

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. Құралбаев З.Қ.

(қолы)

« \_\_\_\_\_ » \_\_\_\_\_ 2014 ж.

ДИПЛОМДЫҚ ЖОБА

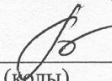
Тақырыбы: ««Шымкент қаласындағы «Ернұр Пресс» ЖШС баспа орталығы  
жұмысының деректер қорын құру»  
5B070400 – «Есептеу техникасы және бағдарламалық қамтамасыз ету»  
мамандығы бойынша

Орындаған Ереханова Алуа Еркінқызы тобы: ВТк-10-1

Жетекші доцент, ф.-м.ғ.к. Аманбаев А.А

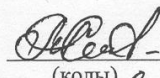
Кеңесшілер :

Экономикалық бөлім бойынша :

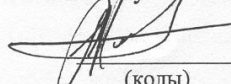
  
доцент Боканова Г.Ш.  
« 20 » 05 2014ж.  
(қолы)

Өмір тіршілігі қауіпсіздігі бойынша:

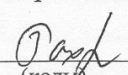
т.ғ.к., аға оқытушы Муташева Г.С.

  
« 15 » 05 2014 ж.  
(қолы)

Есептеу техникасын қолдану бойынша :

  
доцент, ф.-м.ғ.к. Аманбаев А.А.  
« 23 » 05 2014 ж.  
(қолы)

Мөлшер бақылаушы:

  
аға оқытушы Рахимжанова З.М.  
« 28 » 05 2014 ж.  
(қолы)

Пікір жазушы :

АТУ, АТ кафедрасының меңгерушісі : т.ғ.д., проф. Заурбеков Н.С.

« \_\_\_\_\_ » \_\_\_\_\_ 2014 ж.  
(қолы)

Алматы 2014

ҚАЗАҚСТАН РЕСПУБЛИКАСЫ БІЛІМ ЖӘНЕ ҒЫЛЫМ МИНИСТРЛІГІ

Коммерциялық емес акционерлік қоғамы  
АЛМАТЫ ЭНЕРГЕТИКА ЖӘНЕ БАЙЛАНЫС УНИВЕРСИТЕТІ

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

жобаны орындауға берілген

ТАПСЫРМА

Студент Ереханова Алуа Еркінқызына

Жоба тақырыбы «Шымкент қаласындағы «Ернұр Пресс» ЖШС баспа орталығы жұмысының деректер қорын құру»  
ректордың «24» Қарсүйек № 115 бұйрығы бойынша бекітілген.

Аяқталған жұмысты тапсыру мерзімі: «29» мамыр 2014 ж.

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

Қазіргі таңда "Ернұр Пресс" ЖШС баспа орталығына тиісті автоматтандырылған бағдарламалық жүйесінің шірісі сезілуде. Компьютер базасында құрылған ақпараттық жүйелер үлкен көлемді ақпараттарды өздері және ақпараттарды өздері қарама-қарсы етеміні санауға басқаруға тиісті мүмкіндіктерді баспа орталығы газеттерді баспа шығару және мақалаларды тиістілікпен айналысады. Көптеген қалай құжаттармен тиістілік пайдаланады.

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

Баспа орталығының құрылымының уақыттың үнемі өзгеріп, тиістілік пен тиімділігін, ақпарат үшін бағдарламалық жүйелердің пайдалануға тиімділік беремін, интеллектуалды әдістер мен қатар өте көптеген қатаңдықпен тиімділікпен қарама-қарсы және экономикалық пайдалану да өте тиімді тиістілік пайдаланады. Қарама-қарсы бағдарламалық жүйелер: SQL Server 2008, Visual Studio. Тиімділіктер құрылымы тиімділік пайдаланады. Тиімділікте техникалық құрылымы көрсетіледі арнайы тиімділік құрылымы.

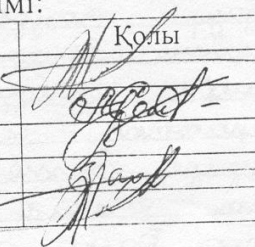
Сызба материалдарының (міндетті түрде дайындалатын сызуларды көрсету) тізімі:

1. Директордың ЕР диаграммасы.
2. Мәсәліге бағдарламаның қосымшаның түрлілік кестісі қабілеттілігін, функционалдығын көрсететін графикалық материал.
3. Бизнес ереже үшін ЕР селлекті.
4. Директор құрылым құрылымының презентациясы сызбасы.
5. Ақпараттың тиімшілігін өмірлік үлгісі сызбасы.

Негізгі ұсынылатын әдебиеттер:

1. Алексей Вишневский. SQL Server . Эффективная работа. – Санкт-Петербург, 2009. – 541с.
2. Полякова Л.Н. Основы SQL. Курс лекций: учеб. пособие. – Москва, 2004. – 368с.
3. Глушаков С.В., Ломотыко Д.В. Базы данных: Учебный курс. – М.: ООО «Издательство АСТ», 2000. – 504с.
4. Агапов В.П. Основы программирования на языке C# . учеб.пособие.- Москва,2012.-128с.
5. Голицына О.Л., Попов И.И. Основы алгоритмизации и программирования. учеб.пособие.-Форум 2008.-432с.

Жоба тараулары бойынша кеңес берушілер және оның мерзімі:

| Бөлім                      | Кеңесші          | Мерзімі    | Қолы  |
|----------------------------|------------------|------------|---|
| Негізгі бөлім              | Аманбаев А.А.    | 20.05.2014 |  |
| Тіршілік қауіпсіздігі      | Муташева Г.С.    | 15.05.2014 |   |
| Экономикалық бөлім         | Боканова Г.Ш.    | 20.05.2014 |   |
| Норма бақылаушы            | Рахимжанова З.М. | 22.05.2014 |   |
| Есептеу техникасын қолдану | Аманбаев А.А.    | 23.05.2014 |   |



ДИПЛОМ ЖОБАСЫН ДАЙЫНДАУ

КЕСТЕСІ

| № р/с | Тарау аттары, әзірленетін сұрақтардың тізімі                  | Жетекшіге ұсыну мерзімдері | Ескерту |
|-------|---|----------------------------|---------|
| 1     | Мобайда зерттеушінің тәуелсіз орнатқыш анықтаушысы            | 06.09.2013                 |         |
| 2     | Мобайдың дерекқорларын, тәуелсізін, тәуелсіздіктерін анықтау. | 25.10.2013                 |         |
| 3     | Деректер моделі   | 11.12.2013                 |         |
| 4     | Дерекқордағы мәліметтің маңызы                                | 12.12.2013                 |         |
| 5     | ДҚ құрудың бастапқы кезеңі                                    | 10.12.2013                 |         |
| 6     | ER моделін құру   | 13.01.2014                 |         |
| 7     | Бизнес ережелер   | 13.01.2014                 |         |
| 8     | Дерек құрыма құрылымын құру                                   | 05.02.2014                 |         |
| 9     | Кезең бойынша тәуелсіздіктерді анықтау.                       | 13.03.2014                 |         |
| 10    | Акциденттің тәуелсіздіктерін анықтау және ережелер            | 20.05.2014                 |         |
| 11    | Мобай тәуелсіздіктерін қорытындылау.                          | 23.05.2014                 |         |

Тапсырманың берілген уақыты « 25 » қазан 2014 ж.

Кафедра меңгерушісі

(колы)

Ф.-М.ғ.д., проф. Құралбаев З.К.

Жоба жетекшісі

(колы)

доцент, Ф.-М.ғ.к. Аманбаев А.А

Орындалатын тапсырманы қабылдаған студент

(колы)

Ереханова Алуа Еркінқызы

Андатпа

Дипломдық жобада Шымкент қ, "Ernur Press" ЖШС баспа орталығы жұмысының деректер қорын құру қарастырылады. Жұмыстың басты мақсаты-баспа орталығы қызметкерлерінің уақытын үнемдейтін,жұмысын жеңілдетедін,аз уақыт ішінде бағдарламаны игеріп, оңай пайдалануға мүмкіндік бередін,интеллектуалды еңбек пен қатар өте көптеген қағаздармен жұмысты қысқартатын және экономикалық жағынан да өте тиімді жүйе жасау.Электронды құжаталмасу жүйесін жобалауда қолданылған бағдарламалық қамтамалар: SQL Server 2008, Microsoft Visual Studio 2012.Мәліметтер қоры MySQL жүйесінде жасалады.

Дипломдық жобаны жазу барысында ақпараттық жүйенің архитектурасын, ДҚ құрылымын жасау жұмыстары жүргізілген.

Өміртіршілік қауіпсіздігі бөлімінде жұмыс бөлмесіндегі жасанды жарықтану жүйесінің және вентиляция жүйесінің есептері келтірілген.

Экономикалық бөлімде енгізілген жүйеден алынған экономикалық пайданы есептеу жүргізілген.

### **Аннотация**

В данном дипломном проекте рассматривается разработка базы данных работы издательства ТОО “Ернур Пресс” г.Шымкент. Цель дипломной работы – разработка информационной системы для сотрудников издательство. Для разработки системы были использованы следующие программные обеспечения: SQL Server 2008, Microsoft Visual Studio 2012.

В процессе написания дипломного проекта были разработаны архитектуры информационной системы, структуры БД.

В части по безопасности жизнедеятельности представлены расчеты искусственного освещения и вентиляции .

В экономической части произведен расчет полученной экономической выгоды от внедряемой системы.

### **Annotation**

In this project the development of a database of Publishing LLP " Yernur Press " Shymkent . Purpose of the project - Development of information system for publishing staff . To develop systems were used the following software : SQL Server 2008 , Microsoft Visual Studio 2012 .

In the process of writing the project developed information system architecture , database structure.

As part of Health and Safety presents calculations of artificial lighting and ventilation .

In the economic part of a calculation of economic benefits derived from the system being implemented .

## Мазмұны

|  |             |
|--|-------------|
| Кіріспе.....   | 8           |
| 1 Техникалық тапсырма.....   | 9           |
| 1.1 Тағайындалуы және пайдалану аймағы.....                                | 9           |
| 1.1.1 Сенімділікке қойылатын талаптар .....                                | 9           |
| 1.1.2 Техникалық құрылғыларға қойылатын талаптар .....                     | 9           |
| 1.1.3 Бағдарламалық құжатқа қойылатын талаптар.....                        | 9           |
| 1.2 Құру кезеңдері мен сатылары.....                                       | 10          |
| 1.2.1 Кезең бойынша жұмыстардың мазмұны.....                               | 10          |
| 1.2.2 Тестілеу.....  | 10          |
| 1.3 Автоматтандырылған жұмыс орнының құрылымы.....                         | <b>Ошиб</b> |
| 1.3.1 АЖО құрылымы.....  | 11          |
| 1.3.2 АЖО-ға қойылатын жалпы талаптар.....                                 | 12          |
| 1.3.3 Дерекордағы шешілетін мәселелер.....                                 | 13          |
| 2 Ақпараттық жүйелер. Деректер қорын жобалау.....                          | <b>Ошиб</b> |
| 2.1 Ақпараттық жүйелер туралы қысқаша мағлұматтар.....                     | <b>Ошиб</b> |
| 2.1.1 Ақпараттық жүйе жобалау сатылары .....                               | 15          |
| 2.1.2 Ақпараттық жүйенің тіршілік циклі.....                               | 16          |
| 2.1.3 АЖ–ның басты сенімділік түсінігі.....                                | 16          |
| 2.2 Деректер қорын жобалау.....  | 20          |
| 2.2.1 Деректер моделі.....   | 341         |
| 2.2.2 Реляционды деректер қорына қойылатын талаптар.....                   | <b>Ошиб</b> |
| 2.2.3 SQL тілі туралы негізгі деректер.....                                | 342         |
| 3 Дерекқорды жобалау және оны іске асыру.....                              | 25          |
| 3.1 ДҚ құрудың бастапқы кезеңі.....  | 25          |
| 3.2 Концептуалды жобалау.....  | 25          |
| 3.2.1 Қажетті ақпарат: пайдаланушылардың талабы.....                       | 25          |
| 3.2.2 Дерекор және қосымшаны жобалау құралдарын таңдаудың негіздемесі..... | 26          |
| 3.3 ER моделін құру.....   | 27          |
| 3.3.1 Бизнес ережелер.....   | 28          |
| 3.3.2 Реляционды қатынастарды құрастыру.....                               | 31          |
| 3.3.3 Логикалық жобалау.....   | 32          |
| 3.4 Физикалық жобалау.....   | 33          |
| 3.4.1 Дерек қорына қосымша құру.....                                       | 33          |
| 4 Тіршілік қауіпсіздігі.....   | 43          |
| 5 Техника-экономикалық негізделуі.....                                     | 55          |
| 5.1 Экономика бөлімі бойынша қорытынды.....                                | 69          |
| Қорытынды.....   | 70          |
| Пайдаланылған әдебиеттер тізімі.....                                       | 71          |

|                  |    |
|------------------|----|
| А қосымшасы..... | 72 |
| Б қосымшасы..... | 74 |

## КІРІСПЕ

Бұл дипломдық жұмыста Шымкент қ,"Ernur Press" ЖШС баспа орталығы жұмысының деректер қоры құру жобаланды. Жұмыстың мақсаты - баспа орталығының жұмыс істеу жүйесін автоматтандыру. Ақпараттық жүйені құрудағы мақсатым қазіргі заман ақпараттық технологиялар ғасыры, сондықтан да жүйені автоматтандырудың маңызы зор. Бұл бағдарламалық шешім техникалық қызмет көрсетуге арналған, қолдау қызметіне түскен ақпараттарды орталықтандырылған жүйесін жасау, құжаттардың орындалуын бақылау, техникалық қолдау қызметінің жұмыс сапасын жоғарылату, қолданушылардан түскен мәліметтерді бақылауға және өңдеуіне уақытты үнемдеу және автоматтандыруға арналған.

Электронды құжатталмасу жүйесін жобалауда қолданылған бағдарламалық қамтамалар: жобалауда SQL Server 2008, Microsoft Visual Studio 2012. Мәліметтер қоры My SQL-жүйесінде жасалады.

Өміртіршілік қауіпсіздігі бөлімінде жұмыс бөлмесіндегі жасанды жарықтану жүйесінің және вентиляция жүйесінің есептері келтірілген.

Экономикалық бөлімде бағдарламалық қамтаманың жүзеге асырылуына қажетті өзіндік құн және жобаның 20 % рентабельділік деңгейіндегі бірлік бағасы есептелінді.

Компьютер базасында құрылған ақпараттық жүйелер үлкен көлемді ақпараттарды өңдеуге және ауқымды ақпараттарды өңдеуді қажет ететін салаларда басқаруды шешуге рұқсат етеді. Енді бұл өз кезегінде бағдарламалық қамтамасыз етуді құруды қажет етеді. Қазіргі таңда “Ernur Press” ЖШС баспа орталығына осындай бағдарламаның жетіспеушілігі сезінуде.

Ұйымдарда қалыптасқан құжаттармен жұмыс істеу мен іс жүргізу және басқарманы құжаттамамен қамсыздандыруда жүйені автоматтандыру тиімді. Қоғамның бүкіләлемдік ақпараттануында, жаңа ақпараттық және коммуникациялық технологияның кең таралуы, нарықтық механизмнің және қазіргі менеджменттің енгізілуі ақпарат рөлінің әлеуметтік-экономикалық үдерістерде ұлғаюына және оны маңызды стратегиялық ресурс ретінде қарастыруға әкеліп соқты. Соған байланысты барлық ұйымдарда құжаттармен жұмыс істеудің жаңа тәсілі пайда болды, яғни жүйені автоматтандыру. Енгізіліп жатқан жүйеміздің басты артықшылығы-баспа орталығы қызметкерлерінің уақытын үнемдейді, жұмысын жеңілдетеді. Аз уақыт ішінде бағдарламаны игеріп, оңай пайдалануға мүмкіндік береді, бұл интеллектуалды еңбек пен қатар өте көптеген қағаздармен жұмысты қысқартады және экономикалық жағынан да өте тиімді.

## **1 Техникалық тапсырмалар**

### **1.1 Тағайындалуы және пайдалану аймағы**

Баспа орталығының ақпараттық жүйесін құру. Программа керекті есептеулерді автоматты түрде жүргізеді, есептеу барысында кететін қателерді азайтады. Сонымен қатар, мынадай бөлімдерді қамтиды:

- клиенттер;
- жіберілім;
- қосымша өзгертулер;
- есеп беру;
- анықтама.

#### **1.1.1 Сенімділікке қойылатын талаптар**

Бағдарлама толық сенімді функционалды жұмыс жасауы үшін тапсырыс беруші ұйымдық – техникалық шаралардың жиынтығын қамтамасыз ету керек. Олар төмендегідей болады:

- техникалық құралдарға үзіліссіз қоректену көзін ұйымдастыру;
- лицензиялық бағдарламалық қамтуды пайдалану;
- ГОСТ 51188–98 талаптарының уақыттылы орындалуы. Ақпаратты қорғау;
- компьютерлік вирусқа бағдарламалық құралдарды тексеру.

#### **1.1.2 Техникалық құрылғыларға қойылатын талаптар**

Сервер қызметін атқаратын дербес компьютер болуы керек. Оның сипаттамалы төмендегідей:

- процессор Pentium–2.0Hz;
- жедел жады 1Гигабайт көлемінен кем емес;
- HDD, 40 Гигабайт;
- операционды жүйе Windows 2000 Server , Windows 7 және Windows 8;
- Microsoft Sql Server 2008.

#### **1.1.3 Бағдарламалық құжатқа қойылатын талаптар**

Бағдарламалық құжаттың құрамы келесідей болу керек:

- техникалық тапсырма;
- бағдарлама және байқаулардың әдістері;
- пайдаланушыға көмек.



## **1.2 Құру кезендері мен сатылары**

Құру сатылары.

Құру келесі үш сатыны қамту керек:

- техникалық тапсырманы құру;
- жобалау жұмысы;
- енгізу.

Құру кезендері .

Техникалық тапсырмаға сай құру жұмыстары кезеңмен орындалу керек.

Құру кезендері келесідей болады:

- ақпарат жүйесін құру;
- бағдарламалық құжатты құру;
- мысалмен бағдарламаның тексерілуі.

Бағдарламаны енгізу кезінде құру кезендері орындалып және бағдарламаны беру болып табылады.

### **1.2.1 Кезең бойынша жұмыстардың мазмұны**

Құру кезеңінде келесі жұмыстар орындалу керек:

- Тапсырманың мақсатын анықтау;
- Техникалық құрылғыларға қойылатын талаптарды анықтау;
- Бағдарламаға қойылатын талаптарды анықтау;

### **1.2.2 Тестілеу**

Тестілеу – бұл ақпараттық жүйеде қателерді іздеу болып табылады.

Қате – бұл бар нәрсенің бір–біріне сәйкес келмеуі.

Сенімділік – бұл әрбір қабылдалмаудың құнын есепке ала отырып есептелінген, белгілі бір уақыт ішіндегі қарсылық жасалмаған жұмыс ықтималдығы.

Тестілеу принциптері.

Тестілеу жақсы жобаланған жүйеде қалған қателерді табу үшін, сонымен қатар жүйенің сенімділігін арттыру үшін және құндылығын арттыру үшін жүргізіледі. Тестілеу арқылы нашар жобаланған жүйенің жоғарғы дәрежедегі сенімділігіне жетуге болмайды.

Егер біз бағдарламаны тестілейтін болсақ, онда қандай да бір жағдайлармен бағдарламаның бағасын көтеру арқылы тестілеуге кеткен шығынды қайтарып алуымыз қажет. Мұны тек бағдарламаның сенімділігін жоғарлату арқылы ғана жасауға болады. Сенімділікті тек үрдіске енгізілген қателерді түзету арқылы жоғарлатуға болады.

Сәтті тест деп, қатені тапқан тестті айтамыз. Егер бірде бір қате табылмаса, онда ол сәтсіз тест деген сөз.

- қателер топтасу қасиетіне ие;
- тестілеуді жеңілдету үшін еш уақытта бағдарламаны өзгертпеңіз;
- бағдарламаны тестілеудің авторымен кездесуден қашу керек. Егер программист бағдарламаны жазуда қате жіберсе, онда оны тестілеуде де дәл сол қатені жіберуі ықтимал;
- тестіні құру – бұл шығармашылық үрдіс, яғни кей жағдайда мидың жұмыс істеуін қажет ететін үрдіс;
- қатені жақсы табатын тест жақсы тест болып табылады;
- бағдарлама неге арналған, яғни тек өзіне ғана тиесілі істі істейді ме, әлде басқа жұмыстарды да істейді ме соны тексеру керек;
- комментариилердің жеткіліксіздігі қателерді іздеуді қиындатады;
- комментариилердің аздығы қатені іздеуді қиындатумен қатар, түсініктеме сөздігінсіз күрделі бағдарламаны талқылау да қиын.

Тестілеуден кейін қатенің жоқ екендігіне кепілдік беруге болмайды, тек белгілі бір деңгейде жүйе жұмысының дұрыстығының сенімділігі жайлы айтуға болады.

Тест – бұл кіріс деректерінің жиынтығы және (немесе нәтижені күтуші қолданушының қызметі) жауап бағдарламалардың іс–әрекеті. Қарапайым бағдарламамен толық, жан–жақты тестілеу жүргізу мүмкін емес, себебі оған уақытта, қажетті ресурстарда жетпейді. Сондықтан тестілеудің қатені табу ықтималдылығына байланысты, тестті құру әдісіне байланысты бірнеше түрі бар.

Тестілеу түрлері:

- құрылымдық тестілеу (ақ жәшік);
- функционалдық (қызметтік) тестілеу (қара жәшік).

Құрылымдық тестілеу.

Берілген тестілеуде бағдарламаның тексті көрініп тұрады. Шешімдердің блоктары, циклдері тестіленеді.

Құрылымдық тестілеудің бірнеше типі бар:

- операторларды жабу;
- шешімдерді жабу;
- шарттарды жабу;
- шарттардың аралас жабулары;
- циклдердің тестіленуі.

Толық түсініктер үшін құрылымдық тестілеудің мысалдарын қарастыруға болады.

Функционалдық (қызметтік) тестілеу.

Берілген тестілеуде бағдарламаның тексті көрінбейді және бағдарлама қара жәшік түрінде, яғни кіріс және шығыс шарттары белгілі, сонымен қоса жұмыстың жалпы схемасы қарастырылады. Бағдарлама оның спецификациясы бойынша тексеріледі.

Функционалдық тестілеудің бірнеше түрі бар:

- эквивалентті кластар;

- шекті мәндерінің анализі;
- белгілі бір мөлшердегі тестілеу;
- белгілі бір көлемдегі тестілеу;
- қорғанысты тестілеу;
- жүйе эксплуатациясы;
- тәжірибелі эксплуатация.

Үлкен түсініктер үшін функционалдық тестілеудің мысалдарын қарастыруға болады [4].

### **1.3 Автоматтандырылған жұмыс орнының құрылымы**

Автоматтандырылған жұмыс орны немесе шетел терминологияда «жұмыс станса» (work-station) бұл анықталған функцияларды автоматтандыру үшін қажетті құралдармен жабдықталған белгілі бір мамандықтағы қолданушы маманның орны. Мұндай құралдар, керек жағдайда басқа қосалқы электронды құрылғылармен, атап айтқанда, дискілік жинақтауыштармен, басу құрылғылармен, оптикалық оқитын құрылғылармен немесе штрих кодасын оқитын, графика құрылғыларымен, басқа автоматтандырылған жұмыс орныны мен және басқа жергілікті есептеу желілерімен т.б. түйіндесетін құрылғылармен толықтырылатын әдетте дербес компьютер болып табылады.

Таралған ИС концепциясына сәйкес олардың ішінде ақпарат көлемінің төменнен жоғары қарай берілуі ИС дың ұйымдық иерархияда жоғарыдағы деңгейлердегі оның қаншалықты қажеттілігіне байланысты болады. Бұл жерде ақпаратты өңдеу нәтижелерінің басым бөлігі және бастапқы деректер деңгейлердің жергілікті дерекқорларда (ДБ) сақталған жөн. Осы таралған басқару идеясын жүзеге асыруда әрбір ИС деңгейі үшін мамандардың автоматтандырылған жұмыс орындарын құруды талап етті [2].

Автоматтандырылған жұмыс орнының негізгі арналуы деп жұмыс орындарындағы ақпараттың орталықтандырылмаған өңдеуді, «өзінің» сәйкес келетін дерекқорларын қолданумен қатар жергілікті АЖО-мен ДК желілеріне, ал кейде қуатты ЭЕМ-лар бар ауқымды есептеу тораптарына ену мүмкіндігімен қолдануды атауға болады.

Автоматтандырылған жұмыс орнында шешілетін мәселелерді шартты түрде ақпараттық және есептеуіш деп бөлуге болады.

Ақпараттық мәселелерге ақпаратты кодалау, классификациялау (жіктеу), жинау, құрылымдық ұйымдастыру, түзету, сақтау жатады. Ақпараттық мәселелер жиі түрде арифметикалық пен мәтіндік сипаттағы және байланыстағы қарапайым есептеу мен логикалық процедуралардан тұрады. Ақпараттық мәселелер әдетте күрделі болып табылады және мамандардың жұмыс уақытының көп бөлігін алады [4].

Есептеу мәселелер формальдандырылатын және жартылай формальдандырылатын болады. Формальдандырылатын мәселелер формальды алгоритм негізінде шешіледі және екі топқа бөлінеді: тура есептеу мәселелер

мен математикалық модельдер негізіндегі мәселелер. Тура есептеу мәселелер қарапайым алгоритмдер көмегімен шешіледі. Одан күрделі мәселелерді шешу үшін әртүрлі математикалық модельдерді қолдану талап етіледі. Соңғы кезде жартылай формальданатын мәселелерді шешуге арналған сематикалық деп аталатын амалдарды жетілдіруге аса назар аударылады. Мұндай мәселелер жиі түрде экономикалық объектілерді оперативті басқару барысында, әсіресе толық емес ақпарат жағдайында шешімді қабылдау кезінде туады.

### **1.3.1 Автоматтандырылған жұмыс орнының құрылымы**

Автоматтандырылған жұмыс орны бұл ұйымдастыру, функционалды және қамтамасыз ету бөліктерінен тұратын кешенді жүйе.

Ұйымдастыру бөлігі автоматтандырылған жұмыс орнының қолданушылардың функцияларын бөліп таратуға арналған ұйымдастыру құрлымынан, сонымен қатар мамандарды даярлау, дамыту мен әкімшілдеу тәсілдерінің жиынтығынан тұрады. Әкімшілдеуге жұмысты жоспарлау, тіркеу, бақылау, сараптау, реттеу, автоматтандырылған жұмыс орнының қолданушыларының құқықтарын мен міндеттерін құжатты түрде рәсімдеуді жатқызады.

Автоматтандырылған жұмыс орнының функционалды бөлігі бұл оперативті және перспективалық жоспарлау мәселелердің шешімін, техника-экономикалық көрсеткіштердің тіркеуін мен сараптауын қамтамасыз ететін экономика-математикалық тәсілдер кешені. Кешен құрылымы автоматтандырылған жұмыс орнының анықталған функцияларын жүзеге асыратын жүйенің біршама дербес ішжүйеден тұратыны болжамдалады.

Ішжүйелердің ерекшеленуі жұмыс орынның функционалды специализациясы мен ұйымдық құрылымдық есеппен функционалды ұйымдық белгісі бойынша жүргізіледі.

Автоматтандырылған жұмыс орнын қамтамасыз ету бөлігі:

- функционалды ішжүйелердің ақпараттық қамтамасыз етуі;
- техникалық құралдар кешені;
- ішжүйелер функцияларын жүзеге асырудың программалық құралдары.

Автоматтандырылған жұмыс орны бұл құжаттардың, ақпарат массивтерінің (ДҚ) және кодалаудың стандартталған және бәрыңғай формалар жиынтығы.

Автоматтандырылған жұмыс орнында қолданылатын ақпарат тиянақтылық, оның арналуы және қозғалу бағыты бойынша жіктеледі. Тұрақтылық бойынша ақпарат шартты тұрақты және айнымалы тұрақты, арналуы бойынша нормативті-анықтамалық, жоспарлау, есептемелік, сараптамалық және оперативті-өндірістік (директивті-оперативті және оперативті) деп бөлінеді. Қозғалу бағыты бойынша ақпарат кіріс, шығыс және аралық деп бөлінеді.

Автоматтандырылған жұмыс орнының техникалық құралдар кешеніне ақпаратты жинау, жеткізу, өңдеу және сақтау процестердің жүзеге асырылуын қамтамасыз ететін құрылғылар жиынтығы кіреді.

Біртекті операцияларды орындайтын құрылғылар техникалық құралдардың функционалды топтарын құрайды. Ішжүйелердің функцияларды жүзеге асырудың программалық құралдарына біріншіден адам-машина АЖО жүйесінде қатынасудың программалық тір құралдарына арнап, «жұмыс ортасын» қалыптастыратын операциялық жүйелерді жатқызуға болады. АЖО программмалық тіл құралдары келесімен қамтамасыз етеді:

- автоматтандырылған жұмыс орнының функционалды мәселелерін шешу;
- автоматтандырылған жұмыс орнының қолданушылар жағынан мәселелердің шешімін ыңғайлы және түсінікті басқару – мәселелер интерфейстері [8].

### **1.3.2 Автоматтандырылған жұмыс орнына қойылатын жалпы талаптар**

Автоматтандырылған жұмыс орнына қойылатын жалпы талаптар келесідей:

- жүйелілік;
- икемділік;
- тұрақтылық;
- эффективтілік (тиімділік).

**Жүйелілік.** Автоматтандырылған жұмыс орнының жұмыс орынның функционалды арналуымен анықталатын құрылымы бар жүйе деп қарастырғаны жөн.

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

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

**Эффективтілік.** Автоматтандырылған жұмыс орнының жүйенің құрылуы мен қолдануына кеткен шығындарына апаратын жоғарыда келтірілген қағидалардың жүзеге асырылу деңгейінің интегралды көрсеткіші де қарастырғаны жөн.

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

Автоматтандырылған жұмыс орнын ендіру компьютер ұйытқы болатын ақпаратты өңдеудің машиналық құралдар мен адам арасындағы функциялар мен жүктеменің дұрыс бөліп тарату жағдайында қалаған эффектті беруі мүмкін.



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

Автоматтандырылған жұмыс орны жеке (дербес) , топтық, коллективті болуы мүмкін. Топтық пен коллективті автоматтандырылған жұмыс орнына қатысты олардың ДК-дің жүйе ретінде эффективті функционалдау мақсатында мамандарға (коллективке) әкімшілдеудің нақты функцияларын анықтау қажет. Автоматтандырылған жұмыс орны адам машиналық жүйе ретінде ашық, икемді, үнемі даму мен жетілдірілуіне бейімдіболу қажет.

Мұндай жүйеде келесі қамтамасыз етілу керек:

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

### **1.3.3 Автоматтандырылған жұмыс орнын жобалау сатылары**

Автоматтандырылған жұмыс орнын дұрыс ұйымдастыру, құрастыру және енгізу мақсатында оның жобалау процесін келесі сатыларға жіктейді:

- бар басқару жүйесін зерттеу мен сараптама жасау;
- автоматтандырылған жұмыс орнын құруының техникалық тапсырманы құрстыру;
- автоматтандырылған жұмыс орнын техникалық жобалау;
- автоматтандырылған жұмыс орнын жұмыстық жобалау.

### **1.3.4 Баспа орталығы жұмысының жалпы сипаттамасы**

Менеджер және админ–баспа орталығы жұмысының негізін қалайды. Администратор менеджерге өзінің жеке беті арқылы кіруді қояды. Ал менеджер өзінің бетіне кіріп ақпараттарды көре алады.

### **1.3.5 Дерекордағы шешілетін мәселелер**

Баспа орталығының жұмысын бақылайтын және бағалайтын орталықтандырылған деректер қорларын тиімді басқаратын жүйе құру, Тапсырыс берілген газеттердің санын есептеп шығару, Шығарылым күнін, сатылым орнын қарастыру, клиенттерді жазылым газеттері бойынша сараптау мүмкіндігін қарастыру.

### **1.3.6 Дерекқордың функционалдығын сипаттау**

Жобаланатын дерекқор және бағдарламалық интерфейс баспа орталықтың қызметкерлеріне арналған. Бағдарламалық қосымшада келесі функциялар іске асырылған:

- ақпараттық функция, клиент, газеттер туралы толық ақпаратты береді;
- енгізу функциясы, дерекқорға жаңа ақпарат енгізуге мүмкіндік береді.

## **2 Ақпараттық жүйелер. деректер қорын жобалау**

### **2.1 Ақпараттық жүйелер туралы қысқаша мағлұматтар**

Ақпараттық жүйе – қандай да бір адам қызметіне арналған ақпараттық моделі болып табылады. Ол жүйе төмендегідей ақпараттық үдерістерді қамтуы керек:

- сақтау;
- ақпарат ауысу;
- ақпаратты түрлендіру.

Қазіргі заманда күннен күнге ақпараттық жүйелер, қызметтердің әр саласына маңыздылығы артуда.

Ақпарат жүйесінің көмегімен шешілетін тапсырмалардың әр түрлі болып келуіне арқасында құрылу жағынан бір–бірінен айырмашылығы бар ақпараттық жүйелердің әр түрлі болып бөлуіне әкеп соқты.

Ақпараттық жүйелерді көптеген белгілері бойынша бөлуге болады. Негізгісі белгісі, функционалдық мүмкіндігі мен құрылу ерекшеліктерімен бөлу болып табылады. Орындайтын тапсырмаға, қолданатын техникалық құралдарға байланысты ақпараттық жүйелер әр түрлі сыныптарға бөлінеді [7].

Деректерді сақтау бойынша ақпараттық жүйелер фактографикалық және құжаттық болып бөлінеді. Фактографикалық жүйелер сан және мәтін түріндегі деректерді сақтауға және өңдеуге арналған. Мұндай деректермен көптеген операциялар орындауға болады. Құжаттық жүйелерде дерек құжат түрінде болып келеді. Алынған құжаттар пайдаланушыға беріледі, ал оны өңдеу құжаттық жүйелерде қарастырылмайды.

Автоматтандырылған ақпараттық жүйелерге негізделе отырып, ақпараттық жүйелер қолмен басқарылатын, автоматтандырылған және автоматтандырылмаған болып бөлінеді.

#### **2.1.1 Ақпараттық жүйе жобалау сатылары**

Тап осы лекцияда қарастырылған типтік тіршілік цикл бірінші кезеңдері ақпараттық жүйе жасау, жүйеге техникалық тапсырма берумен және өңдеумен аяқталады.

Күрделі жүйені жобалау процесі өз алдында шығармашылық жұмыс, ал бұнда қолданылатын технология – ұйымдасқан шығармашылық технология – пәндік облыстағы фундаментальдік зерттеу сатысы мен дайындау және қолдану жүйесі арасындағы процестердің жиынтығы.

Бұндай технологияның бірінші сатысы ақпараттық жүйе құру концепциясымен аяқталатын пәндік облысты зерттеу болып табылады.

Көптеген шынайы жүйе иерерхиялық құрылымға ие. Бұл жүйенің ерекшелігі:

– арнайыланған жеке жұмыстарды белгілеу және оны арнайы дайындалған элементтермен бөлек беру;

– координация – жүйенің негізгі мақсатына жетуге бағытталған арнайы элементтің жиынтығын қамтамасыз ету.

Иерархиялық жүйені қолдана отырып жеке шешім қабылдай алатын, иерархиялық элементтің өзара әсерінің жүйесіндегі ұйымды анықтауға болады. Ұйымның өз бағытында қолданып отырған қиындықтарды шешу ережелерінің жиынтығы стратегия деп аталады.

Барлық ақпараттық жүйе қабылдау сатысында жүйелік жақындау қолданылады. Бұнда жүйелік модельдің көмегімен іске асатын әр түрлі саладағы, типтегі, кластардағы ұйымдардың және пәндік облыстардың зерттелуі мен өңдеуі қарастырылады. Мамандар жүйелік жақындау жоқ деп қабылдау бұл осы жүйеге енудің жақсы жолы деп айтады. Жүйелік жақындау уақытпен тексерілген және қарапайым зерттеу іздеу әдісіне қарама – қарсы келеді.

Бұндай жүйе кәдімгі коммерциялық әдіспен алынбайды. Бәсекелестік бұл жағдайда тек ұсынысты зерттеу және алдын ала өңдеумен ғана шектеледі. Шешуші техникалық және жұмысты жобалау тек бір ғана ұйыммен келісім жасалған соң орындалады. Тапсырыс беруші тапсырманы бекіткеннен кейін нарыққа ұсынысын көрсете алмайды. Сондықтан жүйелік талдау сатысы АЖ жоғарғы мәнге ие. Мысалы, АҚШ – тың қорғаныс министрлігі 1965 жылы №32009 директивасын шығарды, онда құрылым концепция сатысынсыз күрделі жүйені өңдеу туралы келісім қабылданбайды делінген.

Концепция түсінігі – ол анық түсінім әдісі, белгілі бір заттың, құбылыстың, процестің жеткізілуі. Кейбір сатыда бір – бірімен байланысқан білім, факт, тапсырма мен қызығушылықтар жиналып қалады.

Бұндай жағдай адам санасы тез дамып келе жатқан облыстарда көп кездеседі; тез дамып келе жатқан облыстарда бұл үрдіс дағдарысқа алып келеді.

Жүйенің ерекшелігі барлық компоненттердің байланысын анықтайтын модельдің болуы. Әсіресе жеке компоненттердің қасиеттерін жүйенің қасиетімен байланыстыратын модель.

Бұл жүйенің шешімі теңдік жүйесін құру арқылы жүзеге асады. Онда түрлі компоненттердің қасиеттерінің бір – біріне және жүйе қасиетіне әсерін көрсетеді. Әр компонент жеке өз қасиетін жақсарта алмайды, себебі ол өзгеріс басқа компоненттерге ісерін тигізеді. Ол ішкі жүйеге кері әсерін тигізеді, тіпті жалпы жүйенің бұзылуына алып соғады.

Осылайша, тапсырма тек теңдік жүйесін құру арқылы ғана шешуге болады. Кейбір компоненттер жобаланады және осы жүйеге арналып жасалады. Барлық осы арнайы компоненттерді қолдану және осылайша түскен табысты компонентті құрғандарға бөліп беруді қамтамасыз ететін нарық болу үшін жүйе жеткілікті үлкен болуы керек.

Тапсырылымның бітетін ақпараттық жүйе құру сатысы рационалды жоспарлау және ұйымды жұмыстардың үйлесімімен ерекшеленеді. Ақпараттық жүйенің тіршілік циклына мына модельдер:

- каскадты;
- аралық бақылаумен жүретін тарауланған модель;
- спиральды модель [7].

### **2.1.2 Ақпараттық жүйенің тіршілік циклі**

Ақпараттық жүйенің тіршілік циклін жүені құру үрдісі және оны пайдалану кезіндегі болатын оқиғалар түрінде көрсетуге болады.

Тіршілік циклі жүйенің әр түрлі күйін көрсетеді. Оның бастапқы қажеттілігенен бастап толық пайдалануынан тыс қалғанша қарастырады.

Қаіргі кезде жоғарыда айтылғандай тіршілік моделінің келесі түрлері қолданылады:

- Каскадты модель жобаның бекітілген ретімен барлық кезеңдерінің орындалуын қамтамасыз етеді. Келесі кезеңге өту алдындағы кезеңдегі жұмыстын толықтай аяқталуын білдіреді.

- Аралық бақылаумен жүретін тарауланған модель ақпараттық жүйенің құрылуы, кезеңдер аралығымен кері байланыс циклының итерациясымен басқарылады. Аралық кезең түзетулері.

- Спиральды модель әр спиральдың орамында өнімнің келесі нұсқасы құрылады, жобаға талаптар қарастырылады және оның сапасы анықталады.

### **2.1.3 АЖ–ның басты сенімділік түсінігі**

АЖ–ның кез келген шығу эталоны тапсырыс берушінің немесе потенциалды қолданушының техникалық тапсырмасы болып табылады (немесе спецификациялық талап ету).

Мұндай құжаттарды қолданушының белгілі шартын және шығу берілгенін алу кезінде, мазмұнын және мәнінің нәтижесінің құрамын құру кезінде алуға болады. Кез келген АЖ–ның шарттарын көрсеткен функционалды нәтижені қайтаруға және соған керек формалды эталонды қателік сияқты квалификация жасауға болады.

Практика жүзінде тапсырыс беруші мен өндірушінің өндіріс процесінде ПЖ–нің шығу талаптары бойынша нақтыланады және детальданады. Базаның мұндай нақтылануы формальданусыз көрсеткіші және мамандар білімі болып табылады, сонымен қатар уақытаралық жоба этабының тексеру нәтижесі болады. Нормативтік–техникалық құжаттарды орнату, бір немесе бірнеше АЖ–ның ережелерімен берілген мінездемелерін орнату қорытындысын АЖ–ның техникалық операциясы тексереді.

Оның қасиетінің бейнеленуі және мүмкіндігінің анықталуы және оны тура белгілеуде эффектті қолдану, кез келген өнім көрсеткіш жиынының сапасын көрсетеді. Өнім күрделі болған сайын, оның қасиетінің мінездемелік саны және оның сапасының адекватты бейнелену көрсеткішінің қажеттігі көп болады. АЖ сапасы және күрделі бағдарламалық құрал жиынтығының



көрсеткіші – критериілермен жазылады, олардың әрқайсысын метрикалық және әдіс өлшемін анықтау қажет. Бұл критериілер және метрикалық бағдарламаларының соңғы өнімнің жетістік тәсілінің қатысуынсыз қасиетін бейнелейді.

Анықтау белгілері, АЖ–ның басты функциялары, тапсырыс беруші немесе потенциалды қолданушының техникалық талап етілуі, атрибуттар жиыны түрінде анықталатын, оның функционалды қажеттілігі АЖ–ның сапасына байланысты болады.

Көптеген деңгейде функционалды қажеттілік АЖ сенімділігінен байқалады. Оған қоса, функционалды қажеттілікті көптеген әр түрлі критериілер бейнелейді. Оларды басты сенімділік көрсеткішіне байланысты бөлім критериі немесе фактор ретінде қарастыруға болады.

АЖ–ның жұмысқа қабілеттілігі немесе олардың ірі компоненттерінен айырылу кедергілерін көрсеткіш сенімділігінде тіркеуге болады. Техникалық құжаттарды талап етуге орнатылған, берілген функция параметрін орындауға қабілетті объект жағдайын сенімділік теориясының жұмысқа қабілеттілігі деп аталады.

Тек уақыт аралығында байқалатын жүйені ішкі қасиеті сенімділік болып табылады.

Сапа критериі динамикалық және мүлкті стохастикалық, АЖ–ның толық немесе ірі компоненттерінің функционалды мінездемелері бола алады. АЖ–ның интегралды сапа көрсеткіші тәжірибе жүзінде тура бағаланып және анықталып өлшенеді.

Жинап қою және тексеруде қолданылатын шығу берілгендерге АЖ–ның жұмысқа қабілеттілігіне толық сенім береді. Нағыз шығу берілгендерінің мәні берілген техникалық тапсырма және тексеруде қолданылатын бағдарламада басқаша болады. Осындайда шығу берілгендері функционалды бағдарламаларын алдын ала болжау қиын, өйткені олар әр түрлі аномалия, қарсы аяқталулар шақыруы мүмкін.

Нормативтік–техникалық және (немесе) конструкторлық құжаттарды жүйеде орнату, бір немесе бірнеше функцияларды бұзу ең болмаса сапа талабының біреуінде орындалу жағдайын – жауапсыз дейді. Нормативтік–техникалық және (немесе) конструкторлық құжаттама және берілген функцияларды орындау жауапсыздығын анықтауды рұқсат етуге орнатылған, белгі немесе жиын белгілері жауапсыз критериілердің бағалануын жағдайға әкеледі.

Бұзудың «жауапсыздықтан» айрықшалығы уақытша көрсеткіштің ұзақтығын қайта құрудан кейін АЖ–ның функционалды бұзуына қатысты болады, бұздан кейін жауапсыздан кейінгіге қарағанда қайта құру уақыты аз болады.

Бұзулар кезінде және қайта құрылу қасиетіне тура қатысты АЖ–ның тұрақтылық мінездемелерінің функционалды қарсы жауапсыздыққа қабілетті болады. Қайта құрылу – АЖ–ның функционалды толық қайта құрудан кейін қайта жіберумен (рестарт) мінездемеледі. Ресурстар және уақыт

қажеттілігі АЖ–ның функционалды нормаларын жаңартуды қамтамасыз етуді қайта жіберуге болады, сондықтан функционалды қайта құрудың толықтығы және ұзақтығы АЖ–ның бейнелеу сапасынан кейінгі бұзылуды және белгілеуді оның тура мүмкіндікте қолдануға болады.

Басқа өндіріс өніміне қабылданған АЖ–ның сенімді көрсеткіші адекватты аналогиялық мінездеменің мәнділік деңгейі болады. Критерииді кең қолдануға қарсы өндірудің ұзақтығы – жұмысқа қабілеттілік жүйе жағдайының уақытаралық жауапсызы немесе оның жақсы функционалды жүйеден кейінгі бастамасы болады. Критерии сенімінің әр түрлілігі мүмкіндік мінездемесінің бірнеше форма көлемінде қолданылады.

Мүмкіндікті қайта құрылу жүйесінен алу жұмысқа қабілеттілік жағдайының соңғы уақытында бейнелейтін, мінездеме жауапсызын қайта құру критерии коэффициентін дайындауға өндіріледі. Жауапсыз және қайта құру құрамы коэффициент мәнінің дайындығы үлкен интервалға жететін жұмыс жүйесінің пайдалысы уақыт уысына байланысты болады.

Интегралды бағдарламаның сапа көрсеткішінің жақсы өлшемін, қатар дәлдігін алуға күрделі АЖ–ның сапасын бағалауды рұқсат етуге сенімділік теориясының басты түсінігі қолданылады. Келтірілген критериилер басты ретте АЖ–ны тексеру және комплексті жинап қоюдың аяқталу фазасында қолданылады.

Сұраныста АЖ қолданушыларының шығу ақпаратында сенімді көрсетілімі барлық АЖ типінің интегралдық көрсеткішінің маңыздыларының бірі болып табылады, сонымен қатар бағарламалық–техникалық комплекстердің әр түрлі белгілері болады ( $P_{\text{нап}}$  – мүмкіндік сенім көрсетілімі).

Қазіргі заманғы, корпоративті АЖ өзінің комплекстілігін көрсетеді:

- бағарламалық–техникалық құрылғылар;
- ақпараттық қамтамасыз ету (берілгендер базасы);
- телекоммуникация құрылғылары;
- оның функционалды қолданылуы үшін өңдеу сұраныстарын және шығу ақпаратының көрсетілімін қамтамасыз етеді.

Автоматтандырылған сұраныс өңдеуімен және эксплуатациялық шартты қолдануды бақылау кезінде алгоритмдік шындалған шығу ақпаратының келісімін көрсетумен және АЖ–ны техникалық қызмет етуімен, АЖ құрылымын қабылдауды қамтамасыз етумен АЖ қолданушыларының шығу ақпаратының сұраныс көрсетілімінің сенімділігімен түсіндіріледі ( $P_{\text{нап}}$  – мүмкіндік сенім көрсетілімі).

АЖ қолданушысының шығу ақпаратының курстық жұмыста сұраныс көрсетілімінің сенімділігін бағалау келесі модельде қолданылады.

АЖ–ның кез келген уақытында (қолданушының көзқарасымен карағанда) қатар екі жағдайдың біреуінде болады:

- жұмысқа қабілеттілік;
- жұмысқа қабілетсіздік.

АЖ–ның математикалық уақытты күтудің жұмысқа қабілеттілік жағдайы жауапсыздың орташа өңделіміне тең. АЖ–ның математикалық уақытты күтудің жұмысқа қабілетсіздік жағдайы жұмысқа қабілеттіліктің жауапсыздан кейінгі орташа қайта құру уақытына тең.

Кездейсоқ уақытта шығу ақпараты көрсетілім сұранысын орындайды (экран дисплейінде, файлды магнитті тасымалдаушыда немесе шығарғанда). 3 варианты мүмкін:

- АЖ–ның функционалды  $t$  уақытында түскен сұраныс, оның жұмысқа қабілеттілік жағдайы үстінен түседі және сұранысты өңдеу қажеттілігі және оған шығу ақпаратына байланысты берілуі АЖ–ның жағдайында барлық уақытында болады;

- АЖ–ның функционалды  $t$  уақытында түскен сұраныс, оның жұмысқа қабілеттілік жағдайы үстінен түседі, бірақ сұранысты өңдеуге қажетті және оған шығу ақпаратына қатысты берілуінде АЖ бұл жағдайда аз уақыт болады;

- АЖ–ның функционалды  $T$  уақытында түскен сұраныс, оның жұмысқа қабілетсіздік жағдайы үстінен түседі.

Бірінші жағдайда сенімді ақпараттық көрсетілім болады, ал екінші және үшінші жағдайларда сұратылған ақпарат көрсетілмейді.

Зерттеулер кезінде алынған бағалар:

- жауапсыздыққа орташа өңделу;
- орташа қайта құрылу уақыты;
- шығу ақпаратының орташа уақыт көрсетілімі (АЖ–ға сұраныс уақыты);

- АЖ қолданушыларының шығу ақпаратына сұрастырылатын сенімді көрсетілім есептеулерін өткізу үшін шығу берілгендері болады. Шығу ақпараты көрсетілуі мүмкін барлық сұрастырылған ақпараттар нақты анықталған формада (немесе мазмұнда) типтерге бөлінеді [5].

## **2.2 Деректер қорын жобалау**

### **2.2.1 Деректер моделі**

Деректер қорын басқару жүйесі (ДҚБЖ) қолдайтын деректердің моделі өзінің логикалық құрылымы болады. Оған келесідей модельдер жатады:

- иерархиялық;
- желілік;
- реляционды.

Сонымен қатар кейінгі уақытта тәжірие жүзінде бастапқы үш модельдерден гөрі белсенді деректер моделі қолданылып жүр. Олар:

- постреляционды;
- көп өлшемді;
- нысанды–бағытталған.

Осы модельдерге негізделе отырып, қазіргі кезде басқа да деректер моделі құрылуда. Олардың қатарына нысанды–реляционды, деактивті–нысанды–бағытталған, семантикалық, концептуалдық және бағытталған моделдерді жатады.

Кейбір ДҚБЖ бірнеше деректер моделін қамти алады. Мысалы, ИНТЕРБАЗА жүйесінде желілік деректер қорын басқарылады және пайдаланушы интерфейсінде SQL және QBE тілдері құрылған.

Реляционды ДҚБЖ нысанды–бағытталған жүйелердің кейбір қасиеттерін қамтиды. Ондай ДҚБЖ көбіне нысанды–реляционды деп атайды. Оның мысалы ретінде Oracle 8.x. алуға болды. Ал, оның алдындағы нұсқасы, яғни Oracle 7.x таза реляционды болып табылады. Microsoft SQL Server 2005, PostgreSQL, сол сияқты Sav Zigzag, IBM Cloudscape, FirstSQL бұлардың бұлардың бәрі нысанды–реляционды болып табылады.

Постреляционды модель.

Постреляционды модель реляционды модельдің кеңейтілген түрі болып есептеледі. Ол кестеде сақталған деректерді бөлуге рұқсат етеді және өріс көп мәндік бола алады, яғни бір мәндің ішінде бірнеше мәндер болу мүмкін. Осылардан құралған кестелер негізгі кестеге бағынады.

Артықшылығы, реляционды кестелерді жиынын бір постреляционды кестемен беруге болады. Сонымен қатар, тағы артықшылықтары оның ақпараттық жоғары дәрежеде көрсетілуі мен оны өңдеудің тиімділігінде.

Постреляционды модельдің мысалы ретінде ДҚБЖ uniVers айтсақ болады.

Көп өлшемді модель.

Көп өлшемді модель, реляционды модельмен бірге пайда болды. Бұрын ол реляционды модель сияқты кеңінен қолданылмаған, тек 90 жылдардың ортасынан бастап оған қызығушылық артуда. Көп өлшемді модель ақпаратты интерактивті өңдеуге арналған.

Оның артықшылығы, реляционды модельге қарағанда нысандардың өзара күрделі байланыстарын қарастырады.

Көп өлшемді модельдің мысалы ретінде мына жүйелерді қарастыруға болады: Essbase (Arbor Software), Media Multi–matrix (Speedware), Oracle Express Server (Oracle) және Cache (InterSystems).

Нысанды–бағытталған модель.

Нысанды–бағытталған модельде деректер өздерінен бөлек жазбаларын идентификациялана алады және де оларды өңдейтін функцияларды анықтай алады. Ал кемшілігі, сұраныстардың жай орындалуында.

90 жылдарды нысанда–бағытталған модельдер тәжірибе жүзінде іске асырылды. Қазіргі кезде олар кеңінен қолданыста. Атап айтатын болсақ: ДҚБЖ: POET (POET Software), Jasmine (Computer Associates), Versant (Versant Technologies), O2 (Ardent Software), ODB–Jupiter («Интелтек Плюс» ғылыми–өндіріс орталығы), Iris, Orion және Postgres.

## 2.2.2 Реляционды деректер қорына қойылатын талаптар

Біз жұмыста реляциялық деректер қорын негізге алатын болғандықтан оған қойылатын талап – деректердің қалыптастырылуы. Қалыптастыру кестелердегі деректердің қайшылықтары және үйлесімсіз болмауын қадағалайды. Қалыптастырудың алты формасы бар, бірақта тәжірибеде деректер алдыңғы үш қалыптың талаптарын қанағаттандыратын болса жеткілікті.

## 2.2.3 SQL тілі туралы негізгі деректер

Деректерге ену мүмкіндігінің реляциялық тәсілі жазбалар тобының амалдарына негізделген. Амалдарды орындауға SQL (Structured Query Language) құрылымдық сұраныс тілінің құралдары қолданылады. Delphi қосымшаларында BDE механизмін қодану барысында деректер жиыны ретінде SQL–сұранысты орындауды беретін Query немесе StoredProc компоненттері қолданылады. SQL құралдарын жергілікті және қашықтағы деректер қорымен амалдар орындауға қолдануға болады.

Бағдарламалаудың процедуралық тілінен айырмашылығы: есептеу процестерін басқару инструкциясы (циклдер, тармақталу, басқа операторға өту) және енгізу/шығару құралдары жоқ.

Microsoft Access, Visual FoxPro немесе Paradox сияқты ДҚБЖ, SQL–де сұраныстарды бағдарламалаумен байланысты әрекеттерді өздері орындайды, мысалы Query By Example (QBE) – үлгі бойынша сұраныс, пайдаланушыға сұранысты визуальды құру құралдарын білдіреді.

SQL бағдарламалау тілінің толық функционалдық мүмкіндіктерін қамтымағанмен, ол деректерге ену мүмкіндігіне арналған және оны бағдарламаны жасау құралына қосады. және Delphi жүйесіне ендірілген. Сонымен қатар, SQL командаларымен жұмыс істеу үшін сәйкес құралдар және компоненттер бар. Delphi–де мұндай компоненттерге Query, SQLQuery және ADOQuery деректер жиыны жатады.

SQL тілінің функциялары.

Статистикалық функциялар:

- AVG() – орташа мән;
- MAX() – ең үлкен мән;
- MIN() – ең кіші мән;
- SUM() – қосынды;
- COUNT() – мәндердің саны;
- COUNT(\*) – нөлдік емес мәндердің саны.

Жолдармен жұмыс істеуге арналған функциялар:

- UPPER(Str) – Str символдық жолды жоғарғы регистрге айыстыру;
- LOWER(Str) – Str символдық жолды төменгі регистрге айыстыру;



- TRIM(Str) – Str жолының бастапқы және соңғы бос орындарын өшіру;
- SUBSTR (Str FROM TO) – Str жолынан, өзі символдардан тұратын, n1 жолынан бастап, n2 жолымен аяқталатын ішкі жолды бөліп алу;
- CAST(<Expression> AS <Type>) – Expression өрнегін Type типіне келтіру.

Ай–күн (Дата) және уақытпен жұмыс істейтін функция:

- EXTRACT(<Элемент> FROM <Өрнек>) – дата және уақыт мәндерінен тұратын өрнектен, сәйкес көрсетілген элементтен мәндер алу; дата және уақыт элементі ретінде YEAR, MONTH, MINUTE және SECOND элементтерін көрсетуге болады.

Кестелермен келесі амалдарды орындауға болады:

- жаңа кесте құру;
- кестені өшіру;
- кесте өрістерінің құрамын өзгерту;
- индексті құру және өшіру.

Бұл әрекеттер SQL тілінің инструкцияларының көмегімен орындалады.

#### 2.2.4 Кесте құру және өшіру

Кестені құру үшін CREATE TABLE инструкциясы қолданылады:

```
CREATE TABLE <Кесте аты>
  (<Өріс аты> <дерек типі>
   ...
   <Өріс аты> <дерек типі>);
```

Файлдың кеңейтілуі бойынша кестенің форматы автоматты түрде, яғни .db – Paradox кестесі үшін, .dbf dBase кестесі үшін анықталады. Кесте файлы дерек қор псевдонимі көрсетілген, МҚ каталогында орналастыру керек.

SQL тілінің көмегімен кесте құру мысалы:

```
CREATE TABLE Student.dbf
  (Numer Integer
   Name CHAR(15)
   Datarogd DATE);
```

Paradox кестесі үшін PRIMARY KEY сипаттамасын және жақшаның ішіне оның өрістерін көрсетіп, негізгі немесе алғашқы кілтті анықтауға болады. Кілттік өріс өрістер тізімінде бірінші болып орналасуы керек. Негізгі кілтті құрумен жаңа кестеге құруға мысал:

```
CREATE TABLE Personal.db
  (Code AUTOINC
   Name CHAR(15)
   Post CHAR(15)
   Oklad Numeric(10,2)
   PRIMARY KEY(Code));
```

Кестені өшіру үшін келесі инструкция қолданылады:

DROP TABLE <Кесте аты>;

Мысалы DROP TABLE <student>; Дискіден student атты кестеге қатысы бар барлық файлдар өшіріледі.

Кесте өрістерінің құрамын өзгерту.

Кесте өрістерінің құрамын өзгерту өрістерді қосу немесе өшіруді білдіреді және оның құрылымын өзгертуге әкеледі, сонымен қатар кестені басқа қосымшалар қолданбауы керек. Кесте өрістерінің құрамын өзгерту ALTER TABLE инструкциясының көмегімен орындалады:

ALTER TABLE <кесте аты>

ADD <1-өріс аты> <1-өріс типі> ,

DROP <1-өріс аты> ,

...

ADD <n-өріс аты> <n-өріс типі > ,

DROP <n-өріс аты>;

ADD операндысы кестеге жаңа өріс қосады, өріс аты және типі CREATE TABLE инструкциясындағыдай беріледі, ал DROP операндысы көрсетілген атымен кестеден өрісті өшіреді. ADD және DROP операндылары бір-бірімен байланысты емес және еркін түрде орналасуы мүмкін [8].

### **3 Дерекқорды жобалау және оны іске асыру**

#### **3.1 ДҚ құрудың бастапқы кезеңі**

Баспа орталығының жүйесі - клиент-серверлік технологиясын қолдана отырып, пайдаланушыларға техникалық қызмет көрсетуге арналған информациялық жүйе болып табылады.

Бұл бағдарламалық шешім техникалық қызмет көрсетуге арналған ақпараттармен жұмыс жасауда, қолдау қызметіне түскен мәліметтерді орталықтандырылған жүйесін жүргізу, ақпараттардың орындалуын бақылау, техникалық қолдау қызметінің жұмыс сапасын жоғарылату, клиенттерден түскен сұраныстарды бақылауына және өңдеуіне уақытты үнемдеу және автоматтандыруға арналған.

Электронды құжаталмасу жүйесін жобалауда қолданылған бағдарламалық қамтамалар: SQL Server 2008, Microsoft Visual Studio 2012. Мәліметтер қоры My SQL-жүйесінде жасалады.

##### **3.1.1 Дерекқордың функционалдығын сипаттау**

Баспа орталығының жұмысын бақылайтын және бағалайтын орталықтандырылған деректер қорларын тиімді басқаратын жүйе құру, Тапсырыс берілген газеттердің санын есептеп шығару, Шығарылым күнін, сатылым орнын қарастыру, клиенттерді жазылым газеттері бойынша сараптау мүмкіндігін қарастыру.

#### **3.2 Концептуалды жобалау**

Жобаланатын дерекқор және бағдарламалық интерфейс баспа орталықтың қызметкерлеріне арналған. Бағдарламалық қосымшада келесі функциялар іске асырылған:

- ақпараттық функция, клиент, газеттер туралы толық ақпаратты береді;
- енгізу функциясы, дерекқорға жаңа ақпарат енгізуге мүмкіндік береді;

##### **3.2.1 Қажетті ақпарат: пайдаланушылардың талабы**

Жоба пайдаланушының барлық негізгі талаптарын қанағаттандыру керек. Бұл жүйенің негізгі пайдаланушылары баспа орталықтың қызметкерлері болады. Негізгі талаптар алынған ақпараттың ұсынылатын тиімділік деңгейінде негізделеді. Жүйеге қойылатын негізгі талаптардың тізімі:

– жүйе қолданыста қарапайым болу керек. интерфейс ыңғайлы болу керек;

- жүйе қауіпсіздікті қамтамасыз ету керек;
- жүйе дерекқордың тұтастығын қамтамасыз ету керек;
- жүйе толығымен интеграциялану керек, яғни мәліметтердің артылуы және артық жаңартулар болмау керек.

### **3.2.2 Дерекқор және қосымшаны жобалау құралдарын таңдаудың негіздемесі**

Дерекқорларды басқару жүйесі.

Барлық талаптарды қамтамасыз ететін информациондық жүйені жүзеге асыру үшін бірінші кезекте реляциондық дерекқорын және серверлі архитектура клиентін қолданудың анық шешіміне әкелетін орталық деректер сақтаудың әдісін таңдау қажет.

Қазіргі кезде қойылған тапсырмалар талаптарын ДҚБЖ мүмкіншілігіне қарай және тапсырыс берушінің таңдауына байланысты таңдауға болатын бірнеше ДҚБЖ көрсетілген [4]:

- орнату және қолданудың жеңілдігі;
- кең таралған;
- қымбат серверлік қуатты орнатуды қажет етпейді;
- өндіргіштігінің жақсы көрсеткіші;
- деректерді сақтау сенімдігі мен қауіпсіздігін қамтамасыз етудің жеке құралдары;
- коммерциялық емес қолдану үшін өнімді сызғышта арзан түрлерінің болуы.

Microsoft SQL Server төменөндіргіштік серверлік қуаттарға есептелген, сонымен қатар қолданушылық жұмыс станциясында қолданушыға өндіргіштіктің төмендеуінің білінуінсіз жұмыс істеуге мүмкіндігі бар.

SQL компьютерлік дерекқорда сақталатын деректерді өңдеу және оқу үшін арналған құрал болып табылады, және қолданушының дерекқормен өзара қатынас ұйымдастыру үшін қолданылатын программалау тілі болып табылады. Шындығында SQL тек бір белгілі типті – реляционды дерекқормен жұмыс істейді. SQL ДҚБЖ қолданушыға ұсынатын барлық функционалдық мүмкіншіліктерді жүзеге асыру үшін қолданылады, соның ішінде:

- деректерді ұйымдастыру. SQL қолданушыға деректердің көрсетілімі құрылымын өзгертуге, сонымен қатар дерекқордың элементтері арасында қатынас құруға мүмкіндік береді;
- деректерді оқу. SQL қолданушыға немесе қосымшаға дерекқордың ішіндегі деректерді оқуға және олармен қолдануға мүмкіндік береді;
- деректерді өңдеу. SQL қолданушыға немесе қосымшаға дерекқорды өзгертуге, яғни оған жаңа деректер енгізуге, оған қоса бұрыннан бар деректерді өшіруге немесе жаңартуға мүмкіндік береді;

– қатынас құруды басқару. SQL-дің көмегімен қолданушының деректерді оқу және өзгерту және рұқсат етілмеген қатынас құрудан қорғауға мүмкіндігін шектеуге болады;

– деректерді бірге қолдану. SQL параллельді түрде жұмыс істейтін қолданушыларға бір-біріне кедергі келтірмес үшін деректерді бірге қолдануын қадағалайды;

– деректердің тұтастығы. SQL дерекқордың тұтастығын қамтамасыз етіп, оны келісімсіз өзгерістерден немесе жүйенің бас тартуынан қираудан қорғауға рұқсат береді.

SQL – дерекқорды программалау тілі. Дерекқорға қатынас құру үшін бағдарлаушылар өздерінің бағдарламаларына SQL командаларын қояды. Бұл әдістеме қолданушылар жазған бағдарламаларда да, дерекқорлардың қызметтік бағдарламаларында да қолданылады (есеп беру генераторы және деректерді енгізу құралдарында). Осылайша, SQL адамдарға, бағдарламаларға және есептеуіш жүйелерге реляционды дерекқорлардағы ақпараттарға қатынас құруды қамтамасыз ететін қажетті және мықты құралға айналды.

### 3.3 ER моделін құру

SQL – дерекқорды программалау тілі. Дерекқорға қатынас құру үшін бағдарлаушылар өздерінің бағдарламаларына SQL командаларын қояды. Бұл әдістеме қолданушылар жазған бағдарламаларда да, дерекқорлардың қызметтік бағдарламаларында да қолданылады (есеп беру генераторы және деректерді енгізу құралдарында). Осылайша, SQL адамдарға, бағдарламаларға және есептеуіш жүйелерге реляционды дерекқорлардағы ақпараттарға қатынас құруды қамтамасыз ететін қажетті және мықты құралға айналды.

ДҚ құрудың бастапқы кезеңінде алғашқы нысандар жиынтығын анықтау қажет. Бұл ақырғы пайдаланушылар мен жобалаушы көзқарасы бойынша жүйе объектілері туралы ең маңызды ақпарат болып табылады. Баспа орталығының деректер қоры 3.1–кестеде көрсетілген нысандарды қолданады. Microsoft SQL Server төменөндіргіштік серверлік қуаттарға есептелген, сонымен қатар қолданушылық жұмыс станциясында қолданушыға өндіргіштіктің төмендеуінің білінуінсіз жұмыс істеуге мүмкіндігі бар.

К е с т е 3.1 - «Баспа орталығының кестесі» дерекқорының нысандары

| Нысан аты  | Сипаттамасы                     |
|------------|---------------------------------|
| Klient     | Клиент туралы мәліметтер        |
| Tochka     | Сатылым орны туралы мәліметтер  |
| Klient_tel | Клиент номері туралы мәліметтер |
| Patronymic | Аты-жөні                        |
| Name       | Клиент аты туралы мәліметтер    |
| Fam        | Тегі                            |
| Vypusk     | Шығарылым номері                |

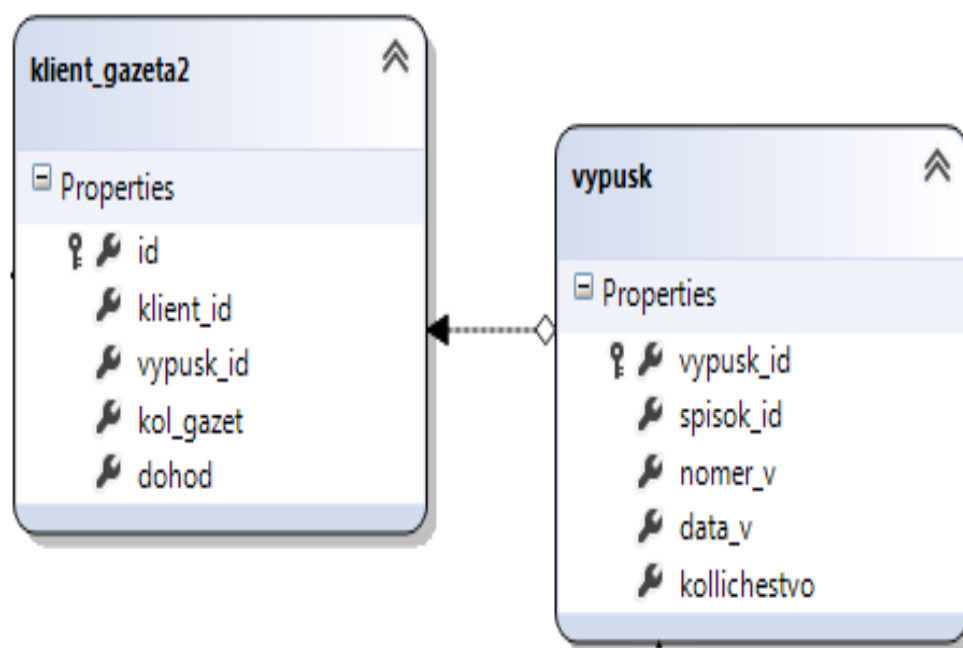
Кесте 3.1-дің жалғасы

|              |                          |
|--------------|--------------------------|
| Kol_gazet    | Газет немесе журнал саны |
| Dohod        | Пайда                    |
| Spisok       | Газеттердің тізімі       |
| Type         | Газет немесе журнал типі |
| Cena         | Соммасы                  |
| Name_tochka  | Сатылым орны аты         |
| Nomer_v      | Шығарылым нөмері         |
| Data_v       | Шығарылым күні           |
| Kollichestvo | Газеттердің саны         |

Жобалаушы және ақырғы пайдаланушы нысандарды анықтау кезінде келісімге келу керек. Жобалаушы нысандар арасындағы байланыстарды операциялардың сипаттамасына негізделіп анықтайды. Нысандар арасындағы байланыстар операциялардың толық сипаттамасы негізінде құрылған бизнес-ережелерге негізделеді. Бұл жерде біз құрып отырылған деректер қорына бизнес-ережелер құрамыз. Алгоритмін құрамыз. Бизнес-ереже 6 бөлімнен тұрады.

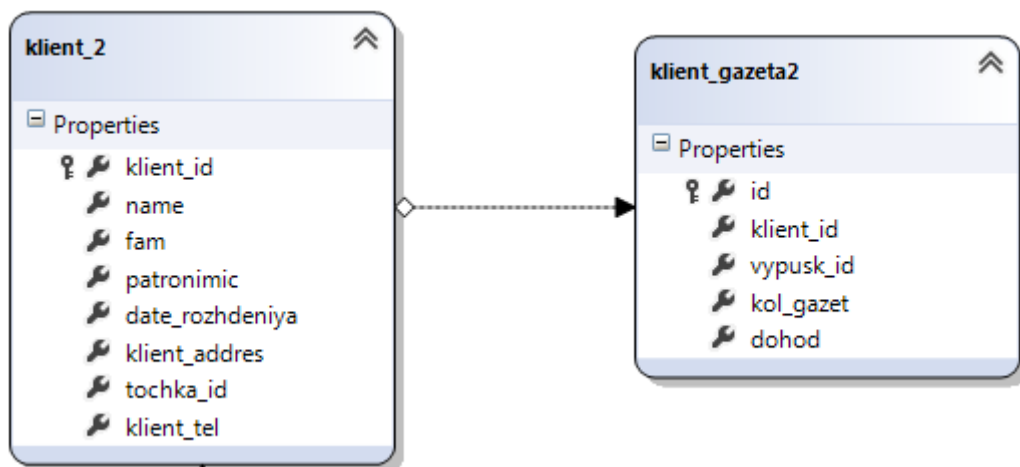
### 3.3.1 Бизнес ережелер

- 1) Клиенттің тапсырыс кестесі болады, кестені 3.1–суреттен көруге болады.



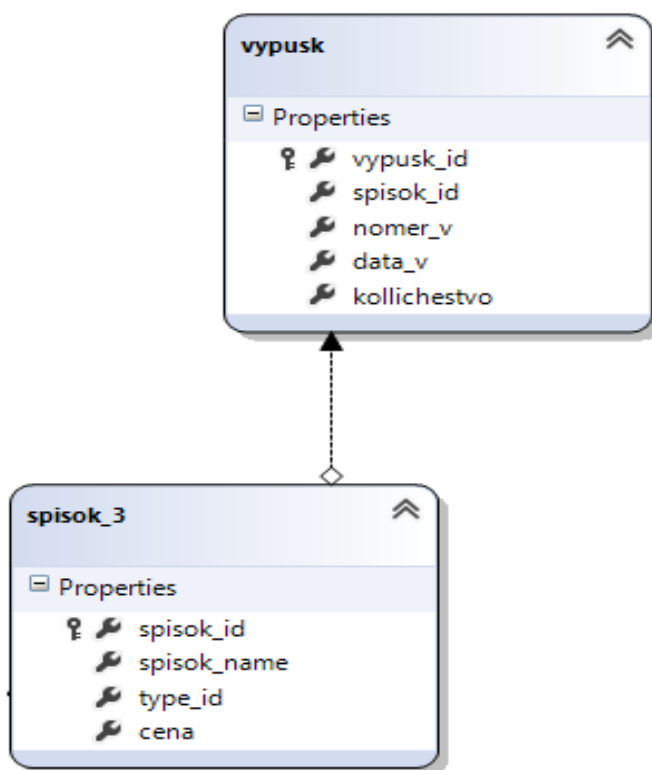
Сурет 3.1 –Клиенттің тапсырыс кестесі бизнес ереже үшін ER сегменті

2) Клиент газеттің барлық түрін сұранысқа бере алады.



Сурет 3.2 – Клиент газеттің барлық түрін сұранысқа беру бизнес ереже үшін ER сегменті

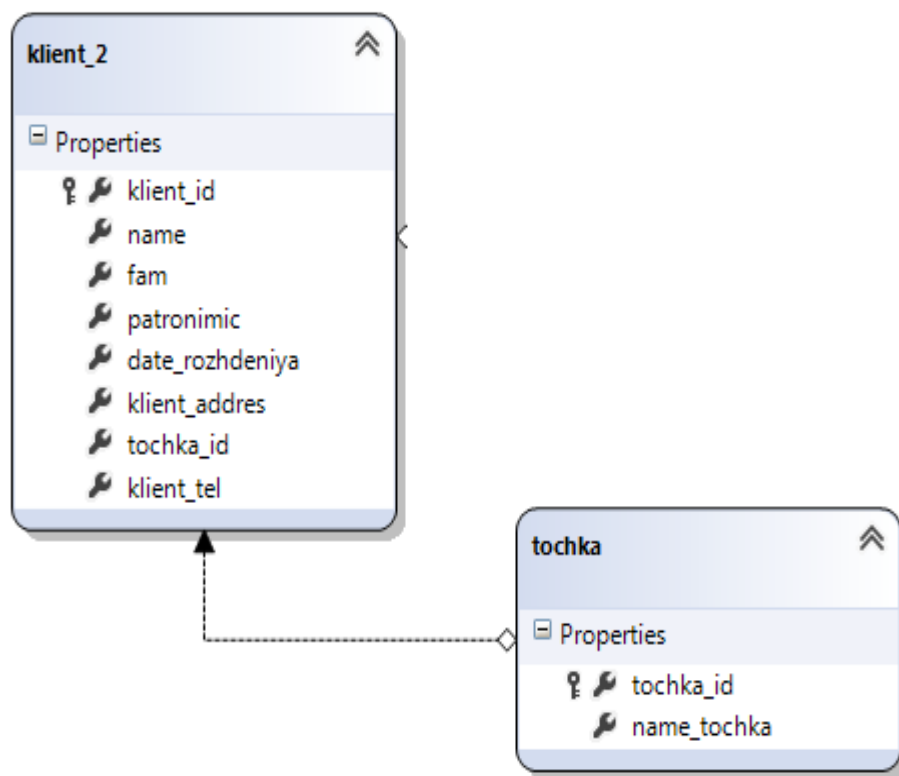
3) Газеттер тізімінде шығарылым нөмерлері болады.



Сурет 3.3 – Газеттер тізімінде шығарылым нөмерлері бизнес ереже үшін ER сегменті

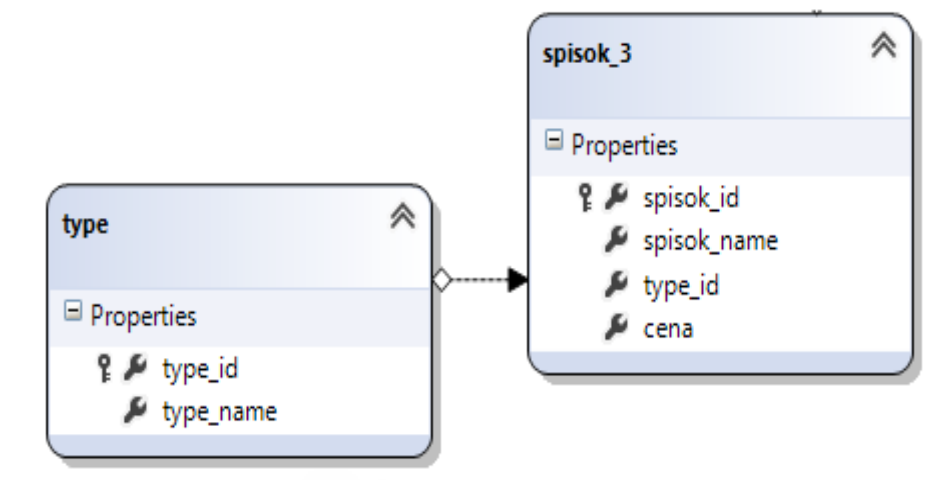


4) Сатылым орны бойынша клиенттерді қарауға болады.



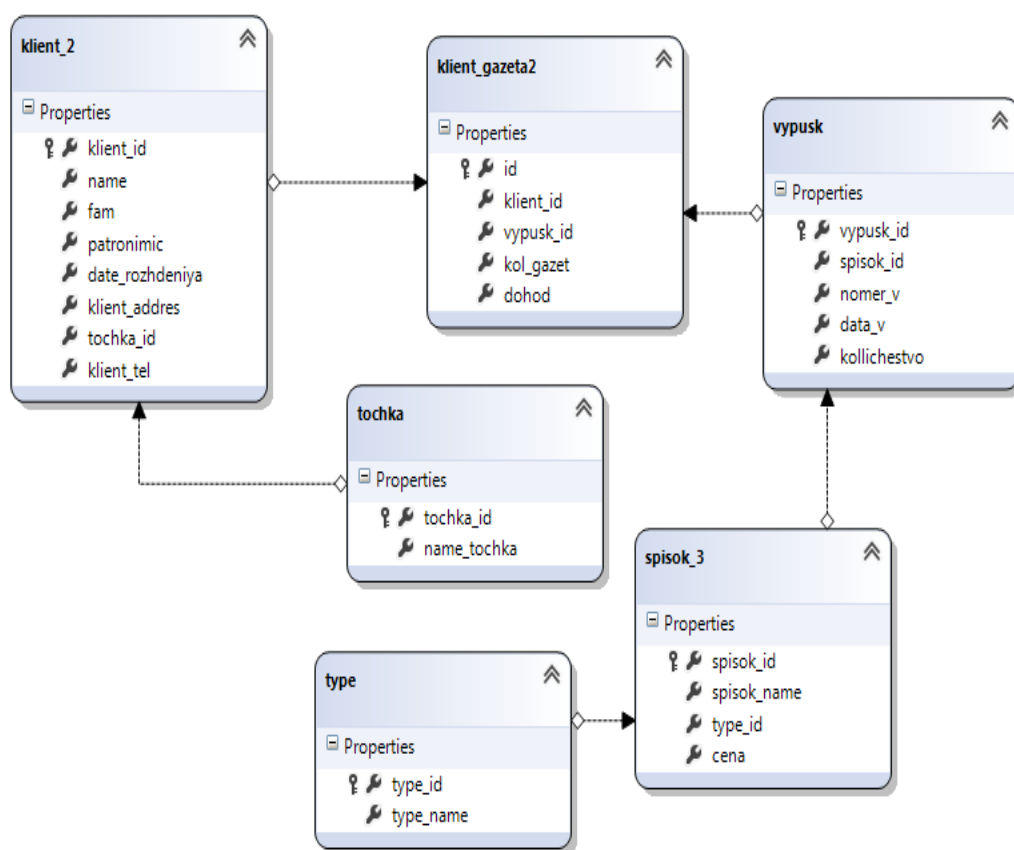
Сурет 3.4 – Сатылым орны бойынша клиенттерді қарау бизнес ереже үшін ER сегменті

5) Газет немесе журнал тізімі.



Сурет 3.5 – Газет немесе журнал бизнес ереже үшін ER сегменті

Жобаның ER диаграммасын 3.6–суреттен көре аласыздар.



Сурет 3.6 – ER диаграмма

### 3.3.2 Реяционды қатынастарды құрастыру

Әрбір реяционды қатынас бір нысанға сәйкес келеді және оған нысанның барлық атрибуттары енгізіледі. Әрбір қатынас үшін алғашқы кілт және сыртқы кілттерді (егер бар болса) анықтау қажет. Қатынастар 3.2–3.6-кестелерінде келтірілген. Әрбір қатынас үшін ішкі аты, типі мен ұзындығы бар атрибуттар көрсетіледі. Мәліметтер типі келесідей белгіленеді: int – сандық, char – символдық, Datetime – мезгіл (оның ұзындығы ДҚБЖ–не байланысты әртүрлі болады, сондықтан ол көрсетілмейді).

К е с т е 3.2 – tochka қатынасы

| Өріс аты           | Атрибут аты | Мәліметтер типі | Ескерту      |
|--------------------|-------------|-----------------|--------------|
| Сатылым коды       | tochka_id   | int             | Алғашқы кілт |
| Сатылым орны атауы | name_tochka | varchar(10)     | Қажетті өріс |

К е с т е 3.3 – type қатынасы

| Өріс аты    | Атрибут аты | Мәліметтер типі | Ескерту      |
|-------------|-------------|-----------------|--------------|
| Тип коды    | type_id     | int             | Алғашқы кілт |
| Типінің аты | type_name   | varchar(20)     | Қажетті өріс |

К е с т е 3.4 – spisok\_3 қатынасы

| Өріс аты     | Атрибут аты | Мәліметтер типі | Ескерту      |
|--------------|-------------|-----------------|--------------|
| Сатылым коды | spisok_id   | int             | Алғашқы кілт |
| Сатылым аты  | spisok_name | varchar(20)     | Қажетті өріс |
| Тип коды     | type_id     | varchar(100)    | Қажетті өріс |
| Бағасы       | sena        | int             | Сыртқы кілт  |

К е с т е 3.5 – klient\_gazeta2 қатынасы

| Өріс аты       | Атрибут аты | Мәліметтер типі | Ескерту      |
|----------------|-------------|-----------------|--------------|
| Коды           | id          | int             | Алғашқы кілт |
| Клиент коды    | klient_id   | int             | Қажетті өріс |
| Шығарылым коды | vyusk_id    | int             | Қажетті өріс |
| Газет саны     | kol_gazet   | int             | Қажетті өріс |
| Соммасы        | dohod       | real            | Қажетті өріс |

К е с т е 3.6 – klient\_2 қатынасы

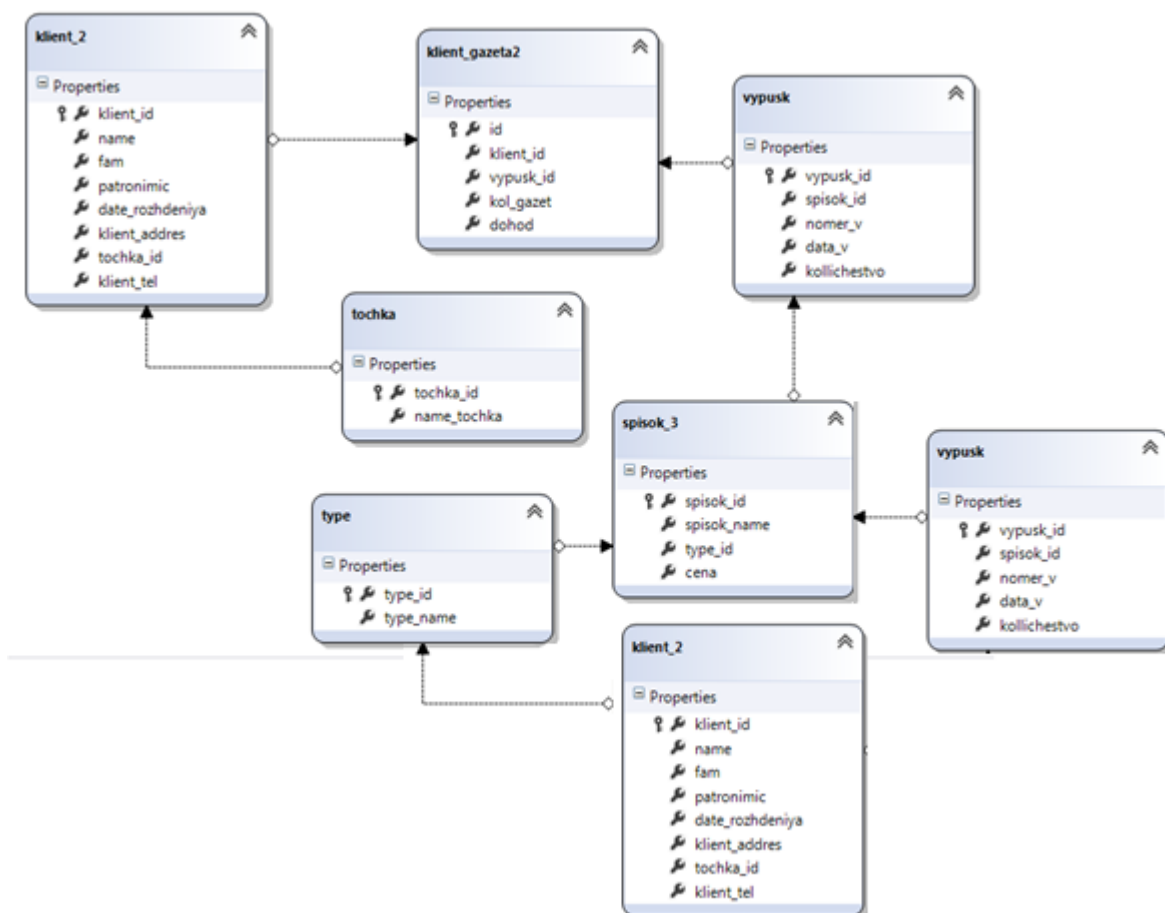
| Өріс аты     | Атрибут аты     | Мәліметтер типі | Ескерту      |
|--------------|-----------------|-----------------|--------------|
| Клиент коды  | klient_id       | int             | Алғашқы кілт |
| Аты          | name            | varchar(30)     | Қажетті өріс |
| Фамилиясы    | fam             | varchar(30)     | Қажетті өріс |
| Әкесінің аты | patronimic      | varchar(30)     | Қажетті өріс |
| Туған күні   | date_rozhdeniya | date            | Қажетті өріс |
| Адресі       | klient_addres   | int             | Сыртқы кілт  |
| Сатылым орны | tochka_id       | int             | Қажетті өріс |
| Телефоны     | klient_tel      | int             | Сыртқы кілт  |

### 3.3.3 Логикалық жобалау

Логикалық жобалау кезеңінде дерекқор моделі жүйені іске асыру кезінде қолданылатын белгілі бір ДҚБЖ (MS SQL Server) үшін жарамды форматқа трансформаланады. Логикалық жобалаудың мақсаты реляционды кестелік құрылымдарды құру болып табылады. Жобаланатын дерекқорда кестелердің және байланыстардың құрылымдарын MS SQL Server көмегімен жасаймыз.

ДҚ диаграммасы.

Концептуалды моделді дерекқордың сәйкес құрылымына енгізілу тәсілі қандай болса да, ДҚ реляционды сұлбасы жобаға сәйкес болу керек. ДҚ диаграммасын 3.7–суретте көрсетілген.



Сурет 3.7 – ДҚ диаграммасы

### 3.3.4 Көрсетілімдер және процедуралар

Көрсетілімдер бұл кестелер, бірақ оның құрамы басқа кестелерден алынады. Олар сұраныстарда дәл негізгі кестелер сияқты жұмыс істейді, бірақ оларда өзінің жеке деректері жоқ.

## 3.4 Физикалық жобалау

### 3.4.1 Дерек қорына қосымша құру

Тапсырманың қойылуы.

Мәліметтер қорымен жұмыс істеу қосымшасын жасау әдістемесі кәдімгі программаны құру әдістемесінен ешқандай айырмашылығы жоқ. Формаға

керекті компоненттер енгізіледі, компоненттердің қасиеттерінің мәндері қойылады, керекті оқиғаны өңдеу процедурасы жасалады.

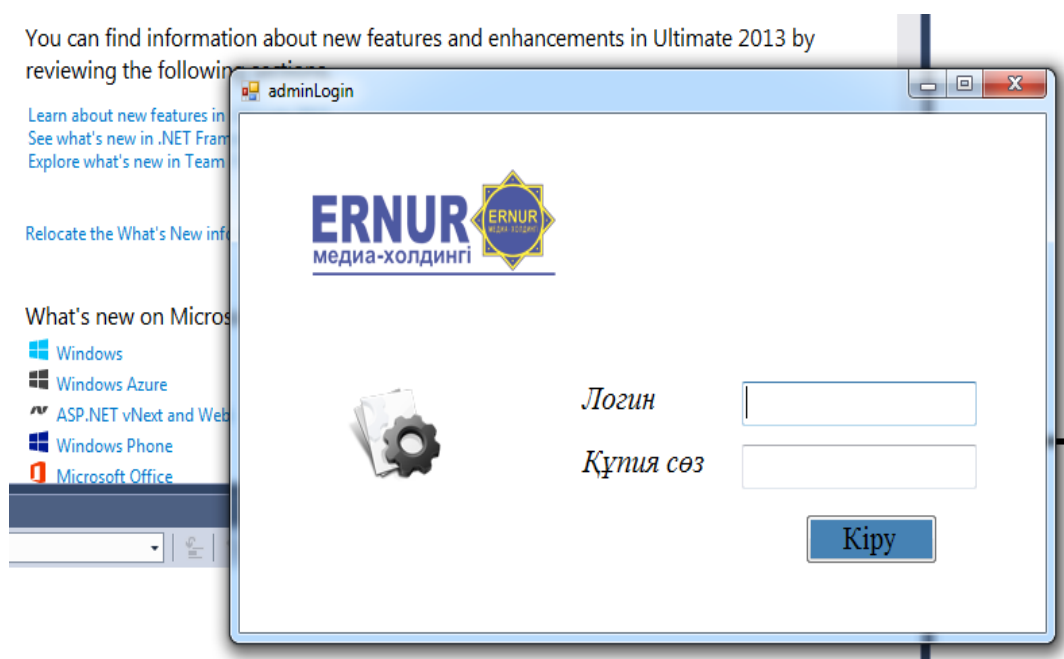
Мәліметтер қорымен жұмыс істеу қосымшасы мәліметтерге енуді қамтамасыз ететін компоненттерден, өрістердің мәліметтерін қарап шығу және редакциялау мүмкіндіктерінен тұрады. Мәліметтерге кіруді қамтамасыз ететін компоненттер Data Access компоненттер палитрасы қосымшасында, ал мәліметтерді бейнелеу компоненттері - Data Controls қосымшасында болады [5].

Database компоненті мәліметтер қорын кестелер жиынтығы ретінде, ал Table компоненті – мәліметтер қорының бір кестесі. DataSource (мәліметтер көзі) компоненті мәліметтерді бейнелеу-редакциялау компоненті (мысалы DBGrid компонентін) мен мәліметтер көзі (Table компоненті немесе SQL-сұраныс нәтижесі) арасындағы байланысты қамтамасыз етеді. DataSource компоненті мәліметтер көзін тез таңдауды, сол компоненттердің біреуін пайдаланып, мысалы DBGrid кестеден мәліметтерді көру үшін немесе осы кестеге SQL-сұранысты орындау нәтижесін көруге мүмкіндік береді [6].

### 3.4.2 Бағдарламаны сипаттау

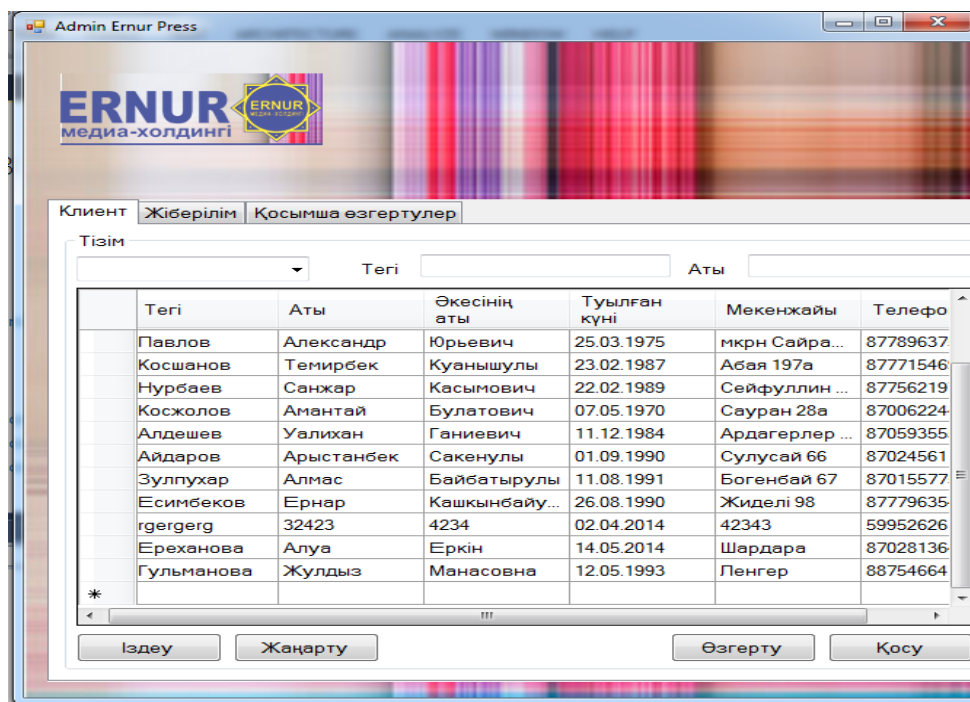
Жүйеге кірудің екі әдісі бар: администратор ретінде, менеджер ретінде. Программаның жұмыс барысы: программа іске қосылмас бұрын пароль арқылы қолданушыға ену рұқсатын береді. Бұл базадағы мәліметтерді қорғау мақсатында құрылған.

Менеджер интерфейсінде газеттер туралы ақпараттар, яғни түскен сұраныстар, 3.8–суретте клиенттер жайлы жалпы ақпарат көрсетілген.



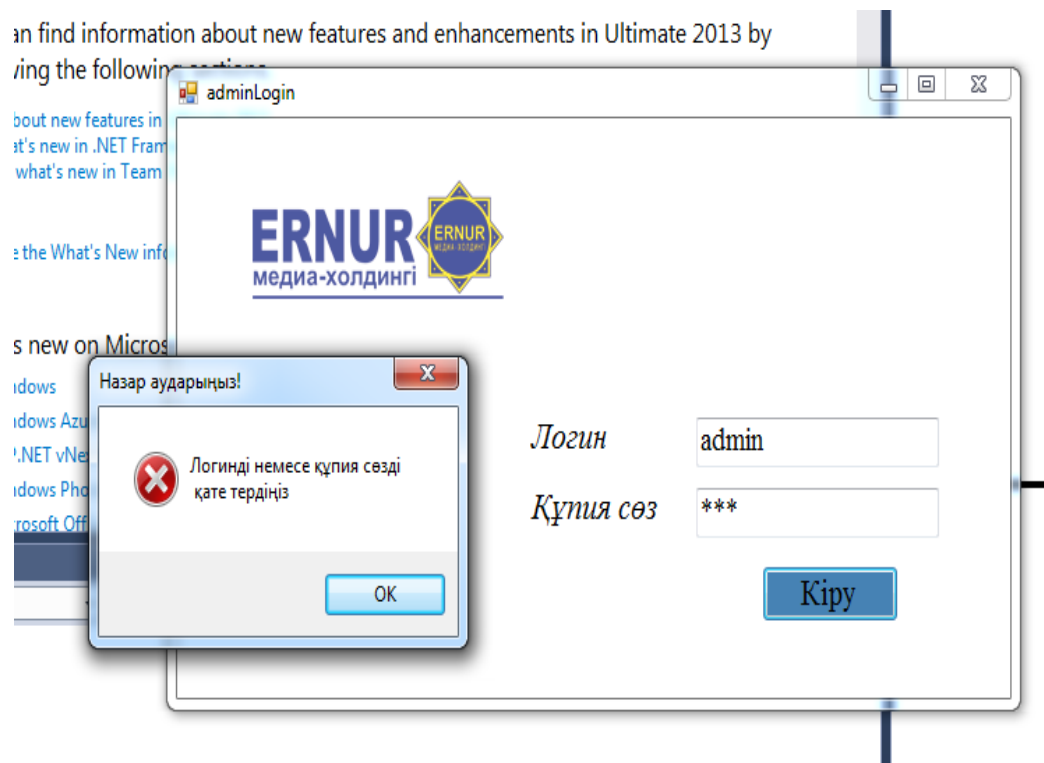
Сурет 3.8 – Админ ретінде жүйеге кіру беті

Администратор бөлімінің басты беті 3.9-суретте келтірілген.



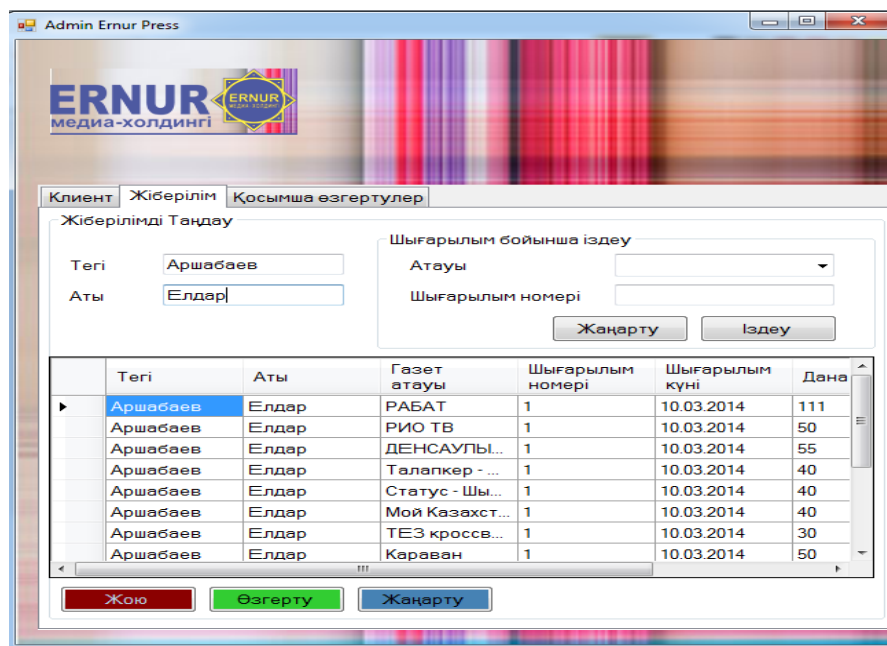
3.9 сурет – Администратор интерфейсі

Администратор терезесінде барлық түскен мәліметтерді көрсету, 3.9-суретте жеке паролін өзгерту функциялары бар.



3.10 сурет – Администратор интерфейсі

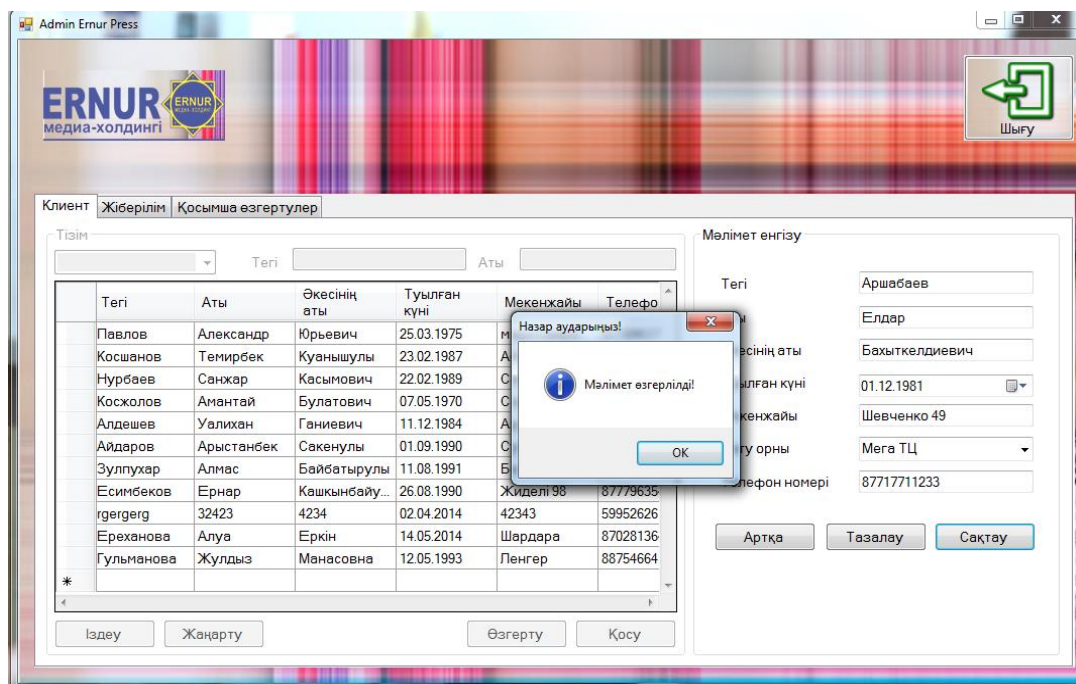
Администратор келіп түскен мәліметтерді іздеу функциясы бар. Іздеу функциясында мәліметтерді мәтін бойынша, аты-жөні бойынша, 3.11–сурет сату орны бойынша жүзеге асыру мүмкіншіліктері бар.



3.11 сурет – Ақпараттарды іздеу беті

### 3.3.2 Администратор интерфейсі

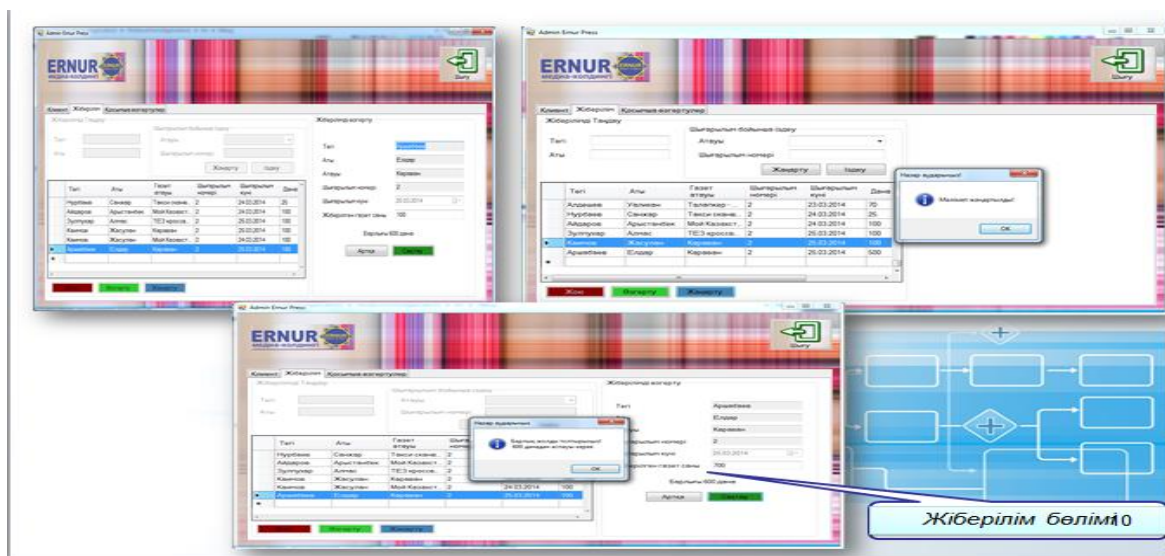
Администратор интерфейсі төрт категориядан тұрады: клиенттер, жіберілім, қосымша өзгертулерді басқару. Осы терезе 3.12–суретте көрсетілген.



3.12 сурет – Администратор интерфейсі

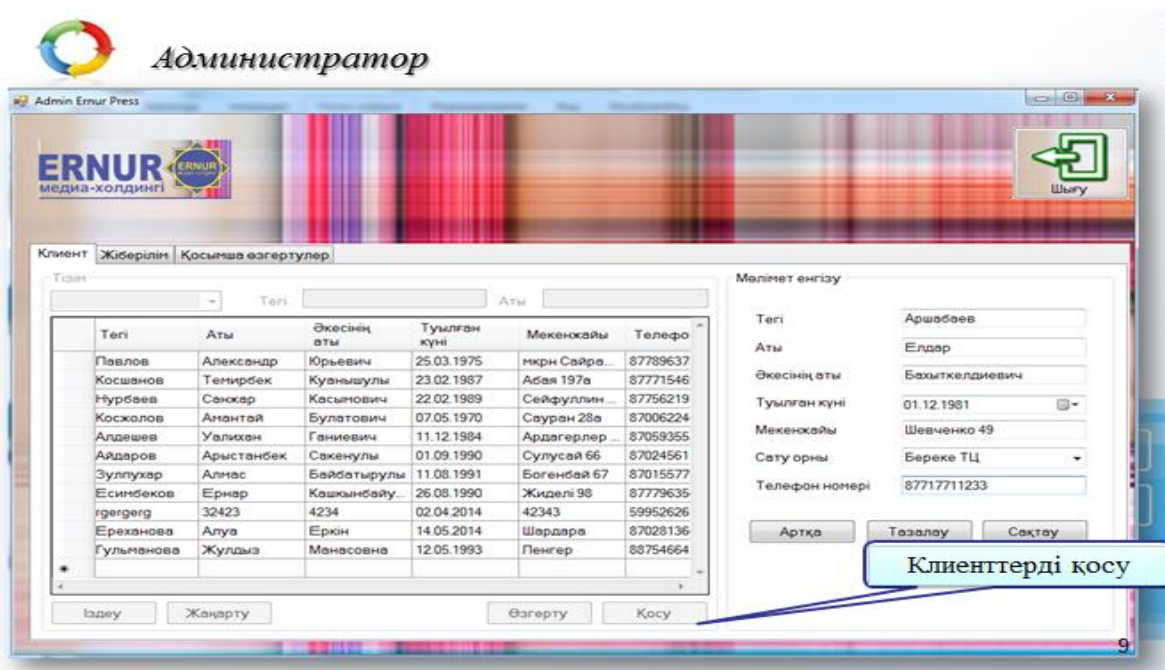


Клиенттердің сұраныстары бойынша шығылымдарды нөмері бойынша жеке сатылым орындарына жіберіп отырамыз.



3.13 сурет – Администратор беті

Администратордың басты міндеті—қолданушылардан келген мәліметтерді қарастырып, олардың өтініштерінің немесе белгілі техникалық қызмет көрсету туралы есеп беру жасау, жүйені басқару.



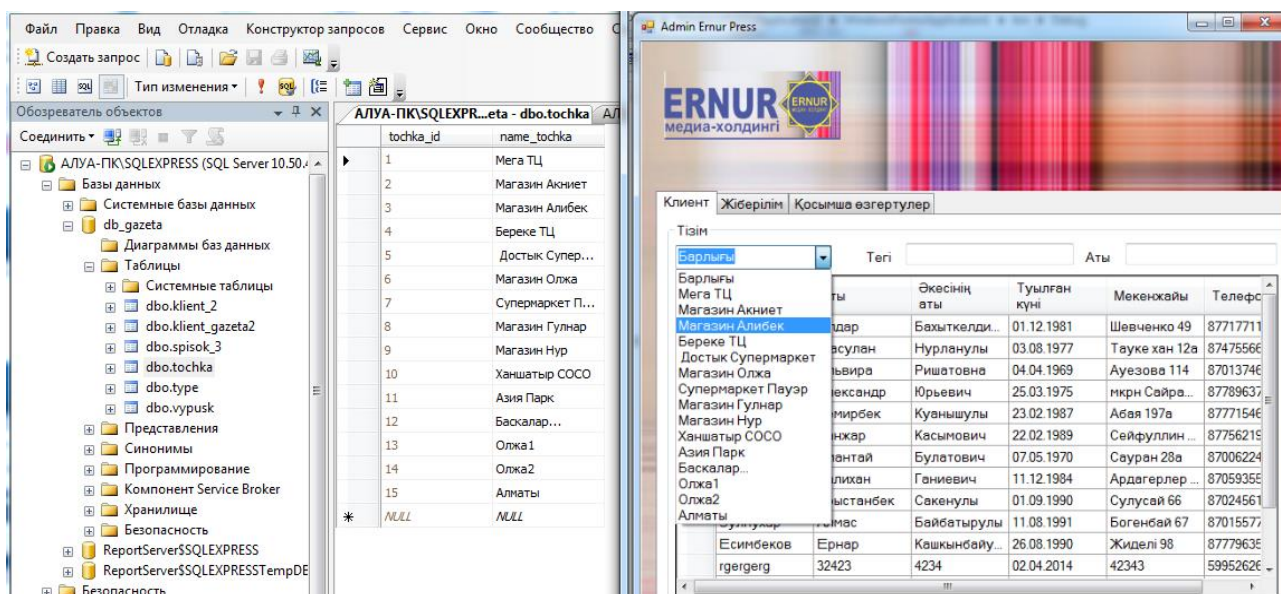
3.14 сурет – Жаңадан келген клиенттерді қосу терезесі



Жүйе администраторы келіп түскен мәліметтерді өңдейді. Клиенттердің сұраныстарын жүйеге енгізеді және жою, өзгерту мүмкіндіктері енгізілген. Администратор бетінде, сонымен қатар, жеке баптауларды өңдеу беті, әрбір қолданушылар туралы мәліметтер бар.

Сату орны беті.

Бұл бет үш бөліктен тұрады: таңдау бөлімі, негізгі кестелер, кестелерді өңдеу бөлімі. Таңдалып алынған кесте негізгі кестелер бөлімінде көрсетіледі. Өңдеу бөлімінде таңдап алынған кестемен жұмыс істейді, яғни жаңа ақпарат қосы, енгізілген өзгерістерді сақтау және қажет емес ақпараттарды өшіруге болады.



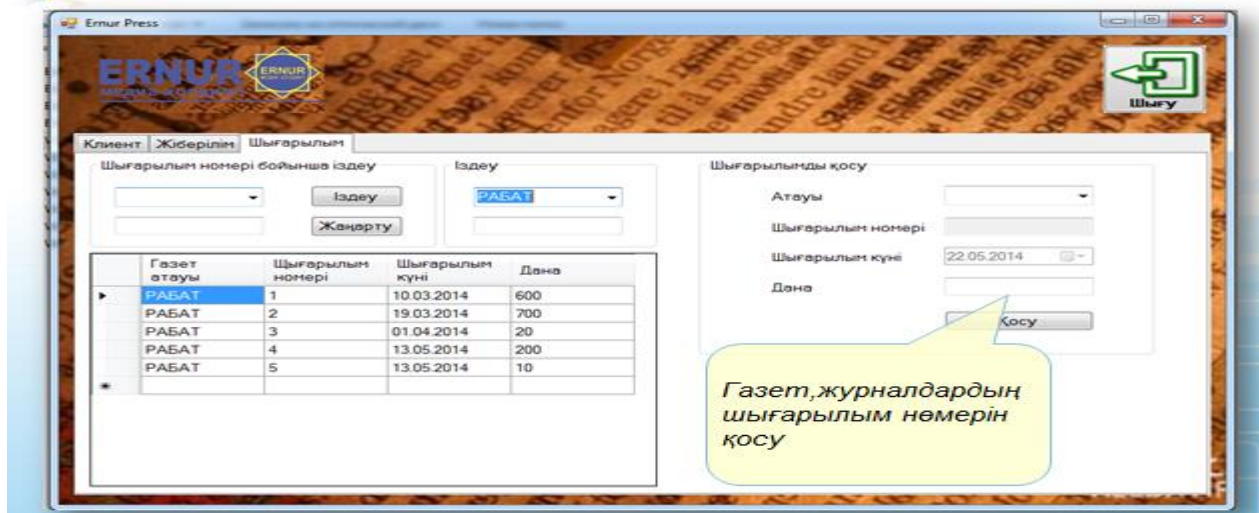
Сурет 3.15 – Сату орындарын көрсететін терезе

Ақпаратты қосу беті.

Баспа орталығы газеттерді шығару және таратумен айналысады. Таралым бойынша және шығарылатын газеттердің саны, сатылым орындары, тапсырыс берген клиенттер жайлы базаға енгізіп отырамыз. Осы қызметті администратор асайды. Администратор жүйені редакциялаумен айналысады. Администратор жүйені редакциялауын 3.16-суретте көрсетілген.

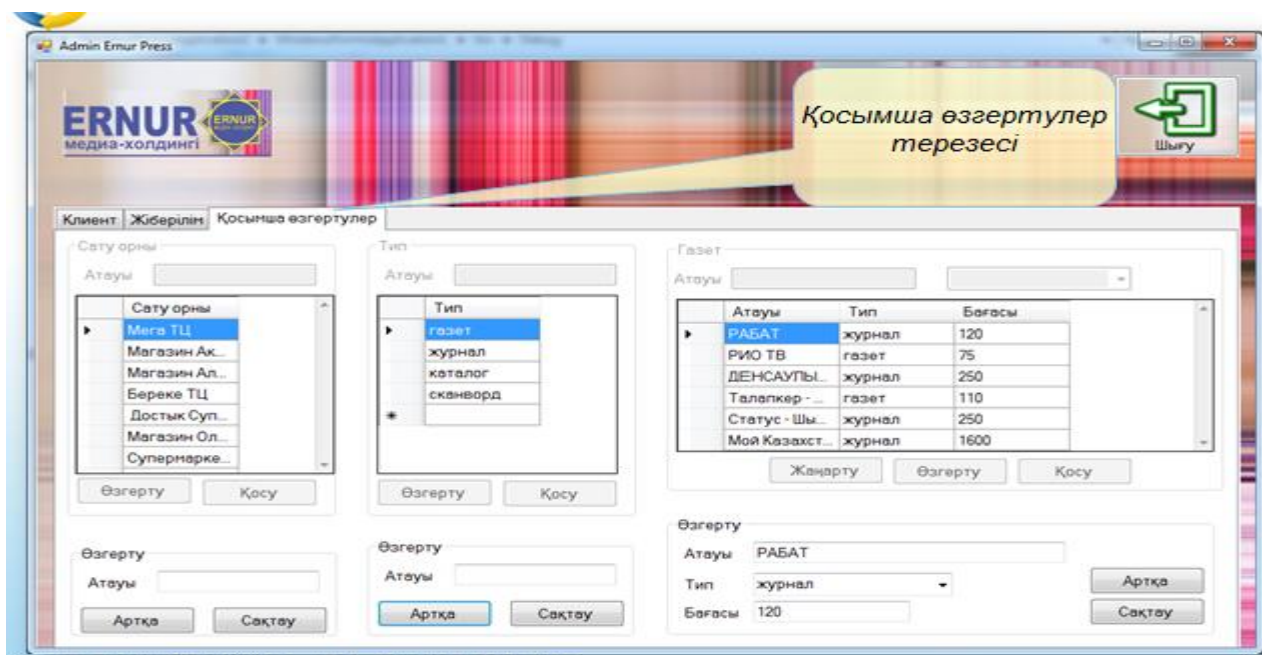


## Ақпараттарды қосу



Сурет 3.16 – Ақпараттарды қосу терезесі

Қосымша өзгертулер терезесі 3.17 -суретте көрсетілген. Қосымша өзгертулерде жаңадан қосылған сатылым орындарын, шығарылатын газеттер немесе журналдар типтерін енгіземіз. Өзгерту батырмасы бойынша газет түрлерін енгіземіз.

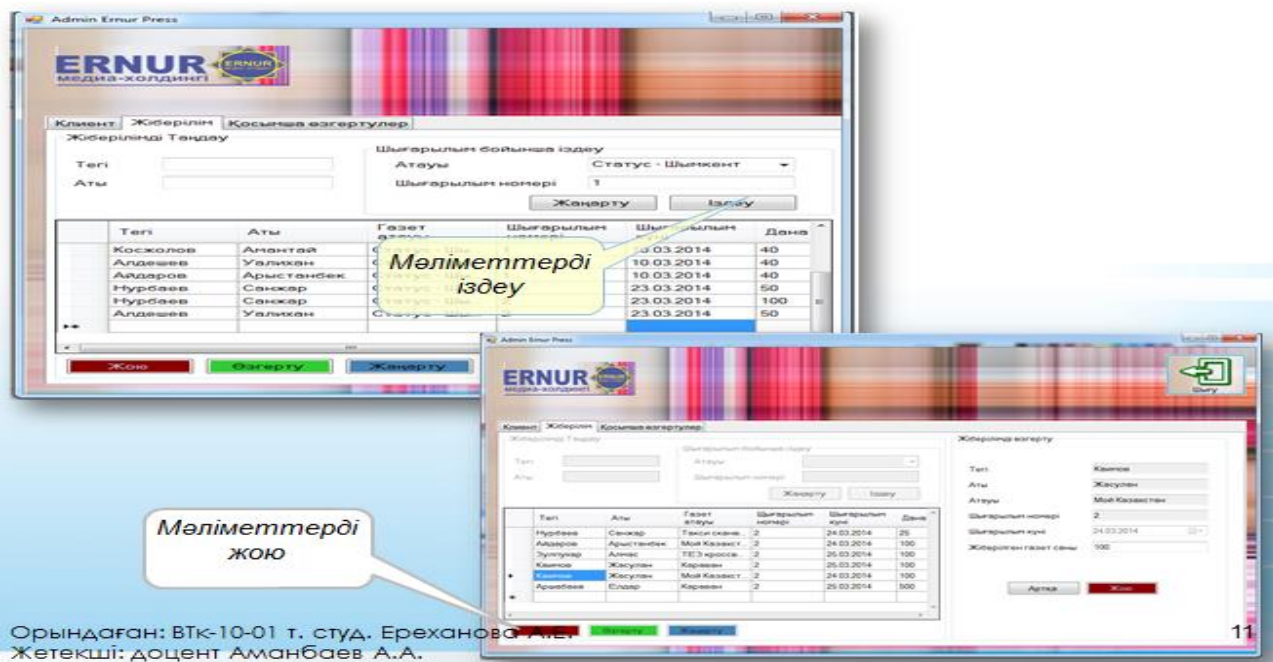


Сурет 3.17 – Қосымша бойынша сұрыптама жасау

Мәліметтерді іздеу беті.

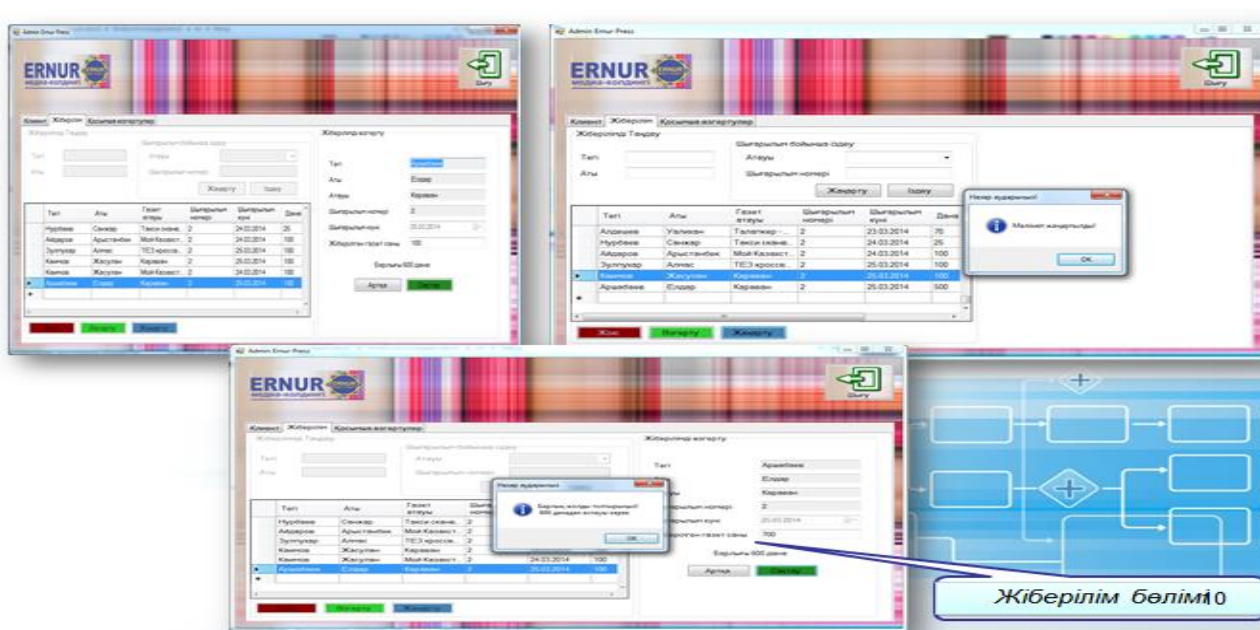
Есеп беру маңызды бөліктерінің бірі болып табылады.

Мысалы 3.18-суретте көрсетілген. Сұрыптаманы атауы бойынша, клиенттің аты бойынша, шығарылым нөмірі бойынша.



Сурет 3.18 – Мәліметтерді іздеу терезесі

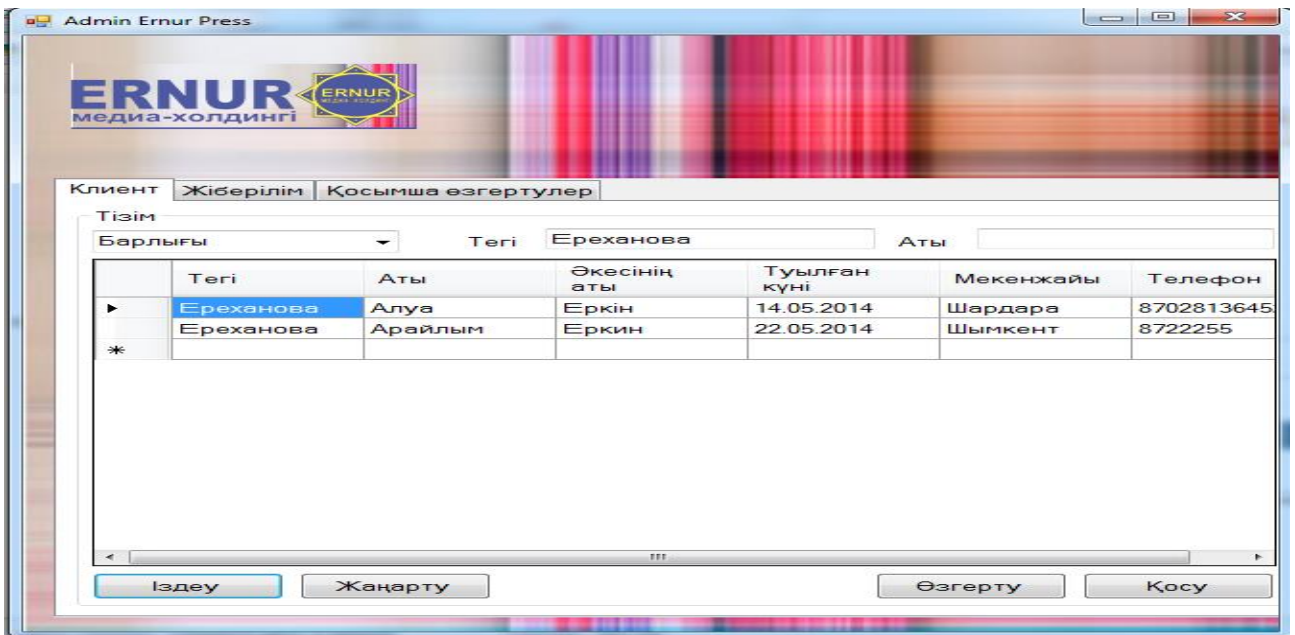
Клиенттердің тапсырысы бойынша сатылым орындарына, клиенттің жеке өзіне газеттерді жібереміз.



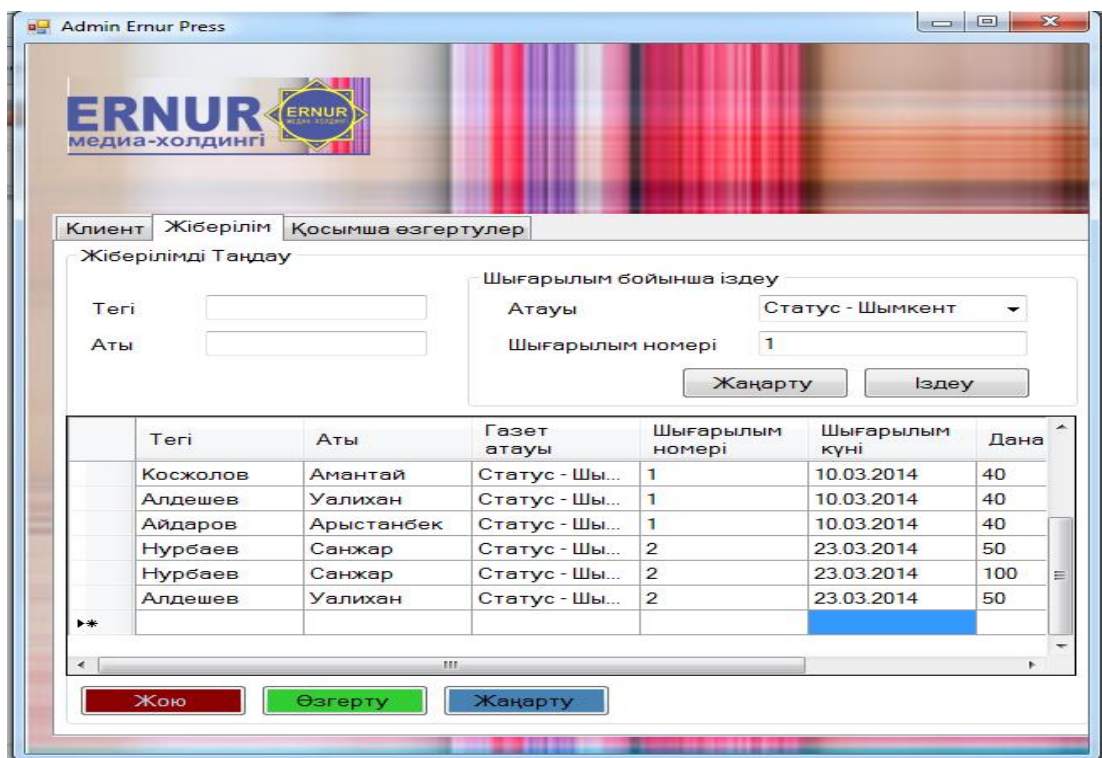
Сурет 3.19 – Жіберілім бөлімі

Керекті мәліметтерді іздеу бөлімі бойынша іздеуге болады. Суретті 3.20-суреттен көре аласыздар.



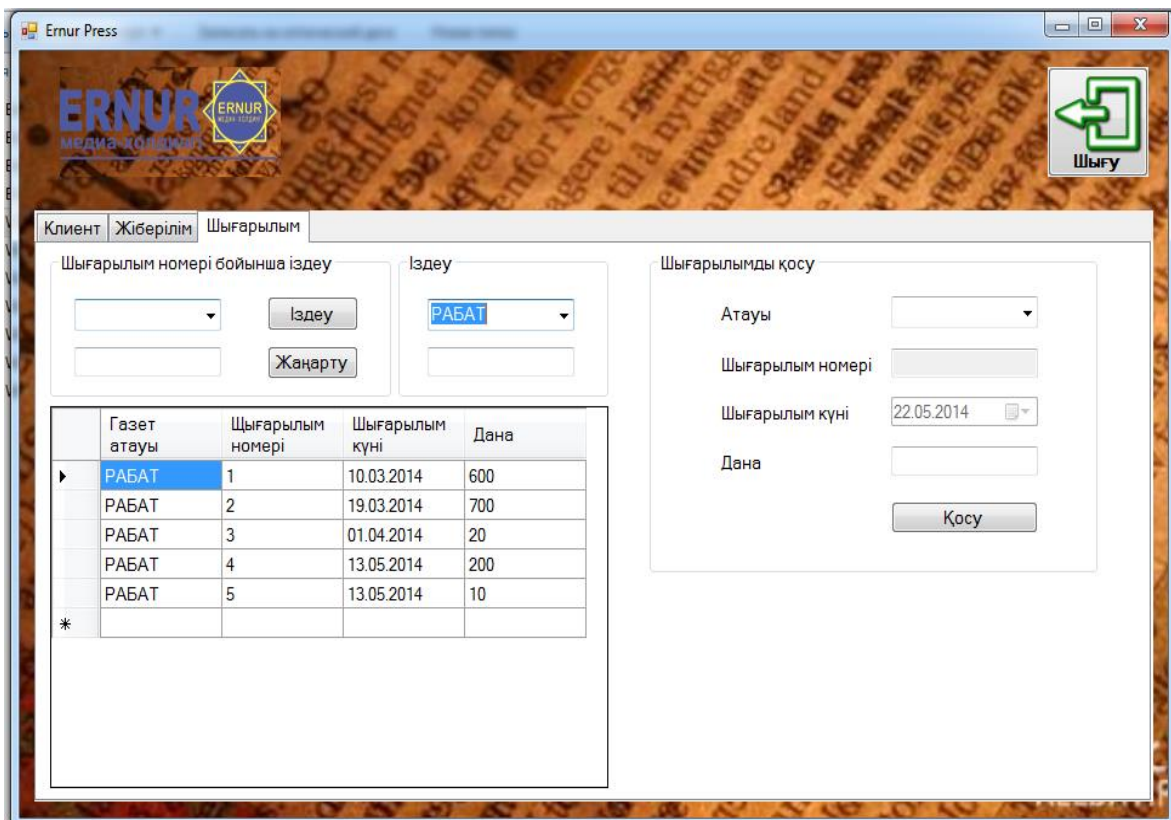


Сурет 3.20 – Мәліметтерді іздеу

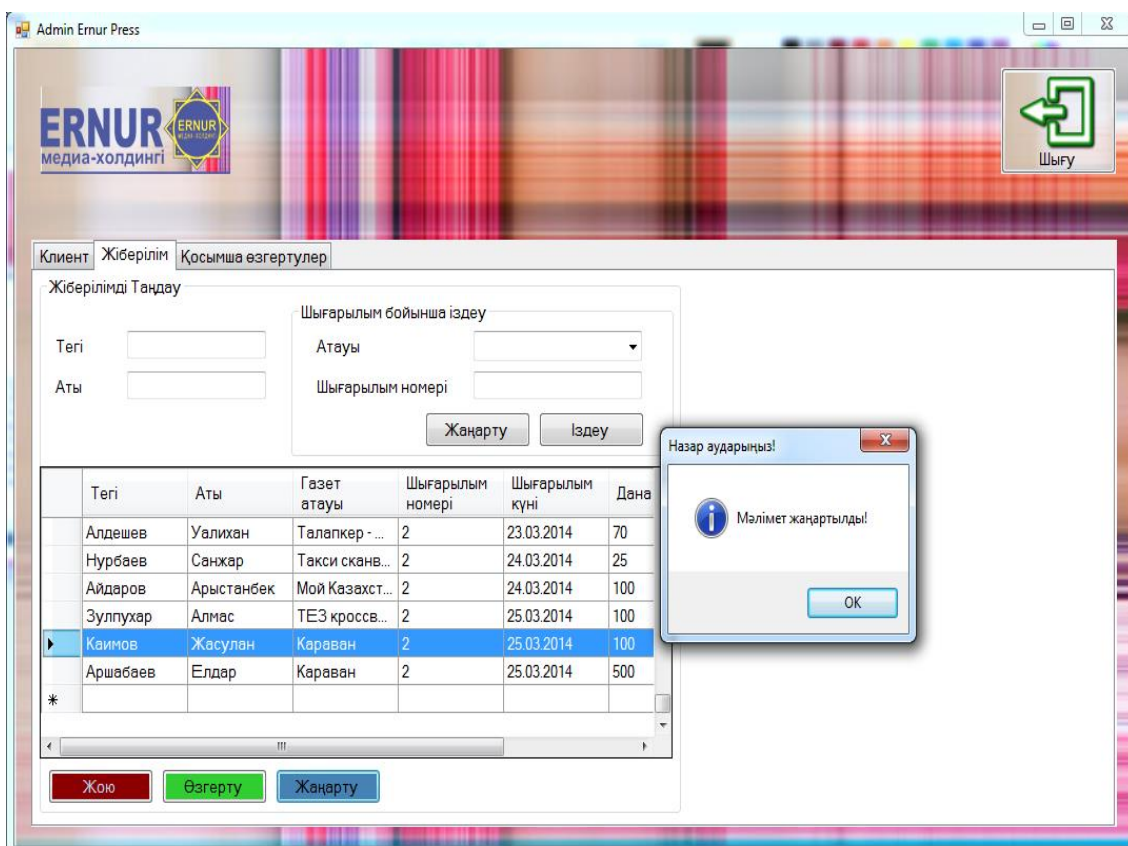


Сурет 3.21 – Жіберілімді өзгерту

Газеттер бойынша іздеудің нәтижесін 3.22-суреттен қарауға болады.



Сурет 3.22 – Рабат газеті бойынша іздеудің нәтижесі



Сурет 3.23 – Жаңарту бөлімі

## 4 Тіршілік қауіпсіздігі

### 4.1 Жалпы жұмыс жағдайын талдау

Дипломдық жобаның тақырыбы «Шымкент қ «Ернұр Пресс» ЖШС баспа орталығы жұмысының деректер қорын құру». Жобаны жасау барысында компьютер негізгі рөл атқарады. Сондықтан, жұмысшылардың компьютермен жұмыс істеген кездегі еңбектің қауіпсіздігімен жұмысқа қабілеттілігін сақтауымыз қажет.

Жұмыс мақсаты-тапсырмаға байланысты жекелеген өндіріс бөлмесінің жасанды жарықтандырылуын орындап есеп жүзінде дәлелдеу,  $E_{min}$  қажетті жарық шамдарының санын анықтап нормаланған жарықтандыруды есептеу.

Бұл жобада қарастырылған жұмыс бөлмесі Шымкент қаласындағы «Ернұр Пресс» ЖШС ғимаратында орналасқан. Сол ғимараттың ішінде бағдарламаны орнату және оны басқару бөлмесі орналасқан. Айтылып жатқан ғимаратымыз 1 этажды болып келеді.

Оператордың жұмыс орнын ұйымдастырғанда жұмыс бөлмесі ауданының маңызы зор. Бұл жобада қарастырылған жұмыс бөлмесінің ұзындығы 8м, ені 4м, биіктігі 3м, бөлмеде 2 терезе бар, оның көлемі 2x2. Бөлмеде 4 адам жұмыс істейді, жұмыс графигі - аптасына 6 күн, күніне 8 сағат.

Дербес компьютер операторының жұмысы ұзақ көру жұмысымен байланысты болғандықтан, Оператор бөлмесіндегі жарықтануды есепке алу қажет. Өндірістік кәсіпорындардағы жасанды жарықтандыру шарттары қызметкерлердің көру қабілетіне, физикалық және моральдық жағдайларына елеулі әсер етеді.

Бұл жобада баспа орталығы жұмысының деректер қорын құру қарастырылған. Жұмысшының негізгі жұмысы ақпаратты енгізу немесе толықтыру, бағдарламалар бойынша дербес компьютерде тапсырманың есептелуін түзету және бақылау. Міне осыған орай, оператор организмінің энергия шығынына байланысты жұмыс категориясын таңдаймыз. ГОСТ 12.1.005-88 ССБТ «Жұмыс орнының ауасы, санитарлық-гигиенаның жалпы талаптары» 4.1–кестеде келтірілген. Оператордың жасайтын еңбегі бірінші категорияға жатады (Ia).

К е с т е 4.1 –Адам организмінің жұмыс категориясы бойынша энергия шығыны

| Жұмыс категориясы         | Категория | Адам организмінің энергия шығыны, кал/сағ | Жұмыс сипаттамасы         |
|---------------------------|-----------|---|---------------------------|
| Жеңіл физикалық жұмыс I a | I a       | 138-172                                   | Жұмыс отыру күйінде өтеді |

## 4.2 Жұмыс орнының микроклимат жағдайы

Жұмысшылардың қолайлы еңбек ету шарттарын қамтамасыз ету үшін микроклимат параметрлерін нормалау қажет. Жұмыс орнының қолайлы жағдайында адамдардың жұмыс істеу қабілеттері жақсарады, жұмысшылардың ауруға шалдығуы азаяды сонымен қатар, еңбек өндірісі артады.

Жұмыс орнындағы нормаланған климаттық параметрлер 4. 2 – кестеде келтірілген.

Кесте 4.2 – Температураның, ылғалдылық пен ауа қозғалысының нормалары ГОСТ 12.1.055.88ССБТ

| Жұмыс мезгілі | Ауа температурасы, °С | Салыстырмалы ауа ылғалдылығы, % | Ауа қозғалысының жылдамдығы, $\frac{м}{с}$ |
|---------------|-----------------------|---------------------------------|--|
| Суық          | 22 - 24               | 40–60                           | 0,1  |
| Жылы          | 23 - 25               | 30–50                           | 0,2  |

Оператор бөлмесіндегі жаз уақыт кезіндегі температура +26°С-ге дейін көтеріледі, ал қыс кезіндегі температура +18-ден +22°С-ге дейін. Бөлменің терезелер арқылы түсетін табиғи жарықтануы және қараңғы уақытта жұмыс істеу мүмкіндігін беретін жасанды жарықтануы бар.

Географиялық ендікке байланысты, жыл мезгілі және тәулік, атмосфераның бұлттылығы мен мөлдірлігіне тәуелді, тікелей күн көзі және көкжиектен түсетін жарықтандыруды табиғи жарықтандыру деп атаймыз. Ал жасанды жарықтандыру деп электр көзінен түсетін жарықтандыруды айтамыз. Операторлардың жұмыс орындарының бөлмесін жасанды жарықтандыру үшін люминесцентті шамдарды қолданамыз, олардың жарық бергіштігі жоғары (75 им/Вт-қа дейін және одан жоғары), жарамдылық мерзімінің ұзақ (10 000 сағатқа дейін). Өрт болу мүмкіндігін болдырмау және оның пайда болған кезінде адамдарға қажетті шараларды қолдану мақсатында өмірге қажетті жүйелерді (жылу, жарықтандыру, вентиляция т.б.) орнатқан кездерде өрт қауіпсіздігі нормаларын сақтауымыз керек. Сол себепті де бөлмеде әртүрлі қорғану жүйелерін пайдалануымыз қажет. Өрт сөндіру құралдары, қолданыстағы нормаларға сәйкес автоматтандырылған өрт сөндіру және байланыс құралдары болуы қажет. Бөлмеде барлық қауіпсіздік шаралары ұйымдастырылған және өрт сөндіргіш құрал орналасқан. Қазіргі кезде электр тоғы өмірде, шаруашылықта жиі пайдаланылады, оның көмегімен ауыр деген жұмыстар жеңілдейді, өндіріс мәдениеті жоғарылайды. Өндірісте электр тоғының айнымалы түрі қолданылады, ол төрт сым арқылы беріледі, жиілігі 50-Гц. Электр тоғының қауіптілігі сонда, өйткені оны көзбен көрмейміз. Сонымен, электр тоғының адам организміне қандай әсері бар десек, олар: жылулық (термикалық); химиялық; биологиялық; механикалық. Адам өміріне

қауіпті жағдайлардың бірі электрлік желілер. Электрлік желілерден қорғаныс құралдары: -жерлендіру; -нөлдеу; -қорғанышты ажырату.

### 4.3 Шудың әсері

Шу адам ағзасына зиянды әсерін тигізе отырып, жұмыс жағдайын нашарлатады. Адамға шу ұзақ уақыт әсер еткен кезде жағымсыз жағдайлар туындайды: көз көру, есту қабілеті нашарлайды, қан қысымы көтеріліп, көңіл бөлу нашарлай түседі. Қатты ұзақ шу жүрек-қан қысымы және жүйке жүйесі қызметінің бұзылуына себепін тигізеді. Шудың зиянды әсерінен қорғайтын жеке сақтық құралдары да бар.

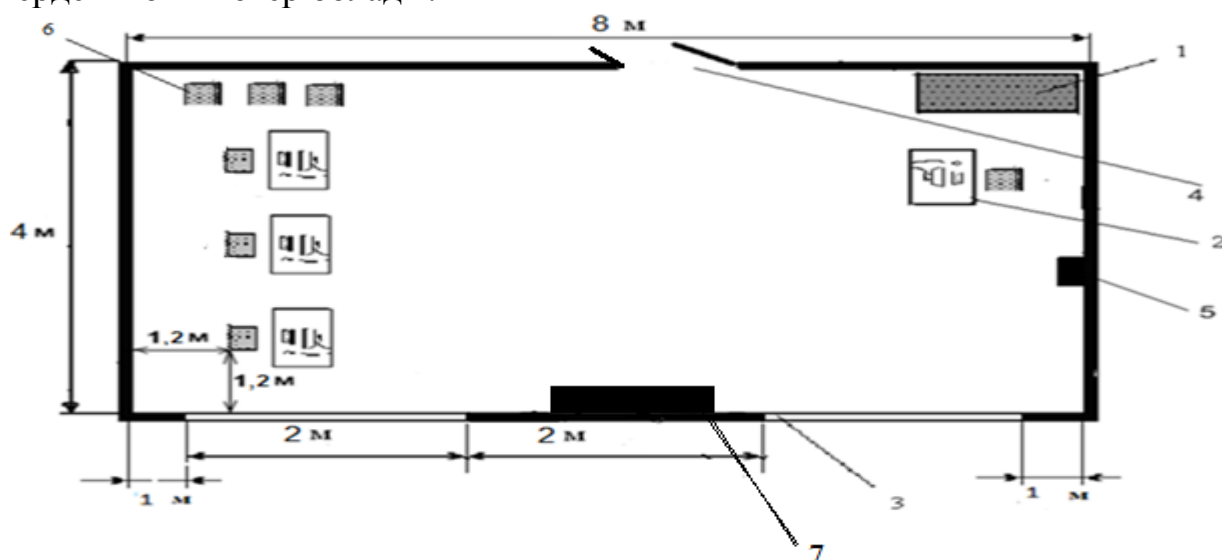
ГОСТ 12.2.003-76 сәйкес жұмыс орнында және одан тыс жерде пайда болатын шудан қорғану үшін келесі әдістерді қолдану керек:

- шу шығатын көздерде оны азайту;
- ұжымдық және жеке қорғаныс құралдарын қолдану;
- жұмыс бөлмелерін ұтымды жоспарлау және акустикалық өңдеуден өткізу.

Шу деңгейін қадағалау операторлар залында қолайлы жағдайды орнатуға көмектеседі. Ал жұмыс орнының қолайлы жағдайларында адамдардың жұмыс істеу қабілеттері жақсарады, жұмысшылардың ауруға шалдығуы азаяды сонымен қатар, еңбек өндірісі артады.

### 4.4 Бөлме жоспары

Компьютерлер енгізілген бөлменің жоспары 4.1-суретте көрсетілген. Бұл жерде 4 компьютер болады.



Сурет 4.1 – Бөлме жоспары:

- 1 – Шкаф.
- 2 – Операторлардың жұмыс орындары.
- 3 – Терезе жақтаулары.



- 4 – Есік.
- 5 – Өрт сөндіргіш.
- 6 – Орындықтар.
- 7 – Кондиционер.

Өндірістік кәсіпорындарындағы жасанды жарықталуының шарттары адамның көру жұмыс қабілеттігіне, физикалық және моральдық күйіне үлкен әсер етеді, ал бұл еңбек өнімділігіне және өнімнің сапасына әсер етуіне әкеледі. Жарық – тіршілік көзінің бір бөлігі.

Ал өндіріс ортасын жарықтандыру дәрежесі еңбеккердің жұмыс жағдайын жақсартады, шаршауын төмендетеді, еңбек өнімділігі мен өнімнің сапасының жоғары болуына тікелей әсері бар. Жарықтың күн көзіне немесе аспан денелерінен пайда болған түрін – табиғи жарық дейміз де, электр шамы арқылы пайда болған жарықты – жасанды жарық дейміз. Күн көзінің жарық шамасы жердің күнді айналуына, атмосфераның оптикалық жағдайына, ауа райына, жарықтың шағылысуы мен жұтылуына байланысты.

Күн сәулесі спектрі өзінің биологиялық әсері бойынша үш түрге бөлінеді. Күн сәулесі спектрінің толқын ұзындығы 380-760 нм бөлігі көздің көру нервтеріне әсер етіп жарық құбылысын тудырады. 100-380 нм ұзындықтағы сәуле толқындар ультра күлгін (радияциялық) 760-340000 нм сәуле толқындар инфрақызыл (жылу сәулелері) деп аталады. Олар (көру) жарық құбылыс тудырмайды. Жарықтандырудың сапасын көрсететін негізгі техникалық жарық көрсеткіштері болып – жарық күші (J), жарық шамасы (E), шағылыстыру коэффициенті  $\alpha_{ж}$ , жұту коэффициенті  $\beta_{ж}$ , өткізу коэффициенті  $\gamma_{ж}$ , жарық ағыны (Ф), жарықтану дәрежесі Я (Яркость), көру нысаны мен фон арасындағы контраст, жарықтың шамасының өзгеруі және жарықтың біргелілік коэффициенті саналады. Жарық шамасын өлшейтін аспап люксметр деп аталады.

Ғимараттарды жарықтандыруды мөлшерлеудің екі әдісі бар: геометриялық және техникалық жарық көрсеткіші арқылы.

Табиғи жарықтандыру күн сәулесін бүйірден түсіру, жоғарыдан түсіру және аралас болып бөлінеді. Бөлмені табиғи жарықтандырудың геометриялық мөлшерлеу мәні болып жарық түсетін ойық пен бөлме едені ауданы арасындағы қатынасты анықтау болып табылады. Ол әрине бөлменің пайдалану жағдайына байланысты. Өндірістік бөлмелер үшін – табиғи жарық коэффициенті  $e = F \cdot n / f = 1/10$ .

Ал жасанды жарықтандыруды мөлшерлеу – жарық шамасы (E) люкспен көрсетіледі. Оның мөлшері көру жұмысының сипаттамасына, көретін нысанның размеріне (0,15мм-ден 5мм-ден артық) контраст, фон көрініске қарай беріледі. Жарық ағыны нысанға бағытталып, шам – жағдайға сәйкес таңдалынып алуы керек.

Жарық адам ағзасының жалпы жағдайына әсер етеді. Сондықтан жұмыс орнындағы жарықтандырудың дұрыс таңдалуы өте қажетті.

## 4.5 Жұмыс орынның жасанды жарықталуы

Табиғи жарықтандыру болмағанда немесе жеткіліксіз болғанда жасанды жарықтандыру қолданылады, қыздырмалық шам, жоғары және төмен қысымды газоразрядтық шамдар, жазық және саңылаулы жарықжолдар ретінде жарықтың осындай көздері іске асырылады.

Көздің шаршаған дәрежесі заттарды көру әсерінен ілесетін процестердің кернеулігіне байланысты болады. Сондай процестерге көруге икемделу, үйлесу және бейімделу жатады.

Еңбектің қолайлы жағдайларын жасау үшін өндірістік жарықтандыру келесі талаптарға тиісті жауап беру қажет:

– жұмыс орындағы жарықтың «Табиғи және жасанды жарықтандыру. Жалпы талаптар» 2.04-05-2002 ҚР СНИП бойынша орындалатын жұмыстың сипаттамасына сәйкес болу қажет.

– жұмысшы беттердегі және қоршаған кеңістік шектеріндегі жарықтың мүмкіндігінше бірқалыпты таралу қажет;

– жұмыс бетінде айқын көлеңкелер болмау қажет;

– түсті дұрыс тарату үшін қажетті жарықтың қажетті спектралдық құрамын жарықтандыру қамтамасыз ету қажет;

– жарықтандыру жүйесі басқа зиянды факторлардың көзі болмау қажет, сондай-ақ электрге және өртке қауіпсіз болу қажет.

Жарықтың көзін таңдау. Жасанды жарық үшін қызу және газоразрядтық шамдар қолданылады. Жалпы жұмыстық қыздыру шамдардың типтерін белгілеу шартындағы келесі әріптер мынаны білдіреді: В- вакуумдық, Г- газбен толтырылған, Б- биспиральдық, БК- биспиральдық криптондық. Шамдардың жарықтық ағымдары, жалпы жұмыстық қыздырушылық және олардың қуаты 4.3-кестеде келтірілген.

К е с т е 4.3 – Қыздырушылық шамның қуаты мен жарықтық ағым

| Шамның типі | Қуат Вт | 220                        | 200-235 | Шамның типі | Қуат Вт | 200                        | 200-235 |
|-------------|---------|----------------------------|---------|-------------|---------|----------------------------|---------|
|             |         | кернеуіндегі жарықтық ағым |         |             |         | кернеуіндегі жарықтық ағым |         |
| В           | 15      | 105                        | 85      | Б           | 150     | 2100                       | 1840    |
| В           | 25      | 220                        | 190     | Г           | 200     | 2800                       | -       |
| Б           | 40      | 400                        | 300     | Б           | 200     | 2920                       | 2540    |
| БК          | 40      | 460                        | -       | Г           | 300     | 4600                       | 4000    |
| Б           | 60      | 716                        | 550     | Г           | 500     | 8300                       | 7200    |
| БК          | 60      | 790                        | -       | Г           | 750     | 13100                      | -       |
| Б           | 100     | 1350                       | 1090    | Г           | 1000    | 18600                      | -       |

## 4.6 Жасанды жарықтандыруды есептеу

Жарықтандыру табиғи және жасанды болуы мүмкін. Жасанды жарықтандыру екі жүйелі болуы мүмкін: жалпы және аралас. Аралас жарықтандыру кезінде жалпыға жергілікті жарықтандыру қосылады.

Жасанды жарықтандыруды есептеу үшін пайдалану коэффициенті және нүктелік әдістер қолданылады. Пайдалану коэффициенті әдісі үлкен көлеңке түсіретін заттар болмаған жағдайдағы біркелкі горизонталь беттердегі жарықтандыруды есептеуге арналған. Бұл әдіс арқылы  $\eta$  коэффициентінің мәні анықталады. Ол есептелінетін бетке түсетін жарық ағынының жарықтандыру құралының толық ағынына қатынасы болып табылады.

Біздің есептейтін бөлменің өлшемдері мынадай:

- габариттері: 8x4x3;
- шағылу коэффициенттері –  $\rho_{\text{төбе}} = 70\%$ ,  $\rho_{\text{қабырға}} = 50\%$ ,  $\rho_{\text{еден}} = 30\%$ ;
- жарықтандырудың бірқалыпсыздық коэффициенті  $\lambda = 1.1 \div 1.2$ ;
- еденнен жұмыс орнына дейінгі биіктік:  $h_{\text{еден}} = 0.8$  м.

Нормаланған жарық қабырғалары, төбесі әктелген, терезелері жабылмаған бөлме үшін  $E=300$ лк. Жарықтандыру жүйесіне люминесцентті ЛД шамдарын қолданамыз, оның қуаты 40 Вт, ал жарықтық ағыны 2340 лм.

Ілінудің есептік ұзындығын анықтаймыз. Жұмыс орнының еденнен биіктігі  $h_{\text{еден}} = 0.8$  м, ал шамның іліну биіктігі  $h_{\text{іліну}} = 0.2$  м болғандықтан, ол келесі шамаға тең:

$$h = h - (h_{\text{еден}} + h_{\text{іліну}}), \quad (4.1)$$

$$h = 3 - (0.8 + 0.2) = 2 \text{ м.}$$

Шырақтар арасындағы ең қолайлы қашықтық төмендегі өрнекпен анықталады:

$$z = \lambda \cdot h, \quad (4.2)$$

$$z = 1.2 \cdot 2 = 2.4 \text{ м.}$$

Бөлме шырақ қатарларының ара қашықтығы 2.4м.

$$n = B/L \quad (4.3)$$

мұнда  $B$  – бөлме ені,  $B = 4$ м;

$L$  – шырақтар арасындағы қашықтық,  $L = 2.4$ м.

Осыдан:

$$n = \frac{4}{2,4} = 1.67 \approx 2$$

Шырақтардың қатарлар санын санайық:

$$i = \frac{S}{h^*(A+B)} \quad (4.4)$$

мұндағы  $A$  – бөлме ұзындығы;

$B$  – бөлменің ені;

$S = A \cdot B = 32 \text{ м}^2$  – ауданы.

$$i = \frac{32}{2 \cdot (8+4)} = 1.3$$

Кесте бойынша пайдалану коэффициенті  $\eta = 45\%$ , қор коэффициенті  $K_k = 1.5$ .

Қажетті шамдар санын анықтаймыз:

$$N = \frac{E \cdot K_k \cdot S \cdot Z}{n \cdot \Phi_l \cdot \eta} \quad (4.5)$$

мұндағы  $E$  – берілген жарықтық;

$K_k$  – қор коэффициенті;

$S$  – бөлменің ауданы,  $S = 32 \text{ м}^2$ ;

$Z$  – жарықтандырудың бірқалыпсыздық коэффициенті.  $Z = 1.1 - 1.2$ .

$$N = \frac{300 \cdot 1.5 \cdot 32 \cdot 1.2}{2 \cdot 2340 \cdot 0.45} \approx 8 \text{ шам}$$

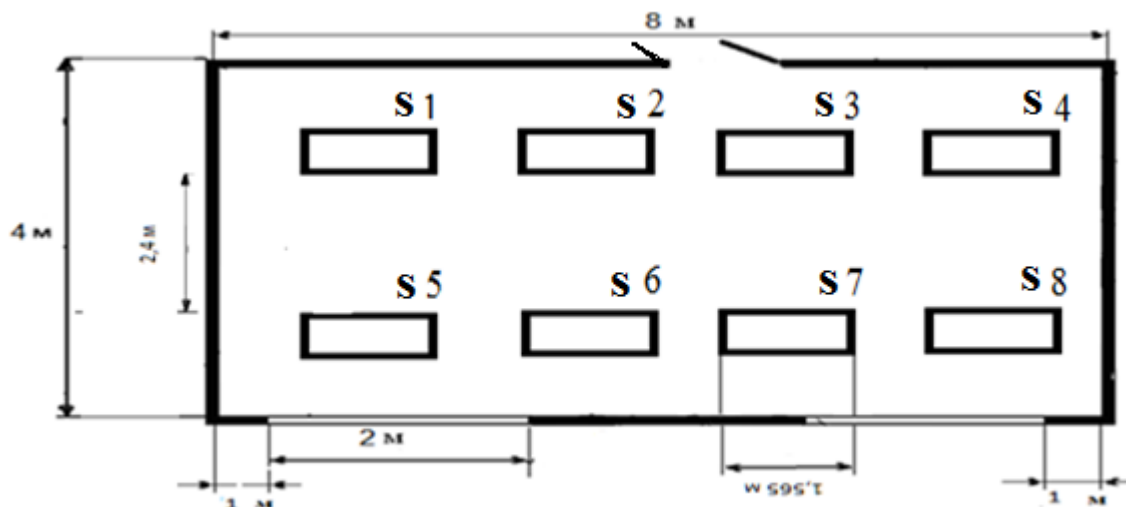
Шыққан есептеулерге байланысты 300 Вт қуатты, жалпы қолдануға арналған, Г қыздыру шамын таңдаймыз.

Барлығы 300 лк нормативтік жарықты құру үшін 300 Вт қуатты Г типті 8 қыздыру шамы қажет. Бір қыздыру шамы бар УСП-35 типті шырақтың ұзындығы  $L_{CB} = 1565 \text{ мм}$ , ал олардың толық ұзындығы мынаған тең:

$$L = N_0 \cdot L_{CB} \quad (4.6)$$

$$L = 2 \cdot 1.65 = 3.13 \text{ м.}$$

Яғни, 8 м ұзындықты бөлмеге екі қатарға төрт шырақтан орналастырамыз, 4.2–суретте келтірілген.



Сурет 4.2 – Жасанды жарықтандыру сұлбасы

#### 4.7 Меншікті қуат әдісі

Меншікті қуат әдісімен жарықтандыруды шешудің маңызы, шырақтардың түрі мен оны орнатудың орнына байланысты, жұмыс бетінен ілгіштің биіктігі, горизонтальды беттегі жарықтандыру мен бөлменің ауданымен меншікті қуат мәні анықталаумен қорытындыланады.

Қыздыру шамдарын орнатуды қабылдаймыз және параллель екі қатарға төртеуден орналастырамыз.

$$N = 2 \cdot 4 = 8 \text{ шам.}$$

Шырақтың меншікті қуаты  $\omega = 40 \text{ Вт/м}^2$ .

Бір шырақтың қуаты:

$$\rho = \frac{\omega \cdot S}{N}, \quad (4.7)$$

$$\rho = \frac{40 \cdot 32}{6} = 160 \text{ Вт.}$$

8 шырақ орнатамыз. Әр шамның қуаты:

$$\rho_g = \frac{160}{2} = 80 \text{ Вт.}$$

Сәйкесінше, келтірілген есептер негізінде 8 шырақты таңдап аламыз, олардың әр қайсысы 80 Вт шамнан тұрады.

#### 4.8 Бөлменің вентиляция жүйесін есептеу

Бөлмеден бір сағатқа шығаруға қажетті ауаның мөлшерін анықтайық  $L$  м<sup>3</sup>/сағ, сонымен бірге келесі формуламен анықталатын артық жылу мөлшерін  $Q_{изб}$  есептеу керек

$$L = \frac{Q_{изб}}{c_s t_{y\epsilon}} \quad (4.6)$$

мұнда  $c_s$  – құрғақ ауаның жылу сыйымдылығы, ккал/кг ( $c_s = 0.24$  ккал/кг град);

$t = t_{yx} - t_{bx}$  есептегенде  $t = 5^\circ\text{C}$  аламыз;

$\rho_v$  – шығарылатын ауаның тығыздығы, температураға байланысты анықталады, кг/м<sup>3</sup> (есептегенде қолданылады  $\rho_v = 1.20$  кг/м<sup>3</sup>).

Артық жылуды анықтаймыз ккал/с:

$$Q_{изб} = Q_n - Q_{от} \quad (4.7)$$

мұнда  $Q_n$  – бөлмедегі ауаға келіп түсетін жылу мөлшері, ккал/с;

$Q_{от}$  – сыртқы қоршаулар арқылы қоршаған ортаға жылу берілуі (жылдың жылы маусымдарында есептеулерде нөлді қабылдауға болады).

Терезе ойықтары арқылы кіретін жылу және жұмыс істейтін адамдар санынан, жабдықтардың қуатына байланысты  $Q_n$  бөлінетін жылу мөлшері

$$Q_n = Q_{ж} + Q_a + Q_p + Q_{бж} \quad (4.8)$$

мұнда  $Q_{өж}$  – өндіріс жабдықтарынан бөлінетін жылу, ккал/сағ;

$Q_a$  – адамдардан бөлінетін жылу, ккал/сағ;

$Q_{кбж}$  – жарық беруші құралдардан бөлінетін жылу;

$Q_p$  – күн радиациясына кіргізілетін жылу, ккал/сағ.

Жұмыс бөлмесіндегі өндіріс жабдықтарынан бөлінетін жылу келесі қатынастармен анықталады

$$Q_{өж} = 860 * P_{жк} * n, \quad (4.9)$$

мұнда 860 жылу эквиваленті 1 кВт\*с, яғни жылу 1 кВт\*с электр энергиясы;

$P_{жк}$  – жабдық қолданатын қуаты,  $P_{жк} = 0.8$  кВт;

$n$  – бөлмеге жылудың таралу коэффициенті,  $n=0.75$ .

$$Q_{ж} = 860 * 0.8 * 0.75 = 516 \text{ ккал/сағ.}$$

Күннен, радиациядан шығатын жылу келесі қатынастармен анықталады

$$Q_p = m * F * g_{ocm}, \quad (4.10)$$

мұнда  $m$  – бөлмедегі терезелер саны;

$F$  – бір терезенің ауданы  $F=2 \text{ м}^2$ ;

$g_{ocm}$  –  $2 \text{ м} \times 2 \text{ м}$  әйнек ауданы арқылы 1 сағат ішінде өтетін жылу мөлшері, яғни әйнек бет арқылы өтетін күн радиациясы.

$g_{ocm}$  мәндері беттердің географиялық бағытына байланысты және терезе сипаттамалары немесе шамдар  $70 - 210$  шектер аралығында қабылдайды. Жұмыс бөлмесінің терезелері қатаң түрде шығыс бағытқа бағытталған, сондықтан  $g_{ocm}$  мәні  $145 \text{ Вт/м}^2\text{C}^\circ$  тең.

$$Q_p = 2 * 2 * 145 = 580 \text{ ккал/с}$$

Адамдардан бөлінетін жылу келесі қатынаспен анықталады

$$Q_l = Q_q \cdot n \quad (4.11)$$

мұнда  $Q_q$  – бір адамнан бөлінетін жылу мөлшері;

$n$  – адам саны.

$$Q_l = 180 * 4 = 720 \text{ ккал/с.}$$

Жарықтандыру құралдарынан бөлінетін жылу

$$Q_{on} = N * N_{on}, \quad (4.12)$$

мұнда  $N$  –  $N = 0,8$  жылуына ауысатын энергия мөлшерін ескеретін коэффициент.

$N_{on}$  – жарықтандыру құралдарының саны.

$$Q_{on} = 0,8 * 8 * 80 = 512 \text{ ккал/с.}$$

Жылу шығару сонда құрайды:

$$Q_{изб} = 516 + 580 + 720 + 512 = 2328 \text{ ккал/ сағ.}$$

Осы сияқты қажетті ауа алмасуы тең болады

$$L = \frac{2328}{0,24 \cdot 5 \cdot 1,2} = 1616,7 \text{ м}^3/\text{сағ.}$$

Бөлменің көлеміне бір сағатқа баспанаға түсетін ауа санының қатынасы ауа алмасу еселігі деп аталады

$$K = \frac{L}{V_n} = \frac{1616,7}{96} = 17 \text{ са.}$$

мұнда  $V_n$  – бөлме көлемі  $V_n = 3 \cdot 4 \cdot 8 = 96 \text{ м}^3$ .

Салқындатқыштың тиісті өнімділігін табамыз

$$W_k = k_3 \cdot L, \quad (4.13)$$

мұнда  $k_3$  – қор коэффициенті,  $k_3 = 1,3, 2,0$ .

$$W_k = 1,8 \cdot 1616,7 = 2910,06 \text{ м}^3/\text{сағ.}$$

Микроклиматтың тиісті параметрлерінің орындалуы үшін  $2910,06 \text{ м}^3/\text{сағ}$  аз болмайтын өнімділікпен жұмыс істейтін салқындатқышты есептеулерге негіздеп жабдықтар тұрған бөлмеге орнату керек. Осы параметрлерді Оңтүстік Кореяда өндірілген Samsung AQ12TFBN салқындатқышы қанағаттандырады және 4.3 – суретте келтірілген.



Сурет 4.3– Samsung AQ12TFBN салқындатқышы

Кондиционердің техникалық сипаттамалары:

- Салқын 9.20 кВт; жылу 15.80 кВт;
- қорек кернеуі 220В, 50 Гц;
- салқынның жұмсайтын қуаты, кВт 1.9;
- жылудың жұмсайтын қуаты, кВт 1.53
- салқын/жылу жұмыс тогы, А 2.3 /3.21;



- жылдық ток пайдалануы 940 кВт\*сағ
- шудың деңгейі, ішкі (жоғ/орт/төм), дБ(А) 40/30/28;
- шудың деңгейі, сыртқы, дБ(А) 47;
- габаритті өлшемдері, Ш/В/Г, ішкі, мм 820 x 285 x 210 ;
- габаритті өлшемдері, Ш/В/Г, сыртқы, мм 660 x 470 x 240 ;
- салмағы, кг 25.

Бөлмеге қажетті ауа мөлшері  $L = 1616.7 \text{ м}^3/\text{ч} = 26.9 \text{ м}^3/\text{мин}$ . Таңдалып алынған салқындатқыш, барлық қажеттіліктерді қанағаттандырады және басқаларымен салыстырғанда бағасы да арзан.

## **5 Жобаның техника– экономикалық негізделуі**

### **5.1 Жоба сипаттамасы**

Қазіргі кезеңде адамзат баласының барлық қызмет салаларында ғылыми-техникалық прогрестің дамуы және сонымен қатар ғылыми-зерттеу жобалары белгілі бір көлемде есептеу техникасының енуіне тәуелді.

Компьютер базасында құрылған ақпараттық жүйелер үлкен көлемді ақпараттарды өңдеуге және ауқымды ақпараттарды өңдеуді қажет ететін салаларда басқаруды шешуге рұқсат етеді. Енді бұл өз кезегінде бағдарламалық қамтамасыз етуді құруды қажет етеді. Дипломдық жобаның негізгі мақсаты Шымкент қаласындағы “Ернұр Пресс” ЖШС баспа орталығының деректер қорын құру қарастырылады. Ақпараттық жүйені MS SQL Server ДҚБЖ (дерекқорларды басқару жүйесінде) жасалған дерекқор және Visual Basic for Applications (VBA), Java бағдарламалық ортасында құрылған клиенттік қосымша құрайды. Жұмыстың нәтижесі оқу орындарында құрылымдық өзгерістерді реттеу үшін пайдаланылады. Сонымен қатар, деректердің қауіпсіздік пен құпиялылық мәселелері қарастырылған.

Жұмысты жазу барысында деректер қорының құрылымын және клиенттік қосымшаны жасау жұмыстары жүргізілген және енгізілген жүйеден алынған экономикалық пайданы есептеу жүргізілген.

Бұл бағдарламада мекеме үшін ақпараттық жүйесін жобалауға өте қолайлы. Бұл интеллектуалды еңбек пен қатар өте көптеген қағаздармен жұмысты қысқартады және экономикалық жағынан да өте тиімді. Сондықтан еңбек етудің негізгі құралы болып – дерекқор орналасқан компьютер есептелінеді. Біз жұмыс орны ретінде жеке кәсіпорынның бөлмесін қарастырамыз. Қарастырылып отырған бөлмеде төрт қызметкер жұмыс істейді, олардың әрқайсысының өз жұмыс орны бар.

Қазіргі таңда, Қазақстандағы ақпараттық технологиялардың дамуына байланысты кез келген оқу орны және кәсіпорындардың құжаттамасы және құжатайналымы қағаз түрінен электрондық түрге ауыса бастады. Электрондық құжаттардың көбісі ресми түрге ауысып, физикалық түпнұсқаларын алмастыруда. Мұндай өзгерістердің құндылығы зор: ұйымдастыру, көшіру және өзгерту техникаларының оңайлылығына қоса, қазіргі заманғы коммуникация желілерін қосатын болсақ, құжатайналым процессінің жылдамдығы артады.

### **5.2 Маркетингтік стратегия**

Маркетингтік стратегияның мақсаты – бағдарламалық қамтамасыз ету арқылы пайда табу. Бағдарламаны енгізу үшін жоспарланып отырған пайданы 5.1–кестеден көре аласыздар.

К е с т е 5.1- Жоспарланып отырған пайда мың.теңге

| Жылдар                             | 2014      | 2015      | 2016       | Барлығы |
|------------------------------------|-----------|-----------|------------|---------|
| Жоспарланған табыс, барлығы,мың.тг | 622952,55 | 1245905,1 | 1868857,65 | 3737715 |

Бағдарламаның сатылым жоспары

Біздің стратегия Шымкент қаласындағы “Ернұр Пресс” ЖШС баспа орталығының деректер қорын құру қарастырылады. Тек біздің бағдарлама толығымен қолданылса және шет ел компанияларымен келісімге келіп, шет мемлекеттерге шығу, нарықтағы өзіндік артықшылығын дамыту. Бұл біздің нақты бағытымызды анықтап және тұрақты тұтынушыларды қамтамасыз етеді. Қазіргі таңда “Ернұр Пресс” ЖШС баспа орталығына осындай бағдарламаның жетіспеушілігі сезінуде.

Бірінші жылы тауарлық белгіні кең түрде тарату және тұрақты тұтынушылардың санын арттыру мақсатында 622952.55 мың қаржы жұмсауды жоспарлап отырмыз.

### 5.3 Бағдарламамен қамтамасыз етудегі еңбек сыйымдылығын есептеу

Еңбек шығыны құрамдасын есептеудегі базалық көрсеткіш мына формуламен есептелінеді:

$$Q = q \times c, \quad (5.1)$$

мұндағы  $Q$  – шартты командалар саны;

$q$  – есеп түріне қарай шартты командалар санын ескеретін коэффициент;

$c$  – бағдарламаның қиындығы мен жаңалығын ескеретін коэффициент.

$$Q = q * c = 1490 * 1.30 = 1937$$

Атап өткен  $q$  коэффициентінің мәнін 1–кестеден (А–қосымшасынан) таңдап алуға болады.

Атап өткен «с» коэффициенті 2-кестеден(А–қосымшасынан) анықталады, ол күрделілік тобы бағанасы мен жаңалықтық дәрежесі бағанасының қиылысуы.

Бағдарламаның күрделілік коэффициенті  $c=1.30$ . Осы әдіспен базалық көрсеткішті табамыз:  $Q = 1937$ .

Ары қарай бағдарламалық өнімді әзірлеуге кететін уақытты есептеу керек.

Жалпы бағдарлама жасауға кеткен уақыт әртүрлі компоненттен тұрады. Бағдарламалық өнімді жасауға кеткен жалпы уақыт құрамы 3–кестеде көрсетілген.

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 сипаттамасы уақыты) келесі формуламен анықталады:

$$T_o = Q \times B / (50 \times K), \quad (5.2)$$

$$T_o = Q \times B / (50 \times K) = 1937 * 1.35 / (50 * 0.8) = 65.37 \text{ адам/сағ.}$$

мұндағы  $B$  – мақсат есебі өзгерісінің коэффициенті,  $B$  коэффициенті мақсат күрделігіне және өзгеріс санына тәуелді – 1.2-ден 1.5–ке дейін (2–кестені қара (А–қосымшасы)).

$K$  – бағдарлама жасаушы білектілігін ескеретін коэффициент.

$K$  коэффициенті мәнін 4–кестеден (А–қосымшасынан) алуға болады.

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

$$T_A = Q / (50 \times K), \quad (5.3)$$

$$T_A = Q / (50 \times K) = 1937 / (50 * 0.8) = 48.425 \text{ адам/сағ.}$$

4)  $T_{bc}$  (блок–сұлба құруға кеткен уақыт)  $T_A$  сияқты 5.3–формуламен есептеледі.

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

$$T_H = Q \times 1.5 / (50 \times K), \quad (5.4)$$

$$T_H = Q \times 1.5 / (50 \times K) = 1937 * 1.5 / (50 * 0.8) = 72.6 \text{ адам/сағ.}$$

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

$$T_{II} = Q / 50, \quad (5.5)$$

$$T_{\Pi} = Q / 50 = 1937 / 50 = 38.74 \text{ адам / сағ.}$$

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

$$T_{OT} = Q \times 4.2 / 50 \times K, \quad (5.6)$$

$$T_{OT} = Q \times 4.2 / 50 \times K = 1937 * 4.2 / 50 * 0.8 = 203.385 \text{ адам/сағ.}$$

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

$$T_D = 24 \text{ адам / сағ.}$$

Еңбек шығындарының сомасы еңбек шығынының құрама сомасы ретінде 5.7–формуламен есептеледі:

$$T = T_{\Pi O} + T_{TO} + T_A + T_{BC} + T_H + T_{\Pi} + T_{OT} + T_D, \quad (5.7)$$

$$T = T_{\Pi O} + T_{TO} + T_A + T_{BC} + T_H + T_{\Pi} + T_{OT} + T_D = 24 + 65.37 + 48.425 + 48.425 + 72.6 + 38.74 + 203.385 + 24 = 524.945 \text{ адам/сағ.}$$

#### 5.4 Бағдарламалық қамсыздандыру шығынының есебі

Бағдарламалық қамсыздандыру шығыны ішіне еңбек ақы шығыны да, еңбек ақидан аударылымдар, амортизациялық және тағы да басқа шығындар кіреді, олар мынандай формуламен анықталады:

$$C = \Phi OT + O_{CH} + A + C_{ЭЭ} + C_{МЖК} + C_{TO} + C_{IP} + C_H \quad (5.8)$$

Еңбек ақы екі жасаушыдан құрылады: негізгі еңбек ақы және қосымша еңбек ақы сомасы (немесе еңбек ақы қоры, ЕАҚ) негізгі еңбек ақы және қосымша еңбек ақы сомасы мына формуламен есептеледі:

$$\Phi OT = Z_{осн} + Z_{доп} \quad (5.9)$$

Негізгі еңбек ақы төмендегідей анықталады:

$$Z_{осн} = T \times TC / t_{орт}, \quad (5.10)$$

$$Z_{осн} = T \times TC / (t_{орт} \times 8) = 524.945 * 2750 * 2.25 / 21 = 154671.295 \text{ тг}$$

Қосымша еңбек ақы негізгі еңбек ақының 20 % құрайды және келесі формуламен есептеледі;

$$З_{\text{доп}} = 0.2 \times З_{\text{осн}}, \quad (5.11)$$

$$З_{\text{доп}} = 0.2 \times З_{\text{осн}} = 0.2 * 154671.295 = 30934.26 \text{ тг}$$

Жалпы еңбек ақысы (еңбек төлеу фонды) негізгі және қосымша еңбек ақының қосындысы ретінде анықталады (9):

$$\text{ФОТ} = З_{\text{осн}} + З_{\text{доп}} = 154671.295 + 30934.26 = 185605.555 \text{ тг.}$$

Әлеуметтік салық ЕАҚ 11 % құрайды (ҚР СК 358 б. 1-тарау) жұмыскердің табысынан, мынандай формуламен есептеледі [1] (А-қосымшасы):

$$О_{\text{сн}} = (\text{ФОТ} - ЗА) \times 11\%, \quad (5.12)$$

$$О_{\text{сн}} = (\text{ФОТ} - ЗА) \times 11\% = (185605.555 - 18560.555) * 0.11 = 18374.95 \text{ тг}$$

мұндағы ЗА – зейнетақы аударылымдар, ЕАҚ-нан 10% құрайды және әлеуметтік салықпен міндеттелмейді [2]:

$$ЗА = \text{ЕАҚ} * 10\%, \quad (5.13)$$

$$ЗА = \text{ЕАҚ} * 10\% = 185605.555 * 0.1 = 18560.5555 \text{ тг}$$

Амортизациялық аударылымдар амортизацияның тағайынды шамаларымен орындалады, пайыздармен жабдықтың баланстық құнына және мына формуламен есептеледі:

$$A = \frac{B_{\text{бас}} \times A_{\text{ш}} \times N}{100 \times 12 \times t} \quad (5.14)$$

Амортизация шамалары ( $A_{\text{ш}}$ ), мына формуламен есептеледі:

$$H_A = \frac{B_{\text{бас}} - K_{\text{тар}}}{T_{\text{норм}} \cdot B_{\text{бас}}} \times 100\%, \quad (5.15)$$

$$H_A = \frac{B_{\text{бас}} - K_{\text{тар}}}{T_{\text{норм}} \cdot B_{\text{бас}}} \times 100\% = \frac{88080 - 4404}{4 * 88080} * 100\% = 23.75\%$$

мұндағы  $K_{\text{тар}}$  – таратылым құны, жабдықтың құнынан 5% құрайды (нұсқа бойынша);  $K_{\text{тар}} = 88080 * 0,05 = 4404 \text{ тг.}$

Құны  $B_{\text{бас}} = 88080$  тг.

Компьютердің қазіргі уақыттағы құны 65160 тг [3].

Принтердің құны 22920 тг [4].

Дербес компьютерде жалпы жұмыс істеу уақыты мына формуламен есептеледі:

$$T = T_A + T_{\text{БС}} + T_H + T_{\text{П}} + T_{\text{ОТ}}, \quad (5.16)$$

$$T = T_A + T_{\text{БС}} + T_H + T_{\text{П}} + T_{\text{ОТ}} = 48.425 + 48.425 + 72.6 + 38.74 + 203.385 = 411.575 \text{ адам/сағ.}$$

Амортизациялық аударылымдар(14):

$$A = \frac{B_{\text{бас}} \times A_{\text{и}} \times N}{100 \times 12 \times t} = \frac{88080 \times 0.2375 \times 21.87}{100 \times 12 \times 411.575} = 0.9263$$

Электрэнергия шығындары мына формуламен есептеледі:

$$C_{\text{ЭЭ}} = K \times k_3 \times T \times C_{\text{кВт-сағ}}, \quad (5.17)$$

$$C_{\text{ЭЭ}} = K \times k_3 \times T \times C_{\text{кВт-сағ}} = 450 \times 0.8 \times 19.37 \times 8 = 55.795 \text{ тг}$$

1 кВт-сағ электрэнергиясының құны 19.37 тг [5].

$$A_{\text{жыл}} = B_{\text{бас}} \cdot \frac{H_A}{100\%} = 88080 \cdot \frac{23.75\%}{100\%} = 20919 \text{ тг}$$

Материалдар мен көмекші бөлшектер шығыны, бағдарламалық өнімді жазу барысында қолданылды ( $C_{\text{МжК}}$ ), сонымен қатар техникалық қызмет көрсету шығыны ( $C_{\text{ТО}}$ ), жабдықтың құнынан 1.5% және 2.5% құрайды және мына формулалар мен есептеледі (18 – 19):

$$C_{\text{МжК}} = 0.015 \times C_{\text{обор}}, \quad (5.18)$$

$$C_{\text{МжК}} = 0.015 \times C_{\text{обор}} = 0.015 \times 88080 = 1321.2 \text{ тг}$$

$$C_{\text{ТО}} = 0.025 \times C_{\text{обор}}, \quad (5.19)$$

$$C_{\text{ТО}} = 0.025 \times C_{\text{обор}} = 0.025 \times 88080 = 2202 \text{ тг}$$

Басқару мен қызмет көрсетуге байланысты үстеме шығындар, сондай-ақ жабдықты пайдалану кезіндегі және де кәсіпорын үдерістері мен айналымдарынан қосымша шығындар еңбек ақы қорынан 50% құрайды және де мына формуламен есептеледі:

$$C_H = 0.5 \times \text{ЕАҚ}, \quad (5.20)$$

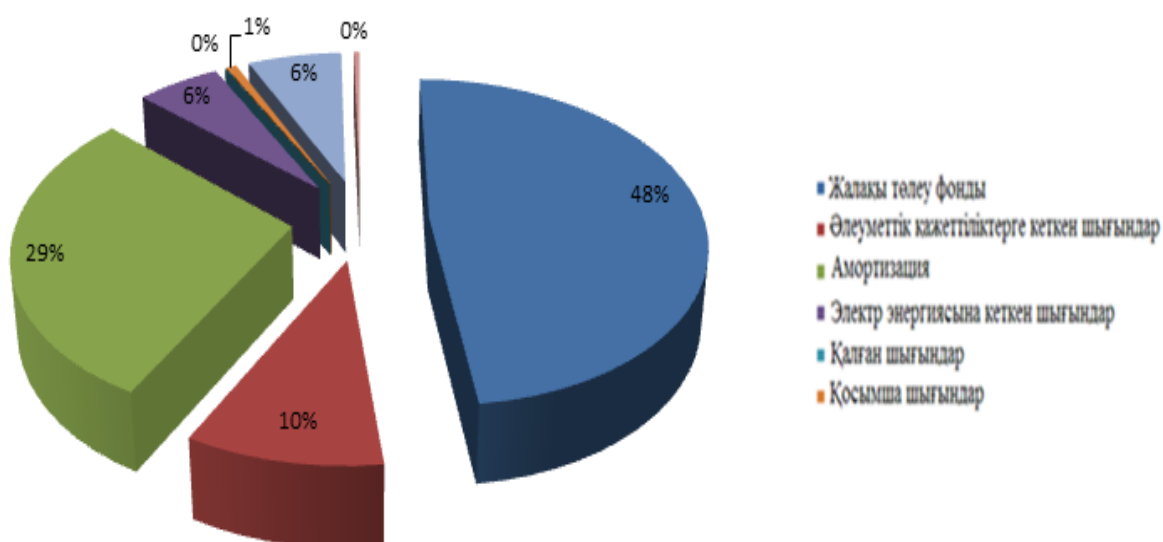
$$C_H = 0.5 \times \text{ЕАҚ} = 0.5 * 185605.555 = 92802.78 \text{ тг}$$

Бағдарламалық өнімнің өзіндік құнының есебінің жиынтық нәтижелерін кесте түрінде ұсыну керек, шығын статьясын атап, және оның ортақ құндағы сыбағаларын пайызбен есептеп, бағдарламалық қамсыздандырудың өзіндік құнының құрылымының диаграммасын сызу керек. Өзіндік құнның қорытынды 5.2–кестеде көрсетілген.

К е с т е 5.2– Өзіндік құнның қорытынды кестесі

| Шығын бабы атауы                           |                   | Сомасы, теңге | Әр баптың үлесі, % |
|--|-------------------|---------------|--------------------|
| ЕАҚ  | ЕА <sub>нег</sub> | 154671,295    | 48,14              |
|  | ЕА <sub>қос</sub> | 30934,26      | 9,63               |
| Үстеме шығындар, С <sub>Накл</sub>         |                   | 92802,78      | 28,88              |
| Әлеуметтік салық шығыны, ӘС                |                   | 18374,95      | 5,72               |
| Пайдалану шығындары                        | С <sub>ээ</sub>   | 55,795        | 0,017              |
|  | С <sub>то</sub>   | 2202          | 0,685              |
|  | А <sub>жыл</sub>  | 20919         | 6,51               |
| Материалдар және көмекші, С <sub>Мжк</sub> |                   | 1321,2        | 0,411              |
| Барлығы:                                   |                   | 321281,28     | 100                |

Өзіндік құн нәтижесінің диаграммасы 5.1–суретте көрсетілген.



Сурет 5.1– Өзіндік құн нәтижесінің диаграммасы



Бағдарламалық өнімді жүзеге асыру бағасы оның құны мен таза кірістің қосындысынан тұрады:

$$\text{Ц} = \text{С} + \text{П} \quad (5.21)$$

мұндағы, С – өнім бағасы;

П – таза кіріс.

Бастапқы бағаны анықтауда бағдарламалық өнімді жүзеге асыру үшін керекті рентабельдік деңгейін анықтау қажет (20%):

$$\text{Ц}_\text{п} = \text{С} \cdot \left(1 + \frac{\text{Р}}{100}\right), \quad (5.22)$$

мұндағы, Р – рентабельдік (20%).

$$\text{Ц}_\text{п} = 321281.28 \cdot (1 + 20/100) = 385537.536 \text{ тг}$$

Бағдарламаның орындалу бағасы келесі формула арқылы табылады.

$$\text{Ц}_\text{р} = \text{Ц}_\text{п} + \text{С} \quad (5.23)$$

Қазіргі таңда ҚР-да ҚҚС 12% құрайды:

$$\text{С} = \text{Ц}_\text{п} \cdot 12\%, \quad (5.24)$$

$$\text{ҚҚС} = 385537.536 \cdot 12\% = 46264.50432 \text{ тг}$$

$$\text{Ц}_\text{р} = 385537.536 + 46264.50432 = 431802.04032 \text{ тг.}$$

### **5.5 Бағдарлама өнімін сатып алуға кеткен бір жолғы шығындар есебі**

Бағдарлама өнімін сатып алуға және оны өндіріске енгізу шығындары келесі шығындардан тұрады:

$$\Sigma \text{З} = \text{С}_\text{С} + \text{С}_\text{ТР} + \text{С}_\text{О} \quad (5.25)$$

Бағдарламаның бағасы:  $\text{С}_\text{С} = 431802.04032 \text{ тг.}$

$\text{С}_\text{ТР}$  – көлік шығыны, жүйе құнынан –25 %, мың тенге;

$$\text{С}_\text{ТР} = 0.25 \cdot \text{С}_\text{С} = 431802.04 \cdot 0.25 = 107950.51 \text{ тг}$$

Жүйе құны үстінде есептелінді, ал қалғандары келесі түрде есептеледі. Өнімді игеруге деген маманды оқыту шығыны, оқытуға кеткен уақыт пен оған деген консалтингті фирмадағы мөлшерлемеден тұрады:

$$C_o = T \times C_{оп}, \quad (5.26)$$

$$C_o = T \times C_{оп} = 4 * 8 * 2600 = 83200 \text{ тг}$$

мұндағы  $T$  – оқытуға кеткен уақыт, сағ;

$C_{оп} = 2600$  тг. Себебі қазіргі таңда сағатына 2500–3000 тг аралығында.

$$\Sigma Z = C_c + C_{тр} + C_o = 431802.04 + 107950.51 + 83200 = 622952.55 \text{ тг}$$

Бағдарлама өнімін сатып алуға кеткен бір жолғы шығындар есебін 5.3-кестеге келтіру керек. Бұл кестеден біз шығын бабын көре аламыз. Жоғарыда есептеген шығындардың қосындысынан тұрады. Шыққан құн біздің бағдарламаның құны болып есептеледі.

**К е с т е 5.3 – Ақпарат жүйелерін енгізуге керекті бір жолымғы шығындар есебінің жиынтығы**

| Шығын бабы атауы           | Сомасы, мың тенге |
|----------------------------|-------------------|
| Жүйенің құны               | 431802,04         |
| Көлік шығыны               | 107950,51         |
| Жүйені оқуға кеткен шығыны | 83200             |
| Барлығы:                   | 622952,55         |

### 5.6 Игеру саласындағы жылдық бір жолғы шығындар есебі

Ақпараттық технологияларын қолданған кездегі жылдық шығындары келесі баптардан тұрады:

- жылдық ЕАҚ;
- әлеуметтік салық аударым;
- басқа да шығындар.

Осының барлығысын енді формула арқылы жазайық:

$$C_э = 3П + O_{сн} + П_p, \quad (5.27)$$

$$C_э = 3П + O_{сн} + П_p = 28224000 + 2794176 + 8467200 = 39485376 \text{ тг}$$

АТ енгізгеннен кейінгі мамандардың жылдық еңбек ақы шығындары келесі формуламен есептеледі:

$$ЗП = (O_C \times t \times K_p) \times Ч \times (1 + K_d), \quad (5.28)$$

$$ЗП = (O_C \times t \times K_p) \times Ч \times (1 + K_d) = (3000 * 8 * 245) * 4 * (1 + 0.2) = 28224000 \text{ тг}$$

$$O_{CH} = (\text{ФОТ} - 3A) \times 11\% = (28224000 - 2822400) * 0.11 = 2794176 \text{ тг}$$

$$3A = \text{ЕАҚ} * 10\% = 28224000 * 0.1 = 2822400 \text{ тг}$$

Басқа да шығындар – материалға деген шығындар, үстеме шығындар жылдық еңбек ақы шығындарынан 30 % құрайды және келесі формуламен есептеледі:

$$Пр = ЗП \times 0.3, \quad (5.29)$$

$$Пр = ЗП \times 0.3 = 28224000 * 0.3 = 8467200 \text{ тг}$$

Ақпараттық технологияларын қолданған кездегі жылдық бір жолғы шығындарды келесі кестеге сомасын және әр баптарын көрсетіп толтырыңыз (5.4 кестені қара).

К е с т е 5.4 – Ақпараттық технологияларын қолданған кездегі жылдық бір жолғы шығындар

| Шығын баптары атауы      | Сомасы, мың теңге |
|--------------------------|-------------------|
| Жылдық ЕАҚ               | 28224000          |
| Әлеуметтік салық аударым | 2794176           |
| Басқа да шығындар        | 8467200           |
| Барлығы:                 | 39485376          |

### 5.7 Ақпараттық жүйе енгізуден үнем мен табыс мөлшерінің есебі

Еңбек өнімділігі өсуінен түскен үнемділікті келтірінді шығындардың базалық ( $C_0$ ) және ұсынылған ( $C_1$ ) нұсқалар айырмасы ретінде шығарып алуға болады.

$$\Delta_{yt} = C_0 - C_1, \quad (5.30)$$

$$\Delta_{yt} = C_0 - C_1 = 59228064 - 39485376 = 19742688 \text{ тг}$$

Ұйымда жүйе енгізілмей тұрғанда (қол еңбегімен айналысқанда) жылдық келтірінді шығындар келесі баптардан тұрады:

- еңбек ақы қоры;
- әлеуметтік салық;

– басқада шығындар.

Осының барлығысын енді формула арқылы жазайық:

$$C_{\Sigma} = ЗП + O_{CH} + П_p, \quad (5.31)$$

$$C_{\Sigma} = ЗП + O_{CH} + П_p = 28224000 + 2794176 + 8467200 = 39485376 \text{ тг}$$

АТ енгізгеннен кейінгі мамандардың жылдық еңбек ақы шығындары келесі формуламен есептеледі:

$$ЗП = (O_c \times t \times K_p) \times Ч \times (1 + K_d), \quad (5.32)$$

$$ЗП = (O_c \times t \times K_p) \times Ч \times (1 + K_d) = (3000 * 8 * 245) * 6 * (1 + 0.2) = 42336000 \text{ тг}$$

$$O_{CH} = (\text{ФОТ} - ЗА) \times 11\% = (42336000 - 4233600) * 0.11 = 4191264 \text{ тг}$$

$$ЗА = \text{ЕАҚ} * 10\% = 42336000 * 0.1 = 4233600 \text{ тг}$$

Басқа да шығындар – материалға деген шығындар, үстеме шығындар жылдық еңбек ақы шығындарынан 30 % құрайды және келесі формуламен есептеледі:

$$П_p = ЗП \times 0.3, \quad (5.33)$$

$$П_p = ЗП \times 0.3 = 28224000 * 0.3 = 12700800 \text{ тг}$$

К е с т е 5.4 –Ұйымда жүйе енгізілмей тұрғанда жылдық бір жолғы шығындар

| Шығын баптары атауы      | Сомасы, мың теңге |
|--------------------------|-------------------|
| Жылдық ЕАҚ               | 42336000          |
| Әлеуметтік салық аударым | 4191264           |
| Басқа да шығындар        | 12700800          |
| Барлығы:                 | 59228064          |

## 5.8 Салыстырмалы экономикалық тиімділіктің көрсеткіштерін есептеу

Нормативтік күрделі қаржы салымының экономикалық тиімділігінің коэффициенті келесі формуламен есептеледі:

$$E_n = \frac{1}{T_n} \quad (5.34)$$

$$E_n = \frac{1}{T_n} = 0.33$$

мұндағы  $T_n$  – нормативтік күрделі қаржы салымын өтелу мерзімі, жыл.  
( $T_n = 1, 2, 3 \dots n$ ) бағдарлама өнімдерінің өтеу мерзімі 4 жыл.

Есептік күрделі қаржы салымының экономикалық тиімділігінің коэффициенті:

$$E_p = \frac{\mathcal{E}_{y^2}}{K}, \quad (5.35)$$

$$E_p = \frac{\mathcal{E}_{y^2}}{K} = \frac{19742688}{39485376} = 0.5$$

Есептік күрделі қаржы салымын өтелу мерзімі:

$$T_p = \frac{1}{E_p}, \quad (5.36)$$

$$T_p = \frac{1}{E_p} = \frac{1}{0.5} = 2$$

Дисконтты есептемегенде өтелу мерзімі 2 жыл.

Бағдарлама өнімін енгізудің салыстырмалы экономикалық тиімділігінің көрсеткіштерін 5.5 кестеден көре аламыз.

К е с т е 5.5 – Бағдарлама өнімін енгізудің салыстырмалы экономикалық тиімділігінің көрсеткіштері

| Көрсеткіштер атауы  | Мәні     |
|---|----------|
| Шартты жылдық шығынды үнемдеу, мың тенге                                  | 19742688 |
| Күрделі қаржы салымының экономикалық тиімділігінің коэффициенті ( $E_p$ ) | 0.5      |
| Күрделі қаржы салымын өтелу мерзімі ( $T_p$ ), жыл                        | 2        |

### 5.9 Динамикалық көрсеткіштер негізінде жобаны өткізуде экономикалық тиімділігін бағалау

Таза дисконттық табыс (ЧДД) келесі формуламен есептеледі:

$$\text{ЧДД} = \sum_{t=0}^T (P_t - 3_t) \times \frac{1}{(1+E)^t} - K \quad (5.37)$$

Есеп аралығы – есептеу кезеңіндегі уақыттың бір бөлігі (жыл, квартал, ай және т. б.).

$$T_{OK} = t + \frac{K - (d\Pi_1 + d\Pi_2)}{d\Pi_3} = 2 + \frac{39485376 - 30162440}{11425167} = 2.816 = 2\text{жыл}10\text{ай}$$

Егер инвестициялық жобадан таза дисконттық табыс (ЧДД) оң болса, жоба тиімді (берілген дисконттау мөлшерінде).

$E$  – тұрақты дисконт мөлшері, 20 % [6].

Онда барлық есеп шегін ескергендегі таза дисконттық табыс сомасы:

$$\begin{aligned} \text{ЧДД} &= \mathcal{E}_1 \frac{1}{1+E} + \mathcal{E}_2 \frac{1}{(1+E)^2} + \mathcal{E}_3 \frac{1}{(1+E)^3} - K = \frac{19742688}{1.2} + \frac{19742688}{1.2^2} + \frac{19742688}{1.2^3} - 39485376 = \\ &= 2102230.67\text{тг} \end{aligned}$$

Таза дисконттық табыстың оң мәні ЧДД > 0, инвестицияның тиімділігін көрсетеді, ақпараттық жүйелер керекті көлемде пайда алып келеді.

Табыстық индексі (ИД) келтірілген әсерлердің сомасының күрделі қаржы салымына қатынасы. Ол келесі формуламен есептеледі:

$$\text{ИД} = \frac{1}{K} \sum_{t=1}^T (P_t - Z_t) \cdot \frac{1}{(1+E)^t}$$

мұндағы  $K$  – күрделі қаржы салымы немесе инвестицияның құны.

$$\text{ИД} = \frac{41587606,67}{39485376} = 1,05.$$

Егер ИД > 1, жоба тиімді, егер ИД < 1 – тиімді емес.

Ішкі табыстық мөлшері (ВНД немесе IRR) - келтірілген әсер, келтірілген күрделі қаржы салымдарға тең болғандағы дисконттау мөлшерін ( $r_{\text{вн}}$ ) көрсетеді.

Бұл көрсеткіш  $r_{\text{вн}}$  (ВНД) келесі теңдеуден шығады:

$$\sum_{t=0}^T \frac{P_t - Z_t}{(1+r_{\text{вн}})^t} = \sum_{t=0}^T \frac{K_t}{(1+r_{\text{вн}})^t}. \quad (5.38)$$

Егер  $r_{\text{вн}}$  мәні инвесторға керек табыс мөлшері салынған капиталдан көп не Ішкі табыстық мөлшері (ВНД):

$$E_1 = 0,20.$$

$$ЧДД_1 = \frac{19742688}{1,2} + \frac{19742688}{1,2^2} + \frac{19742688}{1,2^3} - 39485376 = 2102230,67 \text{ тг};$$

$$E_2 = 0,25.$$

$$ЧДД_2 = \frac{19742688}{1,25} + \frac{19742688}{1,25^2} + \frac{19742688}{1,25^3} - 39485376 = -947649,024 \text{ тг};$$

$$E_1 \rightarrow ЧДД_1 > 0;$$

$$E_2 \rightarrow ЧДД_2 > 0;$$

$$IRR = 0,20 + \frac{2102230,67}{2102230,67 + 947649,024} \cdot (0,25 - 0,20) = 0,234 .$$

Сонымен, дисконттау мөлшері келесі аралықта 20 %...23,44 %.

IRR=23.44% Егер мына шарт орындалса, формула дұрыс:  $r_a < IRR < r_b$  және  $NPV_a > 0 > NPV_b$ . Егерде осы шарт орындалса салып жатқан ақшамызды қайтарып ала аламыз.

К е с т е 5.6 – Бағдарлама өнімін әзірлеуінің және енгізуінің экономикалық пайдалылығының көрсеткіштері

| Көрсеткіштер атауы   | Мәні       |
|--|------------|
| Бағдарлама өнімін әзірлеуге және енгізуге шығын, мың теңге     | 39485376   |
| Бағдарлама өнімін енгізгеннен кейінгі болжалды үнем, мың теңге | 19742688   |
| Таза дисконттық табыс, мың теңге                               | 2102230,67 |
| Табыстық индекс  | 1,05       |
| Ішкі табыстық мөлшері  | 0,234      |
| Дисконтталған өтелу мерзімі, жыл                               | 2жыл10ай   |
| Моральдық ескіру мерзімі, жыл                                  | 3          |

Ақшаның өтелу мерзімін тауып алдық. Енді ақшалар қозғалысын кесте арқылы көрсетуімізге болады. Ақшалар қозғалысын 5.7–кестеден көре аласыздар. Мұнда біз 4 жыл қатарынан алдық 2017 жылға дейін. Осы жылдар

аралығындағы ақшалардың қозғалысын қарауға болады. 1-2 жылдар салынған ақша өтелмейді. Тек 3-ші жылдан бастап пайда келе бастайды.

Ақшаның жылжуы (қозғалысы)

К е с т е 5.7 – Ақшаның жылжуы

| Атауы  | Жылдар    |            |            |             |
|--|-----------|------------|------------|-------------|
|  | 2014      | 2015       | 2016       | 2017        |
| Бірреттік шығын,кВ<br>мың,теңге              | 39485376  |            |            |             |
| Үнемдеуден шыққан<br>пайда                   |           | 19742688   | 19742688   | 19742688    |
| Пайдаға салық                                |           | 3948537,6  | 3948537,6  |             |
| Таза пайда                                   | -39485376 | 15794150,4 | 15794150,4 | 15794150,4  |
| Ағымдағы шығын                               |           | 39485376   | 39485376   |             |
| Дисконттау еселігі<br>(20%<br>мөлшерлеменен) | 1         | 0,83       | 0,694      | 0,58        |
| Таза дисконтталған<br>пайда                  | -39485376 | 16452240   | 13710200   | 11450759,04 |
| Жалпы өсім<br>бойынша ТДП                    | -39485376 | -23033136  | -9322936   | 2127823,04  |



## Қорытынды

Қазіргі таңда, Қазақстандағы ақпараттық технологиялардың дамуына байланысты кез келген оқу орны және кәсіпорындардың құжаттамасы және құжатайналымы қағаз түрінен электрондық түрге ауыса бастады. Электрондық құжаттардың көбісі ресми түрге ауысып, физикалық түпнұсқаларын алмастыруда. Мұндай өзгерістердің құндылығы зор: ұйымдастыру, көшіру және өзгерту техникаларының оңайлылығына қоса, қазіргі заманғы коммуникация желілерін қосатын болсақ, құжатайналым процессінің жылдамдығы артады.

Бұл дипломдық жобада барлық қойылған мақсаттар іске асырылды. Шымкент қ."Egnet Press" ЖШС баспа орталығы жұмысының деректер қоры құрылды. Дерекқорды құру барысында барлық нюанстар ескерілді.

Дерекқор MS SQL Server дерекқорларды басқару жүйесінде жасалды, жаңа түскен ақпаратқа байланысты деректерді тек администратор енгізе алады. Мұндай қатынау принципі пайдаланушылар үшін онша ыңғайлы емес, сондықтан олар үшін Visual Studio бағдарламалық ортасында клиенттік қосымша жасалды. Сонымен қатар, авторизация қолдану арқылы деректердің қауіпсіздік пен құпиялылық мәселелері қарастырылды.

Бұл бағдарламалық өнімді кез келген жоғары оқу орнында пайдалануға болады.

Тіршілік қауіпсіздігі бөлімінде өндірістік жарықтануды есептеудің нәтижесінде бөлмеде 80 Вт қуатты, 2 люминесцентті ЛХБ шамдары бар 8 ЛПО-02 типті шырақ қондырылды. Табиғи жарықтануды нормаға сәйкес келмейді, сондықтан күндіз қосымша жасанды жарықтануды қолдану ұсынылады. Ауа баптау жүйесін есептеудің нәтижесінде бөлмеде Samsung AQ12TFBN жүйесі кондиционері орнатылды.

Техника-экономикалық негізделуі бөлімінде жасалған есептеулер нәтижесінде бағдарламалық өнімді сатудан түскен таза пайда жасалған шығындардан 2 есе артық болып шықты.

Студенттер және мұғалімдерге арналған электрондық веб-журнал – оқу орындарына арналған үлгерімді есептеу жүйесі. Ол әкімшілік және мұғалімдер үшін оңтайлы қамтама.

Дипломдық жобада жасалған бағдарлама баспа орталықтың жұмысын басқаруға арналған. Жаңа технологияларды пайдалана отырып, жұмыс жасауды жеңілдеттік. Пайдаланылған технологиялар мен бағдарламалар тиімді іске асырылды. Нәтижесінде жұмыс қолданушыға тиімді, қызмет көрсету уақытының үнемділігі қамтамасыз етілді. Жұмыс барысында мүмкін болатын қателерді азайтатын, ыңғайлы әрі уақытты үнемді пайдалануға мүмкіндік беретін бағдарлама құру мақсатымыз орындалды. Бағдарламаны құру қиын әрі ұзақ уақытты қажет етуіне қарамастан баспа орталығы жұмысына таптырмас

бағдарлама болып табылады. Дипломдық жобада барлық қойылған мақсаттар іске асырылды.

### **Пайдаланылған әдебиеттер тізімі**

- 1 Никита Культин. Основы программирования в Delphi 2010. – Санкт-Петербург, 2010. – 448с.
- 2 Алексей Вишнеvский. SQL Server . Эффективная работа. – Санкт-Петербург, 2009. – 541с.
- 3 Никита Культин. Основы программирования в Delphi 2010. 2-е издание. – Санкт-Петербург, 2010. – 640с.
- 4 Полякова Л.Н. Основы SQL. Курс лекций: учеб. пособие. – Москва, 2004. – 368с.
- 5 Глушаков С.В., Ломотько Д.В. Базы данных: Учебный курс. – М.: ООО «Издательство АСТ», 2000. – 504с.
- 6 Ибраева Л.К., Сатимова Е.Г. Проектирование баз данных. МУ к выполнению лабораторных работ. – Алматы: АИЭС, 2004.
- 7 Айтхожаева Е.Ж. Стандартный язык баз данных SQL: Учебное пособие. – Алматы: АИЭС, 2005. - 48 с.
- 8 Дунаев В.В. Базы данных. Язык SQL. – СПб.: БХВ – Петербург, 2006.
- 9 Кандзюба С.П., Громов В.Н. Delphi 6/7. Базы данных и приложения. – СПб: ООО «ДиаСофт», 2002.
- 10 Хакиmжанов Т.Е. Расчет аспирационных систем. Дипломное проектирование. Для студентов всех форм обучения всех специальностей. – Алматы: АИЭС, 2002. – 30 с.
- 11 Ломакин В.К. Мировая экономика: Учебник для вузов. – М.: Финансы, ЮНИТИ, 2008. – 727 с.
- 12 Борисов, Е. Ф. Основы экономики: Учебное пособие / Е. Ф. Борисов. – М.: Юрайт – Издат, 2009. – 316 с.
- 13 Современная экономика: Учебное пособие /Под ред. О. Ю. Мамедова. – Ростов-на-Дону: Феникс, 2011. – 456 с.
- 14 Экономика: Учебник / Под ред. Р. П. Колосовой. – М.: Норма, 2011. – 345 с.
- 15 Экономика: Учебное пособие /Под ред. А.С. Булатова. – М.: Юристь, 2009. – 896 с.

## А қосымшасы

### Экономикалық бөлімге қажетті кестелер

К е с т е 1 – q коэффициентінің мәні

| Тапсырма түрлері                | Коэффициенттің өзгеру аралығы |
|---------------------------------|-------------------------------|
| Есептеу тапсырмалары            | 1400 ден 1500                 |
| Оперативті басқару тапсырмалары | 1500 ден 1700                 |
| Жоспарлау тапсырмалары          | 3000 ден 3500                 |
| Көп вариантты                   | 4500 ден 5000                 |
| Комплекстік тапсырма            | 5000 ден 5500                 |

К е с т е 2 – Еңбек сыйымдылығын есептейтін коэффициент

| Бағдарлама тілі | Күрделілік тобы | Жаңалықтық дәрежесі |      |      |      | B коэффициенті |
|-----------------|-----------------|---------------------|------|------|------|----------------|
|                 |                 | A                   | Б    | В    | Г    |                |
| Жоғарғы деңгей  | 1               | 1,38                | 1,26 | 1,15 | 0,69 | 1,2            |
|                 | 2               | 1,30                | 1,19 | 1,08 | 0,65 | 1,35           |
|                 | 3               | 1,20                | 1,10 | 1,00 | 0,60 | 1,5            |
| Төменгі деңгей  | 1               | 1,58                | 1,45 | 1,32 | 0,79 | 1,2            |
|                 | 2               | 1,49                | 1,37 | 1,24 | 0,74 | 1,35           |
|                 | 3               | 1,38                | 1,26 | 1,15 | 0,69 | 1,5            |

К е с т е 3 – Бағдарламалық өнімді жасауға жалпы уақыт құрамы

| Кезең № | Дәл кезеңдегі уақыт белгісі | Кезеңнің мазмұны   |
|---------|-----------------------------|--|
| 1       | T <sub>по</sub>             | Мақсат сипатын дайындау  |
| 2       | T <sub>о</sub>              | Мақсат сипаттамасы   |
| 3       | T <sub>а</sub>              | Алгоритм құру  |
| 4       | T <sub>бс</sub>             | Алгоритмнің блок-схемасын құру                                       |
| 5       | T <sub>н</sub>              | Бағдарламаны ... тілде жазу  |
| 6       | T <sub>п</sub>              | Бағдарламаны теру  |
| 7       | T <sub>оп</sub>             | Бағдарламаны реттеу және тестілеу                                    |
| 8       | T <sub>д</sub>              | Құжаттарды рәсімдеу, пайдаланушыға нұсқаулар және түсіндірмелер жазу |

*А қосымшасының соңы*

К е с т е 4 – Бағдарлама жасаушы білектілігін ескеретін коэффициент

| Жұмыс тәжірибиесі | Білектілік коэффициенті |
|-------------------|-------------------------|
| Екі жылға дейін   | 0.8                     |
| 2-3 жыл           | 1                       |
| 3-5 жыл           | 1.1 – 1.2               |
| 5-7 жыл           | 1.3 – 1.4               |
| 7 жылдан көп      | 1.5– 1.6                |

К е с т е 5 – Бірыңғай тарифтік сеткадан көшірме (БТС)

| Еңбек ақы разряды    | 1   | 2    | 3    | 4    | 5    | 6    | 7    | 8    | 9    | 12   |
|----------------------|-----|------|------|------|------|------|------|------|------|------|
| Тарифтік коэффициент | 1,0 | 1,07 | 1,15 | 1,24 | 1,33 | 1,43 | 1,54 | 1,66 | 1,78 | 2,25 |

## Б қосымшасы

### Жобада қолданылатын бағдарламаның кодтары

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.ComponentModel.Design;
using System.Data;
using System.Data.Linq.SqlClient;
using System.Data.SqlClient;
using System.Drawing;
using System.Linq;
using System.Runtime.InteropServices;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
namespace WindowsFormsApplication1
{
    public partial class Form1 : Form
    {
        private static int doc_id, ssd, td, td1, td2, td3, td4, vpi, vpi2, kg, s3, s6, s7,s8;
        private static DateTime dr, dv;

        public Form1()
        {
            InitializeComponent();
            groupBox2.Visible = false;
            groupBox4.Visible = false;
            button12.Visible = false;
            button21.Visible = false;
            groupBox6.Visible = false;
            groupBox8.Visible = false;
            groupBox10.Visible = false;
            comboBox1.Items.Add("Барлығы");
            var mesto = (from i in Program.db.tochka select i.name_tochka);
            foreach (var i in mesto)
            {
                comboBox1.Items.Add(i);
            }

            var mesto1 = (from i in Program.db.tochka select i.name_tochka);
            foreach (var i in mesto1)
            {
                comboBox2.Items.Add(i);
            }
        }
    }
}
```

## Б қосымшасы жалғасы

```
foreach (var i in type1)
{
    comboBox4.Items.Add(i);
}

comboBox3.Items.Add("Барлығы");
var type2 = (from i in Program.db.type select i.type_name);
foreach (var i in type2)
{
    comboBox3.Items.Add(i);
}

var s_n2 = (from i in Program.db.spisok_3 select i.spisok_name);
foreach (var i in s_n2)
{
    comboBox5.Items.Add(i);
}

a1 = "select v.klient_id, v.fam as 'Тегі', v.name as 'Аты', v.patronimic as 'Әжесінің аты',
v.date_rozhdeniya as 'Туылған күні', v.klient_adres as 'Мекенжайы', v.klient_tel as 'Телефон'
from klient_2 v";
connect();
dataGridView1.Columns["klient_id"].Visible = false;
groupBox1.Enabled = true;
try
{
    dataGridView1.Rows[0].Cells[1].Selected = true;
}
catch (Exception)
{
    dataGridView1.Enabled = false;
    button1.Enabled = false;
}

b1 = "select v.tochka_id, v.name_tochka as 'Сағу орны' from tochka v";
connect2();
dataGridView2.Columns["tochka_id"].Visible = false;
groupBox3.Enabled = true;

c1 = "select v.type_id, v.type_name as 'Тип' from type v";
connect3();
dataGridView3.Columns["type_id"].Visible = false;
groupBox5.Enabled = true;

d1 = "select v.spisok_id, v.spisok_name as 'Атауы', k.type_name as 'Тип', v.sena as
'Бағасы' from spisok_3 v inner join type k on v.type_id=k.type_id";
connect4();
```

```
dataGridView4.Columns["spisok_id"].Visible = false;  
groupBox7.Enabled = true;
```

*Б қосымшасы жалғасы*

```
e1 = "select g.id, v.vypusk_id, k.fam as 'Тегі', k.name as 'Аты', s.spisok_name as 'Газет  
атауы', v.nomer_v as 'Шығарылым номері', v.data_v as 'Шығарылым күні', g.kol_gazet as  
'Дана', g.dohod as 'Сомма, тг' from klient_gazeta2 g inner join klient_2 k on  
g.klient_id=k.klient_id inner join vypusk v on v.vypusk_id=g.vypusk_id inner join spisok_3 s on  
s.spisok_id=v.spisok_id";
```

```
connect5();  
dataGridView5.Rows[0].Cells[1].Selected = true;  
dataGridView5.Columns["id"].Visible = false;  
dataGridView5.Columns["vypusk_id"].Visible = false;
```

```
}
```

```
private string a, a1, b, b1, c, c1, d, d1, e, e1, cn;
```

```
public void connect()  
{  
    SqlConnection cn = Program.conn;  
    a = a1;  
    DataSet ds = new DataSet();  
    SqlDataAdapter da = new SqlDataAdapter(a, cn);  
    da.Fill(ds);  
    dataGridView1.DataSource = ds.Tables[0];  
}
```

```
public void connect2()  
{  
    SqlConnection cn = Program.conn;  
    b = b1;  
    DataSet ds1 = new DataSet();  
    SqlDataAdapter da1 = new SqlDataAdapter(b, cn);  
    da1.Fill(ds1);  
    dataGridView2.DataSource = ds1.Tables[0];  
}
```

```
public void connect3()  
{  
    SqlConnection cn = Program.conn;  
    c = c1;  
    DataSet ds3 = new DataSet();  
    SqlDataAdapter da3 = new SqlDataAdapter(c, cn);  
    da3.Fill(ds3);  
    dataGridView3.DataSource = ds3.Tables[0];  
}
```

```
public void connect4()
```

```

{
    SqlConnection cn = Program.conn;
        Б қосымшасы жалғасы

    DataSet ds4 = new DataSet();
    SqlDataAdapter da4 = new SqlDataAdapter(d, cn);
    da4.Fill(ds4);
    dataGridView4.DataSource = ds4.Tables[0];
}
public void connect5()
{
    SqlConnection cn = Program.conn;
    e = e1;
    DataSet ds5 = new DataSet();
    SqlDataAdapter da5 = new SqlDataAdapter(e, cn);
    da5.Fill(ds5);
    dataGridView5.DataSource = ds5.Tables[0];
}

private void Form1_Load(object sender, EventArgs e)
{
}

private void dataGridView1_CellContentClick(object sender, DataGridViewCellEventArgs e)
{
}

private void button1_Click(object sender, EventArgs e)
{
    doc_id = Convert.ToInt32(dataGridView1.CurrentRow.Cells[0].Value);
    dr = (DateTime) (from i in Program.db.klient_2 where i.klient_id == doc_id select
i.date_rozhdeniya).SingleOrDefault();
    td =(from i in Program.db.tochka where i.name_tochka == comboBox2.Text select
i.tochka_id).SingleOrDefault();
    //var c2 = (from i in Program.db.tochka where i.tochka_id == td select
i.name_tochka).SingleOrDefault();
    //comboBox2.Text = c2;
    textBox1.Text = (dataGridView1.CurrentRow.Cells[1].Value).ToString();
    textBox2.Text = (dataGridView1.CurrentRow.Cells[2].Value).ToString();
    textBox3.Text = (dataGridView1.CurrentRow.Cells[3].Value).ToString();
    dateTimePicker1.Value = dr;
    textBox5.Text = (dataGridView1.CurrentRow.Cells[5].Value).ToString();
    textBox6.Text = (dataGridView1.CurrentRow.Cells[6].Value).ToString();
    var ti = (from i in Program.db.klient_2 where i.klient_id == doc_id select
i.tochka_id).SingleOrDefault();
    var ti1 = (from o in Program.db.tochka where o.tochka_id == ti select
o.name_tochka).SingleOrDefault();
    comboBox2.Text = Convert.ToString(ti1);
    groupBox2.Visible = true;
}

```



```

        groupBox2.Enabled = true;
        groupBox1.Enabled = false;
                                Б қосымшасы жалғасы
button7.Visible = false;
        button5.Visible = true;

    }

    private void button5_Click(object sender, EventArgs e)
    {
        var datepic = dateTimePicker1.Value.ToString("dd.MM.yyyy");
        td = (from i in Program.db.tochka where i.name_tochka == comboBox2.Text select
i.tochka_id).SingleOrDefault();
        if (string.IsNullOrEmpty(textBox1.Text) || string.IsNullOrEmpty(textBox2.Text) ||
string.IsNullOrEmpty(textBox3.Text) || string.IsNullOrEmpty(textBox5.Text) ||
string.IsNullOrEmpty(textBox6.Text) || string.IsNullOrEmpty(comboBox2.Text))
        {
            MessageBox.Show("Барлық жолды толтырыңыз!", "Назар аударыңыз",
MessageBoxButtons.OK, MessageBoxIcon.Information);
        }
        else
        {
            SqlConnection SqlConnection1 = Program.conn;
            SqlConnection1.Open();
            SqlCommand cmd = new SqlCommand();
            cmd.CommandType = System.Data.CommandType.Text;
            cmd.CommandText = "UPDATE klient_2 SET fam='" + textBox1.Text + "', name='" +
textBox2.Text + "', patronimic='" + textBox3.Text + "', date_rozhdeniya=" + datepic + "',
klient_addres='" + textBox5.Text + "', tochka_id=" + td + "', klient_tel=" + textBox6.Text + "' where
klient_id=" + doc_id;
            cmd.Connection = SqlConnection1;
            cmd.ExecuteNonQuery();
            SqlConnection1.Close();

            MessageBox.Show("Мәлімет өзгерілді!", "Назар аударыңыз!",
MessageBoxButtons.OK,
MessageBoxIcon.Information);
            a1 = "select v.klient_id, v.fam as 'Тегі', v.name as 'Аты', v.patronimic as 'Әкесінің аты',
v.date_rozhdeniya as 'Туылған күні', v.klient_addres as 'Мекенжайы', v.klient_tel as 'Телефон'
from klient_2 v";
            connect();
            dataGridView1.Columns["klient_id"].Visible = false;
            groupBox1.Enabled = true;
            groupBox2.Visible = false;
            dateTimePicker1.Value = DateTime.Now.AddDays(1);
            try
            {
                dataGridView1.Rows[0].Cells[1].Selected = true;
            }
            catch (Exception)
            {

```

```

        dataGridView1.Enabled = false;
        button1.Enabled = false;
                                Б қосымшасы жалғасы
    }

    textBox1.Clear();
    textBox2.Clear();
    textBox3.Clear();
    textBox5.Clear();
    textBox6.Clear();
    comboBox2.Items.Clear();
    comboBox2.Text = "";
    var mesto1 = (from i in Program.db.tochka select i.name_tochka);
    foreach (var i in mesto1)
    {
        comboBox2.Items.Add(i);
    }
}

}

private void comboBox1_SelectedIndexChanged(object sender, EventArgs e)
{
    ssd=(from i in Program.db.tochka where i.name_tochka == comboBox1.Text select
i.tochka_id).SingleOrDefault();
    if (comboBox1.Text == "Барлығы")
    {
        a1 ="select v.klient_id, v.fam as 'Тегі', v.name as 'Аты', v.patronimic as 'Әкесінің
аты', v.date_rozhdeniya as 'Туылған күні', v.klient_adres as 'Мекенжайы', v.klient_tel as
'Телефон' from klient_2 v";
        connect();
        dataGridView1.Rows[0].Cells[1].Selected = true;
        dataGridView1.Columns["klient_id"].Visible = false;
    }

    else
    {
        a1 = "select v.klient_id, v.fam as 'Тегі', v.name as 'Аты', v.patronimic as 'Әкесінің аты',
v.date_rozhdeniya as 'Туылған күні', v.klient_adres as 'Мекенжайы', v.klient_tel as 'Телефон'
from klient_2 v where tochka_id = " + ssd;
        connect();
        dataGridView1.Rows[0].Cells[1].Selected = true;
        dataGridView1.Columns["klient_id"].Visible = false;
        comboBox1.Enabled = true;
    }
}

private void button2_Click(object sender, EventArgs e)
{

```

```

    groupBox1.Enabled = false;
    groupBox2.Visible = true;

```

*Б қосымшасы жалғасы*

```

button5.Visible = false;
    button7.Visible = true;
    textBox4.Clear();
    textBox7.Clear();
}

private void button4_Click(object sender, EventArgs e)
{
    textBox1.Clear();
    textBox2.Clear();
    textBox3.Clear();
    textBox5.Clear();
    textBox6.Clear();
    dateTimePicker1.Value = DateTime.Now.AddDays(1);
    comboBox2.Items.Clear();
    comboBox2.Text = "";
    var mesto1 = (from i in Program.db.tochka select i.name_tochka);
    foreach (var i in mesto1)
    {
        comboBox2.Items.Add(i);
    }
}

private void button3_Click(object sender, EventArgs e)
{
    groupBox2.Visible = false;
    groupBox1.Enabled = true;
    dateTimePicker1.Value = DateTime.Now.AddDays(1);
    textBox1.Clear();
    textBox2.Clear();
    textBox3.Clear();
    textBox5.Clear();
    textBox6.Clear();
    comboBox2.Items.Clear();
    comboBox2.Text = "";
    var mesto1 = (from i in Program.db.tochka select i.name_tochka);
    foreach (var i in mesto1)
    {
        comboBox2.Items.Add(i);
    }
}

private void textBox4_TextChanged(object sender, EventArgs e)
{

```

```

a1 = "select v.klient_id, v.fam as 'Тегі', v.name as 'Аты', v.patronimic as 'Әкесінің аты',
v.date_rozhdeniya as 'Туылған күні', v.klient_addres as 'Мекенжайы', v.klient_tel as 'Телефон'
from klient_2 v where v.fam like '"+ textBox4.Text + "%'";

```

*Б қосымшасы жалғасы*

```

connect();

```

```

    dataGridView1.Columns["klient_id"].Visible = false;

```

```

    groupBox1.Enabled = true;

```

```

}

```

```

private void textBox7_TextChanged(object sender, EventArgs e)

```

```

{

```

```

    a1 = "select v.klient_id, v.fam as 'Тегі', v.name as 'Аты', v.patronimic as 'Әкесінің аты',
v.date_rozhdeniya as 'Туылған күні', v.klient_addres as 'Мекенжайы', v.klient_tel as 'Телефон'
from klient_2 v where v.name like '" + textBox7.Text + "%'";

```

```

    connect();

```

```

    dataGridView1.Columns["klient_id"].Visible = false;

```

```

    groupBox1.Enabled = true;

```

```

}

```

```

private void button6_Click(object sender, EventArgs e)

```

```

{

```

```

    a1 = "select v.klient_id, v.fam as 'Тегі', v.name as 'Аты', v.patronimic as 'Әкесінің аты',
v.date_rozhdeniya as 'Туылған күні', v.klient_addres as 'Мекенжайы', v.klient_tel as 'Телефон'
from klient_2 v where v.fam like '" + textBox4.Text + "%'and v.name like '" + textBox7.Text +
"%'";

```

```

    connect();

```

```

    dataGridView1.Columns["klient_id"].Visible = false;

```

```

    groupBox1.Enabled = true;

```

```

}

```

```

private void button7_Click(object sender, EventArgs e)

```

```

{

```

```

    var datepic = dateTimePicker1.Value.ToString("dd.MM.yyyy");

```

```

    td = (from i in Program.db.tochka where i.name_tochka == comboBox2.Text select
i.tochka_id).SingleOrDefault();

```

```

    if (string.IsNullOrEmpty(textBox1.Text) || string.IsNullOrEmpty(textBox2.Text) ||
string.IsNullOrEmpty(textBox3.Text) || string.IsNullOrEmpty(textBox5.Text) ||
string.IsNullOrEmpty(textBox6.Text) || string.IsNullOrEmpty(comboBox2.Text))

```

```

    {

```

```

        MessageBox.Show("Барлық жолды толтырыңыз!", "Назар аударыңыз",
MessageBoxButtons.OK, MessageBoxIcon.Information);

```

```

    }

```

```

    else

```

```

    {

```

```

        SqlConnection SqlConnection1 = Program.conn;

```

```

        SqlConnection1.Open();

```

```

        SqlCommand cmd = new SqlCommand();

```

```

        cmd.CommandType = System.Data.CommandType.Text;

```

```

        cmd.CommandText = "insert into klient_2(fam, name, patronimic,

```

```

date_rozhdeniya,klient_addres,tochka_id, klient_tel) values('" + textBox1.Text + "','" +

```

```
textBox2.Text + "," + textBox3.Text + "," + datepic + "," + textBox5.Text + "," + td + "," +  
textBox6.Text + ")";
```

```
cmd.Connection = SqlConnection1;
```

*Б қосымшасы жалғасы*

```
cmd.ExecuteNonQuery();
```

```
SqlConnection1.Close();
```

```
MessageBox.Show("Мәлімет енгізілді!", "Назар аударыңыз!",  
MessageBoxButtons.OK,
```

```
MessageBoxIcon.Information);
```

```
a1 = "select v.klient_id, v.fam as 'Тегі', v.name as 'Аты', v.patronymic as 'Әкесінің аты',  
v.date_rozhdeniya as 'Туылған күні', v.klient_adres as 'Мекенжайы', v.klient_tel as 'Телефон'  
from klient_2 v";
```

```
connect();
```

```
dataGridView1.Columns["klient_id"].Visible = false;
```

```
groupBox1.Enabled = true;
```

```
groupBox2.Visible = false;
```

```
dateTimePicker1.Value = DateTime.Now.AddDays(1);
```

```
try
```

```
{
```

```
    dataGridView1.Rows[0].Cells[1].Selected = true;
```

```
}
```

```
catch (Exception)
```

```
{
```

```
    dataGridView1.Enabled = false;
```

```
    button1.Enabled = false;
```

```
}
```

```
comboBox2.Items.Clear();
```

```
comboBox2.Text = "";
```

```
var mesto1 = (from i in Program.db.tochka select i.name_tochka);
```

```
foreach (var i in mesto1)
```

```
{
```

```
    comboBox2.Items.Add(i);
```

```
}
```

```
textBox1.Clear();
```

```
textBox2.Clear();
```

```
textBox3.Clear();
```

```
textBox5.Clear();
```

```
textBox6.Clear();
```

```
}
```

```
}
```

```
private void button8_Click(object sender, EventArgs e)
```

```
{
```

```
    a1 = "select v.klient_id, v.fam as 'Тегі', v.name as 'Аты', v.patronymic as 'Әкесінің аты',  
v.date_rozhdeniya as 'Туылған күні', v.klient_adres as 'Мекенжайы', v.klient_tel as 'Телефон'  
from klient_2 v";
```

```
    connect();
```

```
    dataGridView1.Columns["klient_id"].Visible = false;
```

```
groupBox1.Enabled = true;
groupBox2.Visible = false;
textBox4.Clear();
```

*Б қосымшасы жалғасы*

```
comboBox1.Items.Clear();
comboBox1.Text = "";
comboBox1.Items.Add("Барлығы");
var mesto = (from i in Program.db.tochka select i.name_tochka);
foreach (var i in mesto)
{
    comboBox1.Items.Add(i);
}

}

private void textBox8_TextChanged(object sender, EventArgs e)
{
    b1 = "select v.tochka_id, v.name_tochka as 'Сағу орны' from tochka v where name_tochka
like '" + textBox8.Text + "%'";
    connect2();
    dataGridView2.Rows[0].Cells[1].Selected = true;
    dataGridView2.Columns["tochka_id"].Visible = false;
}

private void button9_Click(object sender, EventArgs e)
{
    td1 = Convert.ToInt32(dataGridView2.CurrentRow.Cells[0].Value);
    textBox9.Text = (dataGridView2.CurrentRow.Cells[1].Value).ToString();
    groupBox3.Enabled = false;
    groupBox4.Visible = true;
    button12.Visible = true;
}

private void button11_Click(object sender, EventArgs e)
{
    groupBox4.Visible = false;
    groupBox3.Enabled = true;
    textBox9.Clear();
}

private void button10_Click(object sender, EventArgs e)
{
    groupBox3.Enabled = false;
    groupBox4.Enabled = true;
    groupBox4.Visible = true;
    button21.Visible = true;
    textBox8.Clear();
}
```

```

private void button12_Click(object sender, EventArgs e)
{
    if (string.IsNullOrEmpty(textBox9.Text))
        Б қосымшасы жалғасы
        {
            MessageBox.Show("Барлық жолды толтырыңыз!", "Назар аударыңыз",
                MessageBoxButtons.OK, MessageBoxIcon.Information);
        }
    else
    {
        SqlConnection SqlConnection1 = Program.conn;
        SqlConnection1.Open();
        SqlCommand cmd = new SqlCommand();
        cmd.CommandType = System.Data.CommandType.Text;
        cmd.CommandText = "UPDATE tochka SET name_tochka=" + textBox9.Text + "where
tochka_id=" + td1;
        cmd.Connection = SqlConnection1;
        cmd.ExecuteNonQuery();
        SqlConnection1.Close();

        MessageBox.Show("Мәлімет өзгертілді!", "Назар аударыңыз!",
            MessageBoxButtons.OK,
            MessageBoxIcon.Information);
        b1 = "select v.tochka_id, v.name_tochka as 'Сату орны' from tochka v";
        connect2();
        dataGridView2.Columns["tochka_id"].Visible = false;
        groupBox3.Enabled = true;
        groupBox4.Visible = false;
    }
    textBox1.Clear();
    textBox2.Clear();
    textBox3.Clear();
    textBox5.Clear();
    textBox6.Clear();
}

private void button21_Click(object sender, EventArgs e)
{
    if (string.IsNullOrEmpty(textBox9.Text))
    {
        MessageBox.Show("Барлық жолды толтырыңыз!", "Назар аударыңыз",
            MessageBoxButtons.OK, MessageBoxIcon.Information);
    }
    else
    {
        SqlConnection SqlConnection1 = Program.conn;
        SqlConnection1.Open();
        SqlCommand cmd = new SqlCommand();
        cmd.CommandType = System.Data.CommandType.Text;
        cmd.CommandText = "insert into tochka(name_tochka) values('" + textBox9.Text + "')";
        cmd.Connection = SqlConnection1;
    }
}

```

```
cmd.ExecuteNonQuery();
SqlConnection1.Close();
```

### *Б қосымшасы жалғасы*

```
    MessageBox.Show("Мәлімет енгізілді!", "Назар аударыңыз!",
    MessageBoxButtons.OK,
    MessageBoxIcon.Information);
    b1 = "select v.tochka_id, v.name_tochka as 'Сағу орны' from tochka v";
    connect2();
    dataGridView2.Columns["tochka_id"].Visible = false;
    groupBox3.Enabled = true;
    groupBox4.Visible = false;
    textBox9.Clear();
}
}

private void button14_Click(object sender, EventArgs e)
{
    td2 = Convert.ToInt32(dataGridView3.CurrentRow.Cells[0].Value);
    textBox11.Text = (dataGridView3.CurrentRow.Cells[1].Value).ToString();
    groupBox5.Enabled = false;
    groupBox6.Visible = true;
    button16.Visible = true;
    button22.Visible = false;
    textBox10.Clear();
}

private void button15_Click(object sender, EventArgs e)
{
    groupBox6.Visible = false;
    groupBox5.Enabled = true;
    textBox11.Clear();
}

private void button13_Click(object sender, EventArgs e)
{
    groupBox5.Enabled = false;
    groupBox6.Visible = true;
    textBox10.Clear();
    button22.Visible = true;
    button16.Visible = false;
}

private void button22_Click(object sender, EventArgs e)
{
    if (string.IsNullOrEmpty(textBox11.Text))
    {
        MessageBox.Show("Барлық жолды толтырыңыз!", "Назар аударыңыз",
        MessageBoxButtons.OK, MessageBoxIcon.Information);
    }
}
```



```

    }
    else
    {
        Б қосымшасы жалғасы

        SqlConnection SqlConnection1 = Program.conn;
        SqlConnection1.Open();
        SqlCommand cmd = new SqlCommand();
        cmd.CommandType = System.Data.CommandType.Text;
        cmd.CommandText = "insert into type(type_name) values('" + textBox11.Text + "')";
        cmd.Connection = SqlConnection1;
        cmd.ExecuteNonQuery();
        SqlConnection1.Close();

        MessageBox.Show("Мәлімет енгізілді!", "Назар аударыңыз!",
        MessageBoxButtons.OK,
        MessageBoxIcon.Information);
        c1 = "select v.type_id, v.type_name as 'Тип' from type v";
        connect3();
        dataGridView3.Columns["type_id"].Visible = false;
        groupBox5.Enabled = true;
        groupBox6.Visible = false;
        textBox11.Clear();

    }
}

private void textBox10_TextChanged(object sender, EventArgs e)
{
    c1 = "select v.type_id, v.type_name as 'Тип' from type v where v.type_name like '"+
    textBox10.Text + "%'";
    connect3();
    dataGridView3.Columns["type_id"].Visible = false;
    groupBox5.Enabled = true;
}

private void button16_Click(object sender, EventArgs e)
{
    if (string.IsNullOrEmpty(textBox11.Text))
    {
        MessageBox.Show("Барлық жолды толтырыңыз!", "Назар аударыңыз",
        MessageBoxButtons.OK, MessageBoxIcon.Information);
    }
    else
    {
        SqlConnection SqlConnection1 = Program.conn;
        SqlConnection1.Open();
        SqlCommand cmd = new SqlCommand();
        cmd.CommandType = System.Data.CommandType.Text;
        cmd.CommandText = "update type set type_name='" + textBox11.Text + "' where
        type_id="+td2;
    }
}

```

```
cmd.Connection = SqlConnection1;
cmd.ExecuteNonQuery();
SqlConnection1.Close();
```

*Б қосымшасы жалғасы*

```
MessageBox.Show("Мәлімет енгізілді!", "Назар аударыңыз!",
MessageBoxButtons.OK,
MessageBoxIcon.Information);
c1 = "select v.type_id, v.type_name as 'Тип' from type v";
connect3();
dataGridView3.Columns["type_id"].Visible = false;
groupBox5.Enabled = true;
groupBox6.Visible = false;
textBox11.Clear();

}
}
```

```
private void button19_Click(object sender, EventArgs e)
{
    groupBox8.Visible = false;
    groupBox7.Enabled = true;
    textBox13.Clear();
    textBox14.Clear();
    comboBox4.Items.Clear();
    comboBox4.Text = "";
    var type1 = (from i in Program.db.type select i.type_name);
    foreach (var i in type1)
    {
        comboBox4.Items.Add(i);
    }
}
```

```
private void button17_Click(object sender, EventArgs e)
{
    groupBox8.Visible = true;
    groupBox7.Enabled = false;
    textBox12.Clear();
    button23.Visible = false;
    button20.Visible = true;

}
```

```
private void button18_Click(object sender, EventArgs e)
{
    groupBox8.Visible = true;
    groupBox7.Enabled = false;
    textBox12.Clear();
    button23.Visible = true;
    button20.Visible = false;
```

```

td3 = Convert.ToInt32(dataGridView4.CurrentRow.Cells[0].Value);
textBox13.Text = (dataGridView4.CurrentRow.Cells[1].Value).ToString();
textBox14.Text = (dataGridView4.CurrentRow.Cells[3].Value).ToString();

```

*Б қосымшасы жалғасы*

```

    var ti2 = (from i in Program.db.spisok_3 where i.spisok_id == td3 select
i.type_id).SingleOrDefault();
    var ti3 = (from o in Program.db.type where o.type_id == ti2 select
o.type_name).SingleOrDefault();
    comboBox4.Text = Convert.ToString(ti3);

}

private void button20_Click(object sender, EventArgs e)
{
    td4 = (from i in Program.db.type where i.type_name == comboBox4.Text select
i.type_id).SingleOrDefault();
    if (string.IsNullOrEmpty(textBox13.Text) || string.IsNullOrEmpty(textBox14.Text) ||
string.IsNullOrEmpty(comboBox4.Text))
    {
        MessageBox.Show("Барлық жолды толтырыңыз!", "Назар аударыңыз",
MessageBoxButtons.OK, MessageBoxIcon.Information);
    }
    else
    {
        SqlConnection SqlConnection1 = Program.conn;
        SqlConnection1.Open();
        SqlCommand cmd = new SqlCommand();
        cmd.CommandType = System.Data.CommandType.Text;
        cmd.CommandText = "insert into spisok_3(spisok_name, type_id, cena) values('" +
textBox13.Text + "','" + td4 + "','" + Convert.ToInt32(textBox14.Text) + "')";
        cmd.Connection = SqlConnection1;
        cmd.ExecuteNonQuery();
        SqlConnection1.Close();

        MessageBox.Show("Мәлімет енгізілді!", "Назар аударыңыз!",
MessageBoxButtons.OK,
        MessageBoxIcon.Information);
        d1 = "select v.spisok_id, v.spisok_name as 'Атауы', k.type_name as 'Тип', v.cena as
'Бағасы' from spisok_3 v inner join type k on v.type_id=k.type_id";
        connect4();
        dataGridView4.Columns["spisok_id"].Visible = false;
        groupBox7.Enabled = true;
        groupBox8.Visible = false;
        textBox13.Clear();
        textBox14.Clear();
        comboBox4.Items.Clear();
        comboBox4.Text = "";
        var type1 = (from i in Program.db.type select i.type_name);
        foreach (var i in type1)

```

```

    {
        comboBox4.Items.Add(i);
    }

```

*Б қосымшасы жалғасы*

```

}
}

```

```

private void textBox12_TextChanged(object sender, EventArgs e)
{
    d1 = "select v.spisok_id, v.spisok_name as 'Атауы', k.type_name as 'Тип', v.cena as
'Бағасы' from spisok_3 v inner join type k on v.type_id=k.type_id where v.spisok_name like
"+textBox12.Text+"%";
    connect4();
    dataGridView4.Columns["spisok_id"].Visible = false;
    groupBox7.Enabled = true;
}

private void comboBox3_SelectedIndexChanged(object sender, EventArgs e)
{
    var ssd1 = (from i in Program.db.type where i.type_name == comboBox3.Text select
i.type_id).SingleOrDefault();
    if (comboBox3.Text == "Барлығы")
    {
        d1 = "select v.spisok_id, v.spisok_name as 'Атауы', k.type_name as 'Тип', v.cena as
'Бағасы' from spisok_3 v inner join type k on v.type_id=k.type_id";
        connect4();
        dataGridView4.Columns["spisok_id"].Visible = false;
        groupBox7.Enabled = true;
    }

    else
    {
        d1 = "select v.spisok_id, v.spisok_name as 'Атауы', k.type_name as 'Тип', v.cena as
'Бағасы' from spisok_3 v inner join type k on v.type_id=k.type_id where v.type_id="+ssd1;
        connect4();
        dataGridView4.Columns["spisok_id"].Visible = false;
        groupBox7.Enabled = true;
    }
}

private void button23_Click(object sender, EventArgs e)
{
    td4 = (from i in Program.db.type where i.type_name == comboBox4.Text select
i.type_id).SingleOrDefault();
    if (string.IsNullOrEmpty(textBox13.Text) || string.IsNullOrEmpty(textBox14.Text) ||
string.IsNullOrEmpty(comboBox4.Text))
    {
        MessageBox.Show("Барлық жолды толтырыңыз!", "Назар аударыңыз",
MessageBoxButtons.OK, MessageBoxIcon.Information);
    }
}

```

```

}
else
{
    Б қосымшасы жалғасы

    SqlConnection SqlConnection1 = Program.conn;
    SqlConnection1.Open();
    SqlCommand cmd = new SqlCommand();
    cmd.CommandType = System.Data.CommandType.Text;
    cmd.CommandText = "update spisok_3 set spisok_name=" + textBox13.Text + ",
type_id=" + td4 + ", cena=" + Convert.ToInt32(textBox14.Text) + " where spisok_id="+td3;
    cmd.Connection = SqlConnection1;
    cmd.ExecuteNonQuery();
    SqlConnection1.Close();

    MessageBox.Show("Мәлімет енгізілді!", "Назар аударыңыз!",
    MessageBoxButtons.OK,
    MessageBoxIcon.Information);
    d1 = "select v.spisok_id, v.spisok_name as 'Атауы', k.type_name as 'Тип', v.cena as
'Бағасы' from spisok_3 v inner join type k on v.type_id=k.type_id";
    connect4();
    dataGridView4.Columns["spisok_id"].Visible = false;
    groupBox7.Enabled = true;
    groupBox8.Visible = false;
    textBox13.Clear();
    textBox14.Clear();
    comboBox4.Items.Clear();
    comboBox4.Text = "";
    var type1 = (from i in Program.db.type select i.type_name);
    foreach (var i in type1)
    {
        comboBox4.Items.Add(i);
    }
}
}

private void button24_Click(object sender, EventArgs e)
{
    this.Close();
}

private void label19_Click(object sender, EventArgs e)
{
}

private void textBox15_TextChanged(object sender, EventArgs e)
{
    e1 = "select g.id, v.vypusk_id, k.fam as 'Тегі', k.name as 'Аты', s.spisok_name as 'Газет
атауы', v.pomer_v as 'Шығарылым номері', v.data_v as 'Шығарылым күні', g.kol_gazet as

```

'Дана', g.dohod as 'Сомма, тг' from klient\_gazeta2 g inner join klient\_2 k on g.klient\_id=k.klient\_id inner join vypusk v on v.vypusk\_id=g.vypusk\_id inner join spisok\_3 s on s.spisok\_id=v.spisok\_id where k.fam like'" + textBox15.Text + "%";

*Б қосымшасы жалғасы*

```

connect5();
dataGridView5.Rows[0].Cells[1].Selected = true;
dataGridView5.Columns["id"].Visible = false;
dataGridView5.Columns["vypusk_id"].Visible = false;
}

private void textBox16_TextChanged(object sender, EventArgs e)
{
    e1 = "select g.id, v.vypusk_id, k.fam as 'Тегі', k.name as 'Аты', s.spisok_name as 'Газет
атауы', v.nomer_v as 'Шығарылым номері', v.data_v as 'Шығарылым күні', g.kol_gazet as
'Дана', g.dohod as 'Сомма, тг' from klient_gazeta2 g inner join klient_2 k on
g.klient_id=k.klient_id inner join vypusk v on v.vypusk_id=g.vypusk_id inner join spisok_3 s on
s.spisok_id=v.spisok_id where k.name like'" + textBox16.Text + "%";
    connect5();
    dataGridView5.Rows[0].Cells[1].Selected = true;
    dataGridView5.Columns["id"].Visible = false;
    dataGridView5.Columns["vypusk_id"].Visible = false;
}

private void textBox17_TextChanged(object sender, EventArgs e)
{
    try
    {
        e1 = "select g.id, v.vypusk_id, k.fam as 'Тегі', k.name as 'Аты', s.spisok_name as 'Газет
атауы', v.nomer_v as 'Шығарылым номері', v.data_v as 'Шығарылым күні', g.kol_gazet as
'Дана', g.dohod as 'Сомма, тг' from klient_gazeta2 g inner join klient_2 k on
g.klient_id=k.klient_id inner join vypusk v on v.vypusk_id=g.vypusk_id inner join spisok_3 s on
s.spisok_id=v.spisok_id where v.nomer_v='" + Convert.ToInt32(textBox17.Text) + "'";
        connect5();
        dataGridView5.Rows[0].Cells[1].Selected = true;
        dataGridView5.Columns["id"].Visible = false;
        dataGridView5.Columns["vypusk_id"].Visible = false;
    }
    catch (Exception)
    {
        e1 = "select g.id, v.vypusk_id, k.fam as 'Тегі', k.name as 'Аты', s.spisok_name as 'Газет
атауы', v.nomer_v as 'Шығарылым номері', v.data_v as 'Шығарылым күні', g.kol_gazet as
'Дана', g.dohod as 'Сомма, тг' from klient_gazeta2 g inner join klient_2 k on
g.klient_id=k.klient_id inner join vypusk v on v.vypusk_id=g.vypusk_id inner join spisok_3 s on
s.spisok_id=v.spisok_id";
        connect5();
        dataGridView5.Rows[0].Cells[1].Selected = true;
        dataGridView5.Columns["id"].Visible = false;
        dataGridView5.Columns["vypusk_id"].Visible = false;
    }
}

```

```

}

private void button27_Click(object sender, EventArgs e)
    Б қосымшасы жалғасы

{
    if (string.IsNullOrEmpty(textBox17.Text) || string.IsNullOrEmpty(comboBox5.Text))
    {
        MessageBox.Show("Барлық жолды толтырыңыз ", "Назар аударыңыз",
        MessageBoxButtons.OK,
        MessageBoxIcon.Information);
    }
    else
    {
        e1 =
            "select g.id, v.vypusk_id, k.fam as 'Тегі', k.name as 'Аты', s.spisok_name as 'Газет
атауы', v.nomer_v as 'Шығарылым номері', v.data_v as 'Шығарылым күні', g.kol_gazet as
'Дана', g.dohod as 'Сомма, тг' from klient_gazeta2 g inner join klient_2 k on
g.klient_id=k.klient_id inner join vypusk v on v.vypusk_id=g.vypusk_id inner join spisok_3 s on
s.spisok_id=v.spisok_id where v.nomer_v=" +
            Convert.ToInt32(textBox17.Text) + "' and s.spisok_name=" + comboBox5.Text + "'
and k.name like" + textBox16.Text + "%' and k.fam like" + textBox15.Text + "%";
        connect5();
        dataGridView5.Rows[0].Cells[1].Selected = true;
        dataGridView5.Columns["id"].Visible = false;
        dataGridView5.Columns["vypusk_id"].Visible = false;
    }
}

private void comboBox5_SelectedIndexChanged(object sender, EventArgs e)
{
}

private void button26_Click(object sender, EventArgs e)
{
    vpi = Convert.ToInt32(dataGridView5.CurrentRow.Cells[0].Value);
    vpi2 = Convert.ToInt32(dataGridView5.CurrentRow.Cells[1].Value);
    dv = (from i in Program.db.vypusk where i.vypusk_id == vpi2 select
i.data_v).SingleOrDefault();
    textBox19.Text = (dataGridView5.CurrentRow.Cells[2].Value).ToString();
    textBox18.Text = (dataGridView5.CurrentRow.Cells[3].Value).ToString();
    textBox21.Text = (dataGridView5.CurrentRow.Cells[4].Value).ToString();
    textBox20.Text = (dataGridView5.CurrentRow.Cells[5].Value).ToString();
    dateTimePicker2.Value = dv;
    textBox22.Text = (dataGridView5.CurrentRow.Cells[7].Value).ToString();
    s7 = Convert.ToInt32(dataGridView5.CurrentRow.Cells[7].Value);
    groupBox10.Visible = true;
    groupBox9.Enabled = false;
    button29.Visible = true;
}

```

```
button30.Visible = false;
textBox18.ReadOnly = true;
textBox19.ReadOnly = true;
```

*Б қосымшасы жалғасы*

```
textBox20.ReadOnly = true;
textBox21.ReadOnly = true;
dateTimePicker2.Enabled = false;
```

```
kg = Convert.ToInt32(dataGridView5.CurrentRow.Cells[1].Value);
var gg = (from i in Program.db.vypusk where i.vypusk_id == kg select
i.kollichestvo).SingleOrDefault();
s3 = 0;
var s2 = (from i in Program.db.klient_gazeta2 where i.vypusk_id == kg select
i.kol_gazet).ToArray();
```

```
for (int j = 0; j < s2.Length; j++)
{
    s3 = s3 + s2[j];
}
```

```
s6 = gg - s3;
s8 = s7 + s6;
```

```
label28.Text = "Барлығы " + Convert.ToString(s8) + " дана";
}
```

```
private void button28_Click(object sender, EventArgs e)
```

```
{
    groupBox10.Visible = false;
    groupBox9.Enabled = true;
    textBox18.Clear();
    textBox19.Clear();
    textBox20.Clear();
    textBox21.Clear();
    textBox22.Clear();
}
```

```
private void button29_Click(object sender, EventArgs e)
```

```
{
    if (string.IsNullOrEmpty(textBox22.Text) || s8 < Convert.ToInt32(textBox22.Text) ||
Convert.ToInt32(textBox22.Text)<0)
    {
        MessageBox.Show("Барлық жолды толтырыңыз! \n "+s8+" данадан аспауы
керек", "Назар аударыңыз", MessageBoxButtons.OK, MessageBoxIcon.Information);
    }
    else
    {
        SqlConnection SqlConnection1 = Program.conn;
        SqlConnection1.Open();
```



```

SqlCommand cmd = new SqlCommand();
cmd.CommandType = System.Data.CommandType.Text;

```

*Б қосымшасы жалғасы*

```

cmd.CommandText = "update klient_gazeta2 set
kol_gazet='"+Convert.ToInt32(textBox22.Text)+"' where id="+vpi;
cmd.Connection = SqlConnection1;
cmd.ExecuteNonQuery();
SqlConnection1.Close();
MessageBox.Show("Мәлімет енгізілді!", "Назар аударыңыз!",
MessageBoxButtons.OK,
MessageBoxIcon.Information);
e1 = "select j.id, v.vypusk_id, d.fam as 'Тегі', d.name as 'Аты', k.spisok_name as 'Газет
атауы', v.nomer_v as 'Шығарылым номері', v.data_v as 'Шығарылым күні', j.kol_gazet as 'Дана'
from vypusk v inner join spisok_3 k on v.spisok_id=k.spisok_id inner join klient_gazeta2 j on
j.vypusk_id = v.vypusk_id inner join klient_2 d on d.klient_id = j.klient_id";
connect5();
dataGridView5.Rows[0].Cells[1].Selected = true;
dataGridView5.Columns["id"].Visible = false;
dataGridView5.Columns["vypusk_id"].Visible = false;
groupBox10.Visible = false;
groupBox9.Enabled = true;
}

textBox22.Clear();
}

private void button30_Click(object sender, EventArgs e)
{
SqlConnection SqlConnection1 = Program.conn;
SqlConnection1.Open();
SqlCommand cmd = new SqlCommand();
cmd.CommandType = System.Data.CommandType.Text;
cmd.CommandText = "delete from klient_gazeta2 where id=" + vpi;
cmd.Connection = SqlConnection1;
cmd.ExecuteNonQuery();
SqlConnection1.Close();
MessageBox.Show("Мәлімет жойылды!", "Назар аударыңыз!", MessageBoxButtons.OK,
MessageBoxIcon.Information);
e1 = "select g.id, v.vypusk_id, k.fam as 'Тегі', k.name as 'Аты', s.spisok_name as 'Газет
атауы', v.nomer_v as 'Шығарылым номері', v.data_v as 'Шығарылым күні', g.kol_gazet as
'Дана', g.dohod as 'Сомма, тг' from klient_gazeta2 g inner join klient_2 k on
g.klient_id=k.klient_id inner join vypusk v on v.vypusk_id=g.vypusk_id inner join spisok_3 s on
s.spisok_id=v.spisok_id";
connect5();
dataGridView5.Rows[0].Cells[1].Selected = true;
dataGridView5.Columns["id"].Visible = false;
dataGridView5.Columns["vypusk_id"].Visible = false;
groupBox10.Visible = false;
groupBox9.Enabled = true;
}

```

```

}

private void button25_Click(object sender, EventArgs e)
    Б қосымшасы жалғасы

{
    vpi = Convert.ToInt32(dataGridView5.CurrentRow.Cells[0].Value);
    vpi2 = Convert.ToInt32(dataGridView5.CurrentRow.Cells[1].Value);
    dv = (from i in Program.db.vypusk where i.vypusk_id == vpi2 select
i.data_v).SingleOrDefault();
    textBox19.Text = (dataGridView5.CurrentRow.Cells[2].Value).ToString();
    textBox18.Text = (dataGridView5.CurrentRow.Cells[3].Value).ToString();
    textBox21.Text = (dataGridView5.CurrentRow.Cells[4].Value).ToString();
    textBox20.Text = (dataGridView5.CurrentRow.Cells[5].Value).ToString();
    dateTimePicker2.Value = dv;
    textBox22.Text = (dataGridView5.CurrentRow.Cells[7].Value).ToString();
    button29.Visible = false;
    button30.Visible = true;
    groupBox9.Enabled = false;
    groupBox10.Visible = true;
    label28.Visible = false;

}

private void comboBox5_SelectedIndexChanged_1(object sender, EventArgs e)
{
    e1 = "select g.id, v.vypusk_id, k.fam as 'Тегі', k.name as 'Аты', s.spisok_name as 'Газет
атауы', v.pomer_v as 'Шығарылым номері', v.data_v as 'Шығарылым күні', g.kol_gazet as
'Дана', g.dohod as 'Сомма, тг' from klient_gazeta2 g inner join klient_2 k on
g.klient_id=k.klient_id inner join vypusk v on v.vypusk_id=g.vypusk_id inner join spisok_3 s on
s.spisok_id=v.spisok_id where s.spisok_name='" + comboBox5.Text + "'";
    connect5();
    dataGridView5.Rows[0].Cells[1].Selected = true;
    dataGridView5.Columns["id"].Visible = false;
    dataGridView5.Columns["vypusk_id"].Visible = false;
}

private void button31_Click(object sender, EventArgs e)
{
    try
    {
        e1 = "select g.id, v.vypusk_id, k.fam as 'Тегі', k.name as 'Аты', s.spisok_name as 'Газет
атауы', v.pomer_v as 'Шығарылым номері', v.data_v as 'Шығарылым күні', g.kol_gazet as
'Дана', g.dohod as 'Сомма, тг' from klient_gazeta2 g inner join klient_2 k on
g.klient_id=k.klient_id inner join vypusk v on v.vypusk_id=g.vypusk_id inner join spisok_3 s on
s.spisok_id=v.spisok_id";
        connect5();
        dataGridView5.Rows[0].Cells[1].Selected = true;
        dataGridView5.Columns["id"].Visible = false;
        dataGridView5.Columns["vypusk_id"].Visible = false;
    }
}

```

```

    }
    catch (Exception)
    {
        Б қосымшасы жалғасы

    }
    textBox16.Clear();
    textBox17.Clear();
    textBox15.Clear();
    comboBox5.Items.Clear();
    comboBox5.Text = "";
    var s_n2 = (from i in Program.db.spisok_3 select i.spisok_name);
    foreach (var i in s_n2)
    {
        comboBox5.Items.Add(i);
    }

}

private void button32_Click(object sender, EventArgs e)
{
    comboBox3.Items.Clear();
    comboBox3.Text = "";
    comboBox3.Items.Add("Барлығы");
    var type2 = (from i in Program.db.type select i.type_name);
    foreach (var i in type2)
    {
        comboBox3.Items.Add(i);
    }
    textBox12.Clear();
    d1 = "select v.spisok_id, v.spisok_name as 'Атауы', k.type_name as 'Тип', v.cena as
'Бағасы' from spisok_3 v inner join type k on v.type_id=k.type_id";
    connect4();
    dataGridView4.Columns["spisok_id"].Visible = false;
    groupBox7.Enabled = true;
}

private void button33_Click(object sender, EventArgs e)
{
    var dwww = (from i in Program.db.klient_gazeta2 select i.id).ToArray();
    for (int t = 0; t < dwww.Length; t++)
    {
        int g_i = (from o in Program.db.klient_gazeta2 where o.id == dwww[t] select
o.vypusk_id).SingleOrDefault();
        int sp_i = (from o in Program.db.vypusk where o.vypusk_id == g_i select
o.spisok_id).SingleOrDefault();
        var g_cena = (from o in Program.db.spisok_3 where o.spisok_id == sp_i select
o.cena).SingleOrDefault();
        SqlConnection SqlConnection1 = Program.conn;

```

```

SqlConnection1.Open();
SqlCommand cmd = new SqlCommand();
cmd.CommandType = System.Data.CommandType.Text;
        Б қосымшасының соңы

cmd.CommandText = "UPDATE klient_gazeta2 SET dohod = kol_gazet*" + g_cena + "
where id = " + dwww[t];
cmd.Connection = SqlConnection1;
cmd.ExecuteNonQuery();
SqlConnection1.Close();
}
MessageBox.Show("Мәлімет жаңартылды!", "Назар аударыңыз!",
MessageBoxButtons.OK,
MessageBoxIcon.Information);
}

}
}

```