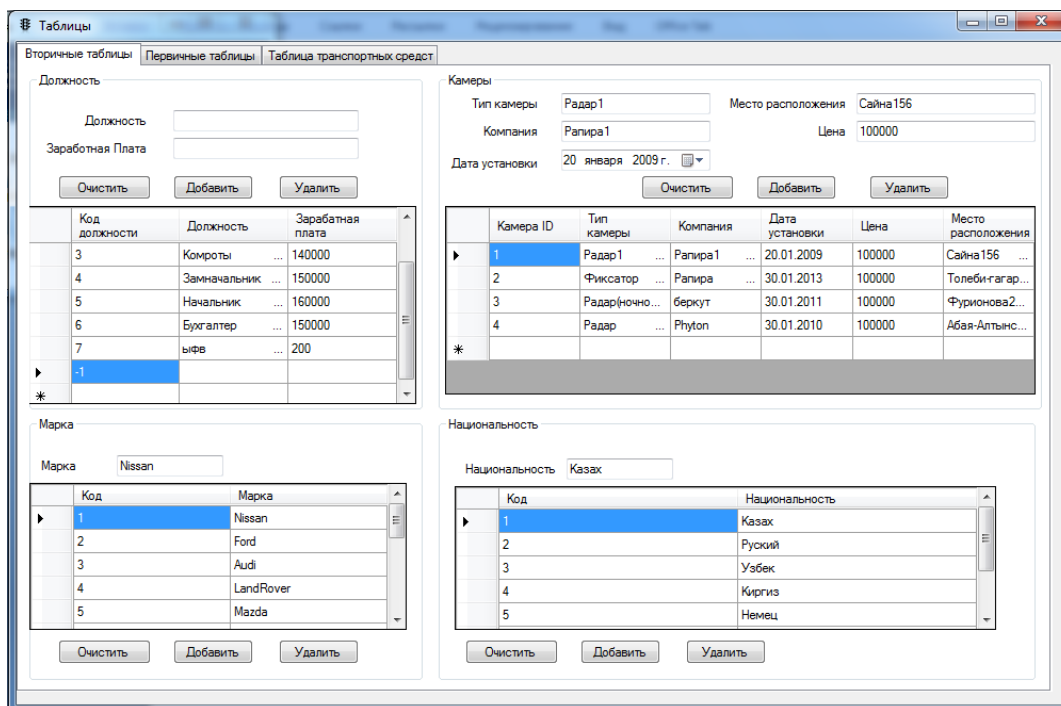
4.2-сурет – Айышпұл кестесін реттеу терезесі

Авто көлік жеке тұлғаға немесе заңды тұлғаға тіркелуі мүмкін. Екеуінің мәліметтері әр түрлі болғандықтан, екі түрлі кесте жобаланды. Осы кестелер арқылы администратор жаңа жеке тұлға мен заңды тұлғаны деректер қорына қосып мәліметтерің еңгізеді. Бұл кестелер толығымен толтырылуы керек, себебі заң бұзушылық кай жерде кай уақытта болғанын біліп белгілі бір адреске жіберілу керек. Жеке тұлға және заңды тұлға ақпараттары бірінші ретті кестеге енгізіледі.



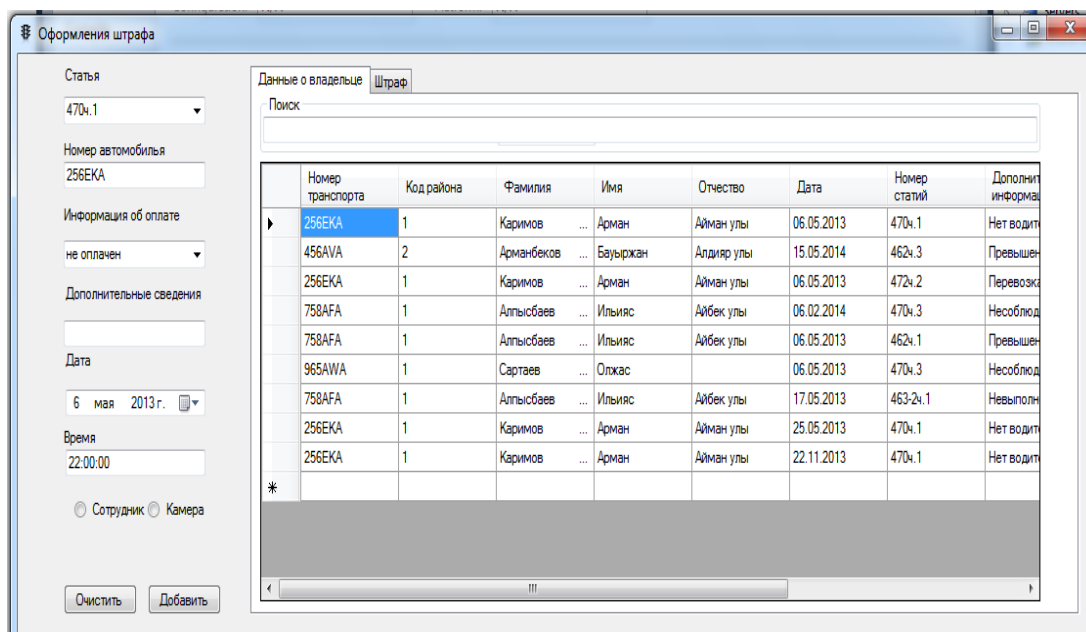
4.3-сурет – Бірінші ретті кестелерді реттеу терезесі

Бірінші ретті кестеден басқа екінші ретті кесте бар. Онда көлік маркасы, түсі, заңды бұзушылық түсірілген бейне бақылау сияқты екінші сортты ақпарат жазылады.

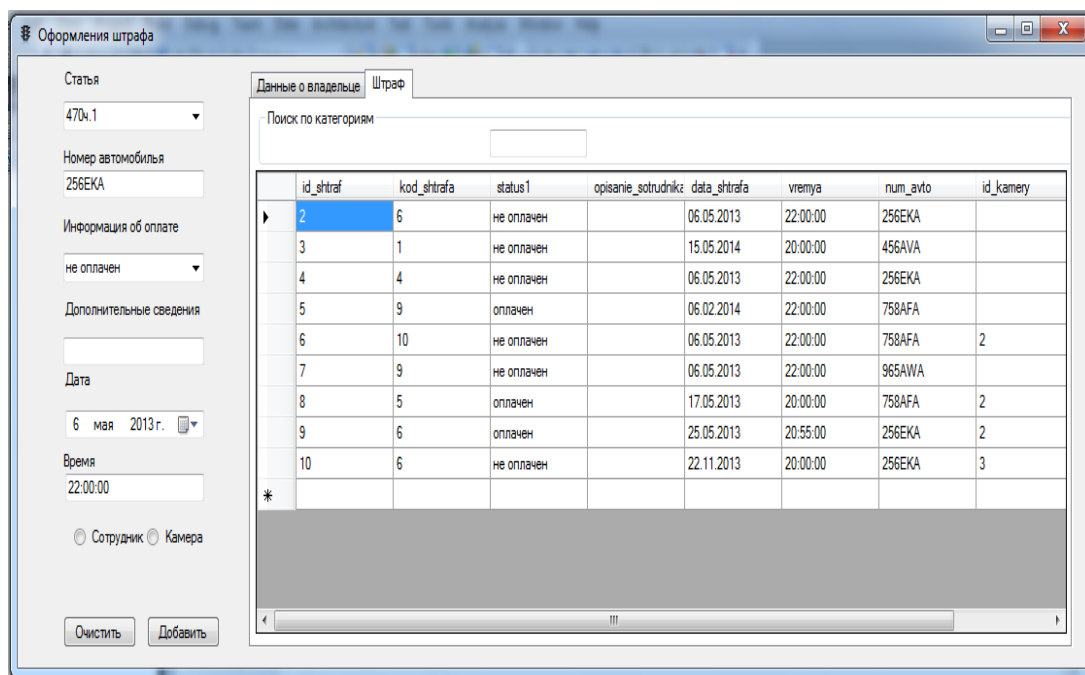


4.4-сурет – Екінші ретті кестелерді реттеу терезесі

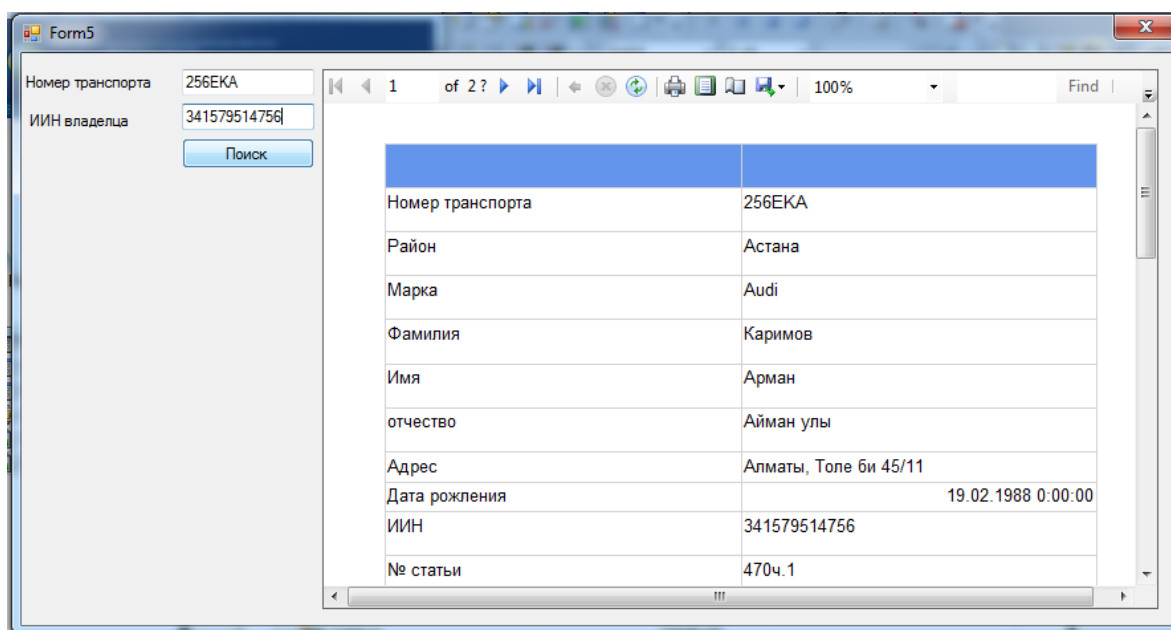
Заңды бұзушылық жасаған жүргізушілер мен айыппұлды рәсімдеу қызметкерлердің басты міндеттерінің бірі болып табылады.



4.5-сурет – Жүргізуші қосу терезесі



4.6-сурет – Айыппұл қосу терезесі



4.7-сурет – Айыппұл іздеу терезесі

4.7-суретте көрсетілген терезе арқылы жүргізуші өз айыппұлдарын көріп қағаз бетіне шығара алады. Оның қолдану интерфейсі жеңіл болғандықтан кез-келген адам оңай қолдана алады. Өз айыппұлдарын көру үшін екі түрлі іздеу мүмкіндігі бар. Жүргізушілер үшін өз авто көлігінің номері немесе куәлігінің ИИН арқылы іздеу қарастырылған.

5 Тіршілік қауіпсіздігі

5.1 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. 5.1-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 монитору;

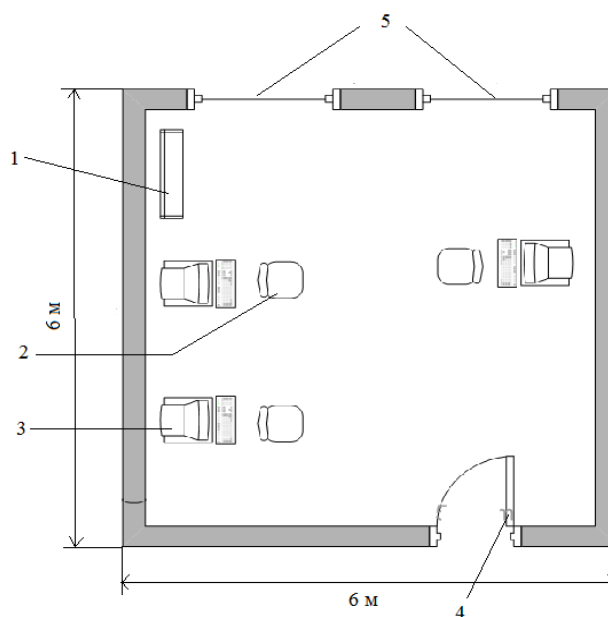
– Мөлшерлер1200x750x1150 мм (дербес компьютер+үстел);

– 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 5.1 – 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



5.1-сурет – Офис жоспары

мұндағы

- 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 ластау 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 2-кестеде келтірілген.

К е с т е 5.2 – Адам ағзасының энергия жұмсау жұмыстарының категориялары

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

5.2 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:

$$Q_{\text{жылу.б}} = Q_{\text{коршау}} + Q_p + Q_a^a + Q_{\text{жарықтану}} + Q_{\text{құрал}}, \text{ Вт} \quad (5.1)$$

мұндағы $Q_{\text{коршау}}$ – 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:

$$Q_{\text{коршау}} = V_{\text{бөлме}} X_0 (t_{\text{шыққан}} - t_{\text{келген}}) \text{Вт} \quad (5.2)$$

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], $t_{\text{ш.есеп}} = 24^{\circ}\text{C}$
 $Q_{\text{коршау}} = 108 \times (27,6 - 24) \times 0,42 = 163,3 \text{ Вт}$.
 Салқын мезгіл үшін: $t_{\text{сырт.есеп}} = -25^{\circ}\text{C}$ [12, кесте 3], $t_{\text{ш.есеп}} = 20^{\circ}\text{C}$
 $Q_{\text{коршау}} = 108 \times 0,42 \times (-25 - 20) = -2041,2 \text{ Вт}$.

Шынылау арқылы күннің сәулеленуінен келетін жылу

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

$$Q_p = (q_{\text{тура}} + q_{\text{шашыр.}}) K_1^c K_2 \beta_{\text{ж.ө.}} n H_0 B_0, \text{ Вт} \quad (5.3)$$

Күннің сәулелері терезеден кірмейтін көлеңке мезгілы үшін (шашыраңқы радиация):

$$Q_{p.} = q_{\text{шашыр.}} K_1^T K_2 \beta_{\text{ж.ө.}} n H_0 B_0, \text{ Вт} \quad (5.4)$$

мұнда $q_{\text{тура}}$; $q_{\text{шашыр.}}$ – 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. $t = 24$ °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:

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

$t = 20$ °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:

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

$t = 24$ °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.

$t = 20$ °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.

Кесте 5.3 – Бөлмедегі адамдардан бөлінетін зиянды заттардың есептелуінің нәтижелері

Жыл мезгілі	Температура °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:

$$Q_{\text{жарықтану}} = \eta N_{\text{жарықтану}}, \text{ Вт} \quad (5.5)$$

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

$N_{\text{жарықтану}}$ – шамдардың орнатылған қуаты 65 Вт/м².

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:

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

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

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

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:

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

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

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

Ауа алмасуды есептеу

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

$$Q_k = \frac{Q_{\text{жылу.б}} \cdot 860}{V_{\text{бөлме}}} = \frac{3,669 \cdot 860}{6 \cdot 6 \cdot 3} = \frac{29,22 \text{ ккал}}{\text{м}^3} \quad (5.6)$$

$Q_k > 20 \text{ ккал/м}^3$ болғанда $\Delta t = 8^\circ \text{C}$.

Бөлмеге қажет ауаның мөлшері жылулық баланстан алынып, мына формуламен анықталады:

$$L = \frac{Q_{\text{жылу.б}} \cdot 860}{C \cdot \Delta t \cdot \gamma} = \frac{3,669 \cdot 860}{0,24 \cdot 8 \cdot 1,206} = 1363 \text{ м}^3/\text{сағ} \quad (5.7)$$

мұнда $C = 0,24 \text{ ккал/кг}^\circ \text{C}$ - ауаның жылу сыйымдылығы;

$\gamma = 1,206 \text{ кг/м}^3$ - ағынды ауаның сыбағалы массасы.

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

Кондиционердің техникалық сипаттамалары:

- салқын $3,5 \text{ кВт}$; жылу $3,5 \text{ кВт}$;
- салқын $\text{кКал/сағ} \ 3 \ 024 \ (3 \ 517)$;
- 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.

6 Технико – экономикалық негізделуі

6.1 Жұмысты сипаттау және қажеттілігін негіздеу

Берілген бітіру жұмысының тақырыбы – ҚР мемлекеттік автомобильді инспекциясының деректер қорын құру.

Берілген өнімді әзірлеудің мақсаты болып деректерді енгізу, сақтау, өңдеу және тасымалдау үшін қолданылатын ақпарат пен қорлардың тұтастығын, қол жеткізулік ыңғайлылығын қолдау болып табылады.

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

- 1) жаңа қызметкерлер туралы ақпаратты қағаз жүзінде емес, бағдарламаға оңай енгізіледі;
- 2) пайдаланушыда қызметкерлер деректер қоры архив сақталады;
- 3) қағаз түріндегі айыппұлды электронды түрге өткізу;
- 4) жылдық есеп беруге оңай қол жеткізуге болады;
- 5) МАИ қызметкерлерінің қатысынсыз айыппұл тізімін алуға мүмкіндік бар.

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 жүргізушілер мен олардың айыппұлдарының деректер қорын қолданатын бағдарламалық қамтама;
- 2) құн (Price)– бұл бағдарламалық қамтаманы пайдаланатын мекемеге қойылған ақша сомасы;
- 3) дистрибуция - бағдарламалық қамтама жеткізілетін ұсыныс берген мекеме.

6.3 Бағдарламаның SWOT-анализі

К е с т е 6.1 – Бағдарламаның SWOT-анализі

Артықшылықтары	Кемшіліктері
Қамтаманың ыңғайлылығы және жеңілдігі. Деректер қорының толығымен МАИ жұмысын басқаруы. Жүргізушілердің өз бетінше айыппұлды қарау мүмкіндігі	Ғаламторға қосылу мүмкіндігі жоқ

Кесте 6.1 соңы

Мүмкіншілігі	Қауіпі
Заманауи талаптарға сай барлық әкімшілік мекемелерде қолданылу мүмкіндігі	Ақпараттың сыртқа шығуы

6.4 Маркетинг-микс элементі

К е с т е 6.2 – Бағдарламаның маркетинг-микс кестесі

Маркетинг-микс элементтері	Қажетті істер мен талдаулар
Product	
МАИ деректер қоры	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	Барлық әкімшілік 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):

$$T_{ПО} = 24 \text{ адам / сағ}$$

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 6.3-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:

$$T_{OT} = Q \times 4,2 / 50 \times K = 3150 \times 4,2 / 50 \times 0,8 = 330,75 \text{ адам/сағ} \quad (6.6)$$

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:

$$T = T_{\text{ПО}} + T_{\text{ТО}} + T_{\text{А}} + T_{\text{БС}} + T_{\text{Н}} + T_{\text{П}} + T_{\text{ОТ}} + T_{\text{Д}}. \quad (6.7)$$

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

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} – (6.7) 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:

$$Z_{\text{осн}} = 835,5 * 19966 / (21 * 8) = 99295,196 \text{ теңге.}$$

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

$$Z_{\text{доп}} = 0,2 * Z_{\text{осн}} = 19859,039 \text{ теңге.} \quad (6.9)$$

Жалпы еңбек ақысы (еңбекті төлеу фонды) негізгі және қосымша еңбек ақының қосындысы ретінде анықталады (10):

$$\text{ФОТ} = Z_{\text{доп}} + Z_{\text{осн}} = 99295,196 + 19859,039 = 119154,235 \text{ теңге.} \quad (6.10)$$

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, жүкқұжатқа 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 $C_{\text{обор}} = 500000$ 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:

$$\begin{aligned} C_{Мжк} &= 0,015 \times C_{обор} = 0,015 * 500000 = 7500 \text{ теңге,} \\ C_{ТО} &= 0,025 \times C_{обор} = 0,025 * 500000 = 12500 \text{ теңге.} \end{aligned} \quad (6.11)$$

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.

$$N = T/24 = 835,5/24 = 34,8 \text{ күн}$$

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

Дербес компьютерде жалпы жұмыс істеу уақыты мына формуламен есептеледі:

$$t = T_A + T_{БС} + T_H + T_{П} + T_{ОТ} = 78,75 + 78,75 + 118,125 + 63 + 330,75 = 669,375 \text{ адам/ сағ}$$

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:

$$C_{ликв} = 0,05 * C_{обор} = 0,05 * 500000 = 25000 \text{ теңге.} \quad (6.14)$$

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

$$H_A = \frac{C_{обор} - C_{ликв}}{T_{норм} \cdot C_{обор}} \cdot 100\% = \frac{500000 - 25000}{4 * 500000} * 100\% = 23,75\%,$$

$$A = \frac{\text{Собор} \times H_A \times N}{100 \times 12 \times t} = 500000 \times 23,75 \times 669,125 / 100 \times 34,812 \times 12 = 190263,02$$

теңге.

Бағдарламалық камтаманы 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 (16):

$$F_{эф} = D_{ном} \cdot d \cdot (1 - f / 100) = 258 \cdot 8 \cdot (1 - 2,2 / 100) = 2019 \text{ сағ.} \quad (6.16)$$

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):

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

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

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

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

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

Сәйкесінше, ЭЕМ-ның жалпы жылдық эксплуатационды шығын құны:

$$\Xi_3 = 433,5 * 71571,307 / 2019 = 15302,68 \text{ теңге}$$

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

$$CC = (\text{ФОТ} - 0,1 * \text{ФОТ}) * 0,11 = (119154,235 - 0,1 * 119154,235) * 0,11 = 11796,269 \text{ теңге} \quad (6.19)$$

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:

$$C_{\text{накл}} = 0,5 * \text{ФОТ} = 5896,37 \text{ теңге} \quad (6.20)$$

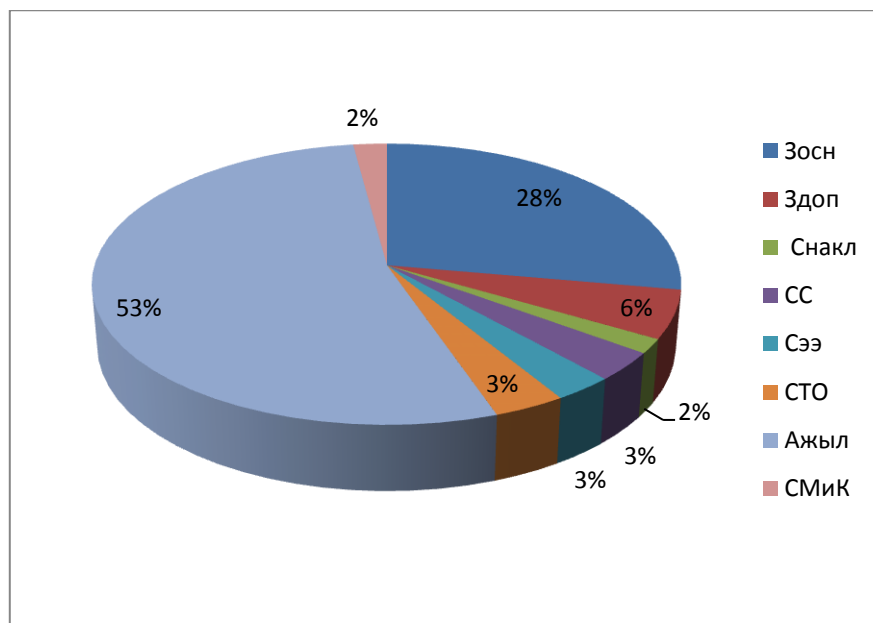
6.5.4 Өзіндік құн нәтижесінің кестесі мен диаграммасы

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 кесте 6.3-те көрсетілген.

К е с т е 6.3 – Өзіндік құн нәтижесінің кестесі

Шығын статьялары		Сомма, теңге	Жалпы соммадан, %
ФЗП	$Z_{\text{осн}}$	99265,49	28
	$Z_{\text{доп}}$	19853,1	6
Жүктеме шығындар, $C_{\text{накл}}$		5896,37	2
Әлеуметтік сақтандыру, CC		11792,7	3
Эксплуатационды Шығындар	$C_{\text{ээ}}^*$	10435,5	3
	$C_{\text{го}}$	12500	3
	$A_{\text{жыл}}^*$	190263	53
Материалдар мен комплектілер, $C_{\text{мик}}$		7500	2
Барлығы:		357506,02	100

Өзіндік құнының диаграммасы:



6.1-сурет – Өнімнің өзіндік құнының диаграммасы

6.6 Бағдарламаның бағасына есептеу

Өнімнің өзіндік құны 357506,02 теңге (жоғарыда кесте 6.3) болды. Өнімнің минималды бағасы келесі формуламен есептеледі:

$$Ц_{\min} = C_{\text{пс}} * (1 + r/100) = 357506,02 * 1,21 = 432582,3 \text{ теңге} \quad (6.21)$$

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

$Ц_{\text{з}}$ – жазба бағасы;

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

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

$$Ц_{\min\text{НДС}} = Ц_{\min} + N_{\text{ндс}} Ц_{\min} = 432582,3 + 0,12 * 432582,3 = 484492,2 \text{ теңге} \quad (6.22)$$

Сонда бағдарламаның құны 484492,2 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 484492,2 теңге (6.22 формула).
Көлік шығындары жүйе құнының 20% құрайды:

$$C_{\text{тр}}=484492,2 * 0,2=96898,43 \text{ теңге}$$

Жұмысшыларды жүйемен жұмыс істеуге оқыту шамамен екі сағатта консалтингті фирманың маманы көмегімен сағатына 30000 теңгедей болып табылады [16]. Соған сәйкес оқу бағасы:

$$C_{\text{обуч}}=30000*10=300000 \text{ теңге}$$

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	484492,2
Көлік шығындары	96898,43
Жұмысшыларды оқыту бағасы	300000
Қорытынды:	1666391

6.6.4 Қолдану саласындағы жылдық шығындарды есептеу 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;

Үнемдеу мекемеде деректер қорын жүргізетін диспетчерлер санын азайту арқасында болады. Бағдарламаны енгізер алдында мекемеде 7 диспетчер болған, бағдарлама енгізгеннен кейін 2 диспетчерге азайды. Одан басқа мекемеде 5 қосымша жұмысшылар бар. Диспетчер ЕА 55000 теңге. Деректер қорын қолмен барысында еңбектің жылдық төлем қоры:

$$C_{\text{дисп}}=55000*2*12=1320000 \text{ теңге.}$$

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

$$CC=(1320000-1320000*0,1)*0,11=120600 \text{ теңге}$$

Қамтаманы қолданысқа енгізгеннен кейін мекеме жылына 1200000 теңге үнемдейді.

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

Шығындар түрлері	Сомма, теңге
Еңбекке төленетін қор	1320000
Әлеуметтік салыққа аударымдар	120600
Қорытынды:	1200000

6.7 Ақшалай құралдардың қозғалысы

К е с т е 6.8– Ақшалай тәсілдердің қозғалысы, теңге

Аты	Жыл		
	2014	2015	2016
Бірмезгілдік шығындар	1666391		
Операциялық кәсіпкерлікте н пайда		1200000	1200000
Дисконттау коэффициенті (21% мөлшерінде)	1	0,826	0,683

Кесте 6.8-нің соңы

Аты	Жыл
-----	-----

	2014	2015	2016
Таза дисконтталған табыс (ТДТ)	-1666391	991735,5	819616,1
ТДТ өспелі нәтижесімен	-1666391	-674655,5	144961,1

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

$$\text{ТДТ}_1 = 1200000 * 0,826 = 991735,5 \text{ теңге}$$

$$\text{ТДТ}_2 = 1200000 * 0,683 = 819616,1 \text{ теңге}$$

6.8 Экономикалық тиімділікті есептеу

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

Шығындары бізмезгілде 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

$$NPV(21\%) = \left(\frac{1200000}{(1+0,21)^1} + \frac{1200000}{(1+0,21)^2} \right) - 1666391 = 144961,1 \text{тг} > 0$$

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

6.8.2 Пайда индексін есептеу (Profitability index, PI)

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

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

$$PI = (1811352/1666391) = 1,09\%$$

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.

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

$$r_b=30\%: \quad PV_1 = 1200000 / (1 + 0,3) = 923076,9 \text{ теңге,}$$

$$PV_2 = 1200000 / (1 + 0,3)^2 = 710059,2 \text{ теңге,}$$

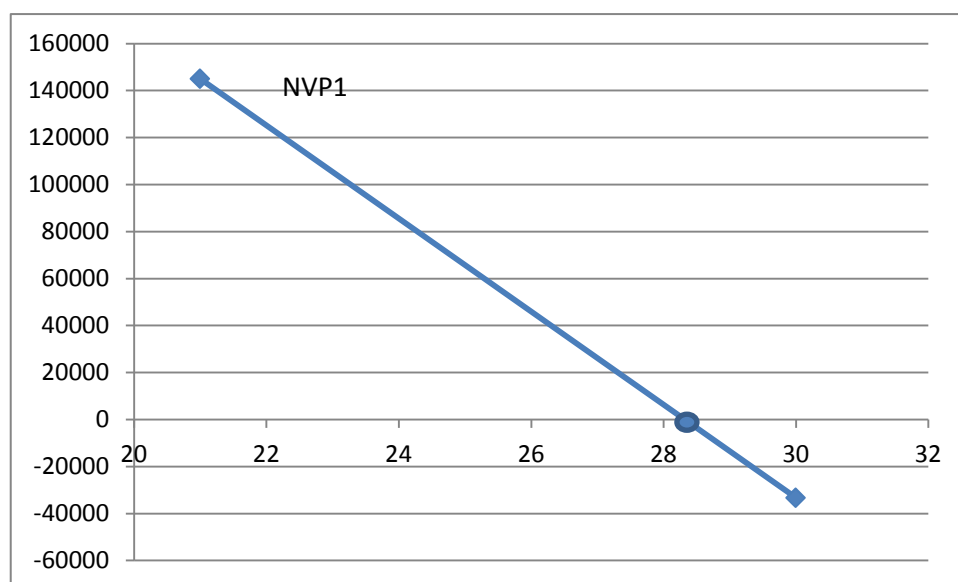
$$NPV(30\%) = (923076,9 + 710059,2) - 1200000 = -33254,5 \text{ теңге}$$

$$NPV(21\%) = 144961,1 \text{ теңге,}$$

$$NPV(30\%) = -33254,5 \text{ теңге,}$$

$$IRR = r_1 + \frac{NPV(r_1)}{NPV(r_1) - NPV(r_2)} \times (r_2 - r_1) = 21 + \frac{144961,1}{144961,1 - (-33255)} * (30 - 21) = 28,3\%$$

Табыстың ішкі нормасы сурет 6.2-де көрсетілген:



Сурет-6.2 - 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:

$$T_{ок} = 1 + \frac{1666391 - 991735,5}{819616,1} = 1,82 = 1 \text{ жыл } 10 \text{ ай.}$$

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

Қорытынды

Бұл дипломдық жобада барлық қойылған мақсаттар орындалды. Қызметкерлер дерекқоры құрылды. Дерекқорды құру барысында барлық кемшіліктер ескерілді. Беріліп отырған бағдарламалық МАИ-ны автоматтандыруға мүмкіндік береді.

Деректер қоры MS SQL Server 2008 нұсқасы дерекқоры басқару жүйесінде жасалды. Жаңа түскен айыппұлдарды тек қызметкер енгізе алады. Мұндай қатынау принципі МАИ қызметкерлері үшін ыңғайлы. Сонымен қатар, авторизация қолдану арқылы деректердің қауіпсіздік пен құпиялылық қамтамасыз етілді.

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

Техника-экономикалық негізделуі бөлімінде жасалған есептеулер нәтижесінде бағдарламалық өнімді жүзеге асыруға кеткен шығын 1666391 теңгені құрады. Әсіресе ірі мекемеде кең таралған, сондықтан жүзеге асырылуының жоғары бағасына қарамастан, берілген өнім экономикалық тиімді болып саналады.

Әдебиеттер тізімі

- 1 Бондарь А.Г. MS SQL Server 2012. Создание баз данных и разработка программ. –БХВ-Петербург, 2013. – 608с.
- 2 Д. Петкович Microsoft SQL Server 2012. Руководство для начинающих. –БХВ-Петербург, 2013. – 816с.
- 3 Джо Майо: Самоучитель Microsoft Visual Studio 2010. – БХВ, 2011 г.– 464с.
- 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 Мамаев Е. MySQL Server. Проектирование и реализация баз данных. Сертификационный экзамен. – СПб.: ВHV, 2004. – 416с.
- 7 Базылов Қ.Б., Алибаева С.А., Нурмагамбетова С.С. Бітіруші жұмысының экономикалық бөлімі үшін әдістемелік нұсқаулар. – Алматы: АИЭС, 2009. – 25 б.
- 8 Иванов И.Н. Экономика промышленного предприятия: Учебник. – М.: ИНФРА–М, 2011. – 395 с.
- 9 Рофе А.И. Экономика труда: Учебник. – М.: КноРус, 2010. – 400 с.
- 10 Остапенко Ю.М. Экономика труда: Учеб. пособие. – М.: ИЦ РИОР, 2010. – 160 с.
- 11 Санатова Т.С., Абдимуратов Ж.С. Өміртіршілігінің қауіпсіздігі. – Алматы: АЭЖБИ, 2008. – 16 б.
- 12 ГОСТ 12.2.032-78 ССБТ. Рабочее место при выполнении работ сидя.
- 13 ГОСТ 21.889-76 ССБТ. Система "человек-машина" Кресло человека – оператора. Общие эргонометрические требования.
- 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.
- 15 СНиП РК 2.04-05-2002. Естественное и искусственное освещение.

Қосымша А

Бағдарламаның MS SQL Server-дегі листингі

```
Drop database GAI2;
```

```
go
```

```
CREATE DATABASE GAI2
```

```
ON PRIMARY
```

```
(NAME = GAI2_data, FILENAME='C:\Program Files\Microsoft SQL  
Server\MSSQL10_50.MSSQLSERVER\MSSQL\DATA\GAI2_data.mdf', size = 5, maxsize =100,  
filegrowth = 2 mb)
```

```
LOG ON
```

```
(NAME = GAI2_log, FILENAME='C:\Program Files\Microsoft SQL  
Server\MSSQL10_50.MSSQLSERVER\MSSQL\DATA\GAI2_log.ldf', size = 5, maxsize = 120,  
filegrowth =2 mb);
```

```
CREATE TABLE nation (
```

```
    kod_nation int identity (1,1) NOT NULL PRIMARY KEY,  
    nation_name nchar (25) NOT NULL);
```

```
go
```

```
CREATE TABLE marka (
```

```
    kod_marki int identity (1,1) NOT NULL PRIMARY KEY,  
    marka_name nchar (25) NOT NULL);
```

```
go
```

```
CREATE TABLE kuzov (
```

```
    kod_kuzova int identity (1,1) NOT NULL PRIMARY KEY,  
    kuzov_name nchar (25) NOT NULL);
```

```
go
```

```
CREATE TABLE svet (
```

```
    kod_sveta int identity (1,1) NOT NULL PRIMARY KEY,  
    svet_name nchar (25) NOT NULL);
```

```
go
```

```
CREATE TABLE raion (
```

```
    kod_raiona int identity (1,1) NOT NULL PRIMARY KEY,  
    raion_name nchar (45) NOT NULL);
```

```
go
```

```
CREATE TABLE vlad (
```

```
    IIN_vlad nchar (12) NOT NULL PRIMARY KEY ,  
    familiya nchar (25) NOT NULL,  
    Imya nchar (20) NOT NULL,  
    Ot4estvo nchar (20) NULL,  
    adres_propiski nvarchar (max) NOT NULL,  
    kod_nation int NULL,  
    data_rojdenya date not null,  
    data_polu4eniya_udo date NOT NULL,  
    data_godnosti_udo date NOT NULL,  
    Kem_vidano nchar (20) NULL,  
    id_prav nchar (10) NULL);
```

go

```
CREATE TABLE kompaniya (  
    bin_komp nchar (12) NOT NULL PRIMARY KEY,  
    nazvanie nchar (30) NOT NULL,  
    adress nchar (40) NOT NULL,  
    telefon nchar (15) NULL,  
    facs nchar (15) NULL,  
    director_IIN nchar (12) NULL);
```

go

```
CREATE TABLE Avto(  
    Num_avto nchar (8) not null PRIMARY KEY ,  
    kod_raiona int NOT NULL,  
    kod_marki int NOT NULL,  
    kod_kuzova int NULL,  
    kod_sveta int NOT NULL,  
    num_kuzova nchar (15) NOT NULL,  
    num_dvig nchar (15) NOT NULL,  
    god_vipuska int NULL,  
    bin_komp nchar (12) NULL,  
    IIN_vlad nchar (12) NULL );
```

go

```
CREATE TABLE Ugon (  
    id_ugon int IDENTITY(1,1) NOT NULL PRIMARY KEY ,  
    num_avto nchar (8) NOT NULL,  
    data_ugona date NOT NULL,  
    status2 nchar (15) NOT NULL,  
    opisanie nvarchar (max) NOT NULL);
```

go

```
CREATE TABLE o_shtrafe (  
    kod_shtrafa int IDENTITY(1,1) NOT NULL PRIMARY KEY,  
    statya nchar (10) NOT NULL,  
    opisanie nvarchar (max) NOT NULL,  
    summa_v_mrp nchar (10) NULL,  
    dop_posletstviya nvarchar (max) NULL);
```

go

```
CREATE TABLE kamery (  
    id_kamery int IDENTITY(1,1) NOT NULL PRIMARY KEY,  
    tip nchar (40) NULL,  
    kompaniya nchar (20) NULL,  
    data_ustanovki date NULL,  
    cena int NULL,  
    Mesto_raspolojeniya char (60) NULL);
```

go

```
CREATE TABLE Shtraf (  
    id_shtraf int IDENTITY(1,1) NOT NULL PRIMARY KEY,  
    kod_shtrafa int NOT NULL,  
    status1 nchar (10) NOT NULL,  
    opisanie_sotrudnika nvarchar (max) NULL,
```

А қосымшаның жалғасы

```
data_shtrafa date NOT NULL,  
vremya time (7) not NULL,  
num_avto nchar (8) NOT NULL,  
id_kamery int NULL,  
iin_sotrud nchar (12) NULL);  
go  
CREATE TABLE doljnost (  
    Kod_doljnosti int IDENTITY(1,1) NOT NULL PRIMARY KEY,  
    doljnost_name nchar (29) NOT NULL,  
    ZP int NOT NULL);  
go  
CREATE TABLE sotrudnik (  
    IIN_sotrud nchar (12) NOT NULL PRIMARY KEY,  
    Imya nchar (20) NOT NULL,  
    Familiya nchar (25) NOT NULL,  
    Ot4estvo nchar (30) NULL,  
    Kod_doljnosti int NOT NULL,  
    Kod_raiona int NULL,  
    data_vida4i_udo date NULL,  
    data_godnos_udo date NULL,  
    data_rojdenya date NOT NULL,  
    adress nchar (60) NULL,  
    telefon nchar (15) NULL);  
  
ALTER TABLE avto  
    ADD CONSTRAINT PK_avto_kuzov FOREIGN KEY (kod_kuzova) REFERENCES kuzov  
(kod_kuzova);  
go  
ALTER TABLE avto  
    ADD CONSTRAINT PK_avto_marka FOREIGN KEY (kod_marka) REFERENCES marka  
(kod_marka);  
go  
ALTER TABLE avto  
    ADD CONSTRAINT PK_avto_raion FOREIGN KEY (kod_raiona) REFERENCES raion  
(kod_raiona);  
go  
ALTER TABLE avto  
    ADD CONSTRAINT PK_avto_svet FOREIGN KEY (kod_sveta) REFERENCES svet  
(kod_sveta);  
go  
ALTER TABLE avto  
    ADD CONSTRAINT PK_avto_vlad FOREIGN KEY (IIN_vlad) REFERENCES vlad  
(IIN_vlad);  
go  
ALTER TABLE avto  
    ADD CONSTRAINT PK_avto_kompaniya FOREIGN KEY (bin_komp) REFERENCES  
kompaniya (bin_komp);  
go  
ALTER TABLE vlad
```

А қосымшаның жалғасы

```
ADD CONSTRAINT PK_vlad_nation FOREIGN KEY (kod_nation) REFERENCES nation
(kod_nation);
go
ALTER TABLE ugon
ADD CONSTRAINT PK_ugon_avto FOREIGN KEY (num_avto) REFERENCES avto
(num_avto);
go
ALTER TABLE shtraf
ADD CONSTRAINT PK_shtraf_avto FOREIGN KEY (num_avto) REFERENCES avto
(num_avto);
go
ALTER TABLE shtraf
ADD CONSTRAINT PK_shtraf_sotrudnik FOREIGN KEY (IIN_sotrud) REFERENCES
sotrudnik (IIN_sotrud);
go
ALTER TABLE shtraf
ADD CONSTRAINT PK_shtraf_o_shtrafe FOREIGN KEY (kod_shtrafa) REFERENCES
o_shtrafe (kod_shtrafa);
go
ALTER TABLE shtraf
ADD CONSTRAINT PK_shtraf_kamery FOREIGN KEY (id_kamery) REFERENCES kamery
(id_kamery);
go
ALTER TABLE sotrudnik
ADD CONSTRAINT PK_sotrudnik_doljnost FOREIGN KEY (Kod_doljnosti) REFERENCES
doljnost (Kod_doljnosti);
go
ALTER TABLE kompaniya
ADD CONSTRAINT PK_kompaniya_vlad FOREIGN KEY (director_IIN) REFERENCES vlad
(IIN_vlad) ;

INSERT INTO raion (raion_name)
VALUES
('Астана'),
('Алматы'),
('Ақмолинская область'),
('Ақтөбінская область'),
('Алматынская область'),
('Атырауская область'),
('Западно-Казахстанская область'),
('Жамбылская область'),
('Қарағандынская область'),
('Қостанайская область'),
('Қызылордынская область'),
('Мангистауская область'),
('Южно-Казахстанская область'),
('Павлодарская область'),
('Северо-Казахстанская область'),
('Восточно-Казахстанская область');
```

А қосымшаның жалғасы

```
go
INSERT INTO doljnost (doljnost_name,ZP)
VALUES ('Инспекторы',100000),
('Замкомроты',120000),
('Комроты',140000),
('Замначальник',150000),
('Начальник',160000);
go
INSERT INTO kamery (tip,kompaniya,data_ustanovki,cena,Mesto_raspolojeniya)
VALUES('Радар1','Рапира1','2009-01-20',100000,'Сайна156'),
('Фиксатор','Рапира','2013-01-30',100000,'Толеби-гагарина'),
('Радар(ночноевидение)','беркут','2011-01-30',100000,'Фурионова25'),
('Радар','Phyton','2010-01-30',100000,'Абая-Алтынсарина');
go
INSERT INTO nation (nation_name)
VALUES ('Казах'),
('Руский'),
('Узбек'),
('Киргиз'),
('Немец'),
('Китаец'),
('Кореец');
go
INSERT INTO kuzov (kuzov_name)
VALUES ('седан'),
('хэч-бэк'),
('универсал'),
('кабриолет'),
('бюик'),
('Минивэн'),
('Купе');
go
INSERT INTO marka (marka_name)
VALUES ('Nissan'),
('Ford'),
('Audi'),
('LandRover'),
('Mazda'),
('BMW'),
('Mercedes'),
('Opel'),
('Dodge'),
('Chevrolet'),
('Lada'),
('Infiniti');
go
INSERT INTO svet (svet_name)
VALUES ('Черный'),
('Белый'),
```


А қосымшаның жалғасы

('Каричневый'),
(Серый),
(Розавый),
(Синий),
(Бежовый),
(Желтый),
(оранжевый),
(Красный),
(Зеленный),
(Фиолетовый);

go

INSERT INTO vlad

(IIN_vlad,familya,Imya,Ot4estvo,adress_propiski,kod_nation,data_rojdenya,data_godnosti_udo,data_polu4eniya_udo,Kem_vidano)

VALUES (124579514132,'Муратбев','Алмас','Максут улы','Алматы, Толе би 45/11',1,'1991-12-23','2010-03-06','2016-08-27','МЮРК'),
(465555514545,'Арманбеков','Бауыржан','Алдияр улы','Алматы, Толе би 45/11',1,'1992-10-02','2010-03-06','2016-12-22','МЮРК'),
(348532514324,'Алпысбаев','Ильяяс','Айбек улы','Алматы, Толе би 45/11',4,'1989-09-12','2009-03-06','2015-12-04','МЮРК'),
(341579514756,'Каримов','Арман','Айман улы','Алматы, Толе би 45/11',3,'1988-02-19','2007-03-06','2017-06-09','МЮРК'),
(545579514654,'Саргаев','Олжас','','Алматы, Толе би 45/11',1,'1980-06-14','2006-03-06','2016-05-11','МЮРК');

INSERT INTO kompaniya (bin_komp,nazvanie,adress,telefon,facs,director_IIN)

VALUES (457894134515,'Megamas','Almaty,Satbaeva 231/169','8727-263-89-45','245-68-78',341579514756),
(565394134575,'Altel','Almaty,Baitursinova 45/54','8727-223-55-75','245-68-78',545579514654),
(957595134475,'Akbulak','Almaty,Шолохова 123/4','8727-213-23-89','245-56-55',348532514324),
(557894454745,'Alatau','Almaty,Алматинская 48/6','8727-274-77-68','245-59-99',46555551454),
(157894755554,'Asyiltas','Almaty,Достык 1/5','8727-226-89-56','248-99-11',124579514132);

go

INSERT INTO o_shtrafe(statya,opisanie,summa_v_mrp,dop_posletstviya)

VALUES ('462ч.3','Превышение скорости движения ТС на величину более 40 км/ч','40','Направляются на сдачу экзамена для проверки знаний ПДД'),
(463-2ч.2','Разворот или движение задним ходом в местах, где такие маневры запрещены','10','Более трех раз в год лишение прав на 6 мес'),
(463-3ч.1','Движение по пешеходным дорожкам, обочинам или тротуарам в нарушение правил дорожного движения','15','Более трех раз в год лишение прав на 6 мес'),
(472ч.2','Перевозка опасных грузов автотранспортными средствами либо спец. автотранспортными средствами с нарушением установленных правил, а равно без спец. разрешения на перевозку опасного груза классов 1, 6 и 7','10','Помещение ТС на штрафстоянку'),
(463-2ч.1','Невыполнение требования правил дорожного движения подать сигнал перед началом движения, перестроения, поворота, разворота или остановки','5','Более трех раз в год лишение прав на 6 мес'),

А қосымшаның жалғасы

('470ч.1','Нет водительского удостоверения вообще , а равно управление ТС водителем, не имеющим права управления соответствующей категорией транспорта, либо передача управления ТС лицу, не имеющему права управления ','15','Помещение ТС на штрафстоянку'),
(470 ч.1','Забыты водительское удостоверение или временное удостоверение, свидетельство о регистрации ТС','5','Помещение ТС на штрафстоянку'),
(470ч.2','Забыт страховой полис','1','Помещение ТС на штрафстоянку'),
(470ч.3','Несоблюдение водителями ТС правил остановок маршрутных ТС, движения в жилых зонах, перевозки пассажиров и грузов, пользования ремнями безопасности при движении на ТС','5','Направляются на сдачу экзамена для проверки знаний ПДД'),
(462ч.1','Превышение скорости движения ТС на величину от 10 до 20 км/ч','10','Более трех раз в год лишение прав на 6 мес.'),
(462ч.2','Превышение скорости движения ТС на величину от 20 до 40 км/ч','15','Направляются на сдачу экзамена для проверки знаний ПДД');

go

INSERT INTO

sotrudnik(IIN_sotrud,Familiya,Imya,Ot4estvo,adress,data_rojdenya,telefon,Kod_doljnosti,Kod_raiona,data_vida4i_udo,data_godnos_udo)

VALUES (547985634552,'Албанбекова','Айжан','Саргази кызы','Алматы,Сатбаева 45/4','1980-10-22','+77015548764',4,6,'2010-03-06','2016-08-27'),

(557985634977,'Турмаганбетов','Алмухаед','Алмас улы','Нурмакова 45/1','1982-09-26','+7078997789',1,10,'2009-04-22','2017-08-21'),

(587985634422,'Алтынбек','Бахтияр','Кайрат улы','Кабанбая 78/22','1976-02-04','+77023324574',1,6,'2007-03-06','2016-02-17'),

(321985634222,'Асылхан','Нуржан','Канат улы','Водазаборная 12/19','1979-01-22','+77758982536',2,2,'2005-03-06','2016-10-07'),

(427985634444,'Абен','Айдар','','Айманова 124/45','1981-11-17','+7778632147',1,3,'2010-03-06','2014-02-05'),

(997985634837,'Байболатов','Анвар','','Майлина 74/44','1963-06-15','+77056658877',2,5,'2007-02-04','2015-09-09');

go

INSERT INTO

Avto(Num_avto,kod_marki,god_vipuska,kod_kuzova,kod_sveta,kod_raiona,IIN_vlad,bin_komp,num_dvig,num_kuzova)

VALUES

('895ASA',4,2000,2,5,6,'124579514132',null,'AS1245671B','as4a5d1a2d48a4'),

('456AVA',6,1998,1,6,2,'46555551454',null,'AS4521879B','HNO4O5BT5'),

('256EKA',3,2001,6,7,1,'341579514756',null,'BD4521879A','ASRE45HJ14HJ15'),

('758AFA',9,1993,5,8,1,'341579514756',null,'BD4521879A','ASRE45HJ14HJ15'),

('965AWA',10,1992,4,9,1,'545579514654',null,'BD4521879A','ASRE45HJ14HJ15'),

('286ABA',7,1999,3,11,1,null,'457894134515','BD4521879A','ASRE45HJ14HJ15');

go

INSERT INTO

Shtraf(num_avto,data_shtrafa,vremya,kod_shtrafa,status1,iin_sotrud,id_kamery,opisanie_sotrudnika)

VALUES ('895ASA','2013-05-06','22:00',6,'неоплочен','547985634552',null,' '),

('456AVA','2014-05-15','20:00',1,'неоплочен','547985634552',null,' '),

('256EKA','2013-05-06','22:00',4,'неоплочен','547985634552',null,' '),

А қосымшаның жалғасы

```
('758AFA','2014-02-06','22:00',9,'оплочен','547985634552',null,' '),  
( '758AFA','2013-05-06','22:00',10,'неоплочен',null,'2',' '),  
( '965AWA','2013-05-06','22:00',9,'неоплочен','547985634552',null,' ');  
go  
INSERT INTO Ugon (num_avto,data_ugona,opisanie,status2)  
VALUES ('286ABA','2010-07-25','Украденная машина была замечена в районе г.Алматы','не  
найдена');
```

Қосымша Ә

Бағдарламаның MS Visual Studio-дегі листингі

Public Class Form3

```
Private Sub DoljnostBindingNavigatorSaveItem_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
```

```
Me.Validate()
```

```
Me.DoljnostBindingSource.EndEdit()
```

```
Me.TableAdapterManager.UpdateAll(Me.GAI2DataSet)
```

```
End Sub
```

```
Private Sub Form3_Load(ByVal sender As System.Object, ByVal e As System.EventArgs)
```

```
Handles MyBase.Load
```

```
'TODO: This line of code loads data into the 'GAI2DataSet.svet' table. You can move, or remove it, as needed.
```

```
Me.SvetTableAdapter.Fill(Me.GAI2DataSet.svet)
```

```
'TODO: This line of code loads data into the 'GAI2DataSet.kuzov' table. You can move, or remove it, as needed.
```

```
Me.KuzovTableAdapter.Fill(Me.GAI2DataSet.kuzov)
```

```
'TODO: This line of code loads data into the 'GAI2DataSet.raion' table. You can move, or remove it, as needed.
```

```
Me.RaionTableAdapter.Fill(Me.GAI2DataSet.raion)
```

```
'TODO: This line of code loads data into the 'GAI2DataSet.Shtraf' table. You can move, or remove it, as needed.
```

```
Me.ShtrafTableAdapter.Fill(Me.GAI2DataSet.Shtraf)
```

```
'TODO: This line of code loads data into the 'GAI2DataSet.Avto' table. You can move, or remove it, as needed.
```

```
Me.AvtoTableAdapter.Fill(Me.GAI2DataSet.Avto)
```

```
'TODO: This line of code loads data into the 'GAI2DataSet.nation' table. You can move, or remove it, as needed.
```

```
Me.NationTableAdapter.Fill(Me.GAI2DataSet.nation)
```

```
'TODO: This line of code loads data into the 'GAI2DataSet.marka' table. You can move, or remove it, as needed.
```

```
Me.MarkaTableAdapter.Fill(Me.GAI2DataSet.marka)
```

```
'TODO: This line of code loads data into the 'GAI2DataSet.vlad' table. You can move, or remove it, as needed.
```

```
Me.VladTableAdapter.Fill(Me.GAI2DataSet.vlad)
```

```
'TODO: This line of code loads data into the 'GAI2DataSet.kompaniya' table. You can move, or remove it, as needed.
```

```
Me.KompaniyaTableAdapter.Fill(Me.GAI2DataSet.kompaniya)
```

```
'TODO: This line of code loads data into the 'GAI2DataSet.kamery' table. You can move, or remove it, as needed.
```

```
Me.KameryTableAdapter.Fill(Me.GAI2DataSet.kamery)
```

```
'TODO: This line of code loads data into the 'GAI2DataSet.doljnost' table. You can move, or remove it, as needed.
```

```
Me.DoljnostTableAdapter.Fill(Me.GAI2DataSet.doljnost)
```

```
End Sub
```

Ә қосымшаның жалғасы

```
Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles Button1.Click
    Me.Validate()
    Me.DoljnostBindingSource.EndEdit()
    Me.TableAdapterManager.UpdateAll(GAI2DataSet)

```

End Sub

```
Private Sub Form3_Shown(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles MyBase.Shown
    DoljnostBindingSource.AddNew()
End Sub

```

```
Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles Button2.Click
    DoljnostBindingSource.RemoveCurrent()
End Sub

```

```
Private Sub TipTextBox_TextChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles TipTextBox.TextChanged

```

End Sub

```
Private Sub Button3_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles Button3.Click
    Me.Validate()
    Me.KameryBindingSource.EndEdit()
    Me.TableAdapterManager.UpdateAll(GAI2DataSet)
End Sub

```

```
Private Sub Button4_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles Button4.Click
    KameryBindingSource.RemoveCurrent()

```

End Sub

```
Private Sub Director_IINTextBox_TextChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Director_IINTextBox.TextChanged

```

End Sub

```
Private Sub FacsTextBox_TextChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles FacsTextBox.TextChanged

```

End Sub

```
Private Sub TelefonTextBox_TextChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles TelefonTextBox.TextChanged

```

Ә қосымшаның жалғасы

End Sub

Private Sub AdressTextBox_TextChanged(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles AdressTextBox.TextChanged

End Sub

Private Sub NazvanieTextBox_TextChanged(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles NazvanieTextBox.TextChanged

End Sub

Private Sub Bin_kompTextBox_TextChanged(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Bin_kompTextBox.TextChanged

End Sub

Private Sub GroupBox3_Enter(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles GroupBox3.Enter

End Sub

Private Sub Button5_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button5.Click

Me.Validate()

Me.KompaniyaBindingSource.EndEdit()

Me.TableAdapterManager.UpdateAll(GAI2DataSet)

End Sub

Private Sub Button7_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button7.Click

Me.Validate()

Me.VladBindingSource.EndEdit()

Me.TableAdapterManager.UpdateAll(GAI2DataSet)

End Sub

Private Sub Button6_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button6.Click

KompaniyaBindingSource.RemoveCurrent()

End Sub

Private Sub Button8_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button8.Click

VladBindingSource.RemoveCurrent()

End Sub

Ә қосымшаның жалғасы

```
Private Sub Button9_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles Button9.Click
    DoljnostBindingSource.AddNew()

```

```
End Sub
```

```
Private Sub Button10_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles Button10.Click
    KameryBindingSource.AddNew()

```

```
End Sub
```

```
Private Sub Button11_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles Button11.Click
    KompaniyaBindingSource.AddNew()

```

```
End Sub
```

```
Private Sub Button12_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles Button12.Click
    VladBindingSource.AddNew()

```

```
End Sub
```

```
Private Sub Button13_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles Button13.Click
    MarkaBindingSource.AddNew()

```

```
End Sub
```

```
Private Sub Button15_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles Button15.Click
    Me.Validate()
    Me.MarkaBindingSource.EndEdit()
    Me.TableAdapterManager.UpdateAll(GAI2DataSet)

```

```
End Sub
```

```
Private Sub Button14_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles Button14.Click
    MarkaBindingSource.RemoveCurrent()

```

```
End Sub
```

```
Private Sub Button16_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles Button16.Click
    NationBindingSource.AddNew()

```

```
End Sub
```

Ә қосымшаның жалғасы

```
Private Sub Button18_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles Button18.Click
    Me.Validate()
    Me.NationBindingSource.EndEdit()
    Me.TableAdapterManager.UpdateAll(GAI2DataSet)
End Sub
```

```
Private Sub Button17_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles Button17.Click
    NationBindingSource.RemoveCurrent()
End Sub
```

```
Private Sub Bin_kompLabel1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs)

End Sub
```

```
Private Sub RadioButton1_CheckedChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles RadioButton1.CheckedChanged
    Bin_kompTextBox1.Visible = True
    IIN_vladTextBox1.Visible = False
End Sub
```

```
Private Sub RadioButton2_CheckedChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles RadioButton2.CheckedChanged
    Bin_kompTextBox1.Visible = False
    IIN_vladTextBox1.Visible = True
End Sub
End Class
```

```
Public Class Form1
```

```
Private Sub ShtrafBindingNavigatorSaveItem_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs)
    Me.Validate()
    Me.ShtrafBindingSource.EndEdit()
    Me.TableAdapterManager.UpdateAll(Me.GAI2DataSet)

End Sub
```

```
Private Sub Form1_Load(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles MyBase.Load
    'TODO: This line of code loads data into the 'GAI2DataSet.vladelec' table. You can move, or
remove it, as needed.
    Me.VladelecTableAdapter.Fill(Me.GAI2DataSet.vladelec)
    'TODO: This line of code loads data into the 'GAI2DataSet.o_shtrafe' table. You can move, or
remove it, as needed.
    Me.O_shtrafeTableAdapter.Fill(Me.GAI2DataSet.o_shtrafe)
```


Ә қосымшаның жалғасы

'TODO: This line of code loads data into the 'GAI2DataSet.Shtraf' table. You can move, or remove it, as needed.

```
Me.ShtrafTableAdapter.Fill(Me.GAI2DataSet.Shtraf)
```

```
End Sub
```

```
Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)  
Handles Button1.Click
```

```
ShtrafBindingSource.AddNew()
```

```
End Sub
```

```
Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)  
Handles Button2.Click
```

```
Me.Validate()
```

```
Me.ShtrafBindingSource.EndEdit()
```

```
Me.TableAdapterManager.UpdateAll(GAI2DataSet)
```

```
ShtrafBindingSource.AddNew()
```

```
End Sub
```

```
Private Sub RadioButton1_CheckedChanged(ByVal sender As System.Object, ByVal e As  
System.EventArgs) Handles RadioButton1.CheckedChanged
```

```
lin_sotrudTextBox.Visible = True
```

```
Id_kameryTextBox.Visible = False
```

```
End Sub
```

```
Private Sub RadioButton2_CheckedChanged(ByVal sender As System.Object, ByVal e As  
System.EventArgs) Handles RadioButton2.CheckedChanged
```

```
Id_kameryTextBox.Visible = True
```

```
lin_sotrudTextBox.Visible = False
```

```
End Sub
```

```
Private Sub Button4_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
```

```
For i = 0 To DataGridView1.ColumnCount - 1
```

```
For j = 0 To DataGridView1.RowCount - 1
```

```
DataGridView1.Item(i, j).Style.BackColor = Color.Black
```

```
DataGridView1.Item(i, j).Style.ForeColor = Color.White
```

```
Next j
```

```
Next i
```

```
For i = 0 To DataGridView1.ColumnCount - 1
```

```
For j = 0 To DataGridView1.RowCount - 1
```

```
If InStr(DataGridView1.Item(i, j).Value, TextBox1.Text) Then
```

```
DataGridView1.Item(i, j).Style.BackColor = Color.AliceBlue
```

```
DataGridView1.Item(i, j).Style.ForeColor = Color.Blue
```

```
End If
```

Ә қосымшаның жалғасы

```
Next j
Next i
End Sub
```

```
Private Sub TextBox1_TextChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles TextBox1.TextChanged
    If TextBox1.Text = String.Empty Then
        VladelecBindingSource.RemoveFilter()
    Else
        VladelecBindingSource.Filter = "num_avto like '%" + _
        TextBox1.Text + "%"
    End If
End Sub
```

```
Private Sub TextBox2_TextChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles TextBox2.TextChanged
    If TextBox2.Text = String.Empty Then
        VladelecBindingSource.RemoveFilter()
    Else
        VladelecBindingSource.Filter = "familiya like '%" + _
        TextBox2.Text + "%"
    End If
End Sub
```

```
Private Sub DateTimePicker1_ValueChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs)

End Sub
```

```
Private Sub TextBox4_TextChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles TextBox4.TextChanged
    If TextBox4.Text = String.Empty Then
        ShtrafBindingSource.RemoveFilter()
    Else
        ShtrafBindingSource.Filter = "status1 like '%" + _
        TextBox4.Text + "%"
    End If
End Sub
```

```
End Class
Public Class Form4
```

```
Private Sub PictureBox1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles PictureBox1.Click
```

Ә қосымшаның жалғасы

End Sub

```
Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles Button1.Click
    If TextBox1.Text = "admin" And TextBox2.Text = "123" Then
        Form3.Show()
    Else
```

```
    End If
End Sub
```

```
Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles Button2.Click
    TextBox1.Clear()
    TextBox2.Clear()
```

End Sub

```
Private Sub Label3_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles Label3.Click
    Form5.Show()
```

End Sub

```
Private Sub Form4_Load(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles MyBase.Load
```

```
End Sub
End Class
Public Class Form5
```

```
Private Sub Form5_Load(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles MyBase.Load
    'TODO: This line of code loads data into the 'GAI2DataSet.vladelec' table. You can move, or
remove it, as needed.
    Me.vladelecTableAdapter.Fill(Me.GAI2DataSet.vladelec)
    'TODO: This line of code loads data into the 'GAI2DataSet.asdf' table. You can move, or
remove it, as needed.
    Me.AsdfTableAdapter.Fill(Me.GAI2DataSet.asdf)
    'TODO: This line of code loads data into the 'GAI2DataSet.Shtraf' table. You can move, or
remove it, as needed.
    Me.ShtrafTableAdapter.Fill(Me.GAI2DataSet.Shtraf)
```

```
    Me.ReportViewer1.RefreshReport()
    Me.ReportViewer1.RefreshReport()
End Sub
```

Ә қосымшаның жалғасы

```
Private Sub ShtrafBindingNavigatorSaveItem_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
    Me.Validate()
    Me.ShtrafBindingSource.EndEdit()
    Me.TableAdapterManager.UpdateAll(Me.GAI2DataSet)

```

End Sub

```
Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles Button1.Click
    If TextBox1.Text = String.Empty Then
        AsdfBindingSource.RemoveFilter()
    Else
        AsdfBindingSource.Filter = "num_avto like '%" + _
            TextBox1.Text + "%"
    End If
    ReportViewer1.RefreshReport()

```

End Sub

```
Private Sub TextBox1_TextChanged(ByVal sender As System.Object, ByVal e As System.EventArgs)

```

End Sub

```
Private Sub TextBox4_TextChanged(ByVal sender As System.Object, ByVal e As System.EventArgs)

```

End Sub

```
Private Sub TextBox5_TextChanged(ByVal sender As System.Object, ByVal e As System.EventArgs)

```

End Sub

```
Private Sub ReportViewer1_Load(ByVal sender As System.Object, ByVal e As System.EventArgs)

```

End Sub

```
Private Sub ReportViewer1_MouseClick(ByVal sender As System.Object, ByVal e As System.Windows.Forms.MouseEventHandler)

```

End Sub

End Class