

ҚАЗАҚСТАН РЕСПУБЛИКАСЫ ҒЫЛЫМ ЖӘНЕ ЖОҒАРЫ БІЛІМ МИНИСТРЛІГІ  
«Қ.И. Сәтбаев атындағы Қазақ ұлттық техникалық зерттеу университеті» коммерциялық  
емес акционерлік қоғамы

Автоматика және ақпараттық технологиялар институты

Автоматтандыру және басқару кафедрасы

6B07103 - Автоматтандыру және роботтандыру

Мырзаш Әділжан Ермұханұлы

«Қойларды есепке алу жүйесін әзірлеу»

Дипломдық жобаға  
**ТҮСІНДІРМЕ ЖАЗБА**

6B07103-Автоматтандыру және роботтандыру

Алматы 2023

ҚАЗАҚСТАН РЕСПУБЛИКАСЫ ҒЫЛЫМ ЖӘНЕ ЖОҒАРЫ БІЛІМ МИНИСТРЛІГІ


«Қ.И. Сәтбаев атындағы Қазақ ұлттық техникалық зерттеу университеті» коммерциялық емес акционерлік қоғамы

Автоматика және ақпараттық технологиялар институты

Автоматтандыру және басқару кафедрасы

**ҚОРҒАУҒА ЖІБЕРІЛДІ**

Автоматтандыру және басқару кафедрасының меңгерушісі, физика-математика ғылымдарының кандидаты

 Алдияров Н.У.  
«02» маусым 2023 ж.



Дипломдық жобаға  
**ТҮСІНДІРМЕ ЖАЗБА**

Тақырыбы «Қойларды есепке алу жүйесін әзірлеу»

6B07103 - «Автоматтандыру және роботтандыру» мамандығы

Орындаған:

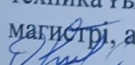
Мырзаш Әділжан Ермұханұлы

Рецензент:

Ғылыми жетекші:

техника ғылымдарының  
магистрі, аға оқытушы

техника ғылымдарының  
магистрі, аға оқытушы

 Оракбаев Е.Ж.

 Зікірбай Қ.Е.

«02» 06 2023 ж

«02» 06 2023 ж.

Алматы 2023

ҚАЗАҚСТАН РЕСПУБЛИКАСЫ ҒЫЛЫМ ЖӘНЕ ЖОҒАРЫ БІЛІМ МИНИСТРЛІГІ  
«Қ.И. Сәтбаев атындағы Қазақ ұлттық техникалық зерттеу университеті» коммерциялық  
емес акционерлік қоғамы

Автоматика және ақпараттық технологиялар институты

Автоматтандыру және басқару кафедрасы

6B07103 - «Автоматтандыру және роботтандыру» мамандығы

**БЕКІТЕМІН**

Автоматтандыру және басқару  
кафедрасының менгерушісі,  
физика-математика ғылымдарының  
кандидаты

Алдияров Н.У.  
«22» маусым 2023 ж.



**Дипломдық жобаны орындауға арналған  
ТАПСЫРМА**

Білім алушы Мырзаш Әділжан Ермұханұлы

Жобаның тақырыбы: «Қойларды есепке алу жүйесін әзірлеу»

Университет проректоры Б.А. Жаутиковтың «23» қараша 2022ж. № «408-П/Ө» бұйрығымен  
бекітілген.

Аяқталған жұмысты тапсыру мерзімі «9» маусым 2023 ж.

Дипломдық жұмысты әзірлеуге жататын мәселелер тізімі:

а) кіріспе;

б) технологиялық бөлім, арнайы бөлім;

Графикалық материалдар тізімі (міндетті сызбаларды дәл көрсете отырып): функционалдық сұлба

Жұмыс презентациясы 17 слайдтарда көрсетілген.

Ұсынылатын негізгі әдебиеттер 24 атаулардан тұрады.

Дипломдық жобаны дайындау  
КЕСТЕСІ

Бөлімдердің атауы, зерттеп дайындалатын мәселелер тізімі	Ғылыми жетекшіге ұсыну мерзімдері	Ескерту
Технологиялық бөлім	23.01.23 - 13.02.23	
Арнайы бөлім	20.02.23 - 10.04.23	

Аяқталған дипломдық жоба үшін, оған қатысты бөлімдердің жобасын көрсетумен,  
кеңесшілер мен норма бақылаушының қойған қолдары

Бөлімдер атауы	Кеңесшілер тегі, аты, әкесінің аты, (ғылыми дәрежесі, атағы)	Қол қойылған күні	Қолы
Технологиялық бөлім	Зікірбай Қ.Е., техника ғылымдарының магистрі, аға оқытушы	25.05.2023	З.Қ.
Арнайы бөлім	Зікірбай Қ.Е., техника ғылымдарының магистрі, аға оқытушы	25.05.2023	З.Қ.
Норма бақылаушы	Жанабаева Ә.Ж., техника ғылымдарының магистрі, ассистент	26.05.23	М.Ә.

Ғылыми жетекшісі З.Қ. Зікірбай Қ.Е.

Тапсырманы орындауға қабылдаған білім алушы М.Ә. Мырзаш Ә.Е.

Күні « 5 » маусым 2023 ж.

## АНДАТПА

Бұл дипломдық жобада-қойларды есепке алу жүйесін әзірлеу болып табылады. Қойларды есепке алу жүйесі-бұл фермадағы қойлардың саны мен сапасын ұйымдастыруға және бақылауға бағытталған іс-шаралар кешені. Ол әр қой туралы,оның сипаттамалары мен денсаулығы туралы ақпарат жинауды, сондай-ақ одан алынған өнім мен пайданы есепке алуды қамтиды.

Бұл дипломдық жобада жүйенің жұмыс істеу принципі көрсетіліп, жүйенің жалпы құрылымы сипатталынды. Жобада, қойлар туралы, олардан алынатын өнімдерді, яғни ет, жүн тері сияқты өнімдерді есептеу жүйесі көрсетілді. Сонымен қатар, бұл жүйеде өндіріс туралы ақпаратты енгізуге болатыны көрсетілді. Жүйеде қойлардың әкелінген күні, олардан өнімнің алынған мерзімі, олардың төлдері туралы ақпараттарды енгізуге болады.

## АННОТАЦИЯ

Этот дипломный проект представляет собой разработку системы учета овец. Система учета овец-это комплекс мероприятий, направленных на организацию и контроль за количеством и качеством овец на ферме. Она включает в себя сбор информации о каждой овце, ее характеристиках и состоянии здоровья, а также учет полученных от нее продукции и прибыли.

В данном дипломном проекте изложен принцип функционирования системы и описана общая структура системы. В проекте показана система учета продуктов получаемых от овец, то есть таких продуктов как мясо, шерсть, кожа. Кроме того, было показано что в этой системе можно вводить информацию о производстве. В систему можно ввести информацию о дате ввоза овец, сроках получения от них продукции и их детенышей.

## ANNOTATION

This project is the development of a sheep accounting system. The sheep accounting system is a set of measures aimed at organizing and controlling the quality of sheep on a farm. It includes collecting information about each sheep, its characteristics and health status, as well as accounting for the products and profits received from it.

In this project, the principle of the functioning of the system is outlined and the general structure of the system is described. The paper shows a system of accounting for products received from sheep, that is products such as meat, wool, leather. In addition it was shown that production information can be entered in this system. You can enter information about the date of importation of sheep the timing of receiving products from them and their cubs into the system.

## МАЗМҰНЫ

Кіріспе	7
1 Қойылған мәселенің қазіргі жағдайы	9
1.1 Қойылған мәселенің қазіргі жағдайы бойынша әдебиетке шолу	9
1.2 Ақпараттық жүйені жасау қажеттілігі	10
1.3 Ақпараттық жүйенің ресурстарына қойылатын талаптар	12
2 Есептің қойылымы	17
2.1 «Қойларды есепке алу» ақпараттық жүйесін құру	23
2.2 Қолданушы интерфейсі	24
2.3 Бағдарламалық қамтамаға қойылатын талаптар	31
3 Жобаның экономикалық негізделуі	32
Қорытынды	40
Пайдаланылған әдебиеттер тізімі	41
ҚОСЫМША	43

## КІРІСПЕ

**Жұмыстың өзектілігі.** Аграрлық секторда ғылымға жеткілікті дәрежеде негізделмей жүргізілген реформалар ауыл шаруашылығы өнімдері өндірісінің құлдырауына алып келді, әсіресе мал шаруашылығы көбірек жапа шекті. Ауыл шаруашылығы саласының құрылымы, әсіресе ондағы өсімдік пен мал шаруашылығы қатынастары мүлдем өзгерді.

Қой шаруашылығындағы автоматтандыру-бұл мал шаруашылығының осы саласын құрылымды етуге қабілетті басқарушылық инновация, ал осы бағытта жұмыс істейтін кәсіпорындар пайда табудың жаңа деңгейіне шығарады. Мүмкін, бұл автоматтандыру арқылы қой шаруашылығына мамандандырылған кәсіпорынның жұмысы аясында жүзеге асырылатын барлық процедуралардың орындалу мерзімін реттеуге болатындығына байланысты.

Қой шаруашылығы Ет, сүт, жүн және жүн сияқты өнімдерді алу үшін айналысады. Бұл өнімдерді алу кезінде бухгалтерлік есеп өте маңызды. Мал азығына, оларға күтім жасауға қанша қаражат жұмсалғанын нақты түсіну қажет. Сондай-ақ, қызметкерлердің жалақысына, басқа шығындарға қанша ақша қажет екенін есептеу керек. Жалпы, қой шаруашылығы ауыл шаруашылығының көп құрылымды саласы деп айтуға болады, сондықтан ондағы істерді оңтайландыру үшін осындай оңтайландырудың әртүрлі құралдарын қолдану қажет. Қой шаруашылығындағы автоматтандыруды өндірістік процестерді оңтайландыру құралы ретінде адам қызметінің мал шаруашылығы салаларында іс жүргізу ерекшеліктерін ескере отырып әзірленетін бағдарламаларды пайдалана отырып қолдану керек. Мұндай бағдарламаны әмбебап есеп жүйесі ұсынады.

УСУ қой шаруашылығын автоматтандыру шеңберінде қойлардың есебін неғұрлым стандартталған көрсеткіштер (салмақ) бойынша да, нақты көрсеткіштер бойынша да (жүн немесе ет нәтижесі бойынша алынған салмақ) жүргізеді. Сонымен қатар, біздің бағдарламамыздың жұмыс істеуі аясында сіз қойлар, олардың жас ерекшеліктері, олар беретін өнімдердің сапасы мен саны және тағы басқалар туралы барлық ақпаратты қамтитын мәліметтер базасын құра аласыз.

Қазіргі заманда ақпарат үлкен орын алып, уақыт өтуіне байланысты қарқынды ұлғайып отыруда. Ақпаратты өз уақытында, сенімді және толық беруді ақпараттық жүйе қамтамасыз етеді.

Ақпараттық жүйелер қолданушылардың сұраныстарына сәйкес ақпаратты жинауға, сақтауға, іздеуге, өңдеуге және жеткізуге мүмкіндік береді және оларды жүйелі ұйымдастыру үшін мәліметтер базасын қолданған жөн.

Мәліметтер базасы деп деректердің электрондық сақтаушысын айтады және ақпаратты сақтауды, мәліметтерге ыңғайлы, тез кіруді қамтамасыз етеді. Мәліметтер базасы белгілі бір ережелерге сай құрылған мәліметтер жиынтығын құрайды және базадағы ақпарат артықсыз, тұтас болуы керек.

Мәліметтер базасын басқару жүйесі мәліметтер базасын құруға, толтыруға, өзгертуге, жоюға арналған программалық жабдық болып табылады.

Мәліметтер базасын жасау кезінде екі негізгі мәселені ескеру қажет:

1. Бағдарламаны өзгертпей-ақ мәліметтерді қосу немесе қайта құру үшін мәліметтер өздерін қолданған бағдарламаға тәуелсіз болуы қажет.

2. Жай бағдарлама тілінде күрделі еңбекті қажет ететін бағдарламасыз-ақ мәліметтер базасынан керекті ақпаратты сұрау және іздеу мүмкіндігін қамтамасыз етеді.

Бүгінгі күннің өзінде мәліметтер базасын қолданбайтын бірде бір білім ортасының және басқа да салалардың жұмысын елестету мүмкін емес.

Қазіргі кезеңнің ерекшелігі үлкен көлемдік материалдық және шикізат ресурстарын пайдалануға негізделген индустриалды экономикадан, негізгі ресурсы ақпарат болып табылатын ақпараттық экономикаға өту. Информатика мамандары қоғамдағы ақпарат рөлінің күшеюін бір ауыздан мойындап отыр. Ақпарат бара – бара қоғамдық әл – ауқат деңгейін анықтайтын шешуші факторға айналуға. Ол бизнестің стратегиялық ресурсы болып қалыптасып келеді.

Бізді қоршаған ортада әрқашан қозғалыста болған ақпараттар жиындары өте үлкен. Уақыт өтуімен олар арту тенденциясына ие. Сондықтан кез келген үлкен немесе кіші ұйымдарда нәтижелі жұмыстарды қамтамасыз ететін мәліметтерді басқару мәселесі туады. Кейбір ұйымдар бұл үшін папкалардан тұратын шкафтарды пайдаланады, бірақ көпшілік ұйымдар компьютерленген мүмкіншіліктерін жоғары бағалайды. Қазіргі күнде көпшілік қаржы ұйымдары, өнеркәсіптер, сауда – саттық және т.б. ұйымдар мәліметтер қорсыз жұмыс істеуі өте қиын. Мәліметтер қоры болмағанда олар ақпараттар көшкінінде жай ғана тұншығып қалар еді.

Ақпараттану концепциясының негізгі идеясы қоғамның әлеуметтік экономикалық дамуына ақпарат пен жаңа ақпараттық технология әсерінің өсе түсуі болып отыр. Осы жерде екі ұғымның түсінігін анықтау проблемасы туады – ақпараттық ұқыптылық және компьютерлік сауаттылық. Бұл екі ұғымды жиі алмастыру байқалады. Ал мұндай ой еңбегі тек жалпы білімді ғана емес, арнайы дайындықты да қажет етеді. Екінші жағынан, әрбір қоғамның ақпараттандыру деңгейі ақпарат саласында жұмыс істеушілер санымен анықталады. Сондықтан да жан – жақты білімді және компьютерлік сауатты мамандар дайындау қазіргі заманның үлкен қажеттілігі болып отыр. Мұндай жағдайларда ақпарат жүйелері бизнес жүргізу құралына, кәсіпорынға немесе бөлімге айналады. Қазіргі кезде қолмен пайдалынатын ақпараттық жүйе (есептеп шығару техникасының көмегінсіз) және автоматтандырылған ақпараттық жүйе болып бөлінеді.

Жұмыстың мақсаты. Қойларды есепке алу бойынша автоматтандырудың автоматтандырылған жүйесі мен қағидатын құру.

Автоматтандыру мал шаруашылығын дамытудың маңызды бағыты болып табылады. Өнеркәсіптік мал шаруашылығындағы өндіріс резервтері өндірісті басқаруды автоматтандыруда қамтылған. Осыған байланысты машиналар жүйесінде автоматтандыру әдістері мен құралдарын сапалы өзгерту қарастырылған.



## **1 Қойылған мәселенің қазіргі жағдайы**

### **1.1 Қойылған мәселенің қазіргі жағдайы бойынша әдебиетке шолу**

Жаңа заман әлемінде адамның кез-келген қызметі оның ақпараттануына және бар ақпаратты нәтижелі пайдалануына тікелей байланысты.

Ақпарат қоғамның маңызды ресурстарының біріне айналған және онсыз адамның алға басуы мүмкін емес. Қызметтің кез-келеген ортасында арнайы рационалды шешім іздеу программалы техникалық құралдарсыз жүзеге аспайтын үлкен көлемді ақпараттарды өңдеуді қажет етеді.

Мәліметтер базасымен ақпарат және ақпараттық жүйе түсінігін қарастырмай жұмыс істей алмаймыз. Кез-келген ақпараттық жүйе негізін мәліметтер базасы құрайды.

Мәліметтер базасы деп, мәліметтердің электрондық сақтаушысын айтады. Оларға қатынас, бір немесе бірнеше компьютерлер көмегімен іске асады. Мәліметтер базасы мәліметтерді сақтау үшін жасалады және ақпаратты сақтау мен мәліметтерге ыңғайлы, тез кіруді қамтамасыз етеді. Мәліметтер базасы өзінен белгілі бір ережелерге сай құрылған мәліметтер жиынтығын құрайды. Мәліметтер базасындағы ақпарат артықсыз, тұтас болуы керек. Қолданушыға қызмет көрсетілу ақпараттық жүйе арқылы орындалады.

Ақпараттық процестер адамдардың барлық қызметінің ортасында жүзеге асады: экономика, техника, ғылым, медицина және т.б. Адамның қызметі көбінесе нәтижелі ұйымдастырылған ақпарат алмасумен байланысты. Ақпарат көлемі шамамен өндіріс көлемінің даму квадратына пропорционалды өсуде деп бекітілген.

Күнделікті өмірде ақпарат түсінігін адамдар арқылы ауызша, жазбаша және басқа да әдістермен берілетін мәліметті түсіну қалыптасқан. Бірақ бұл толық анықтамаға сай емес.

Ақпаратты қабылдау, жіберу, түрлендіру, сақтау және қолдану құралы бола алатын кез-келген мәліметті түсінеді. Ақпарат көлемінің өсуі және күрделенуі мәліметтерді сақтау және жіберу кезіндегі реттеу қажеттілігін туғызды.

Алғашқы уақытта процесс жеке объектілерді автоматтауға бағытталған болатын, себебі сақталынатын мәліметтер көлемі шағын болатын. Бұл электронды техниканың дамуы негізінде мүмкін болды. Дегенмен, ЭЕМ қолдану үлкен көлемді ақпаратты сақтауға мүмкіндік берді. Бұл өз кезегінде ақпараттың өсуіне, сақталынатын мәліметтің құрылысының күрделенуіне ақпараттық жүйелердің автоматтану қажеттілігіне үлгі болды.

Ақпараттық жүйелер қазіргі кезде біздің өміріміздің ажырамас бөлігіне айналды. Сонымен қатар, қазіргі кезде ақпараттың көп бөлігі электронды түрде пайда болуда, бұл коммуникацияның бүгінгі даму барысында оны жоғары мобильділікпен, қол жетімділікпен және көпшілікке ортақтықпен қамтамасыз етеді.

Ақпараттық жүйе бұл – мәліметтер базасының жинағы. Ақпараттық жүйе-жинақтау, мәліметтерді басқару, өңдеу, сақтау, ақпаратты іздеу және құрамында

мәліметтерді өңдейтін техникалық құралдары бар, программалық қамтаманы жүзеге асыратын жүйе. Бұл мағынада ақпараттық жүйе бұл- мекеме, жоғары оқу орны, банк бөлімі және т.б. Мәліметтер базасы теориясында ақпараттық жүйе термині автоматталған ақпараттық жүйе қатынасында қолданылады.

Ең басында мәліметтерді өңдеу жүйесі тапсырмалы әдіс негізінде құрылған. Жеке, жинақталған тапсырмалар өз бетінше автоматталған. (еңбек ақы есептемесі, құрал-жабдықтар, кадрлар есебі және т.б). Әрбір тапсырма үшін өзінің мәліметтер блогы және қолданбалы программасы болады.

Ақпараттық жүйені құратын қазіргі заманғы әдіс олардың интеграциясына, яғни бүкіл мекеменің мәліметтер базасын сақтайтын біртұтас қоймасы, динамикалық жаңартылуы және қолданушыға әртүрлі қатынау мүмкіндігін беретін біртұтас ақпараттық жүйесін құруға негізделген. Бұл тиімді әдіс болып табылады, себебі әртүрлі қолданушылар үшін бірдей мәліметтерді сақтаудан және оларды жаңарту жағдайынан құтылуға мүмкіндік береді. (мысалы, бухгалтерияның мәліметтер базасы).

Ақпаратты-іздеу жүйесі негізінен көптеген сақталған мәліметтердің ішінен іздеудің белгілі бір критерийін қанағаттандыратын мәліметті бөліп алуға және сақтауға бағытталған. Бұл жерде қолданушы үшін мәліметті өңдейтін процесс емес, ақпараттың өзі маңызды болып табылады. Мысалы: анықтамалар, сөздіктер, анықтамалық қызметтердің ақпараттық жүйелері және т.б.

Мәліметтерді өңдейтін жүйелерде мәліметтермен жұмыс істеу процесі маңызды: белгілі бір уақыттағы жаңартылу, есеп беру құрастырылуы, шығарылатын мәліметтердің болмауы немесе сақталынған ақпараттың программалық өңделінген шешімінің көрсетілуі. Мысалы: Жинақ банкінің ақпараттық жүйесі (жаңартылу, қорытынды-жинақ бойынша есептеме, қаржылық есеп), Бухгалтерлік есептің ақпараттық жүйесі және т.б.

Ақпараттың үш түрі және ақпараттық жүйенің үш типі қарастырылады: құжаттық, иконографиялық, фактографиялық, сол себепті ақпараттық жүйе сақталынатын мәліметтің типі бойынша құжаттық, иконографиялық, фактографиялық деп бөлінеді. Құжаттық жүйеде әртүрлі мағынадағы құжаттар сақталынады. Бұл жерде көрсетілген баған бойынша сумманың ортақ мәнін шығаратын іздеу қажет, осы бойынша құжатты ақпараттық жүйе құжаттың өзін емес, құжаттың ізделінген үлгісін тауып береді әрі оның орналасқан орнын көрсетеді.

## **1.2 Ақпараттық жүйені жасау қажеттілігі**

Әрбір ақпараттық жүйе қорыта келгенде адамдар, мекемелер, бөлімдер және ұйымдар үшін негізделген. Сондықтан автоматталынатын ақпараттық жүйені құру мұқият зерттеу мен формализацияланған объектіні сипаттаудан басталады.

Бұл кезде біз мыналарды жүзеге асырамыз:

- автоматизация объектісінің мақсаттық қызметінің санын анықтау;

- автоматизация объектісіне берілетін алдан-ала анықталған талаптар;
- автоматизация объектісінің ұйымдық және топологиялық құрылмын анықтау;

- автоматизация объектісінің ішкі жүйесі бойынша функцияны орналастыру анализі.

Бұл кезде мыналар анықталады:

- әрбір ішкі жүйенің функционалды қызметі және олардың арасындағы функционалды ара-қатынас;

- ішкі жүйедегі ақпараттық ағындар.

Сонымен қатар автоматизация объектісінің функционалды моделінің алдын ала варианты құрылады. Оның құрамында:

- сыртқы объектінің идентификациясын және олармен ақпараттық ара-қатынасы;

- автоматизация объектісін негізгі қызмет деңгейіне дейін және осы қызмет арасындағы ақпараттық байланысты нақтылау.

Алынған мәліметтерді өңдеу нәтижесінде автоматизация объектісі қызметінің моделін құру жүзеге асырылады .

Модель өзінің құрамына толық структуралық, функционалды қызмет моделін және ақпараттық моделді қосады. Модель включает в себя полную структурную функциональную модель деятельности и информационную модель.

Жүйелік жобалаудың келесі кезеңінде ақпараттық жүйе туралы айтылады. Бұл кезеңде мыналар анықталады:

- жүйе архитектурасы, оның қызметі, оның қызмет атқаруының сыртқы шарттары, аппараттық және программалық бөлімдер арасында функцияның үлестірілуі интерфейстер және адам мен жүйе арасындағы функцияның үлестірілуі;

- жүйенің ақпараттық және программалық компоненттеріне қойылатын талаптар, қажетті аппараттық ресурстар, мәліметтер базасына қойылатын талаптар, жүйе компонентінің физикалық сипаттамасы;

- жүйеге қатынасы бар адамдар мен жұмысшылар құрамы;

- жасау процесіндегі шектеу (құрамында ресурсы бар жеке кезеңдерді директивті аяқтау мерзімі және ақпаратты қорғауды қамтамасыз ететін ұйымдастырылған процедуралар).

Техникалық жобаның мәліметтер базасын жасау процесінде мыналар орындалады:

- мәліметтер базасының концептуалды моделін жасау;

- мәліметтер базасының логикалық моделін құру Аппаратты конфигурацияны жасау және программалық платформаны таңдау;

- мәліметтер базасының физикалық моделін жобалау;

- ақпараттық жүйенің автоматтандырылған жұмыс орнын жобалау.

**Ақпараттық жүйені құруға қойылатын талаптар**

Ақпараттық жүйе – қиын жүйелер және элементтерді бөлек бөлімдерге бөлуін талап етеді. Барлық жүйелерге бірден жасау мүмкін болмаса, олардың бөлек бөлімдеріне жасау мүмкін болады. Бөлімдердің белгіленуі толығырақ

және нақтығырақ болса, жүйе элементтері, бөлімдерінің ара-қатынасы және тұрған жері анықталса, онда жүйенің қызмет істеуі және құру процесі тиімдірек болады.

Ақпараттық жүйе (АЖ) – қолданушылардың сұраныстарына сәйкес түрде ақпаратты жинау, сақтау, іздеу, өңдеу және жеткізу үшін арналған. АЖ программалық құралдар кешенінен шешілетін есептерді өзіне қосады. Қолданушы үшін жүйе ыңғайлы және жақсы интерфейске, санкцияланбаған әрекеттерден қорғану мүмкіндігіне ие болуы қажет.

АЖ құрылымы – ішкі жазықтық-уақыттық байланыстардың тұрақты тәртібі және элементтер мен ішкі жүйе бөлімдерінің арасындағы қатынасы. АЖ жабдықтаушы және функционалды бөлімдерден тұрады. Жабдықтаушы бөлімі – есептердің және есептер комплексінің немесе ішкі жүйелерінің жиынтығы, бұл басқарудың маңызды бөлігін құрайды.

Ақпараттық жүйе құрылымы – жүйешіктердің функционалды міндеті мен олардың сыртқы ортамен өзара әрекетін анықтайтын, оның жеке жүйешіктері мен элементтері арасындағы қатынастар мен ішкі кеңістіктік – уақыттық байланыстардың салыстырмалы орнықты реті.

Ақпараттық жүйені құраушы элементтер өзінің алатын орнына қарай функционалды және жабдықтайтын болып екіге бөлінеді. Ақпараттық жүйенің функционалды бөлімі – ішкі жүйе мен кешен жиынынан және басқарудың маңызды бөлімін құрайтын құрылымнан құралған жиынтық.

### **1.3 Ақпараттық жүйенің ресурстарына қойылатын талаптар**

Ақпараттық ресурстар деп материалдық ресурстар ретінде жүретін және қандай да бір форма үшін құндылықты білдіретін, мәліметтердің жиынтығын атайды. Оларға сыртқы жадыда сақталатын мәліметтердің негізгі және көмекші массивтері, және кірістік құжаттар жатады.

Автоматтық тәртіпте жүйені қалыптастыру үшін жүйенің қамтамсыз ету бөлігі қажет, мына ресурстардан тұрады – техникалық жабдықтаудан, математикалық жабдықтаудан, программалық ақпараттық және ұйымдастырушы жабдықтаудан.

Программалық және математикалық жабдықтау машиналық программалар жиынтығын, алгоритмдерді, экономика-математикалық әдістерді және модельдерді қосады.

Ақпараттық жабдықтау ақпараттық ресурстарды құрайды (кіру, уақытаралық және шығу тапсырмалары), олардың ену құралдары.

Ұйымдастырушы жабдықтау АЖ жоспарлау әдісі және құралы ұйымдастырушы жабдықтауға кіреді.

Өңделетін жүйеге келесі талаптар қойылуы қажет:

- Жүйелік программалық жабдықтау нақты уақытта жұмысқа еркіндік беру және кіру/шығу, оқу, жазу, баспаға шығару операцияларын жүргізуге мүмкіндік береді;

- Жүйелік программалық құралдар қосымша программалар пакетін ешқандай өзгеріссіз жаңа аппараттық платформаға ауысуына мүмкіндік беру қажет;

- Қосымша программалы құралдар мәліметтер базасымен байланысты қамтамасыз ету керек;

- Ақпараттық жүйені басқару оңай және көрсетілмелі болуы қажет;

- Жүйені аппараттық өткізу қолайлы арзан болу керек.

Жоғарыдағы талаптарды жүзеге асыру қарапайым және тиімді автоматты жүйе құруға септігін тигізеді. Бұл жүйе мықты бақылау үрдісін ұйымдастырып, кафедра құжаттарын ұйымдастырудың сапасын жоғарылататын іс-әрекеттердің дұрыс орындалуын қадағалап отыруға көмектеседі.

Ақпараттық қамтамаларға қойылатын талаптар

Басқару үрдісі ақпаратты пайдаланусыз мүмкін болмағандықтан, ақпараттық жүйеде ақпараттық қамтамасыз етуді ұйымдастыру әсіресе маңызды.

Ақпараттық қамтамасыздандыруды (АҚ) болып автоматтандырылған жүйеде пайдаланылатын, ақпарат массивтерінің және құжаттаманың үйлестірілген жүйелерінің, экономикалық ақпаратты кодтау және жіктеудің ортақ жүйесінің жиынтығы табылады.

Ақпараттық қамтамасыздандырудың негізгі міндеті – кафедра құжаттары бойынша ақпаратты жинауда, сақтауда, ақпараттық базаға өзгерістер енгізуде, басқарушылық шешімдерді қабылдау үшін шығыстық ақпаратты беруде құралады.

Ал олар өз кезегінде өзіне төмендегілерді қосады:

- ақпараттың сандық және сапалық сипаттамалары;

- ақпараттың құрамы қандай;

- ақпаратты іздеу уақытының азаюы;

- келіп түсуші ақпараттың өз уақыттылығы мен сенімділігі;

- қандай да бір шарттардың өзгеруінің нәтижесінде ақпараттық қамтамасыз етудің бүкіл жүйесінің даму мүмкіндігі.

Ақпараттық қамтамасыз етуді ұйымдастырудың принциптері және құрылымы, құрамы АЖ-нің есептер орналымымен сәйкес түрде келесі талаптарды қанағаттандыруы тиіс:

- мәліметтер базасының барлық элементтеріне жататын құрылым деңгейінен тәуелсіз түрде оңай қол жеткізу;

- мәліметтер базасын ұйымдастырудың алгоритмнің функционалдық міндетінен тәуелсіздіктері;

- басқару жүйесінің барлық функцияларын жүзеге асыру үшін ақпараттың жеткіліктілігі;

- басқару жүйесін дамытудың бағыттарын ескеріп, мәліметтер базасын кеңейтудің мүмкіндіктері;

- ақпаратты сақтаудың кедергілерден қорғалғандығы мен сенімділіктері.

Ақпараттық қамтамасыз етуге қойылатын талаптарда өздерінің ерекшеліктері бар:

1. Машинаға енгізілетін мәліметтер қалыптастырылған және бір мәнді көрсетілуі тиіс;

2. Машинаға енгізілген мәліметтер олардың ары қарай өңделуі жүретін программалық құралмен дұрыс қабылдануы тиіс;

3. Машиналық массивтер машинаның шектелген техникалық мүмкіндіктерін ескере отырып, егер мұндайлар бар болса, ұйымдастырылуы қажет.

Ақпаратпен жабдықтау – бұл ақпараттық жүйеде қолданылатын экономикалық ақпаратты жіктеудің және шартты белгілеудің бірінғай жүйелері мен құжаттардың және ақпарат массивтердің үйлестірілген жүйелерінің бірігуі болып табылады. Ақпаратпен жабдықтаудың негізгі міндетіне – ақпаратты сақтау мен жинау, ақпараттар қоймасына өзгерістер енгізу, басқару шешімдерін қабылдау үшін шығатын ақпаратты беру жатады.

Ақпараттық жүйедегі ақпаратпен жабдықтау, жүйеде қолданылатын мөлшерлік-анықтама ақпаратты (МАА) және экономикалық ақпарат пен құжаттардың жіктелуін қамтиды.

Ақпарат қоймасы – ақпараттық жүйенің жұмыс істеуде қолданылатын реттелген ақпараттың бірігуі.

Машинадан тыс ақпарат қоймасы – ақпараттық жүйені жұмыс істетуде қолданылатын құжаттар мен белгілердің, мәлімдемелердің бірігуін бейнелеп, есептеу техникасынсыз (ЕТ) адамға түсінікті болатын ақпарат қоймасының бір бөлігі. Машинадан тыс ақпарат қоймасына жіктеу мен шартты белгілеу жүйелері мөлшерлік-анықтама құжаттар, жедел ақпарат жатады. Жіктеу мен шартты белгілеу жүйелері мәліметтерді бірімәнді жазып, ақпаратты есептеу желісінің жадында тиімді іздеп және белгілеу үшін қолданылады. Ал мөлшерлік-анықтама құжаттар (МАҚ) – шартты-тұрақты ақпарат қатарына жатады жатады және ұзақ уақыт бойы есеп айырысуда өзгеріссіз қолданылатын әртүрлі тәртіптегіш мөлшерлерді және басқа да мәліметтерді құрайды.

Машина ішіндегі ақпарат қоймасы – бұл ақпараттық жүйе жұмыс істеуде қолданылатын, машиналық тасығыштардағы мәліметтер массивінің бірігуін бейнелейтін ақпарат қорының бөлігі.

Ақпараттық жүйелердің классификациясы.

Ақпараттық жүйе – қойылған мақсатқа жетуде қолданылатын ақпаратты өңдеу, сақтау, тасымалдауда қолданылатын әдіс, құралдардың өзара байланысы. Ақпараттық жүйенің құрылымы көптеген жүйелерді қамтушы жүйе ретінде ақпараттық, техникалық, математикалық, программалық қамтамалардан тұрады.

1. Тапсырманың құрылымы бойынша сипатталатын ақпараттық жүйелердің классификациясы:

- құрылымдық тапсырмаларға арналған ;

-бөліктей құрылымдалған және құрылымдалған тапсырмалар арналған:

а) басқарушы есеп;

ә) өңделетін альтернативті шешімдер (модульдік және эсперттік).

2. Басқару деңгейі және функционалдығы бойынша ақпараттық жүйелердің классификациясы:

- өндіріс жүйелер;
- маркетинг жүйелер;
- финанстық және тіркеу жүйелер;
- кадрлар жүйесі.

3. Автоматтандырылу дәрежесіне байланысты классификациясы:

- қолмен;
- автоматты;
- автоматтандырылған.

4. Ақпаратты қолдану мінездемесіне байланысты классификациясы:

- ақпаратты іздеу жүйесі;
- ақпаратты шешу жүйе;
- басқарушы ақпараттық жүйе;
- мақұлданған ақпараттық жүйе.

5. Қолдану сферасы бойынша классификациясы:

- ұйымдастырушы басқарудың ақпараттық жүйесі;
- техникалық процессті басқарудың ақпараттық жүйесі;
- автоматтандырылған жобалаудың ақпараттық жүйесі;
- интеграцияланған ақпараттық жүйе.

Автоматтандырылған ақпараттық жүйе. Автоматтандырылған ақпараттық жүйе ақпаратты сақтау, өңдеу, тасымалдауға байланысты әрекеттерді автоматтандыруға арналған бағдарламалық ақпараттық құралдардың жиынтығы.

Автоматтандырылған ақпараттық жүйе – қолданушыларға шешім қабылдауда қолданылатын ақпаратты ұйымдастырып ұсынудағы автоматтандырылған ақпараттық технологиялардың кешені. Автоматтандырылған ақпараттық жүйелердің классикалық мысалы банк жүйесін, мекемелерді басқарудың автоматтандырылған жүйесі, авиациялық немесе теміржол билеттеріне алдын-ала тапсырыс беретін, кафедра құжаттарын автоматтандыру жүйелері жатады.

Ақпараттық программалық жүйелердің спецификациясы

Нақты облысқа байланысты ақпараттық жүйелерді қолдану құрылымдары, архитектурасы бойынша ерекшеленеді.

Дегенмен барлық ақпараттық жүйелер үшін жалпы деп саналатын екі қажетті ерекшелеуге болады. Біріншіден, кез-келген ақпараттық жүйе ақпаратты өңдеу, сақтау және жинақтауға арналған. Сондықтан кез-келген ақпараттық жүйені қолдану облысына сәйкес келетін сақтаудың сенімділігін және мәліметтерге қатынауын қамтамасыз етуі тиіс. Айта кету керек, есептеу программалық жүйелерінде бұндай ортаның болуы міндетті емес болып табылады. Сандық есептеулер жүргізуші программаға қойылатын басты талап, оның жылдам жұмыс істеуі: программа берілген уақыт ішінде нақты нәтиже беруі керек. Күрделі есептерді шешу кезінде, тіпті суперкомпьютерлерді пайдалансақ та, шешуге кететін уақыт апталап, кейде айлап саналуы мүмкін. Сондықтан, есептеуші-программистер өте көп уақыт аралығында өңделуші мәліметтер компьютердің негізгі жадысында сақталатындай етіп программаны ұйымдастырады, өйткені олар мәліметтерді сыртқы жадыда сақтауға өте күмәнді

қарайды. Сыртқы жады, әдетте есептеудің аралық нәтижелерін сақтау үшін ғана қолданылады. Автоматтық тәртіпте жүйені қалыптастыру үшін жүйенің қамтамсыз

ету бөлігі қажет, мына ресурстардан тұрады – техникалық жабдықтаудан, математикалық жабдықтаудан, программалық ақпараттық және ұйымдастырушы жабдықтаудан.

Екіншіден, ақпараттық жүйелер ақырғы қолданушыға бағытталады, мысалға банктік клеркті алсақ болады. Бұндай қолданушылардың компьютерден хабары өте аз болуы мүмкін. Олар үшін, терминал, персоналды компьютері немесе жұмыс орталығы тек өзінің профессионалды қызметінің құралы болып табылады. Сондықтан, ақпараттық жүйе қарапайым, ыңғайлы, жеңіл игеруге болатындай интерфейске ие болуы керек және ол интерфейс оны қолданушының жұмыс жасауына қажетті барлық функцияларды қамти алуы керек. Бірақ, сонымен бірге қолданушыға артық іс-әрекет жасауға да мүмкіндік бермеуі керек. Кей-кезде, осы интерфейс меню, батырмалар және т.б-мен графикалық түрде болуы мүмкін. Қазіргі кезде графикалық интерфейсдер кең танымал және қазіргі заманғы ақпараттық қосымшаларды құратын көптеген құралдар ең алдымен графикалық интерфейссті құруға бағытталған. Бір жағынан көптеген қолданушылар (мысалы, банк жұмысшылары) графикалық терминалдарды ұнатпайды. Оның орнына ақпараттық жүйеге қатынайтын бұрынғы интерфейсстік құралдарды қолдануды жөн көреді. Дегенмен, дамыған интерфейсстік құралдарды қолдану әрбір қазіргі заманғы ақпараттық жүйе үшін міндетті болып табылады.



## 2 Есептің қойылымы

Кәсіпорында малмен және құжаттармен жұмыс істеу көп уақыттың көп бөлігін алады. Сондықтанда бұл салада жұмыс істеу барысындағы іс-әрекеттердің тиімділігін арттыру үшін ақпараттық жүйелердің қажеттілігі өте маңызды.

Осы ақпараттық жүйені құру барысында объектілі- бағытталған бағдарламалау тілі C# және microsoft .NET Framework платформасында SQLite 3 дерекқоры қолданылды.

C# бағдарламалау бағдарламалау ортасы

C# бағдарламалау тілі жаңа .NET технологияларын қолдайтын объекті-бағытталған бағдарламалау тілі болып табылады. Кез келген тілде сияқты C# тілінде алфавит– символдар жиыны бар, олардың көмегімен бағдарлама коды (мәтін) жазылады. C# тілінде бағдарламаны жазу үшін келесі символдар қолданылады:

- кез келген әріптерді, C# тілі бас және кіші әріптерді ажыратады;
- 0 - 9 дейінгі цифрлар;
- қызметтік белгілер;
- қызметтік сөздер;
- бағдарлама мәтіні бойынша түсініктемелер.

Әріптер мен цифрлар бойынша барлығы түсінікті, ал қызметтік белгілер мен сөздерді (олардың саны өте көп) тілді оқу барысында меңгереміз.

Бағдарлама кодын жазу барысында оның әр түрлі конструкциясының (операторлар, функциялар, объекттер, жазулар, айнымалылар, т.б.) аттары, яғни идентификаторлары анықталуы тиіс. Олар әріптерден (латын алфавитін қолдану қажет), цифрлардан (0 – 9), астын сызу символынан тұруы мүмкін, бірақ идентификатордың бірінші символы әріп немесе астын сызу символы болуы керек, сан болмауы тиіс. Мысалы:

- дұрыс – A\_9, B34, TУTI, Max; – дұрыс емес – A-9, B 34, 13FIG.

Айнымалыларды сипаттау

Бағдарламалау тілінің «айнымалы» ұғымының алуан түрлі анықтамасы бар, бірақ компьютердің аппараттық қамтамасыз ету тұрғысынан ең дұрыс анықтамасы мынандай – бағдарламаны орындау уақытының әр түрлі мезетінде түрлі мәндерді болуы мүмкін компьютер жадысының аймағы. Осы жадының аймағына сәйкес қолданылатын символикалық ат айнымалы аты немесе айнымалы идентификаторы деп аталады.

Айнымалы - идентификатормен белгіленген компьютер жадысының аймағы, онда бағдарламаның жұмысы кезінде өзгертін деректер сақталады.

Бағдарламада айнымалыны қолданбас бұрын оны жариялау керек – оның атын және типін көрсету керек.

Айнымалы типі оған қандай мәндерді жазуға болатынын көрсетеді – бүтін, нақты, символдық, т.б.

Айнымалы аты идентификатор болып келеді.

Мысалы: int ТҮТ; float  
sum, kol;

Мұнда, ТҮТ айнымалысы – бүтін, ал sum, kol айнымалылары – нақты (бөлшек) типте екені көрсетілген.

Бір типті айнымалылар нүктелі үтірмен аяқталуы тиіс.

Айнымалыны жариялау кезінде жариялау операторы қолданылады - нүктелі үтірмен аяқталатын әрекет. Бірақ әдебиеттерде бұл амалды жариялау операторы деп атаудың орнына жай ғана бағдарлама айнымалысын жариялау деп атайды.

Айнымалыны бірінші рет қолдану жариялау болып есептелетін BASIC бағдарламалау тілінен немесе айнымалылары бағдарлама басында, арнайы бөлімде жарияланатын PASCAL бағдарламалау тілінен өзгеше C# бағдарламалау тілінде айнымалы бағдарламаның кез келген орнында жариялана алады. Сонымен қатар айнымалыны PASCAL тіліне ұқсас бағдарламаның басында немесе BASIC тіліне ұқсас бірінші рет қолдану барысында жариялауға болады.

C# тілінің деректер типі

.NET бағдарламалау технологиясының жаңа бір міндеті CTS

(Common Type System) – жалпы типтер жүйесін қолдану болып табылады.

Ол осы технологиямен жұмыс істейтін кез келген бағдарламау тілі үшін компьютер жадында деректер көрсетімін стандарттауға мүмкіндік береді. Шартты түрде барлық CTS деректер типі мәнді (бүтін, нақты, т.б.) және сілтемелік (массивтер, кластар, т.б.) болып бөлінеді. Компьютер жадысы мәнді типтегі айнымалыларға бағдарлама компиляциясы кезінде, ал сілтемелік типтегі айнымалыларға new операторының көмегімен бағдарлама орындалу барысында бөлінеді.

.NET технологиясында деректер типін белгілеу күрделі иерархиялық құрылымға ие, мысалы, System.Int32. Сондықтан C# тілінде кейбір жиі қолданылатын типтердің жазуын жеңілдету үшін қарапайым типтер ұғымы енгізілді. Қарапайым типтер бұл – кейбір мәнді және сілтемелік типтердің қысқартылып жазылған түрі. 2.1-кестесінде C# тілінің қарапайым типтері және осы типтерге сәйкес CTS типтері көрсетілген.

2.1-кестесі – C# тілінің қарапайым типтері

C# тілінің қарапайым типтері	.NET платформасының CTS типтері
byte	System.Byte
sbyte	System.SByte
short	System.Int16
int	System.Int32
long	System.Int64
ushort	System.UInt16
uint	System.UInt32

ulong	System.UInt64
float	System.Single
double	System.Double
object	System.Object
char	System.Char
string	System.String
decimal	System.Decimal
bool	System.Boolean

Бағдарламада деректер типін таңдауда есептеу дәлдігінің талаптары және компьютердің айнаымалыларға бөлінетін жады көлемі ескеріледі.

Бағдарламалауды меңгерудің бірінші кезеңінде біз тек қана қарапайым типтерді ғана қолданамыз, ал қалған басқа деректер типтерін материалды меңгеру барысында қарастырамыз.

C# тілінің тұрақтылары

C# тілінде тұрақтыларды жариялау айнаымалыларды жариялаумен бірдей, бірақ, const қызметтік сөзі қосылып жазылады.

Мысалы:

```
const char CIMV = 'y'; const int MAX = 640;
```

Тұрақты бұл – идентификатормен белгіленетін, бағдарлама жұмысының барысында өзгермейтін деректердің мәні сақталатын компьютер жадысының аймағы.

Тұрақтыны жариялаған кезде оның типі ғана емес, сонымен бірге оның мәні де көрсетілуі керек.

C# тілінің атаулар кеңістігінің ұғымы

C# тіліндегі кез келген бағдарлама using операторының көмегімен бағдарлама кодына кейбір атаулар кеңістігін көрсетуден басталады.

Бұл ретте бір using операторына тек бір атаулар кеңістігі тиісті бола алады.

Әрбір атаулар кеңістігі .NET платформасына тиісті кластардың белгілі бір тобына сәйкестенеді (әрбір класс - белгілі бір тип). Сонымен, .NET платформасының барлық типтер жиынтығы (CTS-те 4000 аса түрлі типтер белгілі) өзінің функционалды міндеті бойынша логикалық байланысқан топтарға біріктірілді, олар атаулар кеңістігі деп аталады. Кейбір міндеттері шешу үшін кейбір кластар, әдістер, функциялар немесе деректер керек болса, онда сіз бағдарламаға тиісті атаулар кеңістігін қосуыңыз керек. Сонымен, C# тілінде NET платформасының бұрында жазылған кластар кітапханасын қолдану мүмкіндігі орындалады.

Ескере кететін бір жағдай, C# тілі бойынша оқулықтардың көптеген авторлары кітапхана терминін қолданбай, атаулар кеңістігі ұғымымен шектеледі. 2.2-кестесінде ең жиі қолданылатын атаулар кеңістігі көрсетілген

Бағдарламалауға жаңадан келген бағдарламашылардың келелі мәселелерінің бірі - керекті атаулар кеңістігінің атын және оны қосу жолдарын анықтау. Атаулар кеңістігімен жұмыс жасау технологиясын C# тілін менгеру барысына қарай қарастырамыз.

Айта кететін жәйт, түрлі Visual Studio орталарының «қосымшаларды дайындау шеберлері» керекті атаулар кеңістігін автоматты түрде таңдап алады, қажеттілік болмаса оларды өзгертпеген дұрыс. Мысалы, егер Visual Studio 2022 ортасында консольде орындалатын қосымша (Fail->New>Project->Console Application) таңдалса, онда қосымшаны дайындау шебері 1.1-суретінде көрсетілген атаулар кеңістігін автоматты түрде қосады.

## 2.2-кестесі – .NET платформасының кейбір атаулар кеңістігі

.NET платформасының кейбір атаулар кеңістігі	Тағайындалуы
System	Object класы бар түпкі атаулар кеңістігі және қарапайым типтегі деректермен, математикалық функция жинағымен, деректерді енгізу–шығарумен, қоқысты жинау операциясымен жұмыс жасауға арналған кластар жиынтығы, т.б.
System.Data System.data.SqlClient және т.б.	Бұл атаулар кеңістігі деректер базасымен жұмыс жасауға арналған
System.IO	Бұл атаулар кеңістігі файлға деректерді енгізу–шығаруға жауап береді, т.б.
System.Drawing System.Drawing. Drawing2D	Бұл атаулар кеңістігінің кластарында графикалық қарапайым құралдары, қаріптер жиыны, сызықтар түрлері, монитор экранында графикалық ақпаратты көрсету құралдарының жиындары бар.
System.Net	Кластар жиыны желілер бойынша деректерді табыстауға жауап береді.
System. Security	Кластар жиыны желілер бойынша деректерді табыстау қауіпсіздігін артыру үшін қолданылады.
System.Web	Кластар жиыны web-қосымшаларда жұмыс жасауға арналады.
System.Windows.Form	Бұл атаулар кеңістігінің кластары Windows интерфейсінің элементтерімен - терезелер, батырмалар, басқа да басқару элементтерімен жұмыс жасауға арналған

Бағдарламаның атаулар кеңістігін анықтағаннан кейін Visual Studio

2022 ортасында консольді қосымшаны дайындау шебері namespace Console Application { . . . } арнайы нұсқауымен бағдарлама кодының аумағын анықтады, онда құрылған қосымшаның деректер типін қолдануға болады.

Айта кететін жәйт, ортаның бас терезесінің сыртқы көрінісі терезені күйге келтіруге байланысты, оны бағдарламашы «өзіне керекті нұсқада» орындайды. Ортаны меңгеру кезеңінде күйге келтіру сипаттамаларын өзгертпеуді ұсынамыз.

Есепті шешуге арналған атаулар кеңістігін анықтап алғаннан кейін бағдарлама класын сипаттау басталады.

Класс дегеніміз бұл деректерді (өрістерді) және функцияларды (әдістерді) біріктіретін, осы деректерді біртұтас өңдейтін – код үзіндісі (тип).

Анықтама бойынша кез келген бағдарлама деректер мен оларды өңдеу әдістері бар код болып есептеледі. Сондықтан C# тілінде түрлі есептерді шешетін бағдарлама кодтары бағдарлама кластарына орналастырылады. Класс сипаттамасы class қызметтік сөзінен кейін басталады, қызметтік сөзден кейін класс атауы, одан кейін фигуралық жақшаның ішіне деректер мен оларды өңдеу әдістері орналастырылады.

C# тіліндегі бағдарлама класының сипаттамасының ішінде Main() атауы бар әдіс орналасуы керек. Осы әдістен бастап бағдарламаның орындалуы басталады. Назар аударыңыз, C# тілінде бас және кіші әріптер ажыратылады.

Егер есеп қарапайым болса, онда барлық әдістер Main() әдісінде ғана жазыла алады. Егер есепті бөлшектеу қажет болса (яғни басқа кластармен немесе әдістермен орындалатын бөлек үзінділерге бөлу), онда бағдарлама класында осы үзінділерді тиісті түрде сипаттау керек, ал Main() әдісінде оларды іске қосу ретін көрсетуіңіз керек. \

Бағдарлама мысалы

Бағдарлама кодын дайындау үшін компьютерде C# бағдарламау тілінің компиляторын орнату керек. Қазіргі уақытта C# тілінің бірнеше компиляторлары белгілі, мысалы, Visual Studio 2022 визуалды бағдарламалау ортасы, Turbo C# Explorer, т.б.

Әрбір компилятордың бағдарламаны дайындаған кезде ескеретін өз ерекшеліктері бар. Оқу құралында Visual Studio 2022 визуалды бағдарламалау ортасында C# компиляторын қолдануға бағытталған.

SQL тілі

SQL мәліметтерді өңдеу және оқу үшін арналған құрылғы болып табылады. Ол компьютердің мәліметтер базасында орналасқан SQL – бұл сұраныстың тілінің құрылымдық атының қысқаруы (Structured Query Language). Атына сәйкес SQL тілі қолданушы мен мәліметтер базасының ара қатынас ұйымдастыруда қолданылатын программалау тілі болып есептеледі.

Шын мәнінде SQL реляциялық типтегі мәліметтер базасымен ғана жұмыс істейді. Суретте SQL-ң жұмыс істеу сызбасы көрсетілген [12]. Бұл сызбаға қарап есептің жүйесінде маңызды ақпараты бар мәліметтер базасын көреміз. Егер есеп жүйесі бизнес сферасына қатысты болса, онда мәліметтер базасында материалдық құндылықтар, шығарылатын өнімдер, сату көлемі және жалақы туралы ақпараттар сақталынады. Дербес компьютердің мәліметтер базасында үлкен есептеу жүйесінен алынған қол қойылған чектер туралы, мекен –жайы, телефондар туралы ақпараттар сақталынған. Мәліметтер базасын басқаратын компьютерлік бағдарлама немесе ДҚБЖ деп аталады.

Егер қолданушыға мәліметтер базасынан мәліметті алуға қажетті болса, SQL көмегімен ДҚБЖ ға сұраныс жібереді. ДҚБЖ сұранысы өңдеп қажетті мәліметті тауып қолданушы жібереді. Мәліметтерді сұрау және нәтижені алу процесі мәліметтер базасына сұраныс деп аталады. Сондықтан жүйеленген сұраныс тілі деген ат осыдан пайда болған.

Бірақ бұл атау шындыққат сәйкес келмейді. Біріншіден, қазіргі SQL карапайым сұранымдарды туғызатын құрылғыданда үлкен, бірақ бастапқыдан тек қана сол үшін арналған болатын. SQLдің ең негізгі функциясының бір мәліметтерді оқи алатындығы болғанымен қазіргі кезде бұл тіл барлық функционалдық тілді іске асыруда ДҚБЖ- ң қолданушыға беретін мүмкіндіктерін айтайық.

Мәліметтерді ұйымдастыру. SQL қолданушыға мәліметтердің жалпы құрлысын өзгертуге және мәліметтер базасының элементтері арасындағы қатынас орнатуға мүмкіндік береді.

Мәліметтерді оқу. SQL қолданушыға немесе қосымшаға мәліметтер базасында бар мәліметтерді оқуға және оларды қолдануға мүмкіндік береді.

Мәліметтерді өңдеу. SQL қолданушыға немесе қосымшаға мәліметтер базасын өзгертуге, яғни жаңа мәліметтерді қосуға, сонымен қоса бұрыннан бар мәліметтерді жаңартуға немесе жоюға мүмкіндік береді.

Қатынасты басқару. SQL-дің көмегімен қолданушыға мәліметтерді оқуда және өзгертуде мүмкіндіктерін шектей отырып санкцияланған қатынастан қорғау.

Мәліметтерді бірлесе отырып қолдану. SQL қолданушылар арасында мәліметтерді бірлесе отырып пайдалануды қамтамасыз етеді. Олар бір – біріне кедерлі келтірмеу үшін параллельді түрде жұмыс істеуді.

Мәліметтердің тұтастығы. SQL мәліметтер базасының тұтастығын қамтамасыз ете отырып, келіспеген өзгертуден және жүйенің бас тртуынан қорғайды. Сондықтан SQL ДҚБЖ –мен қатынасуда ең мықта тіл болып табылады.

Екіншіден, SQL – бұл COBOL, FORTRAN немес Стипіндегі толық емес компьютерлік тіл. SQL–де шарттарды тексеретін IF операторы, тасмалдауды ұйымдастыратын GOTO және циклдарды құратн DO немесе FOR операторы жоқ. SQL мәліметтер базасы тілінің астары болып табылады. Оның құрамына 30-ға жуық мәліметтер қорын басқаратын операторлар кіреді. SQL операторы базалық тәлге қосымша болады. Мысалы, COBOL, FORTRAN немесе С және мәліметтер базасына қатынас жасауға мүмкіндік береді. Онымен қоса С сияқты мұндай тілдерден SQL операторын ДҚБЖ-ға анық түрде жіберуге болады. Ол функцияларды шақыру интерфейімен іске асады.

Қорыта келсек SQL бұл С немес Pascal сияқты тілдермен салыстырғанда әлсіз жүйеленген тіл болып есептелінеді. SQL-дің операторы ағылшын сөйлемдерін елестетеді және оператордың мәнне әсер етпейтін сөздерден тұрады. Бірақ бұл сөздер оның оқылын жеңілдетеді. SQL-де логикасы жоқ сөздер тіптен кездеспейді. Онымен қоса SQL операторларын жасамайтын арнайы ережелер тобы кездеседі. Олар дұрыс болғанымен ешқандай мәні жоқ.

Өзінің нақты аты болмаса, SQL- реляциондық мәліметтер базасымен жұмыс істейтін жалғыз стандартты тіл болып есептелінеді. SQL- бұл айтарлықтай мықты және оқуға оңай тіл.

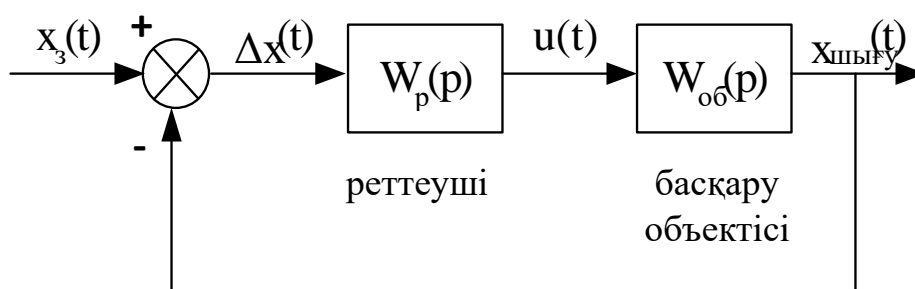
## 2.1 «Қойларды есепке алу» ақпараттық жүйесін құру

Кез келген ақпараттық жүйенің мақсаты - оның қолданылу аумағынан құрудың әдістерінен, техникалық базадан, байланысты қолдау аппаратынан тәуелсіз түрде толық, сенімділікті және өз уақытында ақпаратты беру.

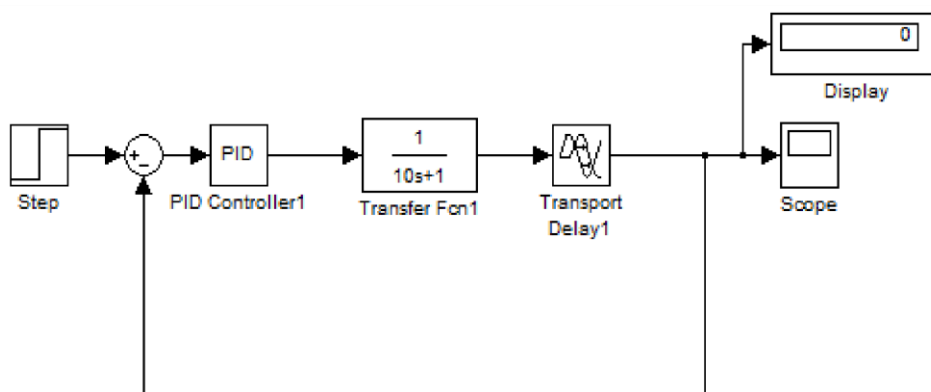
Жүйенің функционалдық және ақпараттық құрылымы ары қарай дамуы мен міндеттердің қосымша кешендерін қосу үшін ашық болуы тиіс.

Кәсіпорын меңгерушісі қойлар есебін реттеуде көптеген қиындықтарға тап болуы мүмкін. Ұйымдастыру барысында негізгі базалық мәліметтерді бірнеше рет қолдануы мүмкін. Сондықтан ақпаратты бірнеше қайтара енгізуді болдырмау үшін автоматтандырылған ақпараттық жүйелерді ендіру қажет.

Кәсіпорынның қойлар есебінің ұйымдастыратын бөлімдердің жұмысының тиімділігін арттыру нәтижесінде жұмыс процессінің графигін және жоспарын ұйымдастыруды жетілдіру процессінің сапасын арттырады.



1 сурет – Бір тізбекті САР схемасы

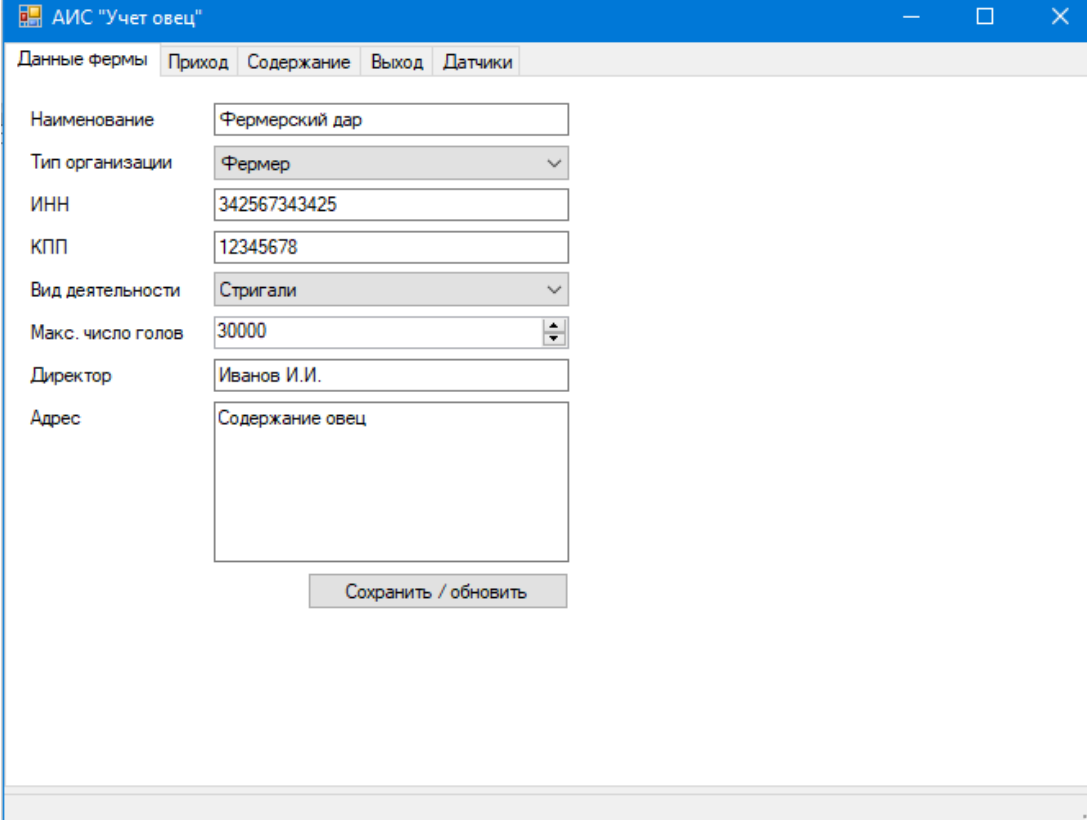


2 сурет – Бір тізбекті САР моделі

## 2.2 Қолданушы интерфейсі

Жұмыс бес формадан тұрады:

- Данные фермы;
- Приход;
- Содержание;
- Выход датчики.



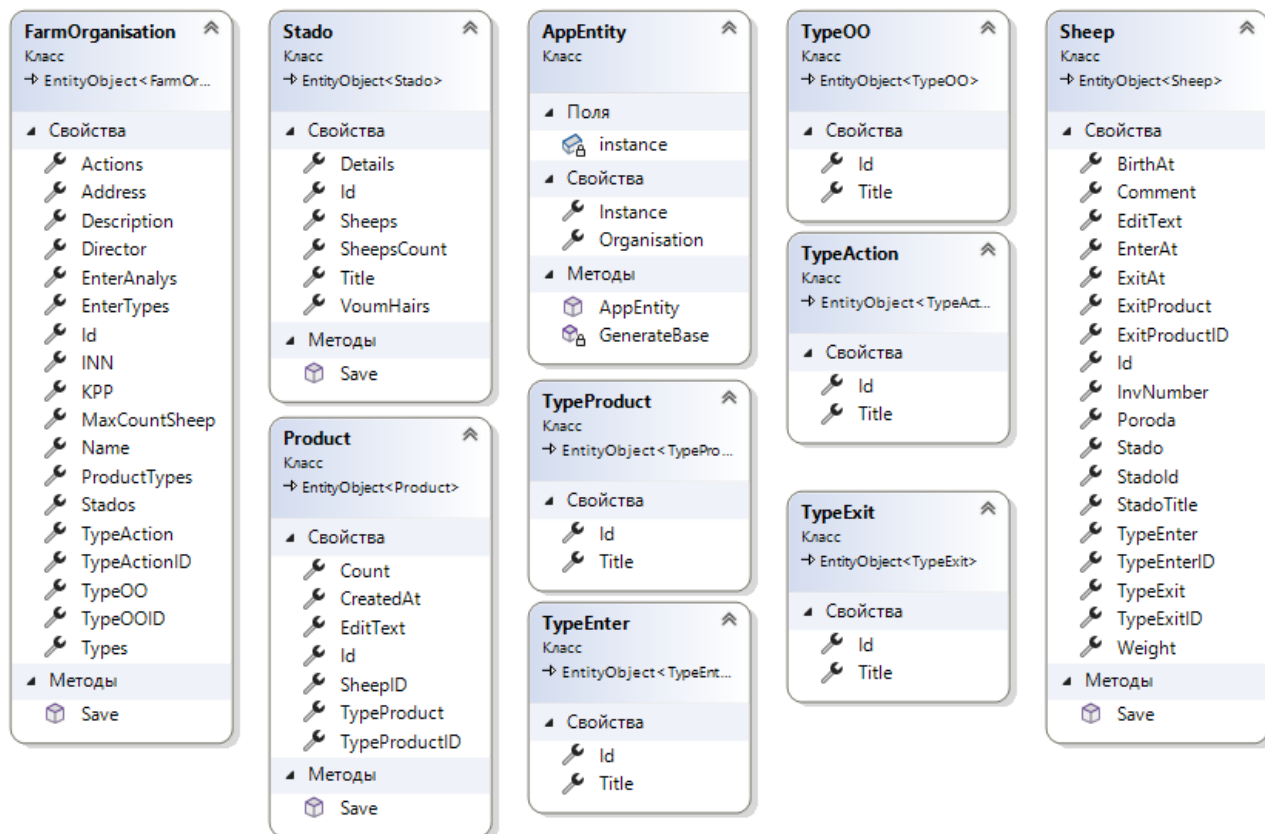
The screenshot shows a software window titled "АИС 'Учет овец'" with a blue header bar. Below the header is a tabbed interface with five tabs: "Данные фермы", "Приход", "Содержание", "Выход", and "Датчики". The "Данные фермы" tab is active. The form contains the following fields:

Наименование	Фермерский дар
Тип организации	Фермер
ИНН	342567343425
КПП	12345678
Вид деятельности	Стригали
Макс. число голов	30000
Директор	Иванов И.И.
Адрес	Содержание овец

At the bottom of the form is a button labeled "Сохранить / обновить".

3 сурет – Мәліметтер қоры бағдарламаның негізгі түрі





4 сурет – Блок-схема (кластық диаграммасы)

Біз әзірлеген бағдарламаның негізгі терезесінде сіз Экономика туралы ақпаратты көре аласыз:- Ұйымның атауы;- Ұйым түрі;- ЖСН;- Қызмет түрі;- Қой бастарының саны;- Ұйым басшысы;- Фирманың мекен-жайы.

АИС "Учет овец"

Данные фермы | Приход | Содержание | Выход

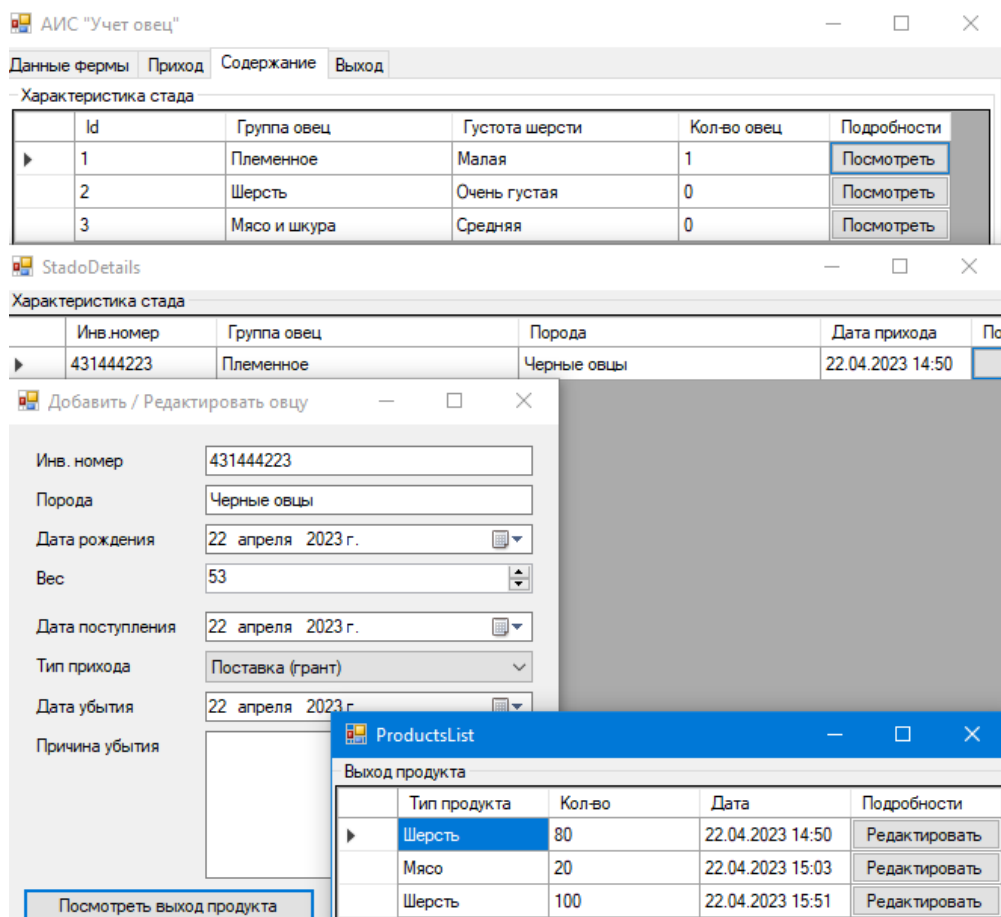
Наименование	Фермерский дар
Тип организации	ИП
ИНН	342567343425
КПП	12345678
Вид деятельности	Племенное хозяйство
Макс. число голов	30000
Директор	Иванов И.И.
Адрес	Содержание овец

Сохранить / обновить

5 сурет – «Данные фермы» терезесінің құрылымы

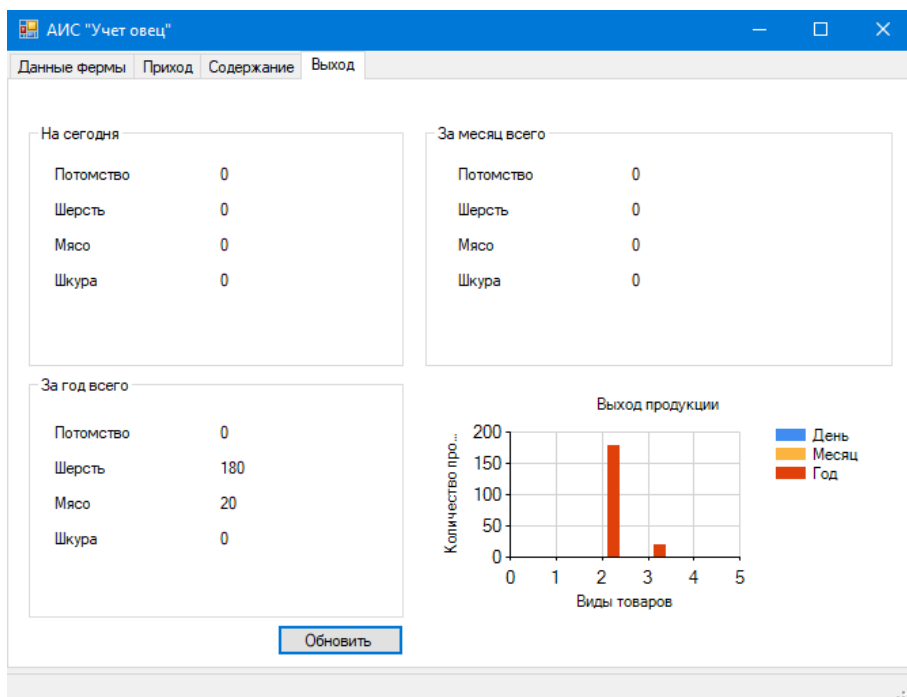
Жұмыстың терезесі - «Содержание» бөлімі. Мұнда біз қойлар туралы ақпаратты басқара аламыз, атап айтқанда:

1. Қойдың жеке нөмірі
2. Тұқым
3. Туған күні
4. Салмақ
5. Келу күні
6. Приход түрі
7. Кету күні
8. Сондай-ақ қойдан алынған өнімнің түрі мен мөлшері (жүн, ет)



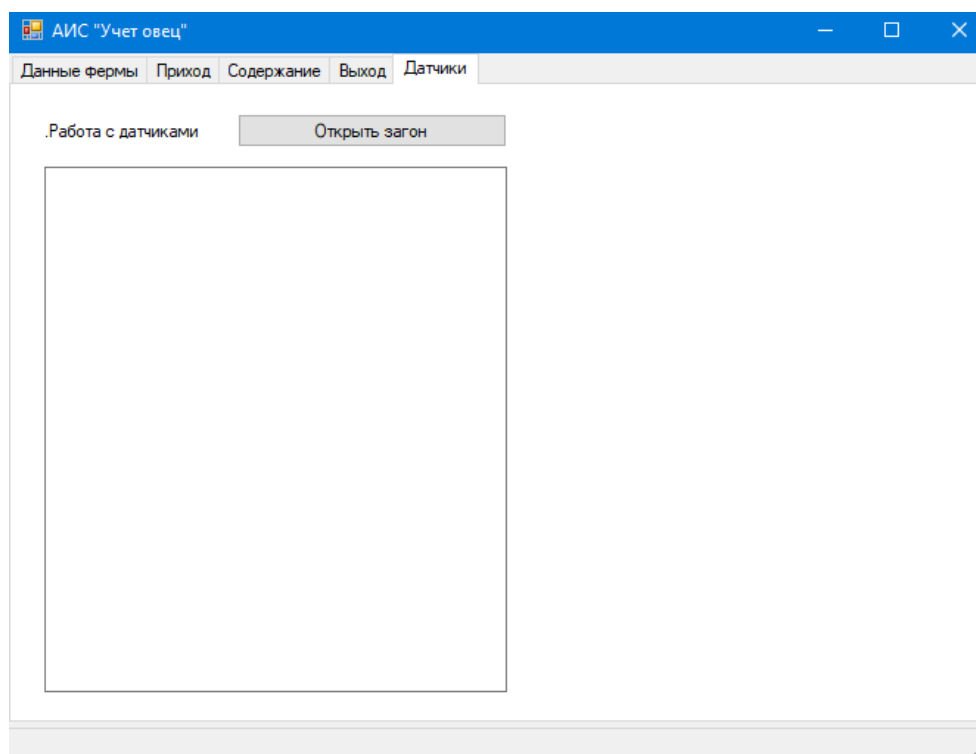
6 сурет – «Содержание» терезесінің құрылымы

Шығу терезесі алынған ұрпақ, жүн, ет, тері (ай мен жыл ішінде) туралы ақпаратты көруге мүмкіндік береді (сурет 5).



7 сурет – «Выход» терезесінің құрылымы

Бағдарламада қойларды есепке алуды автоматтандыру «Датчики» терезесінде ұсынылған.



8 сурет – «Датчики» терезесінің құрылымы

Жұмыстың мәні келесідей:

Әр қойдың құлағына F4R маркалы электронды құлақ белгісі бекітілген, ол қойларды анықтауға мүмкіндік береді.

F4R HDX/M2L тегтер жинағы қайта пайдалануға болады. Жартылай дуплекс.



9 сурет – Қойларды анықтауға арналған F4R электронды құлақ белгісі

Электрондық құлақ белгісі HDX-қайта пайдалануға болатын (ана) •  
өлшемдері: диаметрі 30 мм салмағы: 6,75 грамм.

M21 визуалды дөңгелек тег (әке) өлшемдері: диаметрі 30 мм. салмағы: 2  
грамм

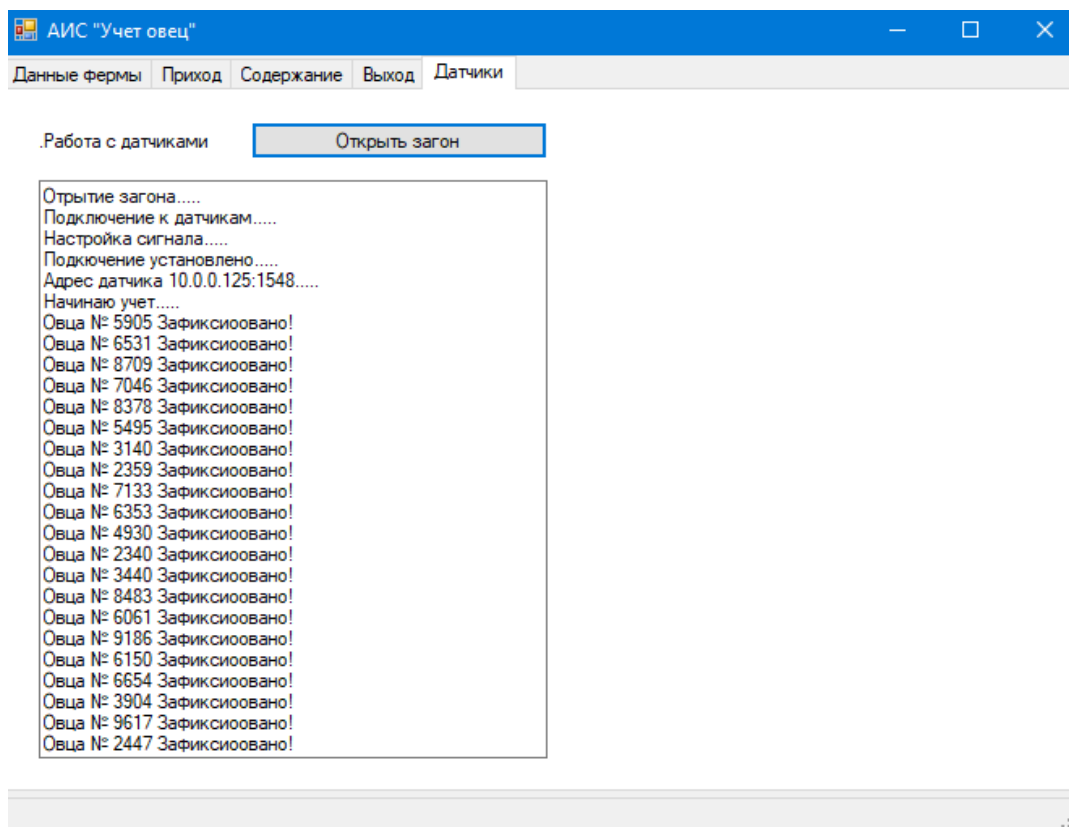
Өндіруші: DATAMARS / Швейцария

Әрі қарай, компьютерге uhf құлақ тегінің rfid оқырма қосылады



10 сурет – UHF құлақ тегінің rfid оқырманы

Бұл құрылғы құлақ белгілерін 300 метр радиуста оқуға мүмкіндік береді, сонымен қатар модем ретінде жұмыс істейді. Құрылғы бағдарламаға қойлардың жанында екендігі туралы сигнал береді, содан кейін бағдарлама қақпаны ашады және олардың сәйкестендіру нөмірлері бойынша өткен қойлардың санын оқиды.



11 сурет – Қақпаны ашу және өткен қойлар туралы ақпаратты оқу процесі

Осылайша, фермер қораға қанша қой кіргенін және қандай қойлар өтпегенін нақты біле алады.

Сонымен, біз әзірлеген қойларды есепке алу бағдарламасы:

Ферманы сауатты дамыту және оның жұмыс істеуі үшін әртүрлі ресурстарды ескеру .

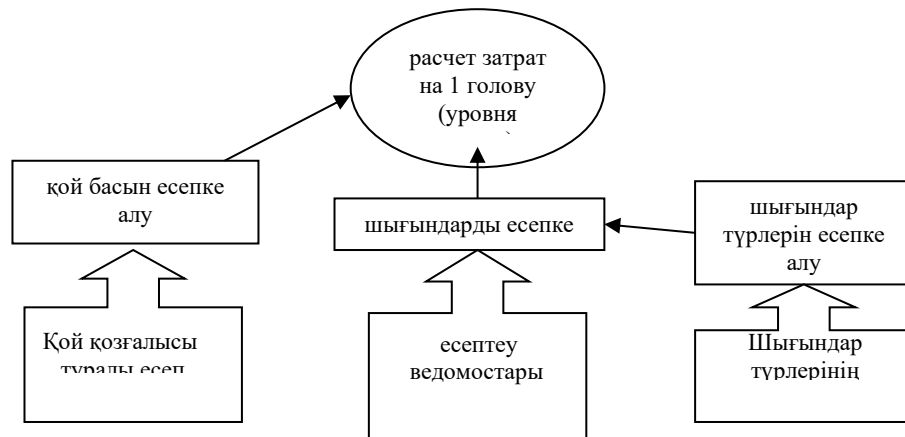
Бағдарлама мал басынан бастап алынған жүн мен ет көлеміне дейінгі көптеген деректерді ескеруге мүмкіндік береді.

Осы бағдарламаның көмегімен мал шаруашылығы өнімдерін есепке алу ферманы дамытудың дұрыс бағытын қамтамасыз ете отырып, әлдеқайда жеңіл және көрнекі болады.

Бағдарламалық әдістерді қолдана отырып, қойларды есепке алу Мал шаруашылығы өнімдерінің әр адамнан шығуын тиімді көрсетуге мүмкіндік береді.

Мал шаруашылығын автоматтандыру қой санын бақылауға мүмкіндік береді.

Бағдарламалық кешен арқылы ферманы автоматтандыру нақты уақыт режимінде есептер арқылы мониторинг жүргізуге мүмкіндік береді.



12 сурет –Тапсырмаларды орындау реттілігі

### 2.3 Бағдарламалық қамтамаға қойылатын талаптар

Программа келесі талаптарды орындауы керек:

- дистрибутивтен жүйеге орнату мүмкіндігіне ие болу;
- барлық қажетті компоненттер (библиотекалар) программамен бірге қолданылу керек;
- clipboard мүмкіндігін қолдау (көшіру, қою және т.б.);
- қолданушы интерфейсі максималды ыңғайлы және қарапайым болуы керек;
- upgrade және қосымша компоненттерді енгізу мүмкіндігі болуы;
- талап етілетін аппараттық анықтамалар операциялық жүйелер анықтамасына сәйкес келуі;
- C# жасау ортасы;
- Windows 7/8/10 операциялық жүйелері.

#### Техникалық құралдарға қойылатын талаптар

Жүйе жұмысы үшін қажетті минималды талаптар: ЭЕМ-персоналды компьютері, Intel Celeron процессоры тактілік жиілігі- 400MHz, жады көлемі-256 МБ, қатты диск 40 Gb, жазатын құрал, VGA- дисплей (SVGA-дисплейі ).

#### Сенімділікке қойылатын талаптар

Берілген ақпараттық жүйені функциялау кезінде енгізілетін және шығарылатын, сонымен қатар қолданушымен енгізілетін және мәліметтер базасының кестесінде бар мәліметтерге бақылау ұйымдастырылуы керек

Егер енгізу және шығару кезінде қате шықса, тиісті хабарлама шығуы қажет. Және де қатені жөндеуге, жұмысты жалғастыруға мүмкіндік берілуі керек.

Мәліметтерді өңдеу және жою операциясын орныдаған уақытта, мәліметтер базасының бүтіндігін қамтамасыз ету керек. Жою операциясы үшін болдырмау мүмкіндігін қарастыру қажет.

### 3 Жобаның экономикалық негізделуі

Жалпы экономика игіліктер мен қызметтер көрсететін өндірістік және өндірістік емес салалардың жиынтығы ретінде түсіндіріледі, онсыз адамзат қоғамы өмір сүре алмайды. Сондай – ақ «экономика» сөзімен экономикалық қатынастар жиынтығы қарастырылады. Осы қатынастар өндіру, бөлу, айырбастау және тұтыну жүйесінде қалыптасады. Қоғамдық қатынастар жүйесінде экономикаға ерекше орын беріледі, себебі ол саяси, құқықтық, рухани және басқада қоғамдық, өмір саласының мазмұнын анықтайды.

Экономикалық тиімділік – адамның қайрат – қуатының ең басты мәселесі. Тиімділікті арттыру – бұл бір эффекті бірлігіне (өндіріс көлемі, өнім бірлігіне деген және т.б.) шығынды қысқартуға қол жеткізу.

Нарықтық экономикаға өтуде ең жоғарғы өндіріс тиімділігін оның дамуының гуманистік мақсатымен қабыстыра қарауды ұйғарады. Қазіргі жағдайда сыртқы экономикалық байланыстар мен әлемдік шаруашылық тиімділігінің мән-мағынасы арта түсуде, сондықтан да тиімділік ұғымы интернационалдық категорияға айналууда.

Шығындар мен тиімділіктің экономикалық бағалаулары қызмет ұйымдар түрін жобалауда маңызды роль ойнайды. Экономикалық көрсеткіштерді анықтау үшін, әртүрлі факторларды ескеру қажет. Бір факторлар өз кезегінде тексеру мен бағалауға мұқтаж болатын басқа бір факторларды туғызады. Шығындарды топтастыру кезінде алдымен шығындардың ең айқын жеңіл есептелінетін түрлерін ерекшелеу қажет.

Басқаратын еңбек нәтижелерін жоғарылату шартының бірі, өзінің иілгіштігімен, мобильділігімен және сыртқы әсерлерге бейімділігімен үйлесімді ақпараттық технология болып келеді. Ақпараттық технология мәліметтермен және есептеуіш техникамен сауатты жұмыс істеуде іскерлікті көрсетеді.

Жасалынушы программалық өнімді енгізу маңызды күрделі қаражатты салуды, соның ішінде тірі еңбек шығындарын, эксплуатациялық есеп-қисаптарды талап етеді. Сондықтан бұл шараларды іске асыру үшін олардың экономикалық тиімділігін негіздеу қажет.

Енгізу мақсаттарының құрылымдалуы жүйедегі жетіспеушіліктердің негізінде болады. Негізгі жетіспеушілік жоғарғы оқу орны деканатындағы студенттер және кафедралардың профессорлық-оқытушылар құрамы туралы мәліметтерді енгізу жұмысы қолмен істелініп, ұзақ уақытты керек етеді.

Әзірленген бағдарламалық өнімді енгізуі нәтижесінде мәліметтерді жинау, сақтау, өңдеу, тапсыру және алу процесстерінің тез жүруі мен қызметкерлердің еңбек өнімділігінің артуы күтілуде.

Бұл программалық өнімді құруға кеткен шығындар программаны жасауға кеткен еңбек ақысы мен оны жасау кезіндегі уақыт шығындары жатады:

$$Ш_{бөк} = Ш_{пеа} + Ш_{убж} + Ш_{жал}, \quad (1)$$



мұндағы

- $Ш_{бөк}$  – бағдарламаны өнімді құруға кеткен шығындар;
- $Ш_{пеа}$  – программа құрушының еңбек ақысына кеткен шығындар;
- $Ш_{муб}$  – машиналық уақыт бірлігіне жұмсалған шығындар;
- $Ш_{жал}$  – жалпы шығындар.

Программа құрушының еңбек ақысына кеткен шығындар

Еңбек ақы ол кәсіпорын қызметкерлерін ынталандыратын, еңбектің артынан төленетін сыйлық.

Программалық өнім бір ай ішінде әзірлену қажет. Программаны құрушының еңбек ақысын төлеуге кететін (программалық өнімді тапсырыспен әзірлегендегі) және программа әзірлеушінің ( программалық өнімді өз күшімен әзірлегендегі) шығындарын құрайық.

Программа құрушының еңбек ақысы нарық шарттарында әр түрлі болуы мүмкін. Есеп үшін  $T_{сағ} = 300$  тг/с болатын орта сағаттап ақы төлеуді аламыз. Сонда күніне 8 сағаттық жұмыс уақытымен және аптасына 6 жұмыс күнін есептегенде, программа құрушының еңбек ақы төлемі құрастырылады:

- 1)  $8 \text{ с/күн} * 6 \text{ күн/апт} = 48 \text{ с/апт}$ ;
- 2)  $48 \text{ с/апт} * 4 \text{ апт/ай} = 192 \text{ с/ай}$ ;
- 3)  $192 \text{ с/ай} + 2 \text{ күн} * 8 \text{ с/күн} = 208 \text{ с/ай}$  (бір айда 30 күн деп есептегенде);
- 4)  $208 \text{ с/ай} * 3000 \text{ тг/с} = 624000 \text{ тг/ай}$ .

Программа құрушының еңбек ақы төлем шығынын оның программалық өнімді жасауға кеткен уақытты оның сағаттық ақысына көбейту жолымен анықталады:

$$Ш_{пеа} = t * T_{сағ}, \quad (2)$$

мұндағы

- $t$  – программаны құру уақыты;
- $T_{сағ}$  – программа құрушының сағаттық ақысы.

Программаны құру уақыты  $t=120$  сағат. Онда программа құрушының еңбек ақысы:

$$Ш_{пеа} = 120 \text{ с} * 3000 \text{ тг/сағ} = 360000 \text{ тг}.$$

Машиналық уақыт бірлігіне жұмсалған шығындар

Уақыт бірлігіне жұмсалған шығындар нақты уақыт бірлігімен программаға арендалық уақытқа кеткен сағаттың бағасына көбейту арқылы анықтаймыз:

$$Ш_{муб} = C_{сағ} * t_{эм}, \quad (3)$$

мұндағы

$C_{\text{сағ}}$  – арендалық уақыт сағаттық бағасы, тг/сағ;

$t_{\text{ЭЕМ}}$  – ЭЕМ программаны құрудың нақты уақыты.

Сағаттық бағасын есептеу. Сағаттық бағасын формула бойынша табамыз:

$$C_{\text{сағ}} = \frac{\text{Ш}_{\text{ЭЕМ}}}{T_{\text{ЭЕМ}}}, \quad (4)$$

мұндағы

$\text{Ш}_{\text{ЭЕМ}}$  – жыл ішіндегі ЭЕМ эксплуатацияға кететін толық шығындары;

$T_{\text{ЭЕМ}}$  – ЭЕМ уақытының нақты жылдық уақыт қоры, сағ/жыл.

ЭЕМ уақытының нақты жылдық уақыт қорын есептейміз. Жыл ішіндегі күндер саны – 365 (52 апта), оның ішінде 65 күні мереке және демалыс. Профилактикалық жұмыстағы тұру уақыты күндік 4 сағатымен анықталады. ЭЕМ жұмыс уақытының жалпы жылдық қоры мынаны құрады:

$$T_{\text{ЭЕМ}} = 8 \cdot (365 - 65) - 52 \cdot 4 = 2192 \text{ сағат.}$$

ЭЕМ - ға кететін эксплуатациялық жалпы шығыны мына формула бойынша анықталады:

$$\text{Ш}_{\text{ЭЕМ}} = (\text{Ш}_{\text{са}} + \text{Ш}_{\text{ам}} + \text{Ш}_{\text{эл}} + \text{Ш}_{\text{қж}} + \text{Ш}_{\text{аж}} + \text{Ш}_{\text{т.б}}), \quad (5)$$

мұндағы

$\text{Ш}_{\text{са}}$  – қызмет көрсету жұмыскерлеріне еңбек ақыға жылдық ұсталымдар, тг/жыл;

$\text{Ш}_{\text{ам}}$  - амортизацияға кеткен жылдық шығындар, тг/жыл;

$\text{Ш}_{\text{эл}}$  - ЭЕМ жылдық электр қуатына кеткен шығындайтын шығындар, тг/жыл;

$\text{Ш}_{\text{қж}}$  – қосымша жабдықтарға кеткен жылдық шығындар, тг/жыл;

$\text{Ш}_{\text{аж}}$  – ағымдық жөндеуге кеткен шығындар, тг/жыл;

$\text{Ш}_{\text{т.б}}$  – жылдық ұсталымдар мен т.б. шығындар, тг/жыл.

а) Қызметкерлердің еңбек ақыға жылдық ұсталымдардың есебі. Қызметкерлердің еңбек ақыға жылдық ұсталымдары оның негізгі еңбек ақысынан, қосымша және еңбек ақы төлеуге бөліп шығарулардан құралады.

Негізгі еңбек ақы төлеу қызметкерлердің жұмыс істейтін ұйымдарында тарифті қойылымымен (лауазымды еңбекақы мөлшерлеріне) және еңбек ақының тұрақты бөлімімен анықталған нормаға лайықты бөлек бағаланады.

Қосымша еңбек ақы ол әртүрлі ақшалай сыйлықтарды, тағайындалған уақыттан тыс жұмыс істегені үшін және т.б. жағдайларда қосады.

Тарифті қойылым - қызметкерлердің категориясына тәуелді уақыт бірлігі бойынша алатын еңбек ақысының абсолютті мөлшері. Ол ең аз тарифті қойылым немесе бірінші дәрежелі тарифті қойылым болып келеді.

Қызметкерлердің еңбек ақысы келесі формула бойынша есептеледі:

$$\text{Ш}_{\text{еа}} = \text{Ш}_{\text{нег}} + \text{Ш}_{\text{қос}}, \quad (6)$$

мұндағы  $\text{Ш}_{\text{нег}}$  – негізгі еңбек ақы сомасы;  
 $\text{Ш}_{\text{қос}}$  – қосымша еңбек ақы сомасы.

Негізгі жалақының сомасы деканата қызмет істейтін қызметкерлердің жалпы санына сәйкес анықталады және келесі формула бойынша есептеледі:

$$\text{Ш}_{\text{нег}} = \sum_{i=1}^n \text{З}_{\text{окл}}^i, \quad (7)$$

мұндағы  $\text{З}_{\text{окл}}^i$  -  $i$  жұмыскерінің бір айдағы тарифті қойылымы, тг.

Деканатта студенттер ісімен декан орынбасары еңбек ақысы айына 300000 тг. және бір хатшы еңбек ақысы айына 200000 тг. Онда қызметкерлердің негізгі еңбек ақыға ұсталымдары (7) формула бойынша есептегенде:

$$\text{Ш}_{\text{нег}} = 300000 + 200000 = 500000 \text{ тг.}$$

Қосымша жалақының сомасы негізгі жалақының 60%-ін құрайды:

$$\text{Ш}_{\text{қос}} = 500000 * 0,6 = 300000 \text{ тг.}$$

Сонда, (6) формула бойынша, қызметкердің жалақысынан жылдық ұсталымдар мынандай:

$$\text{Ш}_{\text{еа}} = (500000 + 300000) * 12 = 9600000 \text{ тг.}$$

Зейнетақы қорына аударымдар сомасы жалпы жалақының 10%-ын құрайды, яғни:

$$\text{ЗҚ} = \text{Ш}_{\text{еа}} * 10\% = 9600000 * 0,1 = 960000 \text{ тг.}$$

Табыс салығына аударымдар сомасы жалпы жалақының 10%-ын құрайды, яғни:

$$\text{ТС} = (\text{Ш}_{\text{еа}} - \text{ЗҚ} - 2 * 12 \text{ АЕК}) * 10\% = (9600000 - 960000 - 2 * 10300 * 12) * 0,1 = 839280 \text{ тг.}$$

б) Амортизациясын есептеу. Амортизация - ол негізгі қорлардың тозық ақшалай айтылуы. Жылдық амортизация жиыны мына формула бойынша анықталады:

$$Ш_{ам} = C_{ЭВМ} * N_{ам}, \quad (8)$$

мұндағы

$C_{ЭВМ}$  - компьютердің нарықтық бағасы, тг/дана;

$N_{ам}$  - амортизация нормасы, %.

Техникалық талаптарға сай амортизация нормасы 25% -ды құрайды. ЭЕМ баланстық бағасын анықтауға жіберу бағасы, тасымалдау шығыны, жабдықты монтаждау кіреді:

$$C_{бал} = C_{нар} + Ш_{орн}, \quad (9)$$

мұндағы

$C_{нар}$  - нарықтық компьютердің бағасы, тг/дана;

$Ш_{орн}$  – компьютерді тасымалдауға және оны орнатуға кеткен шығындар, тг/дана.

Жұмыс істеген компьютер  $C_{нар} = 90000$  тг бағасымен сатып алынған, ал орнатуға және тасымалдауға кеткен шығын шамамен 10% компьютердің бағасынан шығындалды:

$$Ш_{орн} = C_{рын} * 10\% = 90000 \text{ тг.}$$

Осыдан (9) формула бойынша баланстық бағасын есептейміз:

$$C_{бал} = 900000 + 90000 = 990000 \text{ тг/дана}$$

(8) формулаға сәйкес амортизациялық шығындарды есептейміз:

$$Ш_{ам} = 990000 * 0,25 = 247500 \text{ тг/жыл}$$

в) Электр қуатына кеткен шығынды есептеу. Жыл ішінде кететін электр қуатының бағасы мына формуламен табылады:

$$Ш_{эл} = P_{зем} * T_{зем} * C_{эл} * A, \quad (10)$$

мұндағы

$P_{зем}$  - ЭЕМ жиынтық қуаты;

$C_{эл}$  - 1кВт\*сағ электр қуатының бағасы;

$A$  - машина қуатын қолдану қарқындылық коэффициенті.

Техникалық паспорт бойынша ЭЕМ  $P_{ЭВМ}=0,22$  кВт\*сағ электр қуатының бағасы  $C_{эл}=5,15$  тг. Машинаны қолдану қарқындылығы  $A=0,98$

Сонда электр қуатына кеткен шығынды есептегенде:

$$Ш_{эл}=0,22*17600*5,15*0,98 = 24340 \text{ тг.}$$

г) Қосымша материалдарға кеткен шығындар. ЭЕМ-ға қалыпты жұмыс істеуге қамтамасыз ететін материалдардың шығыны ЭЕМ-нің бағасының 1% құрайды:

$$Ш_{қж} = 0,01 * 990000 = 9900 \text{ тг.}$$

д) Ағымдық жөндеуге кеткен шығынды есептеу. Ағымдық және профилактикалық жөндеуге кететін шығын ЭЕМ-нің бағасының 5% құрайды:

$$Ш_{аж} = 0,05 * C_{бал}$$

$$Ш_{аж} = 0,05 * 990000 = 49500 \text{ тг.}$$

е) ЭЕМ-ді қолдануға кеткен басқа да шығындар. ЭЕМ-ді қолдануға байланысты басқа ұйымдардың қызмет көрсету құнынан және ЭЕМ құнының 5%-ін құрайды:

$$Ш_{т.б} = 0,05 * 990000 = 49500 \text{ тг.}$$

*ЭЕМ-ді бір жыл ішінде эксплуатациялауға кеткен шығындар (5) формула бойынша есептесек:*

$$Ш_{ЭЕМ} = 960000 + 247500 + 24340 + 9900 + 49500 + 49500 = 9980740 \text{ тг.}$$

Сонда жалға алынған машиналық уақыт сағаттың бағасы (4) формулаға сәйкес келесідей болады:

$$C_{сағ} = 9980740 / 21920 = 4550 \text{ тг.}$$

Ал машиналық сағаттың төлеу шығындары (3) формулаға сәйкес келесідей болады:

$$Ш_{муб} = 455 * 120 = 546000 \text{ тг.}$$

Жалпы шығындарды есептеу

Жалпы шығындар бұл – жарыққа, жылуға, коммуналдық қызметке кеткен шығындар, оны 13000 теңге деп аламыз.

Сонда программалық өнімді жасау шығындары (1) формулаға сәйкес келесідегідей болады:

$$Ш_{бөк} = 36000 + 546000 + 130000 = 1036000 \text{ тг.}$$

Программалық өнімді енгізгеннен кейінгі тиімділікті есептеу

Қызметкердің еңбек ақыға жылдық ұсталымдарын программалық өнімді енгізгенге дейінгі жылдық ұсталымдар  $Ұ_{енд} = 10950000$  тг., ал программалық өнімді енгізгеннен кейінгі ұсталымдар  $Ұ_{енк} = 9600000$  тг. тең.

Сонда программалық өнімді енгізгеннен кейінгі еңбек ақы ұсталымдар сомасы келесідегідей кішірейеді:

$$\Delta Y_{ен} = Y_{енд} - Y_{енк} = 10950000 - 9600000 = 1350000 \text{ тг.}$$

Экономикалық тиімділікті есептеу

Экономикалық тиімділік – адамның қайрат–қуатының ең басты мәселесі. Тиімділікті арттыру – бұл бір әсер бірлігіне (өндіріс көлемі, өнім бірлігіне деген және т.б.) шығынды қысқартуға қол жеткізу.

Жылдық экономикалық әсерін келесі формула бойынша есептейміз:

$$\mathcal{E}_{жыл} = Y_{ен} - (Ш_{бөк} * (E_n + H_a)), \quad (11)$$

мұндағы

$P_{ен}$  – күтілетін әсер, ол программалық өнімді енгізгеннен кейінгі еңбек ақы ұсталымдардың кішірейген сомасына тең;

$Ш_{бөк}$  - программалық өнімді жасау шығындары;

$E_n$  – 0,12 ге тең тиімділік нормасы;

$H_a$  – 0,25 ке тең амортизация нормасы.

(11) формулаға сәйкес жылдық экономикалық әсер келесідегідей болады:

$$\mathcal{E}_{жыл} = 135000 - (103600 * (0,12 + 0,25)) = 966680 \text{ тг.}$$

Келесі формуланы қолдана отыра енгізілген программалық өнімнің өтімділік уақытын есптейміз:

$$T_{өт} = Ш_{бөк} / \mathcal{E}_{жыл}, \quad (12)$$

бұдан

$$T_{өт} = 103600 / 966680 = 1,0 \text{ жыл.}$$

Келесі формула бойынша экономикалық тиімділік коэффициентін анықтаймыз:

$$K_{эТ} = \mathcal{E}_{жыл} / Ш_{бөк}, \quad (13)$$

бұдан

$$K_{\text{эт}} = 96668 / 103600 = 0,93.$$

Берілген бөлімдегі барлық есеп нәтижелерін қарай отырып, программалық өнімді деканатқа енгізу, экономикалық көзқараспен тиімді деген қортынды жасай аламыз. Бұл программалық өнім қызметкерді оның жұмыс орнынан ауыстыруға арналмаған, бірақ оның ақпараттарды өңдеуге кететін уақытын қысқарту арқылы еңбек өнімділігін арттырып және жұмысын жеңілдетеді.

Берілген программалық өнімнің жылдық экономикалық тиімділігі енгізу барысында 966680 тг. құрайды. Программалық өнімнің өтімділігі 1,0 жылды құрайды, ал экономикалық тиімділік коэффициенті 0,93 тең.

## ҚОРЫТЫНДЫ

Қазіргі заманның жетіліп келе жатқан экономиканың сұранысы мен талабы - тиімді басқару. Бұл тиімді басқару дегеніміз дүние жүзіндегі кәсіпорындар, мекемелер, үлкен оқу орындары, іс құжаттары көп мекемелер және т.б. да көптеген қызмет орындарының жұмысын тиянақты жүргізіп, тиімді басқару болып табылады. Тиімді басқару, сонымен қатар уақытты үнемдей білуді қажет етеді. Ал уақытты үнемдеуге, қазіргі заманда автоматтандырылған жұмыс орнын (бағдарламма) құру болып табылады.

Кәсіпорындарда ақпараттық құжаттар көп, мәліметтерді өзгерту керек болған жағдайда, өте көп қиындықтар туындайды. Мәселен, оны іздеп табу: аты – жөні бойынша іздеу, ескі немесе қазіргі нөмері бойынша, коды бойынша т.б. , осындай қиындықтар туады. Демек, бұл бір құжатты іздеу үшін қаншама уақытпен шыдамдық керектігін көрсетеді. Онда бұл құжатты толтыру үшін қанша уақыттымызды жоғалтатынымызды елестетеді аламаймыз.

Қазіргі таңда осындай мәселелерді шешуге жол ашылды, ол автоматтандырылған қойларды есепке алу жүйесі әзірлеу. Автоматтандырылған қойларды есепке алу жүйесі әзірлеу дегеніміз мәліметтерді бір мәліметтер қорына жинап, мәліметтерді ортадан пайдалану. Әрі еш қиындықсыз өзгерістер мен жаңартулар, өшірулер мен қосулар т.б. мәліметтерді енгізе аламыз.

Менің дипломдық жұмысым осы мақсатты алға қоя отырып, бір нәтижеге келу. Осы дипломдық жұмыстың негізгі мақсаты болып, қойларды есепке алу жүйесі есебіне автоматтандырылған бағдарламаны әзірлеу болып табылады. Барлық мәліметтерді бағдарламаға енгізу арқылы түпкілікті нәтижені, әрі жинақты түрде ала аламыз. Информатика мамандары қоғамдағы информация рөлінің күшеюін бір ауыздан мойындап отыр. Ақпарат бара-бара қоғамдық әлеуметтік деңгейін анықтайтын шешуші факторға айналуда. Ол бизнестің стратегиялық ресурсы болып қалыптасып келеді. Дипломдық жұмыста құрылған ақпараттық жүйе кәсіпорындарда, мекемелерде т.б. шағын ұйымдарда қолданысқа ие бола алады. Бұл бағдарламмамен жұмыс істеу көптеген қиындықтардан арылуға, яғни құжаттарды жылдам, әрі оңай пайдалануға мүмкіндік береді. Сондықтан кез келген үлкен немесе кіші ұйымдарда осы бағдарламаны қолдана отырып, керекті нәтижені алуға болады. Кейбір ұйымдар бұл үшін папкалардан тұратын шкафтарды пайдаланады.

Еліміздің дамуы осы қазіргі заманға сай ақпараттық технологиялардың жетілуіне байланысты. Жұмыс нәтижесінде біз алдымызға қойған мақсаттарымызға қол жеткіздік және бұл ақпараттық жүйені кәсіпорын құжаттарын рәсімдеуде кең қолданысқа енетініне сенімдіміз. Автоматтандадыру ұғымының негізгі идеясы қоғамның әлеуметтік экономикалық дамуына ақпарат пен жаңа ақпараттық технология әсерінің өсе түсуі болып отыр.



## ПАЙДАЛАНЫЛҒАН ӘДЕБИЕТТЕР ТІЗІМІ

- 1 Густав Олссон, Джангуидо Пиани. Цифровые системы автоматизации и управления.
- 2 Промышленная электроника. Для студентов вузов/Под ред. В.А. Лабунцова. – М.: Энергоатомиздат, 1988. – 320 с.
- 3 Аристова, Н. И. Промышленные программно-аппаратные средства на казахстанском рынке АСУТП: научное издание / Н.И. Аристова, А.И. Корнеева. Юревич Е.И. Теория автоматического управления. Москва: Энергия, 1969.-375б.
- 4 Сьрашинин Е.Э. Основы теории автоматического управления. Екатеринбург: УГТУ, 2000-217б.
- 5 Программа развития сельскохозяйственного производства на 2000-2002 годы. 6 Республиканское совещание работников агропромышленного комплекса (8 февраля 2000 г., г.Астана). // Панорама № 6.- 11 февраля 2000 г.
- 7 Айтуганов Б. Развитие агропромышленного комплекса. – Транзитная экономика, 2008 г.
- 8 Баймухамедова Г. Сущность, содержание и функции рынка, средств производства и услуг в условиях рыночной экономики // Вестник с-х науки Казахстана № 8.- 2003 г.
- 9 Баймуратов У.Б. Национальная экономическая система.-Алматы: Ғылым, 2000 ж.
- 10 Гиззатова А.И. Дифференциация регионов Казахстана по уровню обеспеченности продовольственными ресурсами.-М.: НИИЭСХ, 2009 г.
- 11 Калиев Г.А. Законодательное обеспечение развития АПК в Казахстане. // Вестник с/х науки Казахстана № 5.- 2003 г.
- 12 Сейтбеков Л.С. Приоритеты научно-технической политики АПК Казахстана. // Вестник с-х науки Казахстана №10.-2003 г.
- 13 Мадиев Г. Тенденции развития сельхоз формирований. // Вестник с-х науки Казахстана №3.- 2004 г.
- 14 Тлеубергенова А. Қазақстан республикасы ауыл шаруашылығы экономикасында қазіргі кезде қалыптасқан проблемалар. Актуальные проблемы экономики Западного Казахстана. АПК и социальная сфера в условиях рынка: Тезисы международной научно-практической конференций. - Орал: ЗКГУ,2001г.
- 15 Шомкекова Б. Кәсіпкерлікті дамыту мәселелері. // Қаржы-қаражат № 8-9.- 1999 ж.
- 16 Регионы Казахстана, 2002. Статистический сборник/Под ред. А.А. Смайлова - Алматы, 2002 г.
- 17 Казахстан и его регионы. // Ежеквартальный журнал №1.-2003 г.
- 18 А.Б. Оспанов. Қазақ ауыл шаруашылығын механикаландыру және электендіру ғылыми-зерттеу институтының еліміздің агроинженерлік ғылымын дамытудағы ролі. // Жаршы 2003 ж №10.

19 Л.С.Сейтпеков Қазақстан агроөндірістік кешенін ғылыми- техникалық қамтамасыздандырудың басты бағыттары // Жаршы 2003 ж №9.

20 А.К.Күрішбаев. Қазақстан Республикасындағы астық өндірісінің жай-күйі және проблемалары //Жаршы 2003 ж №11.

21 А.А. Колашников, М.М. Мүсекенов. Ауыл шаруашылықтың қазіргі жағдайда егін суару ресурстарын үнемдеу. // Жаршы 2004 ж. № 8

22 А. Мұқашев. Ауыл шаруашылығы өндірісінің экономикалық тиімділігін арттырудың ілімдік негіздері.2003 ж. №5.

23 Н. Я. Коваленко. Экономика сельского хозяйства. Курс лекций Москва 1998 г.

24 Казахстан и его регионы // Ежеквартальный журнал № 1-2003 г

М. Н. Малыш. // Аграрная экономика. Учебник. Санк-Петербург 2002 г.

## ҚОСЫМША

Бағдарлама листингі

```
namespace RamUchet
{
    partial class Form1
    {
        /// <summary>
        /// Обязательная переменная конструктора.
        /// </summary>
        private System.ComponentModel.IContainer components = null;

        /// <summary>
        /// Освободить все используемые ресурсы.
        /// </summary>
        /// <param name="disposing">истинно, если управляемый ресурс должен быть удален;
        иначе ложно.</param>
        protected override void Dispose(bool disposing)
        {
            if (disposing && (components != null))
            {
                components.Dispose();
            }
            base.Dispose(disposing);
        }

        #region Код, автоматически созданный конструктором форм Windows

        /// <summary>
        /// Требуемый метод для поддержки конструктора — не изменяйте
        /// содержимое этого метода с помощью редактора кода.
        /// </summary>
        private void InitializeComponent()
        {
            this.components = new System.ComponentModel.Container();
            System.Windows.Forms.DataVisualization.Charting.ChartArea chartArea1 = new
            System.Windows.Forms.DataVisualization.Charting.ChartArea();
            System.Windows.Forms.DataVisualization.Charting.Legend legend1 = new
            System.Windows.Forms.DataVisualization.Charting.Legend();
            System.Windows.Forms.DataVisualization.Charting.Series series1 = new
            System.Windows.Forms.DataVisualization.Charting.Series();
            System.Windows.Forms.DataVisualization.Charting.DataPoint dataPoint1 = new
            System.Windows.Forms.DataVisualization.Charting.DataPoint(0D, 0D);
            System.Windows.Forms.DataVisualization.Charting.DataPoint dataPoint2 = new
            System.Windows.Forms.DataVisualization.Charting.DataPoint(0D, 0D);
            System.Windows.Forms.DataVisualization.Charting.DataPoint dataPoint3 = new
            System.Windows.Forms.DataVisualization.Charting.DataPoint(0D, 0D);
            System.Windows.Forms.DataVisualization.Charting.DataPoint dataPoint4 = new
            System.Windows.Forms.DataVisualization.Charting.DataPoint(0D, 0D);
        }
    }
}
```

```

System.Windows.Forms.DataGridViewCellStyle dataGridViewCellStyle1 = new
System.Windows.Forms.DataGridViewCellStyle();
System.Windows.Forms.DataGridViewCellStyle dataGridViewCellStyle2 = new
System.Windows.Forms.DataGridViewCellStyle();
System.Windows.Forms.DataGridViewCellStyle dataGridViewCellStyle3 = new
System.Windows.Forms.DataGridViewCellStyle();
System.Windows.Forms.DataVisualization.Charting.ChartArea chartArea2 = new
System.Windows.Forms.DataVisualization.Charting.ChartArea();
System.Windows.Forms.DataVisualization.Charting.Legend legend2 = new
System.Windows.Forms.DataVisualization.Charting.Legend();
System.Windows.Forms.DataVisualization.Charting.Series series2 = new
System.Windows.Forms.DataVisualization.Charting.Series();
this.tabControl1 = new System.Windows.Forms.TabControl();
this.tabPage1 = new System.Windows.Forms.TabPage();
this.numericUpDown1 = new System.Windows.Forms.NumericUpDown();
this.farmOrganisationBindingSource = new
System.Windows.Forms.BindingSource(this.components);
this.comboBox2 = new System.Windows.Forms.ComboBox();
this.actionsBindingSource = new System.Windows.Forms.BindingSource(this.components);
this.comboBox1 = new System.Windows.Forms.ComboBox();
this.typesBindingSource = new System.Windows.Forms.BindingSource(this.components);
this.button1 = new System.Windows.Forms.Button();
this.label5 = new System.Windows.Forms.Label();
this.textBox5 = new System.Windows.Forms.TextBox();
this.label6 = new System.Windows.Forms.Label();
this.textBox6 = new System.Windows.Forms.TextBox();
this.label7 = new System.Windows.Forms.Label();
this.label8 = new System.Windows.Forms.Label();
this.label3 = new System.Windows.Forms.Label();
this.textBox3 = new System.Windows.Forms.TextBox();
this.label4 = new System.Windows.Forms.Label();
this.textBox4 = new System.Windows.Forms.TextBox();
this.label2 = new System.Windows.Forms.Label();
this.label1 = new System.Windows.Forms.Label();
this.textBox1 = new System.Windows.Forms.TextBox();
this.tabPage2 = new System.Windows.Forms.TabPage();
this.button2 = new System.Windows.Forms.Button();
this.chart1 = new System.Windows.Forms.DataVisualization.Charting.Chart();
this.groupBox3 = new System.Windows.Forms.GroupBox();
this.lblY4 = new System.Windows.Forms.Label();
this.lblY3 = new System.Windows.Forms.Label();
this.lblY2 = new System.Windows.Forms.Label();
this.lblY1 = new System.Windows.Forms.Label();
this.label17 = new System.Windows.Forms.Label();
this.label18 = new System.Windows.Forms.Label();
this.label19 = new System.Windows.Forms.Label();
this.label20 = new System.Windows.Forms.Label();
this.groupBox2 = new System.Windows.Forms.GroupBox();
this.lblM4 = new System.Windows.Forms.Label();
this.lblM3 = new System.Windows.Forms.Label();
this.lblM2 = new System.Windows.Forms.Label();

```

```

this.lblM1 = new System.Windows.Forms.Label();
this.label13 = new System.Windows.Forms.Label();
this.label14 = new System.Windows.Forms.Label();
this.label15 = new System.Windows.Forms.Label();
this.label16 = new System.Windows.Forms.Label();
this.groupBox1 = new System.Windows.Forms.GroupBox();
this.lblD4 = new System.Windows.Forms.Label();
this.lblD3 = new System.Windows.Forms.Label();
this.lblD2 = new System.Windows.Forms.Label();
this.lblD1 = new System.Windows.Forms.Label();
this.label9 = new System.Windows.Forms.Label();
this.label10 = new System.Windows.Forms.Label();
this.label12 = new System.Windows.Forms.Label();
this.label11 = new System.Windows.Forms.Label();
this.tabPage3 = new System.Windows.Forms.TabPage();
this.groupBox4 = new System.Windows.Forms.GroupBox();
this.dataGridView1 = new System.Windows.Forms.DataGridview();
this.idDataGridViewTextBoxColumn = new
System.Windows.Forms.DataGridviewTextBoxColumn();
this.colID = new System.Windows.Forms.DataGridviewTextBoxColumn();
this.colName = new System.Windows.Forms.DataGridviewTextBoxColumn();
this.colWeight = new System.Windows.Forms.DataGridviewTextBoxColumn();
this.colCount = new System.Windows.Forms.DataGridviewTextBoxColumn();
this.colEdit = new System.Windows.Forms.DataGridviewButtonColumn();
this.stadosBindingSource = new System.Windows.Forms.BindingSource(this.components);
this.tabPage4 = new System.Windows.Forms.TabPage();
this.button3 = new System.Windows.Forms.Button();
this.chart2 = new System.Windows.Forms.DataVisualization.Charting.Chart();
this.groupBox5 = new System.Windows.Forms.GroupBox();
this.label21 = new System.Windows.Forms.Label();
this.label22 = new System.Windows.Forms.Label();
this.label23 = new System.Windows.Forms.Label();
this.label24 = new System.Windows.Forms.Label();
this.lblPY4 = new System.Windows.Forms.Label();
this.lblPY3 = new System.Windows.Forms.Label();
this.lblPY2 = new System.Windows.Forms.Label();
this.lblPY1 = new System.Windows.Forms.Label();
this.groupBox6 = new System.Windows.Forms.GroupBox();
this.label25 = new System.Windows.Forms.Label();
this.label26 = new System.Windows.Forms.Label();
this.label27 = new System.Windows.Forms.Label();
this.label28 = new System.Windows.Forms.Label();
this.lblPM4 = new System.Windows.Forms.Label();
this.lblPM3 = new System.Windows.Forms.Label();
this.lblPM2 = new System.Windows.Forms.Label();
this.lblPM1 = new System.Windows.Forms.Label();
this.groupBox7 = new System.Windows.Forms.GroupBox();
this.lblPD4 = new System.Windows.Forms.Label();
this.lblPD3 = new System.Windows.Forms.Label();
this.lblPD2 = new System.Windows.Forms.Label();
this.lblPD1 = new System.Windows.Forms.Label();

```

```

this.label29 = new System.Windows.Forms.Label();
this.label30 = new System.Windows.Forms.Label();
this.label31 = new System.Windows.Forms.Label();
this.label32 = new System.Windows.Forms.Label();
this.statusStrip1 = new System.Windows.Forms.StatusStrip();
this.tabPage5 = new System.Windows.Forms.TabPage();
this.button4 = new System.Windows.Forms.Button();
this.label33 = new System.Windows.Forms.Label();
this.txtLog = new System.Windows.Forms.TextBox();
this.tabControl1.SuspendLayout();
this.tabPage1.SuspendLayout();
((System.ComponentModel.ISupportInitialize)(this.numericUpDown1)).BeginInit();

((System.ComponentModel.ISupportInitialize)(this.farmOrganisationBindingSource)).BeginInit();
((System.ComponentModel.ISupportInitialize)(this.actionsBindingSource)).BeginInit();
((System.ComponentModel.ISupportInitialize)(this.typesBindingSource)).BeginInit();
this.tabPage2.SuspendLayout();
((System.ComponentModel.ISupportInitialize)(this.chart1)).BeginInit();
this.groupBox3.SuspendLayout();
this.groupBox2.SuspendLayout();
this.groupBox1.SuspendLayout();
this.tabPage3.SuspendLayout();
this.groupBox4.SuspendLayout();
((System.ComponentModel.ISupportInitialize)(this.dataGridView1)).BeginInit();
((System.ComponentModel.ISupportInitialize)(this.stadosBindingSource)).BeginInit();
this.tabPage4.SuspendLayout();
((System.ComponentModel.ISupportInitialize)(this.chart2)).BeginInit();
this.groupBox5.SuspendLayout();
this.groupBox6.SuspendLayout();
this.groupBox7.SuspendLayout();
this.tabPage5.SuspendLayout();
this.SuspendLayout();
//
// tabControl1
//
this.tabControl1.Anchor =
((System.Windows.Forms.AnchorStyles)((((System.Windows.Forms.AnchorStyles.Top |
System.Windows.Forms.AnchorStyles.Bottom)
| System.Windows.Forms.AnchorStyles.Left)
| System.Windows.Forms.AnchorStyles.Right)));
this.tabControl1.Controls.Add(this.tabPage1);
this.tabControl1.Controls.Add(this.tabPage2);
this.tabControl1.Controls.Add(this.tabPage3);
this.tabControl1.Controls.Add(this.tabPage4);
this.tabControl1.Controls.Add(this.tabPage5);
this.tabControl1.Location = new System.Drawing.Point(0, 0);
this.tabControl1.Name = «tabControl1»;
this.tabControl1.SelectedIndex = 0;
this.tabControl1.Size = new System.Drawing.Size(668, 455);
this.tabControl1.TabIndex = 0;
//

```

```

// tabPage1
//
this.tabPage1.Controls.Add(this.numericUpDown1);
this.tabPage1.Controls.Add(this.comboBox2);
this.tabPage1.Controls.Add(this.comboBox1);
this.tabPage1.Controls.Add(this.button1);
this.tabPage1.Controls.Add(this.label5);
this.tabPage1.Controls.Add(this.textBox5);
this.tabPage1.Controls.Add(this.label6);
this.tabPage1.Controls.Add(this.textBox6);
this.tabPage1.Controls.Add(this.label7);
this.tabPage1.Controls.Add(this.label8);
this.tabPage1.Controls.Add(this.label3);
this.tabPage1.Controls.Add(this.textBox3);
this.tabPage1.Controls.Add(this.label4);
this.tabPage1.Controls.Add(this.textBox4);
this.tabPage1.Controls.Add(this.label2);
this.tabPage1.Controls.Add(this.label1);
this.tabPage1.Controls.Add(this.textBox1);
this.tabPage1.Location = new System.Drawing.Point(4, 22);
this.tabPage1.Name = «tabPage1»;
this.tabPage1.Padding = new System.Windows.Forms.Padding(3);
this.tabPage1.Size = new System.Drawing.Size(660, 429);
this.tabPage1.TabIndex = 0;
this.tabPage1.Text = «ДАННЫЕ фермы»;
this.tabPage1.UseVisualStyleBackColor = true;
//
// numericUpDown1
//
this.numericUpDown1.DataBindings.Add(new System.Windows.Forms.Binding(«Value»,
this.farmOrganisationBindingSource, «MaxCountSheep», true));
this.numericUpDown1.Location = new System.Drawing.Point(124, 145);
this.numericUpDown1.Maximum = new decimal(new int[] {
999999,
0,
0,
0});
this.numericUpDown1.Name = «numericUpDown1»;
this.numericUpDown1.Size = new System.Drawing.Size(217, 20);
this.numericUpDown1.TabIndex = 38;
//
// farmOrganisationBindingSource
//
this.farmOrganisationBindingSource.DataSource =
typeof(RamUchet.Models.FarmOrganisation);
//
// comboBox2
//
this.comboBox2.DataBindings.Add(new
System.Windows.Forms.Binding(«SelectedValue», this.farmOrganisationBindingSource,
«TypeActionID», true));

```

```

        this.comboBox2.DataSource = this.actionsBindingSource;
        this.comboBox2.DisplayMember = «Title»;
        this.comboBox2.DropDownStyle =
System.Windows.Forms.ComboBoxStyle.DropDownList;
        this.comboBox2.FormattingEnabled = true;
        this.comboBox2.Location = new System.Drawing.Point(124, 118);
        this.comboBox2.Name = «comboBox2»;
        this.comboBox2.Size = new System.Drawing.Size(217, 21);
        this.comboBox2.TabIndex = 37;
        this.comboBox2.ValueMember = «Id»;
        //
        // actionsBindingSource
        //
        this.actionsBindingSource.DataMember = «Actions»;
        this.actionsBindingSource.DataSource = this.farmOrganisationBindingSource;
        //
        // comboBox1
        //
        this.comboBox1.DataBindings.Add(new
System.Windows.Forms.Binding(«SelectedValue», this.farmOrganisationBindingSource,
«TypeOOID», true));
        this.comboBox1.DataSource = this.typesBindingSource;
        this.comboBox1.DisplayMember = «Title»;
        this.comboBox1.DropDownStyle =
System.Windows.Forms.ComboBoxStyle.DropDownList;
        this.comboBox1.FormattingEnabled = true;
        this.comboBox1.Location = new System.Drawing.Point(124, 41);
        this.comboBox1.Name = «comboBox1»;
        this.comboBox1.Size = new System.Drawing.Size(217, 21);
        this.comboBox1.TabIndex = 36;
        this.comboBox1.ValueMember = «Id»;
        //
        // typesBindingSource
        //
        this.typesBindingSource.DataMember = «Types»;
        this.typesBindingSource.DataSource = this.farmOrganisationBindingSource;
        //
        // button1
        //
        this.button1.Location = new System.Drawing.Point(181, 301);
        this.button1.Name = «button1»;
        this.button1.Size = new System.Drawing.Size(160, 23);
        this.button1.TabIndex = 16;
        this.button1.Text = «СОХРАНИТЬ / ОБНОВИТЬ»;
        this.button1.UseVisualStyleBackColor = true;
        this.button1.Click += new System.EventHandler(this.button1_Click);
        //
        // label5
        //
        this.label5.AutoSize = true;
        this.label5.Location = new System.Drawing.Point(9, 200);

```



```

this.label5.Name = «label5»;
this.label5.Size = new System.Drawing.Size(38, 13);
this.label5.TabIndex = 15;
this.label5.Text = «Адрес»;
//
// textBox5
//
this.textBox5.DataBindings.Add(new System.Windows.Forms.Binding(«Text»,
this.farmOrganisationBindingSource, «Address», true));
this.textBox5.Location = new System.Drawing.Point(124, 197);
this.textBox5.Multiline = true;
this.textBox5.Name = «textBox5»;
this.textBox5.Size = new System.Drawing.Size(217, 98);
this.textBox5.TabIndex = 14;
//
// label6
//
this.label6.AutoSize = true;
this.label6.Location = new System.Drawing.Point(9, 174);
this.label6.Name = «label6»;
this.label6.Size = new System.Drawing.Size(57, 13);
this.label6.TabIndex = 13;
this.label6.Text = «Директор»;
//
// textBox6
//
this.textBox6.DataBindings.Add(new System.Windows.Forms.Binding(«Text»,
this.farmOrganisationBindingSource, «Director», true));
this.textBox6.Location = new System.Drawing.Point(124, 171);
this.textBox6.Name = «textBox6»;
this.textBox6.Size = new System.Drawing.Size(217, 20);
this.textBox6.TabIndex = 12;
//
// label7
//
this.label7.AutoSize = true;
this.label7.Location = new System.Drawing.Point(9, 148);
this.label7.Name = «label7»;
this.label7.Size = new System.Drawing.Size(101, 13);
this.label7.TabIndex = 11;
this.label7.Text = «Макс. число голов»;
//
// label8
//
this.label8.AutoSize = true;
this.label8.Location = new System.Drawing.Point(9, 122);
this.label8.Name = «label8»;
this.label8.Size = new System.Drawing.Size(99, 13);
this.label8.TabIndex = 9;
this.label8.Text = «Вид деятельности»;
//

```

```

// label3
//
this.label3.AutoSize = true;
this.label3.Location = new System.Drawing.Point(9, 96);
this.label3.Name = «label3»;
this.label3.Size = new System.Drawing.Size(30, 13);
this.label3.TabIndex = 7;
this.label3.Text = «КПП»;
//
// textBox3
//
this.textBox3.DataBindings.Add(new System.Windows.Forms.Binding(«Text»,
this.farmOrganisationBindingSource, «KPP», true));
this.textBox3.Location = new System.Drawing.Point(124, 93);
this.textBox3.Name = «textBox3»;
this.textBox3.Size = new System.Drawing.Size(217, 20);
this.textBox3.TabIndex = 6;
//
// label4
//
this.label4.AutoSize = true;
this.label4.Location = new System.Drawing.Point(9, 70);
this.label4.Name = «label4»;
this.label4.Size = new System.Drawing.Size(31, 13);
this.label4.TabIndex = 5;
this.label4.Text = «ИИИ»;
//
// textBox4
//
this.textBox4.DataBindings.Add(new System.Windows.Forms.Binding(«Text»,
this.farmOrganisationBindingSource, «INN», true));
this.textBox4.Location = new System.Drawing.Point(124, 67);
this.textBox4.Name = «textBox4»;
this.textBox4.Size = new System.Drawing.Size(217, 20);
this.textBox4.TabIndex = 4;
//
// label2
//
this.label2.AutoSize = true;
this.label2.Location = new System.Drawing.Point(9, 44);
this.label2.Name = «label2»;
this.label2.Size = new System.Drawing.Size(94, 13);
this.label2.TabIndex = 3;
this.label2.Text = «Тип организации»;
//
// label1
//
this.label1.AutoSize = true;
this.label1.Location = new System.Drawing.Point(9, 18);
this.label1.Name = «label1»;
this.label1.Size = new System.Drawing.Size(83, 13);

```

```

this.label1.TabIndex = 1;
this.label1.Text = «Наименование»;
//
// textBox1
//
this.textBox1.DataBindings.Add(new System.Windows.Forms.Binding(«Text»,
this.farmOrganisationBindingSource, «Name», true));
this.textBox1.Location = new System.Drawing.Point(124, 15);
this.textBox1.Name = «textBox1»;
this.textBox1.Size = new System.Drawing.Size(217, 20);
this.textBox1.TabIndex = 0;
//
// tabPage2
//
this.tabPage2.Controls.Add(this.button2);
this.tabPage2.Controls.Add(this.chart1);
this.tabPage2.Controls.Add(this.groupBox3);
this.tabPage2.Controls.Add(this.groupBox2);
this.tabPage2.Controls.Add(this.groupBox1);
this.tabPage2.Location = new System.Drawing.Point(4, 22);
this.tabPage2.Name = «tabPage2»;
this.tabPage2.Padding = new System.Windows.Forms.Padding(3);
this.tabPage2.Size = new System.Drawing.Size(660, 429);
this.tabPage2.TabIndex = 1;
this.tabPage2.Text = «Приход»;
this.tabPage2.UseVisualStyleBackColor = true;
//
// button2
//
this.button2.Location = new System.Drawing.Point(189, 387);
this.button2.Name = «button2»;
this.button2.Size = new System.Drawing.Size(93, 23);
this.button2.TabIndex = 15;
this.button2.Text = «Обновить»;
this.button2.UseVisualStyleBackColor = true;
this.button2.Click += new System.EventHandler(this.button2_Click);
//
// chart1
//
chartArea1.Name = «ChartArea1»;
this.chart1.ChartAreas.Add(chartArea1);
legend1.Name = «Legend1»;
this.chart1.Legends.Add(legend1);
this.chart1.Location = new System.Drawing.Point(298, 200);
this.chart1.Name = «chart1»;
series1.ChartArea = «ChartArea1»;
series1.Legend = «Legend1»;
series1.Name = «Series1»;
series1.Points.Add(dataPoint1);
series1.Points.Add(dataPoint2);
series1.Points.Add(dataPoint3);

```

```

series1.Points.Add(dataPoint4);
this.chart1.Series.Add(series1);
this.chart1.Size = new System.Drawing.Size(344, 170);
this.chart1.TabIndex = 12;
this.chart1.Text = «chart1»;
//
// groupBox3
//
this.groupBox3.Controls.Add(this.lblY4);
this.groupBox3.Controls.Add(this.lblY3);
this.groupBox3.Controls.Add(this.lblY2);
this.groupBox3.Controls.Add(this.lblY1);
this.groupBox3.Controls.Add(this.label17);
this.groupBox3.Controls.Add(this.label18);
this.groupBox3.Controls.Add(this.label19);
this.groupBox3.Controls.Add(this.label20);
this.groupBox3.Location = new System.Drawing.Point(6, 191);
this.groupBox3.Name = «groupBox3»;
this.groupBox3.Size = new System.Drawing.Size(276, 179);
this.groupBox3.TabIndex = 11;
this.groupBox3.TabStop = false;
this.groupBox3.Text = «3а год всего»;
//
// lblY4
//
this.lblY4.AutoSize = true;
this.lblY4.Location = new System.Drawing.Point(149, 108);
this.lblY4.Name = «lblY4»;
this.lblY4.Size = new System.Drawing.Size(13, 13);
this.lblY4.TabIndex = 14;
this.lblY4.Text = «0»;
//
// lblY3
//
this.lblY3.AutoSize = true;
this.lblY3.Location = new System.Drawing.Point(149, 82);
this.lblY3.Name = «lblY3»;
this.lblY3.Size = new System.Drawing.Size(13, 13);
this.lblY3.TabIndex = 13;
this.lblY3.Text = «0»;
//
// lblY2
//
this.lblY2.AutoSize = true;
this.lblY2.Location = new System.Drawing.Point(149, 56);
this.lblY2.Name = «lblY2»;
this.lblY2.Size = new System.Drawing.Size(13, 13);
this.lblY2.TabIndex = 12;
this.lblY2.Text = «0»;
//
// lblY1

```

```

//
this.lblY1.AutoSize = true;
this.lblY1.Location = new System.Drawing.Point(149, 30);
this.lblY1.Name = «lblY1»;
this.lblY1.Size = new System.Drawing.Size(13, 13);
this.lblY1.TabIndex = 11;
this.lblY1.Text = «0»;
//
// label17
//
this.label17.AutoSize = true;
this.label17.Location = new System.Drawing.Point(16, 30);
this.label17.Name = «label17»;
this.label17.Size = new System.Drawing.Size(73, 13);
this.label17.TabIndex = 0;
this.label17.Text = «Собственное»;
//
// label18
//
this.label18.AutoSize = true;
this.label18.Location = new System.Drawing.Point(16, 56);
this.label18.Name = «label18»;
this.label18.Size = new System.Drawing.Size(95, 13);
this.label18.TabIndex = 2;
this.label18.Text = «Другие фермеры»;
//
// label19
//
this.label19.AutoSize = true;
this.label19.Location = new System.Drawing.Point(16, 108);
this.label19.Name = «label19»;
this.label19.Size = new System.Drawing.Size(80, 13);
this.label19.TabIndex = 6;
this.label19.Text = «Закупка ягнят»;
//
// label20
//
this.label20.AutoSize = true;
this.label20.Location = new System.Drawing.Point(16, 82);
this.label20.Name = «label20»;
this.label20.Size = new System.Drawing.Size(94, 13);
this.label20.TabIndex = 4;
this.label20.Text = «Поставка (город)»;
//
// groupBox2
//
this.groupBox2.Controls.Add(this.lblM4);
this.groupBox2.Controls.Add(this.lblM3);
this.groupBox2.Controls.Add(this.lblM2);
this.groupBox2.Controls.Add(this.lblM1);
this.groupBox2.Controls.Add(this.label13);

```

```

this.groupBox2.Controls.Add(this.label14);
this.groupBox2.Controls.Add(this.label15);
this.groupBox2.Controls.Add(this.label16);
this.groupBox2.Location = new System.Drawing.Point(298, 6);
this.groupBox2.Name = «groupBox2»;
this.groupBox2.Size = new System.Drawing.Size(344, 179);
this.groupBox2.TabIndex = 10;
this.groupBox2.TabStop = false;
this.groupBox2.Text = «За месяц всего»;
//
// lblM4
//
this.lblM4.AutoSize = true;
this.lblM4.Location = new System.Drawing.Point(150, 108);
this.lblM4.Name = «lblM4»;
this.lblM4.Size = new System.Drawing.Size(13, 13);
this.lblM4.TabIndex = 14;
this.lblM4.Text = «0»;
//
// lblM3
//
this.lblM3.AutoSize = true;
this.lblM3.Location = new System.Drawing.Point(150, 82);
this.lblM3.Name = «lblM3»;
this.lblM3.Size = new System.Drawing.Size(13, 13);
this.lblM3.TabIndex = 13;
this.lblM3.Text = «0»;
//
// lblM2
//
this.lblM2.AutoSize = true;
this.lblM2.Location = new System.Drawing.Point(150, 56);
this.lblM2.Name = «lblM2»;
this.lblM2.Size = new System.Drawing.Size(13, 13);
this.lblM2.TabIndex = 12;
this.lblM2.Text = «0»;
//
// lblM1
//
this.lblM1.AutoSize = true;
this.lblM1.Location = new System.Drawing.Point(150, 30);
this.lblM1.Name = «lblM1»;
this.lblM1.Size = new System.Drawing.Size(13, 13);
this.lblM1.TabIndex = 11;
this.lblM1.Text = «0»;
//
// label13
//
this.label13.AutoSize = true;
this.label13.Location = new System.Drawing.Point(16, 30);
this.label13.Name = «label13»;

```

```

this.label13.Size = new System.Drawing.Size(73, 13);
this.label13.TabIndex = 0;
this.label13.Text = «Собственное»;
//
// label14
//
this.label14.AutoSize = true;
this.label14.Location = new System.Drawing.Point(16, 56);
this.label14.Name = «label14»;
this.label14.Size = new System.Drawing.Size(95, 13);
this.label14.TabIndex = 2;
this.label14.Text = «Другие фермеры»;
//
// label15
//
this.label15.AutoSize = true;
this.label15.Location = new System.Drawing.Point(16, 108);
this.label15.Name = «label15»;
this.label15.Size = new System.Drawing.Size(80, 13);
this.label15.TabIndex = 6;
this.label15.Text = «Закупка ягнят»;
//
// label16
//
this.label16.AutoSize = true;
this.label16.Location = new System.Drawing.Point(16, 82);
this.label16.Name = «label16»;
this.label16.Size = new System.Drawing.Size(94, 13);
this.label16.TabIndex = 4;
this.label16.Text = «Поставка (город)»;
//
// groupBox1
//
this.groupBox1.Controls.Add(this.lblD4);
this.groupBox1.Controls.Add(this.lblD3);
this.groupBox1.Controls.Add(this.lblD2);
this.groupBox1.Controls.Add(this.lblD1);
this.groupBox1.Controls.Add(this.label9);
this.groupBox1.Controls.Add(this.label10);
this.groupBox1.Controls.Add(this.label12);
this.groupBox1.Controls.Add(this.label11);
this.groupBox1.Location = new System.Drawing.Point(6, 6);
this.groupBox1.Name = «groupBox1»;
this.groupBox1.Size = new System.Drawing.Size(276, 179);
this.groupBox1.TabIndex = 9;
this.groupBox1.TabStop = false;
this.groupBox1.Text = «На сегодня»;
//
// lblD4
//
this.lblD4.AutoSize = true;

```

```

this.lblD4.Location = new System.Drawing.Point(149, 108);
this.lblD4.Name = «lblD4»;
this.lblD4.Size = new System.Drawing.Size(13, 13);
this.lblD4.TabIndex = 10;
this.lblD4.Text = «0»;
//
// lblD3
//
this.lblD3.AutoSize = true;
this.lblD3.Location = new System.Drawing.Point(149, 82);
this.lblD3.Name = «lblD3»;
this.lblD3.Size = new System.Drawing.Size(13, 13);
this.lblD3.TabIndex = 9;
this.lblD3.Text = «0»;
//
// lblD2
//
this.lblD2.AutoSize = true;
this.lblD2.Location = new System.Drawing.Point(149, 56);
this.lblD2.Name = «lblD2»;
this.lblD2.Size = new System.Drawing.Size(13, 13);
this.lblD2.TabIndex = 8;
this.lblD2.Text = «0»;
//
// lblD1
//
this.lblD1.AutoSize = true;
this.lblD1.Location = new System.Drawing.Point(149, 30);
this.lblD1.Name = «lblD1»;
this.lblD1.Size = new System.Drawing.Size(13, 13);
this.lblD1.TabIndex = 7;
this.lblD1.Text = «0»;
//
// label9
//
this.label9.AutoSize = true;
this.label9.Location = new System.Drawing.Point(16, 30);
this.label9.Name = «label9»;
this.label9.Size = new System.Drawing.Size(73, 13);
this.label9.TabIndex = 0;
this.label9.Text = «Собственное»;
//
// label10
//
this.label10.AutoSize = true;
this.label10.Location = new System.Drawing.Point(16, 56);
this.label10.Name = «label10»;
this.label10.Size = new System.Drawing.Size(95, 13);
this.label10.TabIndex = 2;
this.label10.Text = «Другие фермеры»;
//

```



```

// label12
//
this.label12.AutoSize = true;
this.label12.Location = new System.Drawing.Point(16, 108);
this.label12.Name = «label12»;
this.label12.Size = new System.Drawing.Size(80, 13);
this.label12.TabIndex = 6;
this.label12.Text = «Закупка ягнят»;
//
// label11
//
this.label11.AutoSize = true;
this.label11.Location = new System.Drawing.Point(16, 82);
this.label11.Name = «label11»;
this.label11.Size = new System.Drawing.Size(94, 13);
this.label11.TabIndex = 4;
this.label11.Text = «Поставка (город)»;
//
// tabPage3
//
this.tabPage3.Controls.Add(this.groupBox4);
this.tabPage3.Location = new System.Drawing.Point(4, 22);
this.tabPage3.Name = «tabPage3»;
this.tabPage3.Padding = new System.Windows.Forms.Padding(3);
this.tabPage3.Size = new System.Drawing.Size(660, 429);
this.tabPage3.TabIndex = 2;
this.tabPage3.Text = «Содержание»;
this.tabPage3.UseVisualStyleBackColor = true;
//
// groupBox4
//
this.groupBox4.Controls.Add(this.dataGridView1);
this.groupBox4.Dock = System.Windows.Forms.DockStyle.Fill;
this.groupBox4.Location = new System.Drawing.Point(3, 3);
this.groupBox4.Name = «groupBox4»;
this.groupBox4.Size = new System.Drawing.Size(654, 423);
this.groupBox4.TabIndex = 0;
this.groupBox4.TabStop = false;
this.groupBox4.Text = «Характеристика стада»;
//
// dataGridView1
//
this.dataGridView1.AllowUserToAddRows = false;
this.dataGridView1.AllowUserToDeleteRows = false;
this.dataGridView1.AutoGenerateColumns = false;
dataGridViewCellStyle1.Alignment =
System.Windows.Forms.DataGridViewContentAlignment.MiddleLeft;
dataGridViewCellStyle1.BackColor = System.Drawing.SystemColors.Control;
dataGridViewCellStyle1.Font = new System.Drawing.Font(«Microsoft Sans Serif», 8.25F,
System.Drawing.FontStyle.Regular, System.Drawing.GraphicsUnit.Point, ((byte)204));
dataGridViewCellStyle1.ForeColor = System.Drawing.SystemColors.WindowText;

```

```

        dataGridViewCellStyle1.SelectionBackColor = System.Drawing.SystemColors.Highlight;
        dataGridViewCellStyle1.SelectionForeColor =
System.Drawing.SystemColors.HighlightText;
        dataGridViewCellStyle1.WrapMode =
System.Windows.Forms.DataGridViewTriState.True;
        this.dataGridView1.ColumnHeadersDefaultCellStyle = dataGridViewCellStyle1;
        this.dataGridView1.ColumnHeadersHeightSizeMode =
System.Windows.Forms.DataGridViewColumnHeadersHeightSizeMode.AutoSize;
        this.dataGridView1.Columns.AddRange(new
System.Windows.Forms.DataGridViewColumn[] {
        this.idDataGridViewTextBoxColumn,
        this.colID,
        this.colName,
        this.colWeight,
        this.colCount,
        this.colEdit});
        this.dataGridView1.DataSource = this.stadosBindingSource;
        dataGridViewCellStyle2.Alignment =
System.Windows.Forms.DataGridViewContentAlignment.MiddleLeft;
        dataGridViewCellStyle2.BackColor = System.Drawing.SystemColors.Window;
        dataGridViewCellStyle2.Font = new System.Drawing.Font(«Microsoft Sans Serif», 8.25F,
System.Drawing.FontStyle.Regular, System.Drawing.GraphicsUnit.Point, ((byte)(204)));
        dataGridViewCellStyle2.ForeColor = System.Drawing.SystemColors.ControlText;
        dataGridViewCellStyle2.SelectionBackColor = System.Drawing.SystemColors.Highlight;
        dataGridViewCellStyle2.SelectionForeColor =
System.Drawing.SystemColors.HighlightText;
        dataGridViewCellStyle2.WrapMode =
System.Windows.Forms.DataGridViewTriState.False;
        this.dataGridView1.DefaultCellStyle = dataGridViewCellStyle2;
        this.dataGridView1.Dock = System.Windows.Forms.DockStyle.Fill;
        this.dataGridView1.Location = new System.Drawing.Point(3, 16);
        this.dataGridView1.Name = «dataGridView1»;
        this.dataGridView1.ReadOnly = true;
        dataGridViewCellStyle3.Alignment =
System.Windows.Forms.DataGridViewContentAlignment.MiddleLeft;
        dataGridViewCellStyle3.BackColor = System.Drawing.SystemColors.Control;
        dataGridViewCellStyle3.Font = new System.Drawing.Font(«Microsoft Sans Serif», 8.25F,
System.Drawing.FontStyle.Regular, System.Drawing.GraphicsUnit.Point, ((byte)(204)));
        dataGridViewCellStyle3.ForeColor = System.Drawing.SystemColors.WindowText;
        dataGridViewCellStyle3.SelectionBackColor = System.Drawing.SystemColors.Highlight;
        dataGridViewCellStyle3.SelectionForeColor =
System.Drawing.SystemColors.HighlightText;
        dataGridViewCellStyle3.WrapMode =
System.Windows.Forms.DataGridViewTriState.True;
        this.dataGridView1.RowHeadersDefaultCellStyle = dataGridViewCellStyle3;
        this.dataGridView1.Size = new System.Drawing.Size(648, 404);
        this.dataGridView1.TabIndex = 0;
        //
        // idDataGridViewTextBoxColumn
        //
        this.idDataGridViewTextBoxColumn.DataPropertyName = «Id»;

```

```

this.idDataGridViewTextBoxColumn.HeaderText = «Id»;
this.idDataGridViewTextBoxColumn.Name = «idDataGridViewTextBoxColumn»;
this.idDataGridViewTextBoxColumn.ReadOnly = true;
this.idDataGridViewTextBoxColumn.Visible = false;
//
// colID
//
this.colID.DataPropertyName = «Id»;
this.colID.HeaderText = «Заголовок»;
this.colID.Name = «colID»;
this.colID.ReadOnly = true;
this.colID.Visible = false;
//
// colName
//
this.colName.DataPropertyName = «Title»;
this.colName.HeaderText = «Группа овец»;
this.colName.Name = «colName»;
this.colName.ReadOnly = true;
this.colName.Width = 150;
//
// colWeight
//
this.colWeight.DataPropertyName = «VounHairs»;
this.colWeight.HeaderText = «Густота шерсти»;
this.colWeight.Name = «colWeight»;
this.colWeight.ReadOnly = true;
this.colWeight.Width = 150;
//
// colCount
//
this.colCount.DataPropertyName = «SheepsCount»;
this.colCount.HeaderText = «Кол-во овец»;
this.colCount.Name = «colCount»;
this.colCount.ReadOnly = true;
//
// colEdit
//
this.colEdit.AutoSizeMode =
System.Windows.Forms.DataGridViewAutoSizeColumnMode.ColumnHeader;
this.colEdit.DataPropertyName = «Details»;
this.colEdit.HeaderText = «Подробности»;
this.colEdit.Name = «colEdit»;
this.colEdit.ReadOnly = true;
this.colEdit.Text = «Посмотреть»;
this.colEdit.Width = 80;
//
// stadosBindingSource
//
this.stadosBindingSource.DataMember = «Stados»;
this.stadosBindingSource.DataSource = this.farmOrganisationBindingSource;

```

```

//
// tabPage4
//
this.tabPage4.Controls.Add(this.button3);
this.tabPage4.Controls.Add(this.chart2);
this.tabPage4.Controls.Add(this.groupBox5);
this.tabPage4.Controls.Add(this.groupBox6);
this.tabPage4.Controls.Add(this.groupBox7);
this.tabPage4.Location = new System.Drawing.Point(4, 22);
this.tabPage4.Name = «tabPage4»;
this.tabPage4.Padding = new System.Windows.Forms.Padding(3);
this.tabPage4.Size = new System.Drawing.Size(660, 429);
this.tabPage4.TabIndex = 3;
this.tabPage4.Text = «Выход»;
this.tabPage4.UseVisualStyleBackColor = true;
//
// button3
//
this.button3.Location = new System.Drawing.Point(195, 400);
this.button3.Name = «button3»;
this.button3.Size = new System.Drawing.Size(93, 23);
this.button3.TabIndex = 17;
this.button3.Text = «Обновить»;
this.button3.UseVisualStyleBackColor = true;
this.button3.Click += new System.EventHandler(this.button3_Click);
//
// chart2
//
chartArea2.Name = «ChartArea1»;
this.chart2.ChartAreas.Add(chartArea2);
legend2.Name = «Legend1»;
this.chart2.Legends.Add(legend2);
this.chart2.Location = new System.Drawing.Point(304, 226);
this.chart2.Name = «chart2»;
series2.ChartArea = «ChartArea1»;
series2.Legend = «Legend1»;
series2.Name = «Series1»;
this.chart2.Series.Add(series2);
this.chart2.Size = new System.Drawing.Size(344, 170);
this.chart2.TabIndex = 16;
this.chart2.Text = «chart2»;
//
// groupBox5
//
this.groupBox5.Controls.Add(this.label21);
this.groupBox5.Controls.Add(this.label22);
this.groupBox5.Controls.Add(this.label23);
this.groupBox5.Controls.Add(this.label24);
this.groupBox5.Controls.Add(this.lblPY4);
this.groupBox5.Controls.Add(this.lblPY3);
this.groupBox5.Controls.Add(this.lblPY2);

```

```

this.groupBox5.Controls.Add(this.lblPY1);
this.groupBox5.Location = new System.Drawing.Point(12, 217);
this.groupBox5.Name = «groupBox5»;
this.groupBox5.Size = new System.Drawing.Size(276, 179);
this.groupBox5.TabIndex = 15;
this.groupBox5.TabStop = false;
this.groupBox5.Text = «3а год всего»;
//
// label21
//
this.label21.AutoSize = true;
this.label21.Location = new System.Drawing.Point(16, 61);
this.label21.Name = «label21»;
this.label21.Size = new System.Drawing.Size(45, 13);
this.label21.TabIndex = 19;
this.label21.Text = «Шерсть»;
//
// label22
//
this.label22.AutoSize = true;
this.label22.Location = new System.Drawing.Point(16, 87);
this.label22.Name = «label22»;
this.label22.Size = new System.Drawing.Size(34, 13);
this.label22.TabIndex = 20;
this.label22.Text = «Мясо»;
//
// label23
//
this.label23.AutoSize = true;
this.label23.Location = new System.Drawing.Point(16, 35);
this.label23.Name = «label23»;
this.label23.Size = new System.Drawing.Size(63, 13);
this.label23.TabIndex = 22;
this.label23.Text = «Потомство»;
//
// label24
//
this.label24.AutoSize = true;
this.label24.Location = new System.Drawing.Point(16, 113);
this.label24.Name = «label24»;
this.label24.Size = new System.Drawing.Size(39, 13);
this.label24.TabIndex = 21;
this.label24.Text = «Шкура»;
//
// lblPY4
//
this.lblPY4.AutoSize = true;
this.lblPY4.Location = new System.Drawing.Point(138, 113);
this.lblPY4.Name = «lblPY4»;
this.lblPY4.Size = new System.Drawing.Size(13, 13);
this.lblPY4.TabIndex = 18;

```

```

this.lblPY4.Text = «0»;
//
// lblPY3
//
this.lblPY3.AutoSize = true;
this.lblPY3.Location = new System.Drawing.Point(138, 87);
this.lblPY3.Name = «lblPY3»;
this.lblPY3.Size = new System.Drawing.Size(13, 13);
this.lblPY3.TabIndex = 17;
this.lblPY3.Text = «0»;
//
// lblPY2
//
this.lblPY2.AutoSize = true;
this.lblPY2.Location = new System.Drawing.Point(138, 61);
this.lblPY2.Name = «lblPY2»;
this.lblPY2.Size = new System.Drawing.Size(13, 13);
this.lblPY2.TabIndex = 16;
this.lblPY2.Text = «0»;
//
// lblPY1
//
this.lblPY1.AutoSize = true;
this.lblPY1.Location = new System.Drawing.Point(138, 35);
this.lblPY1.Name = «lblPY1»;
this.lblPY1.Size = new System.Drawing.Size(13, 13);
this.lblPY1.TabIndex = 15;
this.lblPY1.Text = «0»;
//
// groupBox6
//
this.groupBox6.Controls.Add(this.label25);
this.groupBox6.Controls.Add(this.label26);
this.groupBox6.Controls.Add(this.label27);
this.groupBox6.Controls.Add(this.label28);
this.groupBox6.Controls.Add(this.lblPM4);
this.groupBox6.Controls.Add(this.lblPM3);
this.groupBox6.Controls.Add(this.lblPM2);
this.groupBox6.Controls.Add(this.lblPM1);
this.groupBox6.Location = new System.Drawing.Point(304, 32);
this.groupBox6.Name = «groupBox6»;
this.groupBox6.Size = new System.Drawing.Size(344, 179);
this.groupBox6.TabIndex = 14;
this.groupBox6.TabStop = false;
this.groupBox6.Text = «За месяц всего»;
//
// label25
//
this.label25.AutoSize = true;
this.label25.Location = new System.Drawing.Point(21, 56);
this.label25.Name = «label25»;

```

```

this.label25.Size = new System.Drawing.Size(45, 13);
this.label25.TabIndex = 19;
this.label25.Text = «Шерсть»;
//
// label26
//
this.label26.AutoSize = true;
this.label26.Location = new System.Drawing.Point(21, 82);
this.label26.Name = «label26»;
this.label26.Size = new System.Drawing.Size(34, 13);
this.label26.TabIndex = 20;
this.label26.Text = «Мясо»;
//
// label27
//
this.label27.AutoSize = true;
this.label27.Location = new System.Drawing.Point(21, 30);
this.label27.Name = «label27»;
this.label27.Size = new System.Drawing.Size(63, 13);
this.label27.TabIndex = 22;
this.label27.Text = «Потомство»;
//
// label28
//
this.label28.AutoSize = true;
this.label28.Location = new System.Drawing.Point(21, 108);
this.label28.Name = «label28»;
this.label28.Size = new System.Drawing.Size(39, 13);
this.label28.TabIndex = 21;
this.label28.Text = «Шкура»;
//
// lblPM4
//
this.lblPM4.AutoSize = true;
this.lblPM4.Location = new System.Drawing.Point(149, 108);
this.lblPM4.Name = «lblPM4»;
this.lblPM4.Size = new System.Drawing.Size(13, 13);
this.lblPM4.TabIndex = 18;
this.lblPM4.Text = «0»;
//
// lblPM3
//
this.lblPM3.AutoSize = true;
this.lblPM3.Location = new System.Drawing.Point(149, 82);
this.lblPM3.Name = «lblPM3»;
this.lblPM3.Size = new System.Drawing.Size(13, 13);
this.lblPM3.TabIndex = 17;
this.lblPM3.Text = «0»;
//
// lblPM2
//

```

```

this.lblPM2.AutoSize = true;
this.lblPM2.Location = new System.Drawing.Point(149, 56);
this.lblPM2.Name = «lblPM2»;
this.lblPM2.Size = new System.Drawing.Size(13, 13);
this.lblPM2.TabIndex = 16;
this.lblPM2.Text = «0»;
//
// lblPM1
//
this.lblPM1.AutoSize = true;
this.lblPM1.Location = new System.Drawing.Point(149, 30);
this.lblPM1.Name = «lblPM1»;
this.lblPM1.Size = new System.Drawing.Size(13, 13);
this.lblPM1.TabIndex = 15;
this.lblPM1.Text = «0»;
//
// groupBox7
//
this.groupBox7.Controls.Add(this.lblPD4);
this.groupBox7.Controls.Add(this.lblPD3);
this.groupBox7.Controls.Add(this.lblPD2);
this.groupBox7.Controls.Add(this.lblPD1);
this.groupBox7.Controls.Add(this.label29);
this.groupBox7.Controls.Add(this.label30);
this.groupBox7.Controls.Add(this.label31);
this.groupBox7.Controls.Add(this.label32);
this.groupBox7.Location = new System.Drawing.Point(12, 32);
this.groupBox7.Name = «groupBox7»;
this.groupBox7.Size = new System.Drawing.Size(276, 179);
this.groupBox7.TabIndex = 13;
this.groupBox7.TabStop = false;
this.groupBox7.Text = «На сегодня»;
//
// lblPD4
//
this.lblPD4.AutoSize = true;
this.lblPD4.Location = new System.Drawing.Point(138, 108);
this.lblPD4.Name = «lblPD4»;
this.lblPD4.Size = new System.Drawing.Size(13, 13);
this.lblPD4.TabIndex = 14;
this.lblPD4.Text = «0»;
//
// lblPD3
//
this.lblPD3.AutoSize = true;
this.lblPD3.Location = new System.Drawing.Point(138, 82);
this.lblPD3.Name = «lblPD3»;
this.lblPD3.Size = new System.Drawing.Size(13, 13);
this.lblPD3.TabIndex = 13;
this.lblPD3.Text = «0»;
//

```



```

// lblPD2
//
this.lblPD2.AutoSize = true;
this.lblPD2.Location = new System.Drawing.Point(138, 56);
this.lblPD2.Name = «lblPD2»;
this.lblPD2.Size = new System.Drawing.Size(13, 13);
this.lblPD2.TabIndex = 12;
this.lblPD2.Text = «0»;
//
// lblPD1
//
this.lblPD1.AutoSize = true;
this.lblPD1.Location = new System.Drawing.Point(138, 30);
this.lblPD1.Name = «lblPD1»;
this.lblPD1.Size = new System.Drawing.Size(13, 13);
this.lblPD1.TabIndex = 11;
this.lblPD1.Text = «0»;
//
// label29
//
this.label29.AutoSize = true;
this.label29.Location = new System.Drawing.Point(16, 56);
this.label29.Name = «label29»;
this.label29.Size = new System.Drawing.Size(45, 13);
this.label29.TabIndex = 0;
this.label29.Text = «Шерсть»;
//
// label30
//
this.label30.AutoSize = true;
this.label30.Location = new System.Drawing.Point(16, 82);
this.label30.Name = «label30»;
this.label30.Size = new System.Drawing.Size(34, 13);
this.label30.TabIndex = 2;
this.label30.Text = «Мясо»;
//
// label31
//
this.label31.AutoSize = true;
this.label31.Location = new System.Drawing.Point(16, 30);
this.label31.Name = «label31»;
this.label31.Size = new System.Drawing.Size(63, 13);
this.label31.TabIndex = 6;
this.label31.Text = «Потомство»;
//
// label32
//
this.label32.AutoSize = true;
this.label32.Location = new System.Drawing.Point(16, 108);
this.label32.Name = «label32»;
this.label32.Size = new System.Drawing.Size(39, 13);

```

```

this.label32.TabIndex = 4;
this.label32.Text = «Шкура»;
//
// statusStrip1
//
this.statusStrip1.Location = new System.Drawing.Point(0, 458);
this.statusStrip1.Name = «statusStrip1»;
this.statusStrip1.Size = new System.Drawing.Size(668, 22);
this.statusStrip1.TabIndex = 1;
this.statusStrip1.Text = «statusStrip1»;
//
// tabPage5
//
this.tabPage5.Controls.Add(this.txtLog);
this.tabPage5.Controls.Add(this.label33);
this.tabPage5.Controls.Add(this.button4);
this.tabPage5.Location = new System.Drawing.Point(4, 22);
this.tabPage5.Name = «tabPage5»;
this.tabPage5.Padding = new System.Windows.Forms.Padding(3);
this.tabPage5.Size = new System.Drawing.Size(660, 429);
this.tabPage5.TabIndex = 4;
this.tabPage5.Text = «Датчики»;
this.tabPage5.UseVisualStyleBackColor = true;
//
// button4
//
this.button4.Location = new System.Drawing.Point(151, 19);
this.button4.Name = «button4»;
this.button4.Size = new System.Drawing.Size(183, 23);
this.button4.TabIndex = 0;
this.button4.Text = «Открыть загон»;
this.button4.UseVisualStyleBackColor = true;
this.button4.Click += new System.EventHandler(this.button4_Click);
//
// label33
//
this.label33.AutoSize = true;
this.label33.Location = new System.Drawing.Point(17, 24);
this.label33.Name = «label33»;
this.label33.Size = new System.Drawing.Size(112, 13);
this.label33.TabIndex = 1;
this.label33.Text = «.Работа с датчиками»;
//
// txtLog
//
this.txtLog.Location = new System.Drawing.Point(20, 55);
this.txtLog.Multiline = true;
this.txtLog.Name = «txtLog»;
this.txtLog.Size = new System.Drawing.Size(314, 357);
this.txtLog.TabIndex = 2;
//

```

```

// Form1
//
this.AutoScaleDimensions = new System.Drawing.SizeF(6F, 13F);
this.AutoScaleMode = System.Windows.Forms.AutoScaleMode.Font;
this.ClientSize = new System.Drawing.Size(668, 480);
this.Controls.Add(this.statusStrip1);
this.Controls.Add(this.tabControl1);
this.Name = «Form1»;
this.Text = «АИС \»Учет овец\»»;
this.FormClosing += new
System.Windows.Forms.FormClosingEventHandler(this.Form1_FormClosing);
this.Load += new System.EventHandler(this.Form1_Load);
this.tabControl1.ResumeLayout(false);
this.tabPage1.ResumeLayout(false);
this.tabPage1.PerformLayout();
((System.ComponentModel.ISupportInitialize)(this.numericUpDown1)).EndInit();

((System.ComponentModel.ISupportInitialize)(this.farmOrganisationBindingSource)).EndInit();
((System.ComponentModel.ISupportInitialize)(this.actionsBindingSource)).EndInit();
((System.ComponentModel.ISupportInitialize)(this.typesBindingSource)).EndInit();
this.tabPage2.ResumeLayout(false);
((System.ComponentModel.ISupportInitialize)(this.chart1)).EndInit();
this.groupBox3.ResumeLayout(false);
this.groupBox3.PerformLayout();
this.groupBox2.ResumeLayout(false);
this.groupBox2.PerformLayout();
this.groupBox1.ResumeLayout(false);
this.groupBox1.PerformLayout();
this.tabPage3.ResumeLayout(false);
this.groupBox4.ResumeLayout(false);
((System.ComponentModel.ISupportInitialize)(this.dataGridView1)).EndInit();
((System.ComponentModel.ISupportInitialize)(this.stadosBindingSource)).EndInit();
this.tabPage4.ResumeLayout(false);
((System.ComponentModel.ISupportInitialize)(this.chart2)).EndInit();
this.groupBox5.ResumeLayout(false);
this.groupBox5.PerformLayout();
this.groupBox6.ResumeLayout(false);
this.groupBox6.PerformLayout();
this.groupBox7.ResumeLayout(false);
this.groupBox7.PerformLayout();
this.tabPage5.ResumeLayout(false);
this.tabPage5.PerformLayout();
this.ResumeLayout(false);
this.PerformLayout();

}

#endregion

private System.Windows.Forms.TabControl tabControl1;
private System.Windows.Forms.TabPage tabPage1;

```

```
private System.Windows.Forms.TabPage tabPage2;
private System.Windows.Forms.StatusStrip statusStrip1;
private System.Windows.Forms.TabPage tabPage3;
private System.Windows.Forms.TabPage tabPage4;
private System.Windows.Forms.Button button1;
private System.Windows.Forms.Label label5;
private System.Windows.Forms.TextBox textBox5;
private System.Windows.Forms.Label label6;
private System.Windows.Forms.TextBox textBox6;
private System.Windows.Forms.Label label7;
private System.Windows.Forms.Label label8;
private System.Windows.Forms.Label label3;
private System.Windows.Forms.TextBox textBox3;
private System.Windows.Forms.Label label4;
private System.Windows.Forms.TextBox textBox4;
private System.Windows.Forms.Label label2;
private System.Windows.Forms.Label label1;
private System.Windows.Forms.TextBox textBox1;
private System.Windows.Forms.GroupBox groupBox1;
private System.Windows.Forms.Label label9;
private System.Windows.Forms.Label label10;
private System.Windows.Forms.Label label12;
private System.Windows.Forms.Label label11;
private System.Windows.Forms.GroupBox groupBox3;
private System.Windows.Forms.Label label17;
private System.Windows.Forms.Label label18;
private System.Windows.Forms.Label label19;
private System.Windows.Forms.Label label20;
private System.Windows.Forms.GroupBox groupBox2;
private System.Windows.Forms.Label label13;
private System.Windows.Forms.Label label14;
private System.Windows.Forms.Label label15;
private System.Windows.Forms.Label label16;
private System.Windows.Forms.GroupBox groupBox4;
private System.Windows.Forms.DataGridView dataGridView1;
private System.Windows.Forms.DataVisualization.Charting.Chart chart1;
private System.Windows.Forms.DataVisualization.Charting.Chart chart2;
private System.Windows.Forms.GroupBox groupBox5;
private System.Windows.Forms.GroupBox groupBox6;
private System.Windows.Forms.GroupBox groupBox7;
private System.Windows.Forms.Label label29;
private System.Windows.Forms.Label label30;
private System.Windows.Forms.Label label31;
private System.Windows.Forms.Label label32;
private System.Windows.Forms.BindingSource farmOrganisationBindingSource;
private System.Windows.Forms.ComboBox comboBox2;
private System.Windows.Forms.ComboBox comboBox1;
private System.Windows.Forms.BindingSource typesBindingSource;
private System.Windows.Forms.NumericUpDown numericUpDown1;
private System.Windows.Forms.BindingSource actionsBindingSource;
private System.Windows.Forms.Label lblD1;
```

```

private System.Windows.Forms.Button button2;
private System.Windows.Forms.Label lblY4;
private System.Windows.Forms.Label lblY3;
private System.Windows.Forms.Label lblY2;
private System.Windows.Forms.Label lblY1;
private System.Windows.Forms.Label lblM4;
private System.Windows.Forms.Label lblM3;
private System.Windows.Forms.Label lblM2;
private System.Windows.Forms.Label lblM1;
private System.Windows.Forms.Label lblD4;
private System.Windows.Forms.Label lblD3;
private System.Windows.Forms.Label lblD2;
private System.Windows.Forms.BindingSource stadosBindingSource;
private System.Windows.Forms.DataGridViewTextBoxColumn
idDataGridViewTextBoxColumn;
private System.Windows.Forms.DataGridViewTextBoxColumn colID;
private System.Windows.Forms.DataGridViewTextBoxColumn colName;
private System.Windows.Forms.DataGridViewTextBoxColumn colWeight;
private System.Windows.Forms.DataGridViewTextBoxColumn colCount;
private System.Windows.Forms.DataGridViewButtonColumn colEdit;
private System.Windows.Forms.Label lblPY4;
private System.Windows.Forms.Label lblPY3;
private System.Windows.Forms.Label lblPY2;
private System.Windows.Forms.Label lblPY1;
private System.Windows.Forms.Label lblPM4;
private System.Windows.Forms.Label lblPM3;
private System.Windows.Forms.Label lblPM2;
private System.Windows.Forms.Label lblPM1;
private System.Windows.Forms.Label lblPD4;
private System.Windows.Forms.Label lblPD3;
private System.Windows.Forms.Label lblPD2;
private System.Windows.Forms.Label lblPD1;
private System.Windows.Forms.Button button3;
private System.Windows.Forms.Label label21;
private System.Windows.Forms.Label label22;
private System.Windows.Forms.Label label23;
private System.Windows.Forms.Label label24;
private System.Windows.Forms.Label label25;
private System.Windows.Forms.Label label26;
private System.Windows.Forms.Label label27;
private System.Windows.Forms.Label label28;
private System.Windows.Forms.TabPage tabPage5;
private System.Windows.Forms.TextBox txtLog;
private System.Windows.Forms.Label label33;
private System.Windows.Forms.Button button4;
}
}

```

ҚАЗАҚСТАН РЕСПУБЛИКАСЫ ҒЫЛЫМ ЖӘНЕ ЖОҒАРЫ БІЛІМ МИНИСТРЛІГІ  
«Қ.И. Сәтбаев атындағы Қазақ ұлттық техникалық зерттеу университеті» коммерциялық  
емес акционерлік қоғамы

**ҒЫЛЫМИ ЖЕТЕКШІНІҢ  
ПІКІРІ**

Дипломдық жоба үшін  
Мырзаш Әділжан Ермұханұлы

6B07103 - Автоматтандыру және роботтандыру  
Тақырыбы: «Қойларды есепке алу жүйесін әзірлеу»

Орындалды:

- а) Қойларды есепке алуды автоматтандырылған жүйесінің интерфейсі қарастырылды;  
Қойларды есептеуді қамтамасыз ететін жүйенің блок схемасы құрылды;  
C# бағдарламасының жүйесі;  
Matlab бағдарламасы ортасы.  
б) түсініктеме жазбасы 69 бет.

Дипломдық жобаны орындау барысында қойларды есепке алу жүйесін әзірлеуді қамтамасыз ететін техникалық құрылғыларды таңдау, талдау, олардың бірлескен әрекеті негізінде біртұтас автоматтандырылған жүйені әзірлеу болып табылады.

Теориялық бөлімде қойларды есепке алуға, оларды анықтауға арналған құлақ белгілері қарастырылып, компьютерге қосылатын, құлақ белгілерін оқуға арналған құрал қарастырылды. Қойларды есепке алуды автоматтандырылған жүйесінің құрамы мен мақсаты анықталған. Сонымен қатар, жүйені құру үшін C# бағдарлама ортасы және Matlab бағдарламасы ортасы қолданылып, қойларды есепке алу жүйесі құрылды.

Студент дипломдық жобаны жасауда өздігінен жұмыс істеу қабілетін көрсете алды. Мырзаш Әділжан Ермұханұлы алдына қойған инженерлік есептерді шеше алатынын, әдебиеттермен жұмыс істей алатындығын көрсетті.

Жалпы дипломдық жобаны «90/А/өте жақсы» деп бағалауға, Мырзаш Әділжан Ермұханұлы 6B07103 - «Автоматтандыру және роботтандыру» мамандығы бойынша техника және технология саласының бакалавр квалификациясын беруге лайықты деп санаймын.

**Ғылыми жетекші:**

Тех. ғыл. маг. аға оқытушы

Зікір Зікірбай Қ

«02» 06 2023 ж

## СЫН – ПІКІР

Дипломдық жоба үшін  
Мырзаш Әділжан Ермұханұлы

6B07103 - Автоматтандыру және роботтандыру  
Тақырыбы: «Қойларды есепке алу жүйесін әзірлеу»

Орындалды:

- а) Қойларды есепке алуды автоматтандырылған жүйесінің интерфейсі қарастырылды; Қойларды есептеуді қамтамасыз ететін жүйенің блок схемасы құрылды; С# бағдарламасының жүйесі; Matlab бағдарламасы ортасы.
- б) түсініктеме жазбасы 69 бет.

## Жобаға ескертулер

Дипломдық жобаны орындау барысында қойларды есепке алу жүйесін әзірлеуді қамтамасыз ететін техникалық құрылғыларды таңдау, талдау, олардың бірлескен әрекеті негізінде біртұтас автоматтандырылған жүйені әзірлеу болып табылады.

Теориялық бөлімде қойларды есепке алуға, оларды анықтауға арналған құлақ белгілері қарастырылып, компьютерге қосылатын, құлақ белгілерін оқуға арналған құрал қарастырылды. Қойларды есепке алуды автоматтандырылған жүйесінің құрамы мен мақсаты анықталған. Сонымен қатар, жүйені құру үшін С# бағдарлама ортасы және Matlab бағдарламасы ортасы қолданылып, қойларды есепке алу жүйесі құрылды.

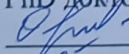
## Жобаны бағалау

Дипломдық жобада бүкіл мәселелер толықтай қарастырылған дей келе, «90/А/өте жақсы» және толық деп бағалап, оны орындаушы Мырзаш Әділжан Ермұханұлы 6B07103 - «Автоматтандыру және роботтандыру» мамандығы бойынша бакалавр лауазымына лайықты деп санаймын.

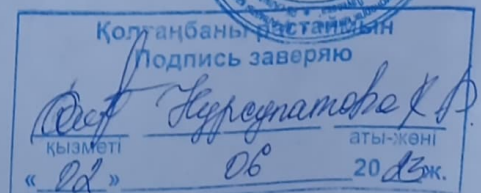
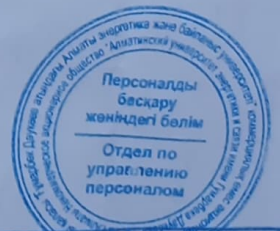
## Сын пікір беруші:

Алматы энергетика және байланыс университеті

PhD докторы, доцент

 Оракбаев Е.Ж

« 02 » 06 2023 ж





**Протокол анализа Отчета подобия  
заведующего кафедрой / начальника структурного подразделения**

Заведующий кафедрой / начальника структурного подразделения заявляет, что я ознакомился(-ась) с Полным отчетом подобия, который был сгенерирован Системой выявления и предотвращения плагиата в отношении работы:

**Автор:** Мырзаш Әділжан Ермұханұлы

**Название:** «Қойларды есепке алу жүйесін әзірлеу»

**Координатор:** Зікірбай Қуаныш Ергараұлы

**Коэффициент подобия 1:** 9.35%

**Коэффициент подобия 2:** 1.35%

**Замена букв:** 8

**Интервалы:** 0

**Микропробелы:** 0

**Белые знаки:** 0

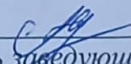
**После анализа отчета подобия заведующий кафедрой / начальника структурного подразделения констатирует следующее:**

- обнаруженные в работе заимствования являются добросовестными и не обладают признаками плагиата. В связи с чем, признаю работу самостоятельной и допускаю ее к защите;
- обнаруженные в работе заимствования не обладают признаками плагиата, но их чрезмерное количество вызывает сомнения в отношении ценности работы по существу и отсутствием самостоятельности ее автора. В связи с чем, работа должна быть вновь отредактирована с целью ограничения заимствований;
- обнаруженные в работе заимствования являются недобросовестными и обладают признаками плагиата, или в ней содержатся преднамеренные искажения текста, указывающие на попытки сокрытия недобросовестных заимствований. В связи с чем не допускаю работу к защите.

Обоснование: В результате проверки на антиплагиат были получены коэффициенты: Коэффициент подобия 1: 9.35% и Коэффициент подобия 2: 1.35%. Работа выполнена самостоятельно и не несет элементов плагиата. В связи с этим, признаю работу самостоятельной и допускаю ее к защите перед государственной комиссией.

« 1 » июня 2023 г.

Дата


  
Подпись заведующего кафедрой /  
начальника структурного подразделения

**Окончательное решение в отношении допуска к защите, включая обоснование:**

Дипломный проект допускается к защите.

« 1 » июня 2023 г.

Дата

  
Подпись заведующего кафедрой /  
начальника структурного подразделения



## Протокол анализа Отчета подобия Научным руководителем

Заявляю, что я ознакомился(-ась) с Полным отчетом подобия, который был сгенерирован Системой выявления и предотвращения плагиата в отношении работы:

**Автор:** Мырзаш Әділжан Ермұханұлы

**Название:** «Қойларды есепке алу жүйесін әзірлеу»

**Координатор:** Зікірбай Қуаныш Ергараұлы

**Коэффициент подобия 1:** 9.35%

**Коэффициент подобия 2:** 1.35%

**Замена букв:**8

**Интервалы:**0

**Микропробелы:**0

**Белые знаки:** 0

**После анализа Отчета подобия констатирую следующее:**

- обнаруженные в работе заимствования являются добросовестными и не обладают признаками плагиата. В связи с чем, признаю работу самостоятельной и допускаю ее к защите;
- обнаруженные в работе заимствования не обладают признаками плагиата, но их чрезмерное количество вызывает сомнения в отношении ценности работы по существу и отсутствием самостоятельности ее автора. В связи с чем, работа должна быть вновь отредактирована с целью ограничения заимствований;
- обнаруженные в работе заимствования являются недобросовестными и обладают признаками плагиата, или в ней содержатся преднамеренные искажения текста, указывающие на попытки сокрытия недобросовестных заимствований. В связи с чем, не допускаю работу к защите.

Обоснование: В результате проверки на антиплагиат были получены коэффициенты: Коэффициент подобия 1: 9.35% и Коэффициент подобия 2: 1.35%. Работа выполнена самостоятельно и не несет элементов плагиата. В связи с этим, признаю работу самостоятельной и допускаю ее к защите перед государственной комиссией.

«31» мая 2023 г.

Дата

*З.К.*

Подпись Научного руководителя