

ҚАЗАҚСТАН РЕСПУБЛИКАСЫ БІЛІМ ЖӘНЕ ҒЫЛЫМ МИНИСТРЛІГІ

Коммерциялық емес акционерлік қоғамы
АЛМАТЫ ЭНЕРГЕТИКА ЖӘНЕ БАЙЛАНЫС УНИВЕРСИТЕТІ

«Компьютерлік технологиялар» кафедрасы

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

(КОЛЫ)

« _____ » _____ 2014 ж.

ДИПЛОМДЫҚ ЖОБА

Тақырыбы: «Smart мобильдік қосымшасын құру»

5В070400 – «Есептеу техникасы және бағдарламалық қамтамасыз ету»
мамандығы бойынша

Орындаған Сапар Ержан тобы: ВТк-10-02

Жетекші аға оқытушы Рахимжанова З. М.

Keңесшілер :

Экономикалық бөлім бойынша :

доцент Боканова Г.Ш.

« 11 » 06 2014ж.

(колы)

Өмір тіршілігі қауіпсіздігі бойынша:

т.ғ.к., аға оқытушы Муташева Г.С.

« 29 » 05 2014 ж.

(колы)

Есептеу техникасын қолдану бойынша :

аға оқытушы Рахимжанова З. М.

« _____ » _____ 2014 ж.

(колы)

Мөлшер бақылаушы:

аға оқытушы Рахимжанова З. М.

« _____ » _____ 2014 ж.

(колы)

Пікір жазушы :

ҚазНУ, информатика кафедрасының доценті, т.ғ.д.

« 29 » мамыр 2014 ж.

(колы)

Муқимбетов М. Ж.

Алматы 2014

ҚАЗАҚСТАН РЕСПУБЛИКАСЫ БІЛІМ ЖӘНЕ ҒЫЛЫМ МИНИСТРЛІГІ

Коммерциялық емес акционерлік қоғамы
АЛМАТЫ ЭНЕРГЕТИКА ЖӘНЕ БАЙЛАНЫС УНИВЕРСИТЕТІ

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

жобаны орындауға берілген

ТАПСЫРМА

Студент Сапар Ержан Жайлаубайұлына

Жоба тақырыбы Smart мобильдік қосымшасын құру
ректордың « ___ » _____ № ___ бұйрығы бойынша бекітілген.

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

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

PHP, HTML басқару жүйесін ортасын пайдалану арқылы компанияның қызметкерлері туралы дерекқор көзі шифрланған ақпаратты ұсынатын қосымша құрылды.

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

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

– Ақпараттық жүйелер.

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

– Техникалық-экономикалық негізделуі.

– Өндірістегі еңбек қорғау мен қауіпсіздік шараларын ұйымдастыру.

Сызба материалдарының (міндетті түрде дайындалатын сызуларды көрсету) тізімі:

ДИПЛОМ ЖОБАСЫН ДАЙЫНДАУ

КЕСТЕСІ

№ р/с	Тарау аттары, әзірленетін сұрақтардың тізімі	Жетекшіге ұсыну мерзімдері	Ескерту
1	Автоматтандырылған қызмет көрсету орнының анықтамасы (арналуы, құрылымы, т.б.).	03.04.14.	
2	Ақпараттық жүйелер, дерекқорлар (ДҚ), дерекқорларды басқарудың жүйелері (ДҚБЖ) түсініктері.	05.04.14.	
3	Жүйенің дерекқорын жобалау және жүзеге асыру, жобалау құралдарын таңдау, ER моделін жобалау.	23.03.14.	
4	Дерекқорды құру, пайдаланушы интерфейсін жобалау, дерекқормен байланысын орнату.	11.05.14.	
5	Шифрлау, деректерді мөлдір шифрлау (TDE) түсініктері мен принциптері.	20.04.14.	
6	Жүйенің дерекқорын шифрлау түрін таңдау, жүзеге асыру	25.04.14.	

Жоба тараулары бойынша кеңес берушілер және оның мерзімі:

Бөлім	Кеңесші	Мерзімі	Қолы
Негізгі бөлім	Рахимжанова З.М.		
Тіршілік қауіпсіздігі	Муташева Г.С.		
Экономикалық бөлім	Боқанова Г.Ш.		
Мөлшер бақылаушы	Рахимжанова З.М.		
Есептеу техникасын қолдану	Рахимжанова З.М.		

Тапсырманың берілген уақыты « _____ » _____ 2014 ж.

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

Жоба жетекшісі _____ аға оқытушы Рахимжанова З.М.
(қолы)

Орындалатын тапсырманы қабылдаған студент _____ Сапар Ержан Жайлаубайұлы
(қолы)

АҢДАТПА

Ұсынылып отырған дипломдық жұмыстың мақсаты “Smart” жарнама агенттігінің мобилді веб сайтының құруы. Қазіргі таңда елімізде смартфондарға сұраныс көбейіп келеді, смартфондар өндірісінде жылдан-жылға сандық та сапалық даму байқалады. Интернетке көп жұртшылық смартфон арқылы қосылатыны баршамамызға аян. Дегенмен, көптеген веб сайттардың мобилді нұсқасы жоқ, сонысымен қолдануға ыңғайсыз. Осы дипломдық жұмыс қолданушылардың жұмысын жеңілдету мақсатында мобилді веб сайт құруға арналған.

Дайындалған мобилді веб сайт негізінде қолданушылар қажеттіліктеріне сай өздеріне кез келген ыңғайлы жер мен уақытта өз смартфондарынан сайтқа қосылып, сайттан керекті ақпарат алып, қызметтер бағасымен, түрімен таныса алады. Сонымен қатар мобилді сайттарда онлайн сатып алу, тапсырыс беру сияқты операцияларды орындауға болады.

Бағдармалық жабдықтауды дайындауда HTML, CSS Mysql и PHP программалық тілдері қолданылды.

Сонымен қатар, дипломдық жұмыста “Smart” мобилді веб сайтының қолдану ережелері де ұсынылған.

АННОТАЦИЯ

Целью данной дипломной работы является разработка мобильного веб сайта рекламного агенства “Smart”. В настоящее время нашей стране увеличивается спрос на смартфоны, в производстве смартфонов год за годом очевиден рост и в качестве и в количестве. Большинство людей подключаются в интернет через смартфоны. Но не у всех веб сайтов есть мобильная версия, тем и неудобны для использования. Данная дипломная работа посвящена для создания мобильного веб сайта с целью облегчения работы пользователей.

Разработанный мобильный веб сайт предоставляет возможность пользователям подключаться на сайт в любом удобном месте и время через свои смартфоны и получить необходимую информацию, и могут ознакомиться видами услуг и их ценам. А также в мобильных веб сайтах можно совершать операции как купить, заказать онлайн.

Для разработки программного обеспечения использованы язык программы HTML, CSS Mysql и PHP.

В дипломной работе даны правила и указания по применению данного интернет магазина “ Smart ”.

ANNOTATION

The given degree work is a development mobile web site of the company "Smart". At present our country internet shops are growing year for year obviously and as and in amount. But now it is not enough to use the last technology, this degree work is created for the reason creation make internet shop.

Designed internet shop gives a chance to client to buy goods which they need without walking out their building. Presented degree work gives a chance consumer to choose goods on taste straight on put.

For software development are used language of the program HTML, CSS Mysql and PHP.

In degree functioning are given rules and instructions on using given internet shop " Smart ".

МАЗМҰНЫ

КІРІСПЕ	8
1 Web технологияның қазіргі жағдайы.....	9
1.1 Интернет туралы түсінік	9
1.2 Сайт туралы түсінік.....	11
1.3 Веб–сайттың түрлері.....	12
1.4 Кле́нт–сервер технологиясы.....	13
2 Сайт жасау құралдары	17
2.1 HTML тілінің негіздері	17
2.2 PHP тілінің негіздері	23
2.3 MySQL дерек қоры негіздері.....	29
2.4 CSS бағдарламалау тілінің ерекшелітері	36
2.5 JavaScript тілінің негіздері.....	38
3 “Smart” жарнама агенттігінің мобильді веб сайты жасау.....	40
3.1 Мәселенің қойлымы	40
3.2 “Smart” мобилді веб сайтының құрылымы мен мазмұны.....	40
3.3 Бағдарламалық жабдықтау	41
3.4 “Smart” мобилді веб сайты қолданушыларға нұсқаулық.....	46
4 Өмір тіршілік қауіпсіздігі.....	51
4.1 Еңбек шартының анализы	52
4.2 Есептеу бөлімі.....	54
4.3 Тіршілік қауіпсіздігі бөлімі бойынша қорытынды	62
5 Техико-экономикалық негізделуі.....	64
5.1 Жоба сипаттамасы.....	64
5.2 Бағдарламаның өңделуінің еңбек өнімділігінің есептелуі.....	67
5.3 Бағдарламалық қамтаманы өңдеуге кеткен шығынны	68
ҚОРЫТЫНДЫ	84
Пайдаланған әдебиеттер тізімі:	85
Қосышасы А.....	85

КІРІСПЕ

Қазіргі таңда ақпараттық технологиялар қарқынды дамыған заманда интернет өмірдің ажырамас бір бөлігі болып табылады. Интернет (ағылш. *Internet — International Network*) — компьютерлік серверлердің бүкіләлемдік желісі. Интернетке қосылу мүмкіндігі болған жағдайда, білім беру мекемелері, мемлекеттік ұйымдар, коммерциялық кәсіпорындар және жеке адамдар сияқты миллиондаған қайнар көзінен ақпарат алуға болады. Қазіргі кезде Интернет сөзін пайдаланғанда, физикалық желінің өзін емес, Дүниежүзілік желі және ондағы ақпаратты айтамыз. Сонымен қатар интернет кез келген компьютерлермен бүкіл әлем бойынша ақпарат алмасу мен беру мүмкіндігі, желілер жүйесі, байланыс араларын өзара біріктіретін, тораптардың жиынтығы.

Интернет технологисы жылдам өзгеріп отырады. Интернетпен жұмыс істеу оңайланғандықтан, қазіргі өзгерістер торапты кім немесе қандай мақсатпен қолдануында болып отыр. Дегенмен, " Web-тен білгім келген нәрсе туралы, информация таба аламын ба?"-деген сұрақ туындайды. Сол себепті бір жаққа телефон соғудан бұрын немесе кітапханаға барар алдында Web-тен информация алады.

Айта кетерлік жағдай – бүгінгі күні смартфондар мен планшеттердің дамып келе жатқан уақыты. Әрбір қолданушы интернетке бұрынғыдай дербес компьютерге қарағанда қолындағы смартфоннан көбірек қосылады. Осыған орай, қолданушыларға қолайлылық мақсатында веб сайттардың мобилді нұсқасын жасау елімізде белең алып отыр. Мобилді веб сайтқа смартфоннан немесе планшеттен қосылудың көптеген артықшылықтары бар.

Қолданушы өзінің қажеттілігіне қол жеткізуі үшін, ыңғайлы жиһазда отырып, интернет-беттерді смартфон арқылы аша алады және де қажетті ақпаратты немесе операцияны орындай алады. Қолданушының алдында мағлұматтар түрлерінің, сонымен қатар әр түрлі қызмет көрсетулердің тізімі бар. Мобилді веб сайт, көпшілік адамдардың пікірі бойынша - бұл өті ыңғайлы сайт нұсқасыларының бірі. Және бұл шынымен ыңғайлы болғандықтан ол дамып келеді. Әр ұйым үшін сайттың дамуы- бұл үлкен артықшылық, себебі қысқа мерзім ішінде базасы үлкееді, компанияның тұтынушылар саны артады, сонымен қатар бұл тиімді болғандықтан уақыт үнемделеді.

1 Web технологияның қазіргі жағдайы

1.1 Интернет туралы түсінік

Интернет (бас әріппен жазылса) – кез келген компьютерді жер шарында орналасқан басқа жұмыс станциясымен, яғни телефон арнасына қосылған басқа компьютермен жылдам байланыстыратын Дүние жүзілік Желі. Оны дүниедегі ең үлкен ауқымды (глобальный) желі деп атайды. Осылай телефон арнасы арқылы байланыса алатын компьютерлер бір-бірімен TCP/IP хаттама (протокол) ережелерімен мәлімет алмасады, оларды бір нұсқада, яғни бір тілде «сөйлейді» деп айтса да болады. Дүние жүзілік Халықаралық телефон желісі сияқты оны ешкім басқармайды, ол ешкімнің жеке меншігі емес. Міне, осы Интернет желісі көмегімен электрондық почта арқылы хабар алып (беріп), басқа компьютерлердегі ақпаратты көріп, қашықтан телеконференцияларға қатынасу жұмыстарын жүргізуге мүмкіндік бар. TCP/IP – Интернет желісіне қосылған компьютерлер арасында ақпарат алмасуды қамтамасыз ететін мәліметтерді бір жүйеге келтіру ережелері немесе оларды құрастыру хаттамасы. IP (Интернет Protocol) – мәліметтерді оны алушының адресі көрсетілген шағын тақырыптары бар бірнеше бөліктерге немесе дестелерге бөлетін желі аралық хаттама. TCP (Transmission Control Protocol) – мәліметті жөнелту ісін басқаратын хаттама, ол желідегі ақпарат дестелерін дұрыс жеткізу үшін жауапты болып саналады. Интернет жүйесін пайдалана отырып, үйден шықпай ақ, көптеген елдерді, қалаларды аралап музейлерді, кітапханаларды көріп, дүниежүзілік мәдени және ғылыми жетістіктермен танысып, оларға өркениетті елдің азаматы ретінде өз үлесіңізді қосып, өзіңізді дүниенің бір кішкене бөлігі ретінде сезіну мүмкіндігіне ие боласыз. Интернет (кіші әріппен жазылса) – TCP/IP хаттамалары негізінде желі аралық байланысу технологиясы.

Интернет желісінің құрылымы – әрбір тұтынушы компьютері кәдімгі телефон арналарымен түйінді машиналармен байланысады. Ал түйінді немесе негізгі машиналар бір-бірімен қуатты оптикалық талшықты немесе спутникті арналармен жалғасады. Түйінді машиналар кез келген жай компьютерлер арасында байланыс орнату үшін қажет, олар: тәулік бойынша үзіліссіз жұмыс істеп, байланыс сеанстарының арасындағы уақытта жолда жүрген ақпараттарды уақытша сақтайды, ақпараттық серверлер деп аталатын мәлімет жинақтаушы компьютермен жылдам істейтін оптикалық түрдегі байланыстыру ісін қамтамасыз етеді. Ақпаратты сервер – дегеніміз қалың көпшілікке арналған, әрбір тұтынушы пайдалана алатын көптеген ақпараттар түрлері жинақталған арнаулы компьютер. Мұнда жаналықтар, мерзімді баспасөз (газет-журналдар), жарнамалар, т.с.с. мәліметтер сақталады. Желіге қосылған әрбір компьютерге қайталанбайтын айрықша өзіндік адрес (IP-адрес) беріледі, адрес компьютердің типімен (IBM, Macintosh), операциялық жүйенің түрімен (MS DOS, Windows 98, Windows NT) байланыста болмайды, демек жіберілген ақпарат тура адрес көрсетілген

компьютерге келіп түседі. Түйінді машина мәліметтерді тасымалдау кезінде хабарды жіберген және оны алатын компьютерлер арасындағы ең қысқа жолды таңдайды, жұмыс барысында ол байланыс арнасының бос аралықтарын тиімді түрде пайдаланады. Бұл мүмкіндік Интернетті қазіргі кездегі ең жылдам, әрі арзан, әрі сенімді байланыс жабдығына айналдырды деуге болады. Ал енді түйінді машиналар жұмысы қалай ұйымдастырылған, Интернетте адресі кім тағайындайды, компьютерлер арасындағы байланыс қалай қамтамасыз етіледі, компьютер телефонмен қалай қосылады деген мәселелерге келсек, бұның бәрін істейтін арнайы қоғам – Интернет провайдері деген ұйым, мекеме бар. Мысалы, Алматыда Интернет Провайдері болып қызмет атқаратын Интернет трейнинг орталығы Nursat, S&G Communications, Parasang, қазақтелеком, Астел Арна Спринт т.с.с. мекемелер жұмыс істейді.

Интернет-те қызмет көрсету провайдері – ISP (Интернет Service Provider), Интернет-пен қарапайым тұтынушылардың тікелей қатынас құруын жүзеге асыратын заңды тұлға.

World Wide Web құрылымы–WWW дүнежүзілік тармақталған желісі–бұл Интернет-тің ең кең таралған, күннен күнге тоқтаусыз өсіп жатқан қызмет түрі. World Wide Web–Интернет–тегі барлық құжаттар және мультимедиалық ресурстарды сипаттайтын термин. Бұл мәліметтерді пайдалану жолында оларды оқып көру үшін Microsoft Интернет Explorer, Netscape Navigator сияқты програмалық жабдықтар қолданылады. Олар Интернет–тен керекті құжаттарды іздеп табуға, қарауға, жазып алуға мүмкіндік береді. «Гипермәтін (байланысқан мәтіндер) мүмкіндігі Web мәліметтерінің бірінен біріне көшуді жеңілдетеді. Файлдардың «HTML–ің (hypertext markup language) форматын іске қосатын гипермәтін мүмкіндігі Web парақтарының бір бетінен екінші бетіне «сілтеу бойынша көшуді жеңіл жүзеге асырады. Бұл беттерде суреттер, бейне жазбалар, дыбыстар, үшөлшемді (кеңістіктегі) нақты бейнелер т.с.с. орналасады. Осындай сілтемелер арқылы байланысқан парақтар World Wide Web–тің негізгі құралымын құрайды. Web жүйесіне қосылғаннан кейін әрбір адам WWW желісіндегі кез келген мәліметті басқалармен бірдей пайдалану құқығына ие болады, бұдан соң басқа компьютермен байланасу үшін немесе қосымша артықшылық, шектеулер үшін ақы төлеудің қажеті жоқ.

World Wide Web (WWW немесе Web) Интернет мәліметтерін жеңіл көруге болатын графикалық интерфейс мүмкіндігін береді. Ондағы мәліметтер мен олардың бір–бірімен байланысын ақпараттық «өрмек деп айтса да болады. Web–тің әр бетінің басқа парақтармен байланысын көрсететін сілтеме белгілері бар, оны бір–бірімен байланысқан парақтардан тұратын өте үлкен кітапхана деуге болады. Бір тораптық компьютерде орналасқан мәліметтер Web кітабы секілді, ал оның беттері кітап парақтарын көзге елестетеді. Бұл беттердегі мәліметтер дүниенің кез келген бұрышында орналаса береді. Солар арқылы жер шарындағы барлық серверлік компьютерлердегі ақпараттар көз алдыңызда орналасады, мұнда қашықтағы–қымбат, жақындағы–арзан деген ұғым жоқ, олардың бағасы тек

мәліметтің көлеміне немесе сіздің байланысып отырған уақытыңыздың ұзақтығына байланысты. Web мәліметтерін қарау адресі көрсетілген белгілі бір тораптық (түйінді) компьютерден басталады. Түйінді компьютердегі мәліметтің бірінші беті кітаптың сырты немесе мазмұны тәрізді, әрбір беттің URL (Universal Resource Locator) форматында берілген қайталанбайтын өзіндік адресі болады. Ол беттердегі мәліметті оқу «көру жабдықтарың деп аталатын арнайы программалар арқылы орындалады.

URL–адрестері–Бұл Интернетте орналасқан құжаттардың адресін жазудың арнайы формасы. Ол–желінің қай серверінде орналасса да, керекті мәліметті айнытпай қатесіз табуды қамтамасыз ететін сөз тіркесі URL адрестерінің жазылуынан мысал келтірейік.

[http:// www.soccer.ru/dinamo/rus/index.html](http://www.soccer.ru/dinamo/rus/index.html)

мұндағы:

а)http протокол, яғни хаттама;

б)www.soccer.ru–адресінің домендік бөлігі, оның ішінде:

в)www–компьютердің (сервердің) аты;

г)soccer.ru–доменнің аты;

к)dinamo/rus–сервердегі каталог аты;

т)index.html–Web–парақтары файлының аты;

Домен–бір компьютермен басқарылатын немесе бір желілік жұмыс машинасының (желі торабының) басшылық етуімен істейтін информациялық желі қорлары (ресурстары) тобы.

1.2 Сайт туралы түсінік

Веб–сайт түсінігі– Веб–сайт немесе жәй сайт (ағыл. Website, web–өрмекшінің торы және site– орын)–бір немесе жинақталған веб–парақтары интернеттеHTTP/HTTPS хаттамалары арқылы мүмкіншілік алады. Сайттың парақтары жалпы түпкі мекен–жайларды, сонымен қатар қарапайым тақырыпты, логикалық құрлымды, рәсімдеуді және авторлауды біріктіреді, жинақталған барлық жалпы мүмкіндік алған сайттар жалпы әлемдік торды құрайды. Сайттың мағынасы– кез келген өзгелерге желі арқылы 1–2 гб мәліметті бере алуы үшін және өзі телеканал болуы үшін, жалпы жергілікті желі арқылы мемлекет тұрғындарын біртұтас күшке біріктіру.Сайттың мағынасы ертеректегі түсінікте физикалық желі түйіні–хостпен сервермен шатасырлған, бірақ интернеттің өсуімен, серверлер технологиясының жақсаруымен бір компьютерде көптеген сайттармен домендерді орналастыруға болатын болды.

Веб–сайт тарихы– Әлемдегі бірінші сайт 1991 жылы 6–тамызда пайда болды, оны жасаған Тим Бернерс–Ли. Сайтта World Wide Web– тың жаңа технологиясы, HTTP мәліметтерді тасымалдау хаттамасының негізі, URI жүйелік адресациясы және HTML гипертекстік белгілеу тілі баяндалған. Сонымен қатар сайтта серверлермен браузердің жұмыс істеу және орнату ережелері де баяндалған. Сайт әлемдегі алғашқы интернет–каталог болды, сонымен кейінірек Тим Бернерс–Ли оған басқа сайттардың да сілтемелік

тізімін орналастырды. Бернерс–Ли дің алғашқы сайтының жұмыс істеуіне қажетті барлық құралдар ертеректе дайындалған–1990 жылдың соңында алғашқы гипертекстік браузер WorldWideWeb веб редактор функциясымен пайда болды, деркқордағы алғашқы сервер NeXTcube және алғашқы веб парағы болды.

Тим Бернерс–Ли, гипертекст негізінен желіде мәліметтер алмалмасуды атқаруы мүмкін деп есептеді, және оған өмірдегі өз ойларын жүзеге асырудың сәті түседі. Тағы 1980 жылы Тим Бернерс–Ли кездейсоқ қауымдастықтың мәліметтерін сақтауға қолданылатын гипертекстік бағдарламалық қамтама Enquire – ні жасаған. Содан соң Женеведегі Ядролық зерттеулердің Европалық орталығында жұмыс істеді, ол қызметтестеріне гипертекстік документтерді жәрылау туралы, өзімен гиперсілтелер арқылы байланысуды ұсыныс етеді. Бернерс–Ли Гипермәтіндік рұқсаттың ішкі іздеулермен құжаттарға, сонымен қатар интернеттің жаңа қорларына мүмкіндігін Демонстрациялап (көпшілікке паш ету) өтті. Нәтижесінде 1991 жылы мамырда Женева (CERN) де WWW стандартты бекітілді.Тим Бернерс–Ли HTTP, URI/URL и HTML веб технологияларының негізін салушы атасы болып табылады. Бүгінгі күнде, интернет желісінде өзінің қызмет сферасында, саудалық немесе саудалық емес қызметтермен шұғылданатынын ұсынғанда кез–келген компанияға сайт жасау керек.Интернет күніне 24 сағат жұмыс істейді, осы уақыт ағымында сіздің бизнесіңізбен өзіңіздің тікелей қатысуыңызсыз мыңдаған потенциалды клиенттер танысуы мүмкін.Сайт–бұл сіздің интернеттегі өкілдік түріңіз, сізге клиенттердің және серіктестердің базасын кеңейтуге, online да сатуды басқаруға, сіздің компанияңыздың танымалдығын жоғарылауға мүмкіндік береді.

Веб–сайт функциялары

– **Жарнамалау** фирманың өзіндік бейнесін жасап және оны қолдап, жарнамалық акциялар өткізіледі.

– **Хабарландыру** жалпы фирма тралы жан–жақты мәліметтер беріп, оның қызмет көрсету бағасы мен әріптестерге жеткізу қызметі жалпыға бірдей хабарланады.

– **Сату** каталогда бағалармен көрсетілген товарларға тапсырыс беруді құжаттаумен ақша төлеудің мүмкіндіктерін береді.Зерттеу– киенттердің тауарлар жәйлі сұрақтарына қатысты ақпараттарды жинайды.

– **Құрлымдау** Фирманың құрлымын кемелдендіріп, әріптестердің фирманың дерекқорларына қатынас орнатуына анағұрлым көп мүмкіндік береді.

1.3 Веб–сайттың түрлері

Визиткалы сайт–қарапайым сайт болып табылады.Ол html құралдарысыз қолданушылармен интерактивті байланысты береді.Сайт фирманың дизайнына бүкілдей сәйкеседі.

Фирма сайтты тек потенциалды клиенттерге жарнамалық ақпараттарды беру ұшын қолданады. Сайттың негізгі беттері: Негізгі бет – Сіздің фирма туралы қысқа да нұсқа ақпараттардан тұрады. Компания туралы – фирманың жетістіктерімен, клиенттермен жұмысы және мұның сізге берер пайдасын көрсетеді. Өнім – Фотолармен берілген товарлар туралы қысқаша ақпараттардан тұрады. Контакт – Әдетте негізгі контакты ақпарат әрбір бетте қайталаынады. ол сіздің компания туралы толық ақпараттан тұрады.

Жарнамалық сайттар – бұл түрдегі сайт товарлармен қызмет өтеуді көрсету ұшын қолданады. ол жайнаған дизайнға толы болады. бұл түрдегі сайт төмендегідей қосымша беттерден құрам табады: Товарлар каталогі – ол визиткалы сайт товарларға қатысты жан-жақты ақпараттар берумен ерекшеленеді. әрі көрсетілген қасиеті бойынша іздеу функциясы бар.

Қайдан, қалай сатып аласыз – сату нүктелері, қойма, офис меңгерушілер туралы ақпараттар бар. Бағалар – жалпы товарлардың бағалар тізімін береді.

Қолдау – компаниямен байланыс орнатудың мүмкіндіктері көрсетіледі.

Ақпараттық сайттар – Ақпараттық сайттар – қажетті ақпараттарды жинаумен ерекшеленеді.

Жаңалықтар – назар аударуға тұрарлық жаңалықтармен мәліметтер береді.

Тақырыптар каталогы – сайттағы мақалалармен материалдардың негізгі тақырыпымен беріледі. қолданушы кілттік сөздермен іздеуді орындайды.

Сілтеме және форум – таныс қолданушыларға арналған ақпараттардан құрам табады.

Интернет-магазин (дүкен) – Бұл түрдегі сайт товар сатумен қызмет көрсетудің тиімді түрі болып табылады. сатып алушылар ұшын мұның ең негізгі ерекшелігі таварға үйінде отырып – ақ тез әрі арзан қолжеткізетіндігінде.

1.4 Кліент-сервер технологиясы

Ереже бойынша ақпараттық жүйенің құрамына кіретін компьютерлер және бағдарламалар тең құқықтық болып табылмайды. Желідегі бір компьютер процессорлар, файлдық жүйе, почталық қызмет, басу (печати) қызметі, мәліметтер базасы сияқты ақпараттық – есептеу ресурстарды орналастырады. Кез – келген ресурсты басқаратын компьютерді сол ресурстың сервері, ал онымен қолданғысы келген компьютерді клиент деп аталуы қабылданған. Нақты сервер өзінің ие болған ресурс түрімен сипатталады. Яғни, егер ресурс мәліметтер базасы болса, онда тағайындалуы – мәліметтер базасын өңдеуге байланысты клиенттердің сұраныстарына қызмет көрсету болып табылатын, мәліметтер базасының сервері туралы айтылады; егер ресурс – файлдық жүйе болса, онда файлдық сервер немесе файл-сервер туралы айтылады және т.с.с. Бұл принцип үрдістердің өзара әсерлесуіне де таралады. Егер олардың біреуі басқаларға сәйкесінше қызметтер жинағын беру арқылы кейбір функцияларды атқаратын болса, мұндай үрдіс сервер ретінде қарастырылады. Осы

қызметтерді пайдаланатын үрдістерді клиенттер деп атау қабылданған. Бүгінгі таңда “клиент–сервер” технологиясын кең қолданыс алуға, алайда өзінен-өзі ол әмбебап рецепттерді ұсынбайды. Ол тек қазіргі заманғы таратылған ақпараттық жүйенің қалай ұйымдастырылу керек екенінің жалпы көрінісін береді. Оның үстіне бұл технологияның нақты бағдарламалық өнімдерде және де бағдарламалық қамтаманың түрлерінде жүзеге асуы әлдеқайда ажыратылады. “Клиент–сервер” технологиясының негізгі принциптерінің бірі әртүрлі табиғаты бар стандарттық бағдарламаның қосымшаның (приложения) функцияларын үш топқа бөлуге негізделген. Бірінші топ – бұл мәліметтерді енгізу және көрсету функциясы. Екінші топ берілген пәндік облысқа сипатталған (характерные) тек қана қолданбалы функцияларды біріктіреді. Үшінші топқа мәліметтерді сақтау және басқару (мәліметтер базасын, файлдық жүйелерді және т.б.) функциялары жатады. Осыған сәйкес кез келген бағдарламада қосымшада (приложения) келесі логикалық компоненттер таңдалады:

- бірінші топтың қызметтерін жүзеге асыратын көрсету компоненті (presentation);

- екінші топтың қызметтерін қолдайтын қолданбалы компонент (business application);

- үшінші топтың қызметтерін қолдайтын ақпараттық ресурстарға қатынау компоненті (resource acces) немесе ресурстар менеджері (Resource manager).

“Клиент–сервер” технологиясының шеңберінде бағдарламаларды қосымшаларды (приложения) жүзеге асыруының айырмашылығы үш фактормен анықталады. Біріншіден, барлық үш топтың қызметтерін жүзеге асыру үшін қандай механизмдер қолданатындығымен. Үшіншіден, жүйеде компьютерлер арасында логикалық компоненттер қалай таратылғандығымен. Әрқайсысы келесі сәйкес келетін модельдерде жүзеге асатын үш жақын келулер (подход) ерекшеленеді (выделяются):

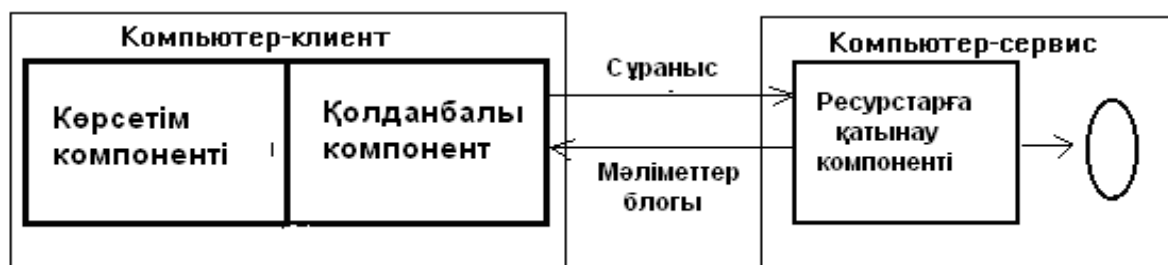
- қашықтықтағы мәліметтерге қатынау моделі (Remote Date Access – RDA);

- мәліметтер базасы серверінің моделі (DateBase Server – DBS);

- приложений (қолданба, қосымша) серверінің моделі (Application Server – AS).

RDA – моделінде көрсетім компонент және қолданбалы компонент кодтары бірлескен және компьютер–клиентте орындалады. Соңғысы мәліметтерді енгізу және көрсету функцияларымен бірге таза қолданбалы функцияларды да қолдайды. Ақпараттық ресурстарға қатынау, ереже бойынша арнайы тілдің операторларымен (SQL тілі, егер мысалы мәліметтер базасы туралы айтылатын болса) немесе арнайы кітапхананың функцияларын шақыруымен қамтамасыз етіледі (егер сәйкес келетін API болса). Ақпараттық ресурстарға сұраныстар қашықтықтағы компьютерге торап бойынша жіберіледі (мысалы, мәліметтер базасы серверіне). Соңғысы сұраныстарды өңдейді және орындайды және клиентке мәліметтер блогын

қайтарады (сурет 1.4.1). “клиент–сервер” архитектурасы туралы айтқанда, көбінесе осы модель туралы айтады.



Сурет 1.4.1 Қашықтықтағы мәліметтерге қатынау моделі.

DBS–моделі (сурет 1.4.2) компьютер–клиентте орындалатын үрдіс көрсетім функцияларымен шектеледі, ал қолданбалы функциялар компиляцияланатын резиденттік процедуралар немесе мәліметтер базасының процедуралары деп аталатын сақталатын процедураларда жүзеге асырылған (stored procedure) деген жорамалда құрылады. Олар тікелей мәліметтер базасында сақталады және мәліметтер базасының компьютер–серверінде орындалады (мұнда мәліметтерге қатынауды басқаратын компонент, яғни МББЖ қызмет етеді). Сақталатын процедуралардың механизмі – DBS–моделінің ерекше сипаттамасы– әзірше тек қана МББЖ– де, онда да бәрінде емес бар болғандықтан ақпараттық ресурстың түсінігі мәліметтер базасына дейін тарылған.



Сурет 1.4.2 Мәліметтер базасы серверінің моделі

Тәжірибеде егер мәліметтер базасының бүтіндігін қолдауды (поддержка) және кейбір қарапайым қолданбалы функцияларды сақталатын процедуралармен (DBS–модель) қолдайтын болса аралас модельдер жиі қолданылады, ал әлдеқайда күрделі функциялар компьютер–клиентте орындалатын (RDA–модель) қолданбалы бағдарламада тікелей жүзеге асырылады. Бірақ бірден екі моделдің элементтерін қамтитын мұндай шешімдер олардың арақатынасы туралы біздің представлений (елестету, көрсетім) принципно (принципшіл) өзгерте алмайды.

AS – моделінде (сурет 1.4.3) компьютер–клиентте орындалатын үрдіс мәліметтерді енгізу және көрсетуге жауап береді (яғни, бірінші топтың қызметтерін жүзеге асырады). Қолданбалы функциялар қашықтықтағы

компьютерде қызмет ететін үрдістер тобымен (қолданба серверлері) орындалады. Қолданбалы есептерді (задач) шешу үшін қажетті ақпараттық ресурстарға қатынау RDA–моделіндегідей әдіс бойынша қамтамасыз етіледі. Қолданбалы компоненттерден әртүрлі түрдегі ресурстарға қатынау бар–мәліметтер базасы, индекстелген файлдар, кезектер және т.б. Ереже бойынша, серверы приложений ресурс менеджері қызмет ететін компьютерде орындалады, бірақ басқа компьютерлерде де орындалуы мүмкін.



Сурет 1.4.3 серверы приложений моделі

Бұл екі модельдің фундаментальное айырмашылығы неде? RDA және DBS– модельдері функцияларды бөлудің екізвенотық сұлбасына тіреледі (опираются). RDA– моделінде қолданбалы функциялар бағдарлама–клиентке приданы (берілген), DBS–моделінде олардың орындалуы үшін жауапкершілікті МББЖ ядросы алады. Бірінші жағдайда қолданбалы компонент көрсетім компонентімен сливается, екінші жағдайда – ақпараттық ресурстарға қатынау компонентіне интегрируется. Керісінше, AS–моделінде функцияларды бөлудің үшзвенотық классикалық сұлбасы жүзеге асырылған, мұнда қолданбалы компонент қолданбалы бағдарламаның маңызды элементі ретінде белгіленген, оны анықтау үшін көпмақсатты (многозадачная) операциялық жүйенің әмбебап механизмдері қолданылады және екі басқа компоненттері бар интерфейстер стандартталған. Бұл ерекшеліктен AS – модельдер тәжірибелік қызмет үшін маңызы үлкен өздерінің артықшылығын көрсетеді

2 Сайт жасау құралдары

2.1 HTML тілінің негіздері

Егер сіз сізге арналған немесе сіз туралы мәлімет сақтайтын Web бет құрғыңыз келсе, сіз HTML (Hypertext Markup Language– гипертекстік белгілеу тілі) тілін үйренуіңіз керек. HTML – бұл беттің құрылымын суреттеу тілі. Ол сізге қарапайым мәтін алып, оны абзац, тақырып, тізім және басқа да құрылымы бойынша форматтауға мүмкіндік береді. Онымен қоса, HTML сіздің оқырмандарыңыз бір құжаттан екінші құжатқа оңай аттап өте алулары үшін, сізге байланысты беттерге сілтемелер құруға мүмкіндік береді. HTML SGML– ға негізделген (standart Genere Lized Markup Language– жалпы белгілеу стандартты тілі). Нақтырақ айтсақ, HTML бұл SGML құжатты типінің анықтаушысы, немесе SGML қағидасының өзіндік қосымшасы. Сіз HTML құжаттарын ақпараттық жүктемені сақтайтын форматталмаған текстің ішіндегі HTML тэгі (tags) арқылы құрасыз. HTML тэгтері–бұл “азырақ” (<) белгісінен басталып, “көбірек”(>)белгісімен аяқталатын символдар тізбегі. Web браузерлер HTML–ды синтаксистік тексерген кезде үш қағидаға сүйенеді:

- Пробел және басқа да “көрінбейтін символдарға” назар аудармайды.
- Форматтау тэгтері кіші немесе үлкен (бас) әріптерімен жазылуы мүмкін.
- Форматтау тэгтерінің көпшілігі сыңарларымен бірге жазылады.

Кейбір ерекше тэгтерді ескермесек, HTML форматтау тегтерінің барлығын да сыңар тегтері бар. Ашатын тег берілген эффекті активтендірсе, жабатын тег оны өшіреді. Тег сыңарларын көбінесе контейнер деп те атайды. Себебі, олар қосып – өшіретін эффект мәтінде сол екі тэгтің арасында ғана пайда болады. Мысалы, мәтінің қатары толық әріпен жазылуы керектігін көрсететін болсақ, былай жазамыз:

 Бұл қатар толық қаріппен жазылады. .

Жабатын тегтің ішінде орналасқан мәтін « кері қисық сызық» символымен шамаланады. Бұл қағидаға бағынбайтын тэгтер <BASE> (негізгі мәлімет),
 (қатар соңы), <HR> (горизанталь сызық) және IMG (бейне). Бұл тегтер жалғыз қолданылады.

HTML –ге тікелей қатысы болмаса да, HTML құжаттарының өте маңызды бөлігі болып URL адрестері есептеледі. Ол әр–түрлі тэгтерде кең көлемде қолданылады. Негізі URL World Wide Web–тіге құжаттың адресі болып есептеледі. URL– дың құрылымы жер шарының кез–келген белігінде орналасқан бірге жұмыс істейтін. Web – сервердің кез–келген типіндегі кез–келген құжаттардың ықшамды идентификациялау тәсілі мүмкіндігін береді. Ақпараттар серверінің хаттамасын ескермесек, URL–дың жалпы формуласы келесі:

Хаттама:// Сервер_ аты: порт/файл–аты

HTML–құжатын белгілеу үшін біз <HTML> ж,не </HTML> тэгтерін қолданамыз. Бұл тегтер өзінде HTML –дың барлық тэгтерін және құжаттың ақпараттық құрамын қамтиды. <HTML> құжаттардың бірінші қатарында, ал <HTML> тәгі құжаттың соңғы қатарында орналасуы керек. Бұл тегтер олардың арасында орналасқан барлық символдардың HTML–коды екенін білдіреді. Құжаттың тақырыбы <HTML> тегінен кейін жазылып, </HEAD> тегімен аяқталады. Сіз ақпарат тақырыбына көп мөлшерде символдар енгізе алсаңыз да, оның құрамында екі заттың болуы міндетті: құжат аты және оның URL–ы.

Сіз құжаттыңызға ат берген кезде, оны <TITLE> және </TITLE> тэгтерінің арасына орналастыра аласыз. Құжат аты Интернет Explorer 3 тақырыб қатарында және журнал (History) және избранные страницы (Favorites) тізімдерінде пайда болады. Сіз құжат атына сол құжатты толық түсіндіретіндей етіп таңдауыңыз керек. Бірақ ол 40 символдан аспауы керек. Сіз <BASE HREF= “базовый_URL”> тегінде URL негіздік адресін көрсетесіз. Ол құжаттың абсолют URL–іне теңестіріледі. Сіз құжат тақырыбын жазып болған соң оның денесін құруды бастайсыз. HTML құжатының денесі ақпаратты сақтайтын толық мәтінде және мәтінді форматтауға арналған барлық HTML тегтерін қамтиды. Құжат денесі тақырыбтан соң басталып, <BODY> және </BODY> тэгтерімен шектеледі. Құжат денесі келесі түрде көрініс табады:

```
<HTML>
<HEAD>
<TITLE> бұл жерде құжат аты орналасады </TITLE>
<BASE HREF= «негіздік_URL»>
</HEAD>
<BODY>
</BODY>
</HTML>
```

Жаңа абзац бастау және оны аяқтау үшін <P> және </P> тэгтері қолданылады. HTML спецификациясын байланысты <P> ALIGN атрибутын қолдайды. Ол абзацтың ішінде мәтінді туырлауға мүмкіндік береді. Абзацтар бас қатармен бөлінеді. Бас қатар тастамай жаңа абзацты бастау үшін
(қатар соңы) тәгік орданлыда. Мысалы:

```
<TITLE> өлең жолдары </TITLE>
<P>
```

You can tell it's Spring with all of the rain we're getting.
It makes me to recite the rhyme.

<!—Вы можете сказать, что уже весна, поскольку идет такой сильный дождь. Поэтому я хочу процитировать стихи—>

```
</P>
```

```
<P>
```

Rain, rain

Go away

Come again

Another day

HTML – тақырыб стилінің алты түрін қолдайды. Мәтінді қандай да бір стильде азу үшін оны <Hn> және </Hn> тәгтерінің арасына орналастыру керек. Олар былай жазылады.

<H1> Heading style 1 </H1>

<H2> Heading style 2 </H2>

<H3> Heading style 3 </H3>

<H4> Heading style 4 </H4>

<H5> Heading style 5 </H5>

<H6> Heading style 6 </H6>

HTML төрт физикалық стильді қолдайды. Физикалық стиль– бұл курсив немесе толықтық сияқты қаріптің атрибуттары. Олар келесі де көрсетілген.

HTML тілінің физикалық стилі

Стиль

Толық шрифт

Курсив

Астын сызу

Машинопись (фиксталген ен)

тег

...

<I>...</I>

<U></U>

<TT>...</TT>

HTML логикалық стиль олар ерекшеленген мәтіннің мағынасын анықтайды. Олардың қаріп атрибуттарына қатысы жоқ болғандықтан, олар әр браузерде әр-түрлі көрініс табуы мүмкін. Логикалық стильдер келесі кестеде көрсетілген.

HTML логикалық стилі

HTML тізімдері сіздің құжатыңыздағы ақпараттарды қызықты етіп көрсетуге мүмкіндік береді. Олар келесі кестеде көрсетілген.

HTML тізімдері кестесі

Тип

тізімі тегі

элемент тегі

Нөмірленген

...

...

Таңбаланған

...

...

Баяндау тізімі

<DL>...</DL>

<DD>...</DD><DT>...</DT>

Меню

<MENU>...</MENU>

...

Директория

<DIR>...</DIR>

...

Келесі HTML кодында тізімге мысал келтірілген:

<html>

<H2> Unordered Lists –маркированный список </H2>

Bulleted LIST items

!—элементы маркированного списка—

LIST items are indented

!—элементы имеют отступ—

<H2>Ordered LISTS– нумерованный список </H2>

Numbered List items

```

<LI>—элементы нумерованного списка—</LI>
<LI>List items are indented </LI>
<LI>элементы имеют отступ </LI>
</OL>
<H2>Description Lists—список списанной </H2>
<DL>
<DT>First term </DT>
<DD> description of first term </DD>
<DD> !—Описания первого термина—</DD>
<DT>Second term </DT>
<DT>!—Второй термин—</DT>
<DD>Description of second term </DD>
<DD>— описание второго термина—</DD>
</DL>
</html>

```

HTML сізге арнайы символдарды қолдану мүмкіндігін береді. «Көбірек» (>), «азырақ» (<) және тырнақша (“) символдары HTML–дың форматты тегтерінде қолданылатындықтан, оларды экранға шығару үшін белгілі бір тізімде орналастыру керек. Ол тізімдерде амперсant белгісі (&) қолданылады.

Оны да экранға шығару үшін белгілі бір тізімде орналастыру керек. Арнайы символдардың тізімі келесі кестеде көрсетілген.

Резервтелген HTML символдары үшін арнайы символдар тізімі

<i>Тізімі</i>	<i>көрініс</i>	<i>мәні</i>
<	<	азырақ
>	>	көбірек
&	&	амперсant
"	“	Тырнақша

Осы HTML символдарын қолдану арқылы біз қарапайым Web бетті құра аламыз. Web бетті көркемірек ету үшін HTML –дың басқа да көптеген тегтерін үйрену қажет. Оларды қолдану арқылы біз Web–бетке графика және кестені кірістіре аламыз.

Web параққа кестені және графиканы қосу.

Web парақты көркемірек ету үшін оған графиканы және кестені енгізу керек. Мәтіндік біркелкілікті бұзатын графикалық элементтер әр–түрлі болуы мүмкін. Web графикасы қарапайым сызықтан күрделі суретке дейінгі диапазонды қамтиды. Графиканы қолданудың негізгі себебі–сіздің Web торабыңыздың қолданушыларына ақпаратты қызықтырақ әрі түсініктірек етіп көрсету.

Горизанталь сызық– графиканың ең оңай элементі. Оны Web бетке кірістіру үшін <HR> (Horizontal Rule, горизанталь сызық) тегі қолданылады. Оған мысал келесі кодта көрсетілген:

```

<HTML>
<H2>демонстрация горизантальной линии </H2>
<P>

```

Здесь HTML–текст, а ниже линия

<P>

<HR>

<P>

<I> для книги «Интернет Explorer 3 в подленнике» </I>

</HTML

Компьютерлік графиканы сајтайтын бірнеше форматтар бар. Бірақ WorldWideWeb–те сіз тек ғана екі форматты қолданасыз. Олар: GIF және JPEG. GIF форматы (Graphics Interchange Format–графикалық мәліметтерді алмасу форматы) алғашында кескіні бар файлдарды сақтау үшін стандарт ретінде CompuServe торабының қолданушыларына арнап құрастырылады. Қазіргі кездегі оның стандартты CIF 89а. CIF форматында сақталатын графика 256 түспен шектеледі. Сол үшін бұл форматта толық түсті суретті сақталмай ақ қайған жөн. CIF–ты белгі, эмблема және басқа да қолдан жасалған бейнелерге қолданған дұрыс. CIF стандартты Web бет авторлары үшін маңызды келесі эффектерді қолдайды:

– Кезекті көрсетілім (Interlacing). Бұл, көрші болмаған бейнелердің бірге сақталатындығын білдіреді. Браузер бұл бейнені оқуына байланысты, екінші көріністің шеті бейнелене береді. Бұл екінші бейне толық шықпай ақ, оның не туралы екенін түсінуге мүмкіндік береді.

– Тұнықтылық (Transparency) Тұнық CIF бейнесінде түстердің бірі тұнық етіп белгіленеді. Бұл құжат фонының одан өте алуына мүмкіндік береді. Сол үшін тұнық CIF–тер өте кең таралған. Ондай кесіндерді құру үшін жаңа програмалар да құрастырылуда. Мысалы, Photo CIF деп аталатын PhotoShop–тың модулі тұнық бетті де, кезекті CIF–ті де құруға мүмкіндік береді.

– Анимация. Анимацияланған CIF анимация құру үшін бір файлда сақталған суреттердің ретімен жасалалады.

CIF 89а–ны толық қолдайтын браузер анимация жасау үшін суреттерде кезегімен шығару керек екендігін білетін болады. Бұл программалар сондай–ақ анимация басталғанға дейінгі уақытты және оның қайталану ретін орнатуға мүмкіндік береді.

– JPEG. JPEG (Joint Photographis Experts Group)–кескін бойынша біріккен эксперттер тобы) толық түсті кескінді қолдап, оларды сығулы күйде сақтауға мүмкіндік беретін форматтар жиынтығы. Кең таралған браузерлер JPEG форматының кескіндерін көрсете алады, бірақ бұрын ол үшін арнай программалар қолданылған. Жақында шыққан прогрессивті JPEG (p–JPEG) форматы, кескінге кезектілік эффектін береді. Бірақ JPEG–де тұнықтылық мүмкін емес. Себебі берілген бейнелердің біраз өзгеруіне түрткі болады. JPEG–форматы толық түсті суреттер үшін таңдалынатын формат болса да, ол түстердің жылдам ауысатын кескіндеріне жарамайды.

Оның себебі түстердің тез ауысуы компрессияға байланысты пайда болатын математикалық манипуляцияға төзе алмайды. Ол үшін ол суреттерді GIF форматында сақталған жөн. Бейнелер HTML құжатының ішінде бейнеленсе де, сіз оларды бөлек файлдарда сақтауыңыз керек. Бейнені Web

бетке кірістіру үшін тегін пайдаланасыз. Оның ең қарапайым көрінісі келесі: . Мұндағы URL– графикалық бейнені сақтайтын файлдың адресі. SRC атрибуты міндетті болып, ал бейнесі бар файлдың URL–ын көрсетеді. Файлда әр–түрлі жерлерде орналасқандықтан, сіз алыстағы серверден де, жақын жердегі серверден де оны кірістіре аласыз.

 тегінің атрибуттары келесі:

 тегінің атрибуттары Атрибут анықтамасы

ALIGN= «орналасу» мәтінің бейнеге байланысты орналасуы (төбесінде (TOP), ортасында (MIDDLE) немесе астында (BOTTOM) болуы мүмкін).

ALT= «мәтін» бейненің орнына көрсетілуі керек мәтін .

ISMAP Карт–бейне құру үшін қолданылады WIDTH= «ені»

Бұл бейнені пиксель бойынша екіге текестіріледі.

HEIGHT= «биіктік» Бұл бейнені пиксель бойынша биіктігіне теңестіреді.

ALIGN атрибутын мысал келесі кодта келтірілген:

```
<HTML><HEAD> <TITLE> IMG –tag <TITLE> </HEAD>
```

```
<img src= «int.gif» align= «TOP»> выравнивание вверх
```

```
<img src= «int.gif» align= «MIDDLE»> выравнивание по центру
```

```
<img src=«int.gif»align=«BOTTOM»> выравнивание вниз (по умалчанию)
```

```
</HTML>
```

Сіздің қарапайым шертуерге көркем альтернатива құру үшін пернелер эффектін құру арқылы, графикалық бейнелерге гиперсілтемелер құруыңыз мүмкін. Бірақ мәтінді және <A> тэгтерінің арасына орналастырудың орнына сіз олардың арасына тегін орнатасыз. Осыған мысал:

```
<HTML>
```

```
<A HREF=http://portal.nienschanz.ru:800/wwwrus/general/profile.html>
```

```
<IMG SRC="nienl.gif"></A>
```

```
<HR>
```

Отсюда вы можете посетить страницу с профайлом фирмы Ниеншанц.

```
<P>
```

```
http://portal.nienschanz.ru.800/wwwrus/general/profile.html
```

```
</HTML>
```

HTML Web бетті көрмірек етіп құру үшін кесте құру мүмкіндігін береді. Кестені құруды бастау үшін <TABLE> тәгі қолданылады. Оны аяқтайтын тэг </TABLE>. Олардың арасында кесте құру үшін керекті тегтердің барлығы орналасады.<TABLE> тәгі BORDER–n түріндегі рамка атрибутына ие болуы мүмкін. Ол ені n–пиксель рамка құрады.<TR> және </TR> тэгтері кесте бағандарын құту үшін қажетті барлық тегтерді қамтиды.<TR> тәгі ALIGN және VALIGN атрибуттарына ие болуы мүмкін.ALIGN ұяшықтағы мәліметтің орналасуын реттейді. Ол LEFT (сол жақ бойынша туырлау), RIGHT(оң жақ бойынша туырлау) және CENTER(ортасына туырлау). мәндеріне ие болуы мүмкін.VALIGN ұяшықтағы мәліметтің вертикаль орналасуын реттейді. ОлTOP(жоғарғы жақ

бойынша туырлау), BOTTOM(төменгі жақ бойынша) және MIDDLE(ортасы бойынша) мәндерін қабылдауы мүмкін. Сіз кестені сызып болған соң, оны толтыруға кірісесіз. Егер ұяшық кестенің мәліметін сақтаса, сіз оны <TD> және </TD> тэгтерімен көрсетесіз. <TD> және </TD> тэгтерінің аралығында орналасқан ұяшықта жазылады. Тақырып жазу үшін сіз <TH> және </TH> тэгтерін қолданасыз.

2.2 PHP тілінің негіздері

PHP тарихы 1995 жылы тәуелсіз Расмус Лердорф (Rasmus Lerdorf) атты программист–келісуші оның онлайндық резюмеcін оқыған сайтқа кіруші адамдар санын есептеуге арналған Perl/CGI сценариін жазуынан басталады. Оның сценариін екі есепті шешуге бағытталған: келушінің деректерін тіркеу және web–бетіне келушілердің санын шығару. WWW дамуы әзір тек басталғанда мұндай есептерді шешу үшін арнайы құралдар болған жоқ, сонда авторды халық сұрақтармен жауып тастады. Лердорф Personal Home Page (PHP) немесе Hypertext Processor (гипермәтінді үрдіс) атты өзінің аспабын тегін тарата бастады.

PHP–дің сәтті жарыққа шығуы Лердорфты PHP кеңейтілуін жобалауға итермеледі. Кеңейтілудің бірі HTML формасында енгізілген мәліметтерді символдық айнымалыларға түрлендіреді, ол оларды басқа да жүйелерге экспорттауға мүмкіндік береді. Берілген мақсатқа қол жеткізу үшін Лердорф келешектегі жобаларда Perl–дан C–ге өтуге шешім қабылдады. PHP –дегі аспаптар жиынын кеңейтілуі PHP 2.0 – нің тууына себепші болды, немесе 2.0, или PHP–FI (Personal Home Page — Form Interpretator). 2.0 нұсқасын дамытуда бүкіл дүниежүзінің программисттері ат салысқан.

PHP–дің жаңа нұсқасы танымал атаққа ие болды, және жақын арада негізгі жобалаушылар командасы құрылды. Олар программалық кодты HTML–ге ендірудің түпнұсқалық концепсиясын сақтай отырып, лексикалық анализ механизмін қайта жазды, осыдан келе PHP 3.0 жарыққа шықты. 3.0 нұсқасының шығуына жуыр 1997 жылы 50 000 астам қолданушы PHP–дің өздерінің web–беттерін жақсатуға қолданылған еді.

1997 жылы PHP–дің қысқартылуы «Personal Home page» деген мағынаны емес, «PHP Hypertext Processor» ұғымын білдіруі керектігі шешілді.

MySQL – дерекқордың көп ағымды ықшам сервері. MySQL үлкен жылдамдық, тұрақтылық және қодланыста жеңлдігімен сиптталады.

MySQL кіші және орта қосышаларды ең ықтимал шешуі болып табылады. Сервердің бастапқы мәтіні платформалардың көбісінде компиляцияланады.

PHP тілінің негіздері – Кодта барлық командалар үтірлі нүктемен (;) аяқталуы тиіс. Сіз үтірлі нүктемен аяқтай отырып бірнеше команданы бір жолда орналастыра аласыз. Бірақ программа кодының анықтылығы үшін оны істеу қажет емес. Сонымен қатар PHP–де басқарушы элементтер

(шартты операторлар, циклдер және т.б.) бар, олардан кейін үтірлі нүкте коюдың қажеті жоқ.

Программа кодына түсініктемелер енгізуге болады, олар HTML браузеріндегідей PHP–интерпретатормен сияқты қабылданбайды. Түсініктемелерді қолдану программистке сол мезетте не ойластырғанын еске түсіруге көмектеседі. Біржолдық түсініктеме екі тік слэш белгісімен (//) немесе «решетка» белгісімен (#), ал көпжолды ашылатын және жабылатын «жақшалармен» (/*және*/) жүзеге асырылады.

Айнымалылар – PHP–де айнымалылар доллар белгісінен (\$) басталады. Айнымалылар аты латын әріптерінен, сандардан символдық астыңғы сызықтардан(_) тұруы мүмкін. Айнымалылар аты әріптер регистріне сезімтал, демек \$a мен \$A – бұл әртүрлі айнымалылар. Айнымалыларды жариялағанда әдетте басқа тілдердегідей белгілі бір тип көрсетілмейді. Айнымалы типі (демек, ол сақтайтын мәндер типі) қолдану контексті бойынша анықталады.

PHP–де келесі айнымалылар типі болады:

- integer
- floating point
- string
- object
- array

«Бос» айнымалы VAR кілттік сөзімен жарияланады:

```
VAR $some_var;
```

Сонымен қатар айнымалы бірінші рет қолданғанда жариялануы мүмкін:

```
$some_var="value";
```

Integer типті айнымалылар құрамында –2 биллионнан +2 биллионға дейінгі ондық, сегіздік және оналтылық санау жүйесіндегі сандар бола алады.

Мысалы:

```
$var1=100;      /* Ондық мән */  
$var2=0144;    /* Сегіздік мән */  
$var3=0x64;    /* Оналтылық мән */
```

Floating point типті айнымалалар – бұл жылжымалы нүктелі (бөлшек) сан. Мысалы:

```
$var1=2.34;  
$var2=234e1;
```

Жол(string типті айнымалы) – бұл әріптер, сандар және арнайы символдар комбинациясы. Ол бірлік (апострофтар) немесе қос тырнақшамен беріледі. Егер жол қос тырнақшамен берілсе, онда ол құрамында айнымалылардың бар–жоқтығына тексеріледі.

Жолдық айнымалыларға мысал:

```
$var="23";  
$str1='Айнымалы құрамында $var мәні бар';  
$str2="Айнымалы құрамында $var мәні бар";
```

Біздің айнымалыларымызды шығаруда сәйкесінше аламыз:

Айнымалы құрамында \$var мәні бар

Айнымалы құрамында 23 мәні бар

Мүмкін, сізге жолға арнайы символдар енгізу керек болар. Мысалы, сіз кері слэшті (\) қолдана отырып қос тырнақшаны (“) шығара аласыз. Егер

```
$str1=" "Графика" фирмасы ";
```

деп жазсақ онда қате тудырады, бірақ егер кері слэшті (\) қолдансақ

```
$str1=" \"Графика\" фирмасы ";
```

онда қате болмайды.

2.2.1 кестеде жолдық мәндерде қолданылатын Escape–тізбектері (кері слэш және арнайы символдар) сипатталған.

2.2.1 кесте – PHP тілінде қолданылатын Escape–тізбектері

Escape–тізбектері	Мәндері
\n	Жаңа жол басы
\r	Каретка ауыстыру
\t	Табуляция символы
\\	(\) символы
\"	(") символы
\\$	(\$) символы
\0	Сегіздік мән
\x	Оналтылық мән

Объекттер (object типті айнымалы) класс экземпляры. Объектті құру үшін ең алдымен классты құру керек. Класс олармен жұмыс істеуге арналған айнымалылар мен функциялардан тұрады.

array типті айнымалылар массивтерді жариялағанда қолданылады. PHP–де массивтің екі түрі болады:

– бүтінсанды индекстері бар массив

– индекстелген жолдары (хэш) бар массив

Мысалы, төрт мәннен тұратын массивті құру үшін келесіні жазу керек:

```
$myarr=array("value1", "value2", "value3", "value4");
```

Массивке орналастырылған әрбір элементке индекс тіркеледі (0–ден бастап). Осылайша, \$myarr[0] бізге “value1”–ді, ал \$myarr[3] “value4”–ті береді, Әрине, массивке \$myarr[]=”value5”; элементін қосуға болады.

Хэшти келесідегідей құруға болады:

```
$myhash=array('size'=>'large', 'style'=>'italic', 'family'=>'Arial');
```

Операциялар – PHP операцияның бірнеше түрін қолдайды:

– меншіктеу;

– арифметикалық операциялар;

– логикалық операциялар;

– конкатенция;

– салыстыру.

Арифметикалық операциялар – 2.2.2 кестеде арифметикалық операциялар қысқа түрде келтірілген

2.2.2 кесте – Арифметикалық операциялар

Опера тор	Опера цияға мысал	Сипаттамасы
+	$\$a+\b	$\$a$ мен $\$b$ айнымалыларының қосындысын есептеу
–	$\$a+-\b	$\$a$ мен $\$b$ айнымалыларының айырмасын есептеу
*	$\$a*\b	$\$a$ мен $\$b$ айнымалыларының көбейтіндісін есептеу
/	$\$a+/\b	$\$a$ мен $\$b$ айнымалыларының бөліндісін есептеу
%	$\$a\%\b	$\$a$ айнымалысын $\$b$ айнымалысына модуль бойынша бөліндінің қалдығын есептеу

Логикалық операциялар (2.2.3 кесте) белгілі бір критериялар бойынша анықтайды, қайтарылатын мән ақиқат (true) не жалған (false) екендігін.

2.2.3 кесте – Логикалық операциялар

Опер атор	Опера цияға мысал	Сипаттамасы
and не &&	$\$a$ and $\$b$ $\$a$ && $\$b$	Егер екі айнымалының да мәндері ақиқат болса, true мәнін қайтарады
or не 	$\$a$ or $\$b$ $\$a$ $\$b$	Егер ең болмағанда бір айнымалының мәні ақиқат болса, true мәнін қайтарады
not не !	not $\$a$! $\$a$	Логикалық терістеу, айнымалы мәнін иверттеу

Салыстыру операциясы екі айнымалы не мәндер арасындағы қатынасты анықтайды және true немесе false мәндерін қайтарады. Салыстыру операторлары 2.2 .4 кестеде келтірілген.

2.2 .4 кесте – Салыстыру операторлары

Оператор	Сипаттамасы
==	Тең
!=	Тең емес

>	Артық
<	Кем
Оператор	Сипаттамасы
>=	Артық не тең
<=	Кем не тең

Басқару құрылымдары – PHP программа жұмысының бірнеше басқару құрылымын қолдайды. Оларға if, for, while, swith және т.б. жатады, if/ elseif

Берілген құрылым программалық логиканы жүзеге асыруға мүмкіндік береді. Сіз түрлі өрнектерді тексере аласыз және тексеру нәтижесіне байланысты сол не басқа әрекеттерді орындай аласыз. Базалық синтаксисті келесідегідей сипаттауға болады:

```
<?php
if (өрнек1)
{
әрекет1;
}
elseif (өрнек2)
{
әрекет2;
}
else
{
үнсiздiк бойынша әрекет;
}
?>
```

әрекет1 мен әрекет2 тек if шартында өрнек1 немесе elseif шартында өрнек2 ақиқат болғанда ғана орындалады. Егер де сипатталған барлық өрнектер жалған болған жағдайда үнсiздiк бойынша әрекет орындалады.

for мен foreach

for цикл операторы берілген код блогын берілген рет (итерация) орындайды.

```
<?php
for (өрнек1; өрнек2; өрнек3)
{
әрекет;
}
?>
```

Параметрлер өрнек1 – қайталау санағышының бастапқы мәні; өрнек2 – сәйкесінше FOR циклінің орындалуы тоқтатылатын критерий сипаттамасы (шарт), өрнек3 – санағыш мәнінің өзгертілуі (көбею не азаю).

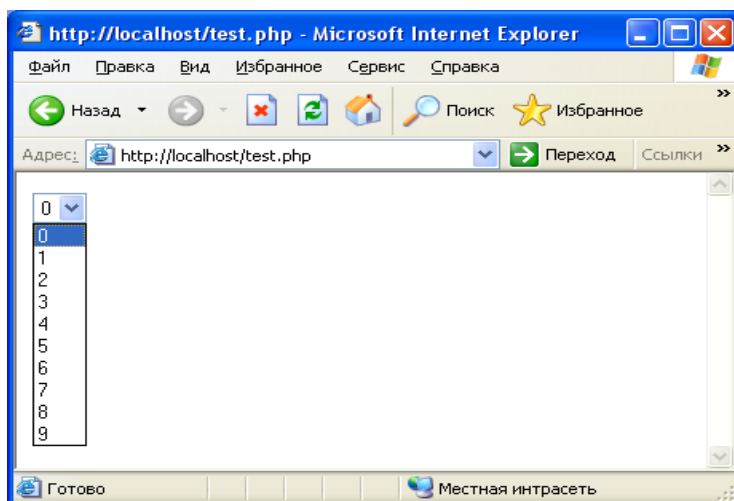
Нәтиже 1.4 суретте келтірілген.

FOREACH құрылымы сандық массив немесе хэштің элементтерін таңдап шығу үшін арналған.

```
<?php  
foreach ($array as $value)  
{ PHP-код }  
?>
```

Хэш болған жағдайда синтаксис келесідегідей болады:

```
<?php foreach ($hash as $key=>$value) { PHP-код } ?>
```



2.2 .1 – сурет. PHP-дегі программа мысалы жұмысының нәтижесі

While -while цикл операторы белгілі бір шарт орындалғанша код блогын қайталай береді. Шарт жалған болған кезде цикл орындалуы тоқтатылады.

Switch-Функциялар – тілдің дәстүрлі конструкциясы. Олар түрлі параметрлерді қолдана отырып, және өзінің жұмысының нәтижесін функцияны шақыру нүктесіне қайтаратын белгілі бір әрекеттер жиынын орындайды. PHP тілі функцияның екі түрін қолдайды:

- программистпен жарияланған функциялар (қолданушылық);
- тілдің стандартты (ішкі) функциялары.

Қолданушы функциялары – Егер кейде командалар сериясын орындау қажет болса, онда сіз бұл командаларды функцияның ішіне орналастыра аласыз. Бұл сіздің кодыңызды ыңғайлы, да оқуға оңай қылады.

Ылғи бір командалар сериясын қайта тергеннің орнына функцияны (алдын-ала жарияланған) шақыра саласыз. Бұл әлдеқайда қысқарақ. Бұдан басқа, егер де өзге өзгертулерді енгізу қажет болса, жарияланған функцияны бір рет түзегеумен шектелуге болады.

PHP-кодта функцияны құру үшін келесі синтаксисті қолданады:

```
function функция_аты(функция_параметрі);  
{ әрекеттер; }
```

Параметрлер: функция_аты – функция аты программистпен тағайындалады; функция_параметрі – функциямен (осы параметрлердің мәндеріне байланысты функция жұмысының нәтижесі өзгеріп отырады)

колданылатын параметрлер; әрекеттер – функция денесі (ол орындауға тиіс командалар жиыны).

Ішкі функциялар – Бұл тілді жобалаушылармен оны құрғандағы жарияланған функциялар, олар жиі колданылатын операциялар. Ішкі функцияларды қолдану үшін сізге бұл функцияның атын және параметрлер тізімін білуге қажет. Функцияға қатынау (шақыру) оның аты бойынша жүзеге асырылады.

2.3 MySQL дерек қоры негіздері

Деректер қоры – бұл құрамында белгілі бір ақпараты бар екіөлшемді өзара байланысқан кестелер жиыны. Деректер қорын басқаратын (жаңа деректер қорын, кестелерді құрады және де құрылған объектілерді қолдануға мүмкіндік береді) программалық қамтама деректер қорын басқару жүйесі (ДҚБЖ) деп аталады. Клиенттердің сұрауларын сипаттау үшін тұтас тіл құрылған – ол SQL (Structured Query Language – құрылымдық сұраулар тілі). SQL сұраулары арқылы сіз келесі әрекеттерді орындай аласыз:

- Деректер қоры мен кестелерді құруға.
- Кестелерге ақпаратты қосуға.
- Ақпаратты өшіруге.
- Ақпаратты модификациялауға.
- Қажет ақпаратты алуға.

Әрине, admin қолданушысы бар болудан басқа сәйкес құқықтарға ие болу керек. MySQL әрбір сұрауы үтірлі нүктемен аяқталуы тиіс.

MySQL — бұл ылғи да компьютерде жұмыс істеп тұратын сервер–программа. Клиенттік программалар (мысалға, сценари) оған арнайы сұрауларды сокет (демек, желілік құралдар арқылы) механизмі арқылы жіберіп отырады, ол оларды өңдеп нәтижесін сақтайды. Осыдан кейін, қайтадан клиенттің арнайы сұрауы бойынша бүкіл нәтиже немесе оның бір бөлігі қайта жіберілед. Неге ылғи тұтас нәтиже жіберілмейді? Өте карапайым: мәселе мәліметтер жиынының нәтижесі өте үлкен болып, оны желі бойынша тасымалдауға көп уақыт кетіп.

Сокеттерді қолдану механизмі клиент–серверлік технологиясына ойластырылған, бұл жүйеде программалардан арнайы сұрауларды қабылдап және өңдейтін арнайы программа – MySQL – сервері қосылуы керек деген сөз.

Деректер қорымен байланыс құру – Бірақ деректер қорымен жұмыс бастамас бұрын онымен желілік байланыс құру, және де қолданушы авторизациясын орындау қажет. Ол үшін арнайы функция бар `mysql_connect()`.

```
int mysql_connect([string $hostname] [,string $username]
[,string $password])
```

`mysql_connect()` функциясы MySQL деректер қорымен желілік байланысты құрады устанавливает сетевое соединение с базой данных

MySQL, ол \$hostname (по умолчанию это localhost, т.е. текущий компьютер) атты хостта орналасқан, және ашық байланыстың идентификаторын қайтарады. Келесі жұмыстардың барлығы дәл осы идентификатормен жүргізіледі. Тіркелгенде қолданушы \$username аты және \$password құпиясөзі (үнсiздік бойынша берілген үрдісті қосқан қолданушы аты мен бос құпиясөз) көрсетіледі. Сонымен қатар \$hostname жолында хост_аты:порт (егер MySQL стандартты емес басқа портқа қосылған жағдайда) форматты порт аты болуы мүмкін. MySQL-серверімен байланыс сценари жұмысының бітісімен не болмаса mysql_close() функциясы шақырылғанда автоматты түрде жабылады. Егер сіз сценаридің бүкіл жұмысы барысында тек бір ғана деректер қорымен байланыс құруды жоспарласаңыз, онда сіз қайтарылған мәндерді сақтамасаңыз да болады, және қалған функцияларды шақырғанда байланыс идентификаторын көрсетпесеңіз болады.

```
int mysql_select_db(string $dbname [,int $link_identifier])
```

MySQL серверіне алғашқы сұрауды жібермес бұрын қандай деректер қорымен жұмыс істейтінімізді көрсетуіміз керек. Осы үшін сипатталған функция қажет. Ол PHP-ді \$link_identifier байланысының келесі операцияларында \$dbname деректер қоры қолданылатынын ескертеді.

Қателерді өңдеу – Егер MySQL-мен жұмыс барысында қателер туса (мысалы, сұрауда жақшалар дұрыс қойылмаса немесе параметрлер жетіспесе), онда қате туралы хабарлама және оның нөмірін келесі екі функция арқылы алуға болады.

```
int mysql_errno([int $link_identifier])
```

Функция соңғы тіркелген қатенің нөмірін қайтарады. \$link_identifier байланыс идентификаторын көрсетпесе болады, егер де сценари жұмысы барысында тек бір байланыс құрылған жағдайда.

```
string mysql_error([int $link_identifier])
```

Бұл функция қате нөмірі емес, құрамында қате хабарламасы бар жолды қайтарады. Оны қателерді жөндеген кезде қолданған ыңғайлы.

Деректер қорына сұрауларды орындау – Ол үшін арнайы жалғыз mysql_query() — функциясы деректер жиынын нәтижелейтін идентификаторды қайтарады.

Есте болсын, біз нәтиже бірден клиентке жіберілмейтінін айтқан болатынбыз? Міне, чтобы оған жету үшін идентификатор керек. Оны параметр ретінде қабылдайтын және сол не басқа деректерді қайтаратын өте көп функциялар бар. Олар туралы кейінірек сөз қозғаймыз.

```
int mysql_query(string $query [,int $link_identifier])
```

Бұл функциясы өзінің қолданысында әмбебап: ол MySQL-серверге з \$query сұрауын жібереді және жауаптың немесе нәтиженің идентификаторын қайтарады. \$query параметрі SQL тілі ережесі бойынша құрылған жол. Алдындағы орнатылған байланыс қолданылады \$link_identifier, ал ол болмаған жағдайда – соңғы ашық байланыс.

Олар сәтті орындалған не орындалмағаны туралы тек белгіні қайтаратын бірнеше SQL командалары (мысалы, ол UPDATE, INSERT және т. б. командалар) бар. Бұл жағдайда белгі функциямен қайтарылады.

Керісінше, SELECT сұрауы үшін шығарылым идентификаторы қайтарылады, егер нөл болса, қате болғандығын білдіреді.

Шын мәнінде сұрауды орындауға арналған тағы бір функция бар, бірақ ол ондай қолдануға ыңғайлы емес. Себебі әр сұрау сайын қатынауға қажет деректер қорының атын көрсету керек.

`int mysql(string $dbname, string $query [,int $link_identifier])`

Қызметі `mysql_query()` сияқты, тек қатныау таңдалған деректер қорына емес, `$dbname` параметріне көрсетілгенге орындалады. Егер сізде бірнеше деректер қоры бар бола бір уақытта қатынасаңыз, бұл функцияны пайдалану сізге тиімді болар. Әдеттегідей, `$link_identifier` параметрін түсіруге болады, онда соңғы ашық байланыс қолданылатын болады.

Кесте құру – `create table` КестеАты (ӨрісАты тип, ӨрісАты тип, ...)

Бұл командамен деректер қорында өздерінің аттарымен (ӨрісАты) және көрсетілген типтерімен анықталатын бағандары (өрістермен) бар жаңа кесте құрылады,.

Өріс типтері :

Бүтін сандар – Бүтін сандардың бірнеше типтері болады, олар сақтауға арналған деректер қорында бөлінентін деректердің байт сндарымен ерекшеленеді. Бұл типтердің барлығында айырмашылық тек аттарында (кейбір қысқартулармен), жазылуы келесідегідей:

префикс `INT [UNSIGNED]`

Міндетті емес `UNSIGNED` жалауы таңбасыз (0–ден үлкен не тең) сандарды сақтайтын өрісті құратынын білдіреді. Типтер аттары жалпы түрде префикс `INT` сияқты 5 кестеде келтірілген.

2.3.1 кесте – Бүтінсанды мәндер типтері

Тип	Сипаттамасы
TINYINT	–128 ден +127 дейінгі сандарды сақтай алады
SMALLINT	–32 768 ден 32 767 дейінгі диапазон
MEDIUMINT	–8 388 608 ден 8 388 607 дейінгі диапазон
INT	–2 147 483 648 ден 2 147 483 647 дейінгі диапазон
BIGINT	–9 223 372 036 854 775 808 ден –9 223 372 036 854 775 807 деінгі диапазон

Бөлшек сандар – Дәл бүтін сандар сияқты MySQL–де бірнеше түрге бөлінеді. Жалпы түрде былай жазылады:

ТипАты[(length,decimals)] [UNSIGNED]

Мұнда `length` — PHP-ге берілгенде бөлшек сандар орналасатын белгіорын саны (өріс ені), ал `decimals` — ескерілетін ондық үтірінен кейінгі белгілер (сандар). Әдеттегідей, `UNSIGNED` таңбасыз сандарды белгілейді.

2.3.2 кесте – Рационалды мәндер типтері

Тип	Сипаттамасы
FLOAT	Үлкен емес дәлдікті жылжымалы нүктелі сан
Тип	Сипаттамасы
DOUBLE	Екілік дәлдікті жылжымалы нүктелі сан
REAL	DOUBLE үшін синоним
DECIMAL	Жол түрінде сақталатын бөлшек сан
L	
NUMERIC	DECIMAL үшін синоним
C	

Жолдар – дегеніміз символдар массиві. Әдетте `SELECT` сұрауы бойынша мәтіндік өрістен іздегенде символдар ергистрі қарастырылмайды, демек «қарындаш» пен «ҚАРЫНДАШ» бір болып саналады. Бұдан басқа, егер мәтінді енгізгенде немесе өзгерткенде дерекқор оны автоматты түрде қайтакодтауға бапталған болса, онда бұл өрістер сіз көрсеткен кодировкада сақталады.

Ең алдымен `length` символдарынан аспайтын жол типімен танысайық, мұндағы `length` 1 ден 255 дейінгі диапазонда жатады.

`VARCHAR(length) [BINARY]`

Осындай типті жолға өзгеріс енгізгенде одан ақырғы пробелдер қиып алынады. (`rtrim()` функциясының шақырылуындай). Егер `BINARY` жалауы көрсетілген болса, онда `SELECT` сұрауында жол регистр есебімен салыстырылады. `VARCHAR` типі 255 символдан аспайтындығымен ыңғайсыз. Оның орнына 6 кестеде келтірілген басқа мәтіндік типтерді қолдануға болады.

2.3.3 кесте – Жолдық мәндер типтері

Тип	Сипаттамасы
TINYTEXT	Максимум 255 символ сақтай алады
TEXT	Максимум 65 535 символ сақтай алады
MEDIUMTEXT	Максимум 16 777 275 символ сақтай алады

LONGTE XT	4 294 967 295 символ сақтай алады
--------------	-----------------------------------

Бинарлы мәліметтер — бұл TEXT форматындағыға ұқсас мәліметтер, бірақ іздеу жүргізгенде символдар регистрі ("abc" мен "ABC" — әртүрлі жолдар). Барлығы бинарлы мәліметтердің 4 типі болады (7 кесте).

2.3.4 кесте – Бинарлы мәндер типтері

Тип	Сипаттамасы
TINYBLOB	Максимум 255 символ сақтай алады
BLOB	Максимум 65 535 символ сақтай алады
MEDIUMBLOB	Максимум 16 777 275 символ сақтай алады
LONGBLOB	4 294 967 295 символ сақтай алады

Уақыт пен дата – MySQL уақыт пен датаны әртүрлі форматта сақтауға арнайы бейімделген бірнеше жол типін қолдайды (8 кесте).

2.3.5 кесте – Уақыт пен датаның дерекқорда көрсетілуі

Тип	Сипаттамасы
DATE	YY–MM–DD форматындағы дата
TIME	HH:MM:SS форматындағы уақыт
DATETIME	YY–MM–DD HH:MM:SS форматындағы уақыт пен дата
TIMESTAMP	timestamp форматындағы уақыт пен дата

Модификаторлар мен типтер жалаулары – Типке модификаторды қосуға болады, олар «тәртіп» пен сәйкес бағандармен орындауға қажет операцияларды тағайындайды. Олардың ең көп тарағандары 9 кестеде келтірілген.

2.3.6 кесте – Негізгі модификаторлар

Модификатор	Сипаттамасы
-------------	-------------

not null	өріс анықталмаған мәнді қабылдамайтынын білдіреді
primary_key	өріс басты кілт екенін білдіреді, яғни жүгінуге болатын жазба идентификаторы екендігі
auto increment	жаңа жазба қойғанда өріс уникалды мәнге иеленеді, сол себепті кестеде екі бірдей нөмірлі өріс болмайды
DEFAULT	қолданылатын өріс үшін мәнді үнсіздік бойынша орнатады

Кестені өшіру – drop table КестеАты

КестеАты кестесін өшіреді. Кесте міндетті түрде бос болудың қажеті жоқ, сондықтан байқаусыздан құрамында ақпараты бар кестені өшіріп қоймаңыз.

Жазбаны қою – insert into КестеАты(ӨрісАты1 ӨрісАты2 ...) values('мән1','мән2',...)

КестеАты кестесіне ӨрісАтыN деп белгіленген өрісті қосады, мұнда мәнN мәндеріне сйкес келеді. Бұл командада аталып өтпеген өрістер «анықталмаған» (анықталмаған мән – бұл бос жол емес, берілген жолда мәндер жоқ екендігі) мәндеріне ие болады. Қайта көрсетілмеген өрісте кестені құрғанда not null берілген жағдайда бұл команда сәтсіз аяқталады. Өрістер мәндерін кәдімгі жақшаға алуға болады, бірақ мұнда апострофтарды қлджанған ыңғайлырақ сияқты. Кестеге бинарлы деректерді енгізгенде (немесе құрамында слэш пен апострофтары бар) кейбір символдар кері слэштермен «қорғалуы» тиіс, демек \, ' символдарыменжәне нөлдік кодты символдармен.

Жазбаны өшіру – delete from КестеАты where Өрнек

КестеАты кесесінен Өрнек орындалған барлық жазбаларды өшіреді. Өрнек параметрі — бұл РНР ережелері бойынша құрастырылған жай ғана логикалық өрнек. Мысалға:

(id<10) and (name regexp 'a*b') and (age=25)

Өрнекте өріс аттары, константалар және операторлардан басқа қарапайым «есептелетін» бөліктер де кездесуі мүмкін, мысалы: (id<10+11*234).

Жалпы айтқанда, өрнек форматы келешекте кездесетін сұраулардың барлық командалары үшін жалпыға бірдей. Мысалы, ол select операциясында да, update операциясында да қолданылады.

Жазбаларды іздеу – select * from Кесте where Өрнек [order by ӨрісАты [desc]]

Бұл команда — негізгі және өте мықты. Ол Өрнек өрнегін қанағаттандыратын барлық жазбаларды іздейді. Оның мүмкіндіктері әлдеқайда молырақ, бірақ оған біз тоқталмаймыз. Егер жазбалар бірнешеу болса, онда order by ұсынысы бойынша олар бұл кілттік сөздің оғ жағында жазылатын аты бойынша іріктеледі (егер desc сипаттаушысы жазылса, онда іріктеу кері ретпен орындалады). Сонымен қатар order by ұсынысында бірнеше өріс берілуі мүмкін. * символының рөлі маңызды. Ол таңдалынған жазбалардан барлық өрістерді көрсету керектігін білдіреді. Басқа жағынан, жұлдызша орнына үтір қойып, көрсетуге қажетті өрістің аттарын жазып шығуға. Бірақ * өзі жиірек қолданылады.

Жазбаларды жаңарту—updateКесте set(ӨрісАты1='мән1', ӨрісАты1='мән2', ...) where Өрнек

Өрнек өрнектерін қанағаттандыратын Кесте кестесінде барлық жазбалар үшін көрсетілген өрістер сәйкес мәндерде орнатылады. Бұл команда егер бір жазбаның барлық өрістерін емес, тек кейбіреулерін ғана жаңарту қажет болған жағдайда қолданылады.

Өрнекті қанағаттандыратын жазбалар санын алу — MySQL—дің тағы бір жиі қолданылатын мүмкіндігін — Өрнекті қанағаттандыратын жазбалар санын алу қарастырған жөн. Жалпы айтқанда, бұны жасаудың бірнеше тәсілі бар. Соның бірі мынау:

```
select count(if(Өрнек,1,NULL)) from Кесте
```

Бұл мысалдың өзі MySQL—дің тілі сипатталғанға қарағанда қаншалықты бай екенін көрсетеді...

Бағанның бірегей мәндерін алу — Деректерқорды қолданғанда, берілген кестедегі бағанда қандай бірегей мәндері бар екендігін білген ыңғайлы болады. Мысалы, егер кез—келген бір статистикалық кестеде әрбір жазбада, оның құрамында адамдар жайында мәлімет болса, және онда Country (ел) өрісі бар болса, онда кестеде 30 жасқа дейін өмір сүріп келген адамдардың барлығы қай елде тұратынын табу үшін келесі сұрауды орындаймыз:

```
select distinct ӨрісАты from Кесте where Өрнек
```

Біздің жағдайда ӨрісАты=Country, ал Өрнек — age>=30. Бұл сұрау ізделіп отырған алдерден тұратын бір бағаннан тұратын нәтижені тудырып береді.

Нәтижені алу — бұл жай ғана деректер жиыны, және оған кіретін жазбалар санын mysql_num_rows() арқылы білуге болады. Мысалы, егер алдыңғы мысалда кестеден таңдау кезінде 30 жасқан 10 адам бар болса, онда бізнәтиже идентификаторында 10 «жолға» «сілтеме» аламыз.

Әрбір жазба — бұл өрістер мәндерінің тізімі, яғни, select ... from Кесте сұрауында көпнүкте орнына (жұлдызша болса, барлық өрістер) көрсетілген реті бойынша. Осылайша, нәтиже — бұл өзіндік екіөлшемді массив: бірінші индекс — жазба нөмірі және екінші — өріс аты.

```
Нәтиже параметрлері — int mysql_num_rows(int $result)
```

mysql_num_rows() функциясы сұрау нәтижесінде жазбалар санын қайтарады. Осылайша, функция «екіөлшемді массив нәтижесінің» вертикальды өлшемін анықтауға мүмкіндік береді.

```
int mysql_num_fields(int $result)
```

Бұл функция нәтиженің бір жолына өрістер санын қайтарады, яғни, число \$result нәтижесінде бағандарды. Сонымен қатар «екіөлшемді массив нәтижесінің» горизонтальды өлшемін анықтауға мүмкіндік береді.

```
Нәтиже өрісін алу – int mysql_result(int $result, int $row, mixed $field)
```

Функция \$field өрісінің нәтижесін \$row нөмірлі жолға қайтарады. \$field параметрі тек өріс атын беріп қоймай, сонымен қатар оның нөмірін — позициясын, кесте құрғандағы «тұрған» бағанда. Дегенмен, мүмкіндігінше барлық жерде өрістің атының өзін қолдануға тырысу керек.

Функция әмбебапты: оның көмегімен бүкіл нәтижені бір-бір ұяшық бойынша «өтіп» шығуға болады. Мұны істеуге тиым салынбағанмен, қолданбауға тырысу керек, өйткені mysql_result() біршама баяу жұмыс істейді.

2.4 CSS бағдарламалау тілінің ерекшелігері

Cascading Style Sheets(CSS) – каскадты кестелер стилі. Кестелер стилі – беттің дизайн детальдарының оның құрылымы мен мазмұнынан бөліп көрсету. Классикалық түрде HTML құрылымы мен дизайны аралас болды: мәтіндік абзацтың жанында оның түсі, шрифт өлшемі, шрифт типі және т.б. Егер сайтта шрифт өлшемі сияқты қарапайым параметрді өзгерту керек болса, онда барлық беттерге жазып шығу керек болады. CSS сипаттамасы бөлек бір файлда сақталатын барлық объекттерге стильді басқаруға көмектеседі. CSS қолдана отырып стильдер сипаттамасы бар файлды түзету арқылы барлық беттерде шрифт өлшемін өзгертуге болады. Стильдер әр түрлі болған сайын дизайнды да жақсырақ өзгертуге болады. Ең бастысы серверде жатқан дайын HTML құжаттарды ауыстырмауға болады. Қолданушы браузері файлға стильдермен (сілтеме бойынша) жүгінеді де сәйкес қалыпты береді.

CSS тарихы. 1996 ж. желтоқсанында W3C Консорциумымен стандарт ретінде қабылданған бірінші деңгейдегі CSS алға жылжыған үлкен қадам болды, себебі WEB-беттердің мазмұнын (мәтін, сурет және т.б..) оның пішімделуінен (макет және мәтін сипаттамасы, мысалы, шрифттар туралы ақпарат) айыруға көмектесті. Бұдан соң HTML тілі қайтадан функционалды-бағытталған болды (формаға бағытталған емес), бірақ қолданушыға бетті басқаруға кедергі келтірмеді[2].

1998 ж. мамырында жоғары деңгейде өңдеушіге WEB-беттерді басқаруды жүзеге асыратын CSS2 стандарты қабылданды. Бұл стандарт бірінші деңгейдегі CSS негізделген және құрамында WEB-беттердің объекттері мен элементтерін тура орналастыру, сонымен қатар арнайы бағдарламалық қамтамасыздандыруға WEB-беттердің мазмұнын оқуға мүмкіндік беретін жаңа функцияларды қосты[2].

CSS құру мақсаты. CSS веб-беттерді құрушылармен түсі, шрифті, бөлек блогтардың орналасуы мен веб-беттердің сыртқы түрінің басқа аспектілерін көрсету үшін қолданылады. CSS өңдеудің мақсаты веб-беттердің логикалық құрылымының негізгі сипаттамасын осы веб-беттің сыртқы түрінің сипаттамасынан (CSS тілі көмегімен) бөлу болып табылады (HTML немесе басқа да белгілеу тілдері көмегімен). Мұндай бөлу құжаттың рұқсатын ұлғайтуға, оның көрсетілуімен басқару мүмкіндігі, сонымен қатар мазмұн құрылымында қиыншылық пен қайталануды төмендетеді. Одан басқа CSS бір құжатты әр түрлі стильдер немес экрандық көрсету, баспа көрсетілімі, дауыспен оқу (арнайы дауыстық браузермен немесе экраннан оқу бағдарламасы) сияқты шығару әдістерінде, немесе Брайль шрифтінің қолданатын құрылғылармен шығару кезінде көрсетуге мүмкіндік береді.

CSS құру ережелері. CSS кестесінің құжатқа қосылуы кезіндегі стильдер кестесіндегі әр CSS ережесінде екі негізгі бөлік бар — селектор және жариялау блогы. Ереженің сол жағында орналасқан селектор ереженің құжаттың қай бөлігіне таралатынын анықтайды. Жариялау блогы ереженің оң жағында орналасады. Ол фигуралық жақшаларға алынады және өз кезегінде «;» белгісімен бөлінген көптеген жариялаулардан тұрады. Әр жариялау «:» белгісімен бөлінген CSS қасиеті мен мәнінің тіркесінен тұрады. Селекторлар бір жолда үтір арқылы топтасуы мүмкін. Ондай жағдайда қасиет әр қайсысына қолданылады.

```
селектор, селектор {  
    қасиет: мәні;  
    қасиет: мәні;  
    қасиет: мәні;  
}
```

Басқа бір жағдайда CSS кестесінің құжатқа қосылуы кезіндегі CSS ережесі «;» белгісімен бөлінген жариялаулар тізімінен тұрады.

Селектор түрлері

CSS ережесінің селекторлары болуы мүмкін:

Элементтердің селекторлары;

– p {font-family: Garamond, serif;}.

Класстардың селекторлары;

– .note {color: red;background: yellow;font-weight: bold;}.

Идентификаторлардың селекторлары;

– #paragraph1 {margin: 0;}.

Атрибуттардың селекторлары;

– a[href="http://www.somesite.com"]{font-weight:bold;}.

Ұрпақ селекторлары (контексттік селекторлар);

– div#paragraph1 p.note {color: red;}.

Балалық элементтер селекторлары;

– p.note > b {color: green;}.

Псевдокласс селекторлары;

– :active {color: yellow;}.

Псевдоэлементтер селекторлары;

- `p:first-letter {font-size: 32px;}`.

Сонымен қатар CSS кестесінде құжатта кездесетін әр элементті белгілейтін әмбебап селектор бар. Мысалы, `* {color:red;}`. Класс немесе идентификатор беретін әр селектор алдында әмбебап селектор белгісін қоюға болады, мысалы, `.first {...}` және `*.first {...}` бірмағынаны білдіреді.

CSS-беттеу. CSS пайда болуына дейін веб-беттерді рәсімдеу тікелей құжат мазмұнының ішінде HTML құралдарымен жүзеге асырылды. Бірақ CSS пайда болысымен құжаттың мазмұны мен көрсетілуін принципті түрде бөлу мүмкін болды. CSS арқылы көптеген ұқсас құжаттарды бір стильдер әсемдеуге және бұл рәсімді тез өзгертуге болады.

Артықшылықтары:

- Әр түрлі қарау құрылғылары үшін беттің бірнеше дизайнларының болуы.
- Бөлек CSS-файлға мәліметтердің көрсету ережесі есебінен сайт беттерінің жүктелу уақытының азаюы.
- Дизайнды өзгерту қарапайымдылығы. Әр бетті емес, тек CSS-файлды ғана өзгерту.
- Рәсімдеудің қосымша мүмкіндіктері.

Кемшіліктері:

- Беттеудің әр браузерде әртүрлі көрсетілуі.
- Жиі кездесетін тек CSS-файлын ғана емес, сонымен қатар HTML тегтерінде түзету қажеттілігі.

Қауіпсіздік. Каскадты стильдер кестесін (CSS) браузерлермен өңдеу кезінде қолданушы компьютерінің жойылған жіберілуі кезіндегі қауіпті кодқа қолданылатын жарияланбаған жады туындауы мүмкін.

2.5 JavaScript тілінің негіздері

JavaScript – бұл объектіге және тілдік құралдарға негзделген, және орта мүмкіндіктері объектімен ұсынылатын, программалау тілі, ал JavaScript сценарилер (программа) бұл өзараәрекеттенуші объектілер жинағы. JavaScript объектісі – бұл реттелмеген қасиеттер жинағы, оның әрбіреуінің ноль немесе одан көп атрибуттары бар, олар бұл қасиет қалай қолданылуы мүмкіндігін анықтайды. Мысалы, егер `ReadOnly` қасиетінің атрибутына `true` (шын) мәні меншіктелсе, онда осы қасиеттерді программалау арқылы өзгерту мүмкіндіктері нәтижесіз болады. Қасиет– бұл басқа объектілерден, көп қолданылатын мәндер мен әдістерден құралатын контейнерлер. Көп қолданылатын мәндер – бұл кез келген енгізілген `Undefined`, `Null`, `Boolean`, `Number` и `String` типтерінің элементі; объект– бұл енгізілген `Object` типінің тағы бір элементі; әдіс– қасиетпен арқылы объектімен ассоциированлған функция.

JavaScript, `Global`, `Object`, `Error`, `Function`, `Array`, `String`, `Boolean`, `Number`, `Math`, `Date`, `RegExp` сыйақты бірнеше енгізілген объектіден құралады. Бұдан

басқа, JavaScript қатаң түрде айтқанда, міндетті түрде функция мен әдіс болып табылмайтын енгізілген операциялар жинағынан, сонымен қатар, программаның орындалу логикасын басқаратын енгізілген операторлардың жинағынан құралады.

JavaScript синтаксисі негізінен Java тілінің синтаксисіне сәйкес келеді, бірақ сценарилер тілін үйренуге жеңілдету үшін онымен салыстырғанда жеңілдетілген. Мысалы, айнымалылар декларациясы оның типінен құралмайды, қасиетте типсіз болады, ал функция декларациясы программа текстінде оны шақырғаннан кейін тұруына болады.

Объектілер жайында жалпы мағлұматтар JavaScript тілі JavaScript және C++ тілдерімен салыстырғанда қатаң түрде объектілер классынан тұрмайды. Оның орнына олар үшін объектілерді құруды жады бөлу жолымен жадыны конструкторларды қолдайды, ол Оның орнына ол объектілерді олар үшін жады бөлу және олардың барлық немесе кейбір қасиеттерін инициализациялау жолымен құратын конструкторларды қолдайды. Барлық конструкторлар объектілер болып табылады, бірақ бар объектілер конструктор болмайды. Әрбір конструктор, прототипті және бөлінетін қасиеттерге негізделген ізін басуды іске асыру үшін қолданылатын prototype қасиетін иеленеді. Объектілер new операциясында конструкторларды шақыру жолымен құрылады; мысалы new String (жаңа жол) String жаңа объектісін құрады. New-сыз конструкторды шақыру нәтижесі конструкторға тәуелді. Сонымен, String("бұл жол"), объект емес көп қолданылатын жолды құрады.

JavaScript прототипке негізделген, із басуды қолдайды. Әрбір конструктормен сәйкес прототип байланысқан, және конструктормен құрылған әрбір объект осы прототипке сілтемені құрайды (объектінің порототипі деп аталынатын). Прототип өз кезегінде, өз прототипіне және ары қарай сілтеме құруы мүмкін. Осылайша прототиптер тізбегі құрылады. Объектінің қасиетіне сілтеме— бұл берілген атпен қасиетті құрайтын объектілер прототипінің тізбегінің бірінші прототипіне сілтеме. Басқа сөзбен айтқанда, егер берілген объектінің берілген атпен қасиеті болса, онда осы қасиетке сілтеме қолданылады, егер жоқ болса, онда осы объектінің қасиеті зерттеледі және т.б..

Объектілер классына негізделген, объектке бағытталған тілдерде, ағымдағы қалып класстар экземплярларымен іске асырылады. Әдістер класстармен, ал із басушылық құраммен және мінез құлқымен . JavaScript та ағымдағы қалып және әдістер объектімен іске асырылады, ал құрамы мен мінез құлқы із басады. Барлық объектілер, олардың прототипінен құралатын, қасиеттерден құралады, осы қасиеттерді және оның тағайындалуын бөледі. Класстарға негізделген тілдерден айырмашылығы, қасиеттер объектіге оларға мәндерді меншіктеу жолымен динамикалық түрде қосылуы мүмкін. Жеке жағдайда, конструкторлар мәндерді барлығына немесе құрылатын объектінің кейбір қасиеттеріне меншіктеуге міндетті емес.

3 “Smart” жарнама агенттігінің мобильді веб сайтының жасау

3.1 Мәселенің қойылымы

Интернет – бұл әлем бойынша әр жерде орналасқан мыңдаған кіші желілерден тұратын өте үлкен, кең ауқымды компьютерлер желісі деп айтсақ қате емес.

“Smart” мобилді веб сайтының құру – электрондық сауда құралын ұсынады және дербес компьютерсіз ақ сайтқа қосылуды жүзеге асырады. Ол тұтынушылардың электронды өнімдерге болған түрлі қажеттіліктеріне үйінде отырып қол жеткізуіне мүмкіндік береді.

Тәжірибеге қарағанда қала орталығындағы жәй дүкен кірісімен салыстырғанда, шын әрекеттенген интернет дүкеннің кірісі әлде қайда жоғары болуы мүмкін. Көп жағдайларда мұндай дүкендер ауыстырмайды, керісінше жәй дүкендерді жақсы толықтырады. Автоматтандырылған интернет дүкендерді аздап немесе толығымен өндеудің мағанасы бар. Аталмыш жерде сіз тауарды қоса аласыз және өзгерте аласыз тіпті олардың бейнесі қоймадағы мәліметтер базасымен байланысты болуы да мүмкін.

Интернет дүкен – интернетте тауарды сату және қызметін көрсетудің өте ыңғайлы жүйесі. Интернет дүкен – үлкен көлемдегі ақпаратты орналастыруға, оперативті түрде тауарлардың түр түрін жаңартуға, жұмыс үрдісін қадағалауға мүмкіндік береді. Мысалы автоматты түрде прайс–листті жаңалауға мүмкіндік береді. Интернет дүкен сіздің дәстүрлік бизнесіңіздің жалғасы. Сонымен қатар тәуелсіз құрлым болуы да мүмкін. Кез келген жағдайда дұрыс басқарылған жұмыс өнімді болады.

“Smart” мобилді веб сайты функционалды қызметтерінің көпсалалығы біздің сайттың тұтынушыға да, қызмет көрсетушілерге де өте қолайлы болып табылады.

3.2 “Smart” мобилді веб сайтының құрылымы мен мазмұны

“Smart” мобилді веб сайты – электрондық сауда құралын ұсынады және жәй дүкеніңіз болмай ақ сатылымды жүзеге асыруға мүмкіндік береді. Ол электрониканың көптеген өнімдерін базарға салып отыр. Интернет дүкен – интернетте тауарды сату және қызметін көрсетудің өте ыңғайлы жүйесі. Интернет дүкен – үлкен көлемдегі ақпаратты орналастыруға, оперативті түрде тауарлардың түр түрін жаңартуға, жұмыс үрдісін қадағалауға мүмкіндік береді. Мысалы автоматты түрде прайс–листті жаңалауға мүмкіндік береді. Интернет дүкен сіздің дәстүрлік бизнесіңіздің жалғасы. Сонымен қатар тәуелсіз құрлым болуы да мүмкін. Кез келген жағдайда дұрыс басқарылған жұмыс өнімді болады.

3.3 Бағдарламалық жабдықтау

3.3.1 Жалпы мағлұматтар

"Smart" мобилді веб сайтының құру HTML, CSS, PHP бағдарламалау тілінде және SQL құрылымды сұраныстар тілінде жазылған.

Мобилді веб сайт құру - электрондық сауда құралын ұсынады және жәй дүкеніңіз болмай ақ сатылымды жүзеге асыруға мүмкіндік береді.

Біздің интернет дүкенмен жұмыс жасау үшін Microsoft Windows 2008/XP операциялық жүйесі, Chrome браузері және сервер қажет.

3.3.2 Функционалдық қызметі

Деректер қоры – бұл құрамында белгілі бір ақпараты бар екіөлшемді өзара байланысқан кестелер жиыны. Клиенттердің сұрауларын сипаттау үшін тұтас тіл құрылған – ол SQL (Structured Query Language – құрылымдық сұраулар тілі).

MySQL — бұл ылғи да компьютерде жұмыс істеп тұратын сервер-программа. Клиенттік программалар (мысалға, сценари) оған арнайы *сұрауларды* сокет (демек, желілік құралдар арқылы) механизмі арқылы жіберіп отырады, ол оларды өңдеп нәтижесін сақтайды.

3.3.3 Логикалық құрылымын баяндау

Программаның негізгі модульдерінің логикалық құрылымы төмендегідей:

3.3.1 кесте – basi.php модулінің логикалық құрылымы

№ жол тізімі	Қызметі
1	Модульдің басы
2	Тақырыптық бөлік
4	CSS –ті іске қосылуы
6- 76	JS тағы мәзір бөлігіне арналған программа
77- 80	Мәзір бөлігіне арналған стилдер
81	Программа денесі
82- 183	Мәзірлік бөлімді орналастыру
184 -218	Таурларды іздеу мен уақыт функциясына арналған бөлік
219 -220	Программа соңы

3.3.2 кесте – smart_index.php модулінің логикалық құрылымы

Жол тізімі	№	Қызметі
	1	Модульдің басы
	2	Тақырыптық бөлік
	3	CSS –ті іске қосылуы
	5	Қаріпті тағайындау

3.3. 2 кесте жалғасы

Жол тізімі	№	Қызметі
	6-7	base.php модулін шақыру
	8	Программа денесі
35	9-	Мәзір мен бағалар тізімі бөлімі
45	36-	Біздің тауарлар мен өтімді тауарлар бөлімі
73	46-	Ортадағы ақпараттық бөлікке арналған программа
107	74-	Оң жақ бөлікті орналастыруға арналған программа
-125	108	Астыңғы адрестік бөліктің программасы
-151	126	Сайт жасаушылар туралы мәлімет беретін бөліктің программасы
-153	152	Программа соңы

3.3.3 кесте – Smart_Loctor.php модулінің логикалық құрылымы

Жол тізімі	№	Қызметі
	1	Модульдің басы
	2	Тақырыптық бөлік
	3	CSS –ті іске қосылуы
	5	Қаріпті тағайындау
	6-7	base.php модулін шақыру

3.3.3 кесте жалғасы

Жол тізімі	№	Қызметі
35	8 9-	Программа денесі Мәзір мен бағалар тізімі бөлімі
45	36-	Біздің тауарлар мен өтімді тауарлар бөлімі
73	46-	Ортадағы тауарларды шығаруға арналған программа
107	74-	Оң жақ бөлікті орналастыруға арналған программа
-125	108	Астыңғы адрестік бөліктің программасы
-151	126	Сайт жасаушылар туралы мәлімет беретін бөліктің программасы
-153	152	Программа соңы

3.3.4 кесте – Smart_reg2.php модулінің логикалық құрылымы

Жол тізімі	№	Қызметі
	1	Модульдің басы
	2	Тақырыптық бөлік
	3	CSS –ті іске қосылуы
	5	Қаріпті тағайындау
	6-7	base.php модулін шақыру
	8	Программа денесі
35	9-	Мәзір мен бағалар тізімі бөлімі

3.3.4 кесте жалғасы

Жол тізімі	№	Қызметі
73	46-	Ортадағы ақпараттық бөлікке арналған программа
107	74-	Оң жақ бөлікті орналастыруға арналған программа

-125	108	Астыңғы адрестік бөліктің программасы
-151	126	Сайт жасаушылар туралы мәлімет беретін бөліктің программасы
-153	152	Программа соңы

3.3.5 кесте – Smart_Loctor.php модулінің логикалық құрылымы

Жол тізімі	№	Қызметі
	1	Модульдің басы
	2	Тақырыптық бөлік
	3	CSS –ті іске қосылуы
	5	Қаріпті тағайындау
	6-7	base.php модулін шақыру
	8	Программа денесі
37	9-	Мәзір мен бағалар яғни сол жақ орналастыруға арналған программа
41	38-	Ортадағы қолданушыларды тіркеуге арналған форма программасы

3.3.5 кесте жалғасы

Жол тізімі	№	Қызметі
47	42-	Оң жақ бөлікті орналастыруға арналған программа
51	48-	Астыңғы адрестік бөліктің программасы мен Сайт жасаушылар туралы мәлімет беретін бөліктің программасы
53	52-	Программа соңы

3.3.6 кесте – tau_kil_tr.php модулінің логикалық құрылымы

Жол тізімі	№	Қызметі
	1	Модульдің басы
	2	Тақырыптық бөлік
	3	Программа денесі
	5-	Метод POST бойынша айнмалыларды меншіктеу

16		
22	17-	Толырылмаған өрістерге ескерту беру программасы
34	23-	Тіркелушінің мәліметтерін дерекқорға жазатындай етіп қарыпты дұрыстау
47	35-	Дерекқор мен байланыс орнатып тіркелушінің мәліметтерін дерекқорға жазу
50	48-	Керекті ақпаратқа сілтемелер
52	51-	Модульдің соңы

3.3.7 кесте – mon_sur1.php модулінің логикалық құрылымы

Жол тізімі	№	Қызметі
	1	Модульдің басы
	2-4	Тақырыптық бөлік
	5	Программа денесі
	6-7	basi.php модулін шақыру
41	8-	Авторизация модулі
53	42-	Талданған тауар туралы мәліметтер
67	54-	Төлем тәсілін таңдауға арналған программа бөлімі
69	68-	Модульдің соңы

3.3.4 Шақыру және жүктеу

Бұл жобаны орындау үшін асыру үшін WebServer программасын жүкеп, WWW қалтасының ішіне жасаған Web парақтарымызды саламыз, одан кейін Internet Explorer браузерін жүктеп адресік жолға localhost деп жазып жүктейміз.

3.3.5 Қажетті техникалық жабдықтар

Курстық жоба Pentium® 4 процессорында, XP professional edition операциялық жүйесінде жасалды (Cpu 1.80Ghz, 256 Mb ОЗУ). Орындалған жұмысты қағаз бетіне түсіру Samsung ML-1210 баспа құрылғысы арқылы жүзеге асырылды.

3.3.6 Кіріс мәліметтер

Кіріс мәліметтер ретінде тіркелген қолданушылардың тауарларды сатып алу үшін өздерінің ақпараттарын енгізуі және олардың ақша төлеу формасын және тауарды таңдауы, тауардың санын белгілеуі және Туран Алем Банкі карт номерін және құпия сөзін енгізуі.

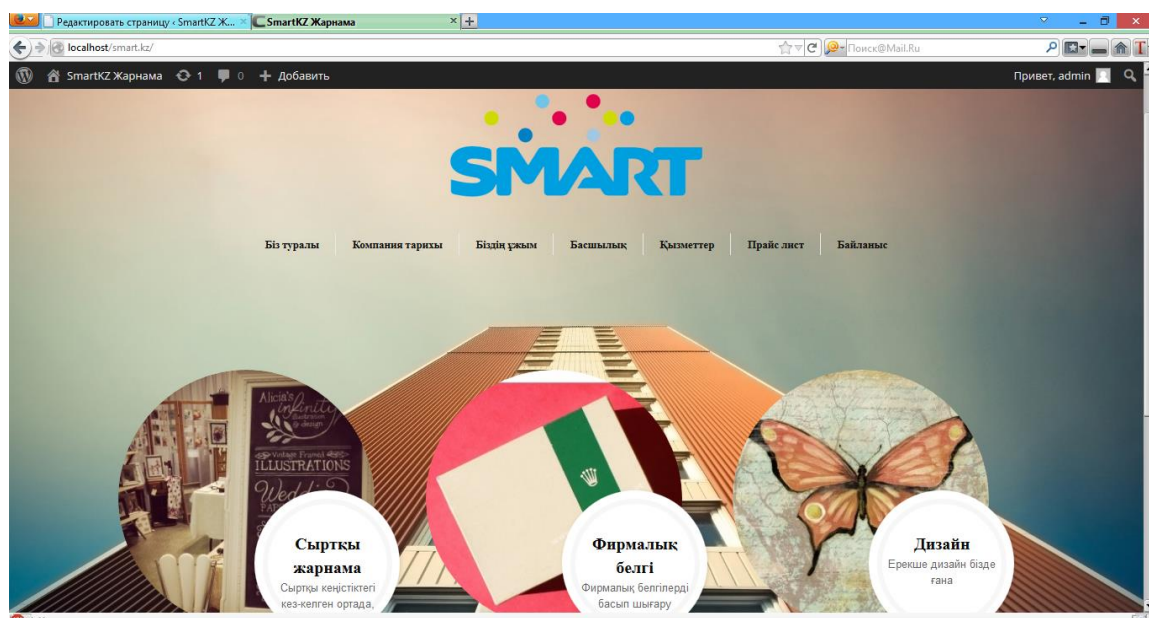
3.3.7 Шығыс мәліметтер

Шығыс мәліметтерге клиенттердің енгізген мәліметтері бойынша авторизация жасап, егерде сұраныстары қабылданған жағдайда оларға керекті мәліметтерді яғни сұраныстарының қабылданғаны және қанша уақытта сұраныс берген тауарды жеткізіп беретіні туралы мәліметтер шығару.

3.4 “Smart” мобилді веб сайты қолданушыларға нұсқаулық

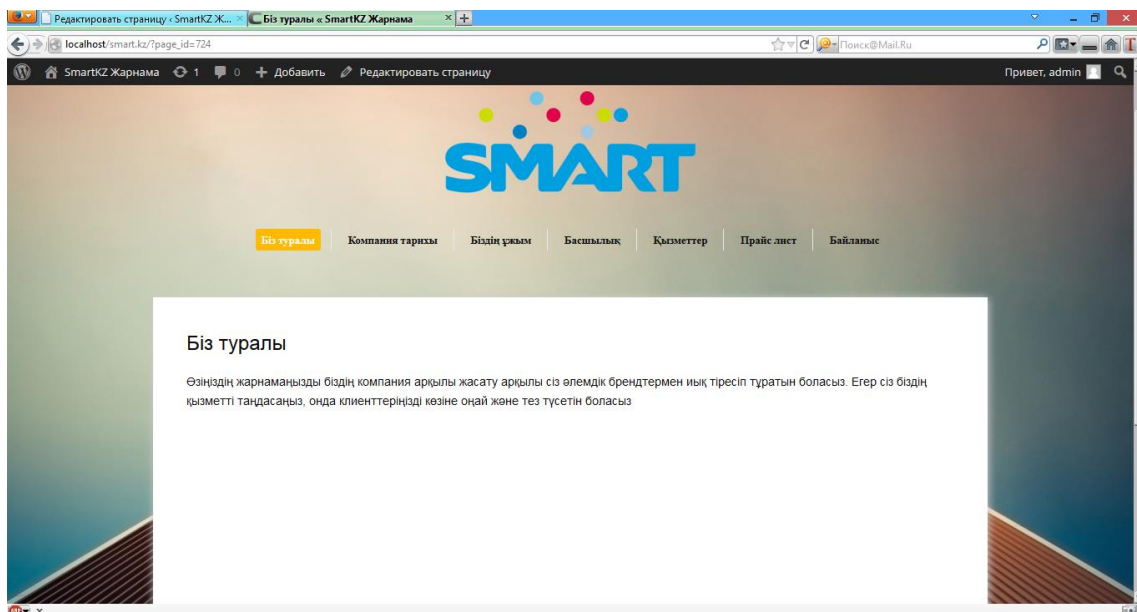
3.4.1 Қолданушыларға нұсқаулық

“Smart” мобилді веб сайтының бас беті төменде көрсетілгендей формада шығады:



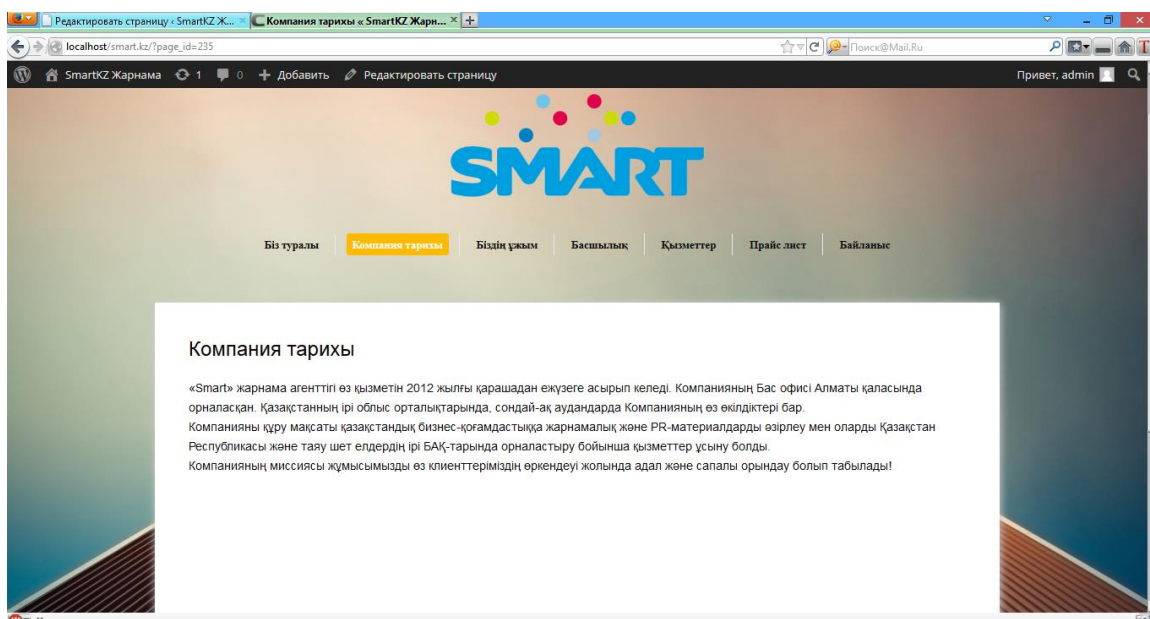
3.1.1 – сурет. Негізгі бет

Әрине бұл негізгі бет көпшілікке бірдей түсінікті болып, қолданушы еш саспастан өзінің ойдағы көргісі келген ақпаратына батырмалардағы ашық жазылған гиперсілтемелер көмегімен көру мүмкіндігін алады. Мәзірдегі сайкесті батырмаға курсор барысымен қосымша мәзір ашылады. Сөйтіп қолданушы “Smart” мобилді веб сайтының барлық тауарлар туралы кең көлемді ақпарат ала алады. Шертілген мәзірге сай (3.1.2 – суретке) ие боламыз.



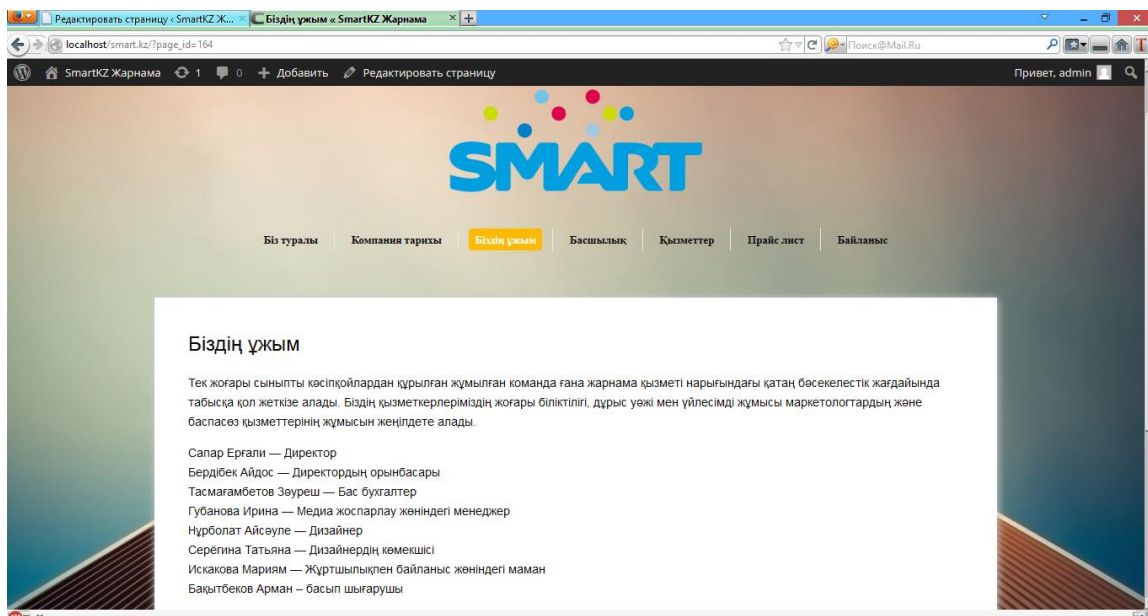
3.1.2 – сурет. Біз туралы

Егер сіз өзіңіздің ойыңыздағыдай өнімді тауып сатып алуға сұраныс бергіңіз келсе әуелі тіркеліп алуды естен шағармаңыз. Ол үшін сіз веб парағының сол жағын ала орналасқан тізімдердегі тіркеу батырмасын шертіңіз.



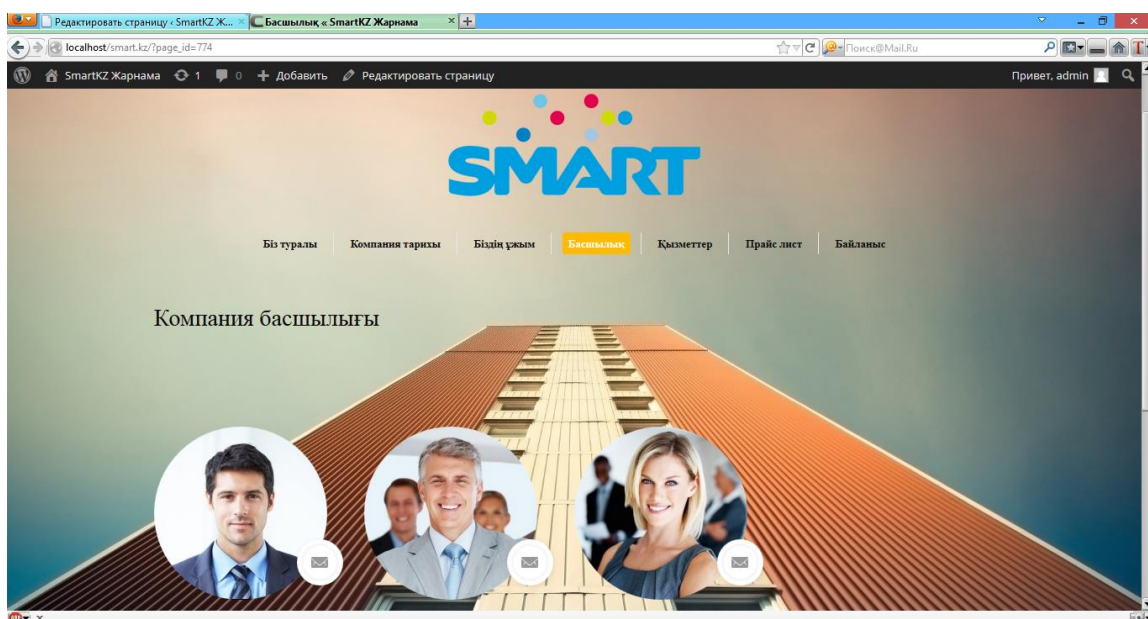
3.1.3 – сурет. Компания тарихы

Ары қарай өзіңіз таңдаған өнімнің жанындағы сатып алу батырмасын шертсеңіз келесідей сатып алуға сұраныс беру беті ашылады:



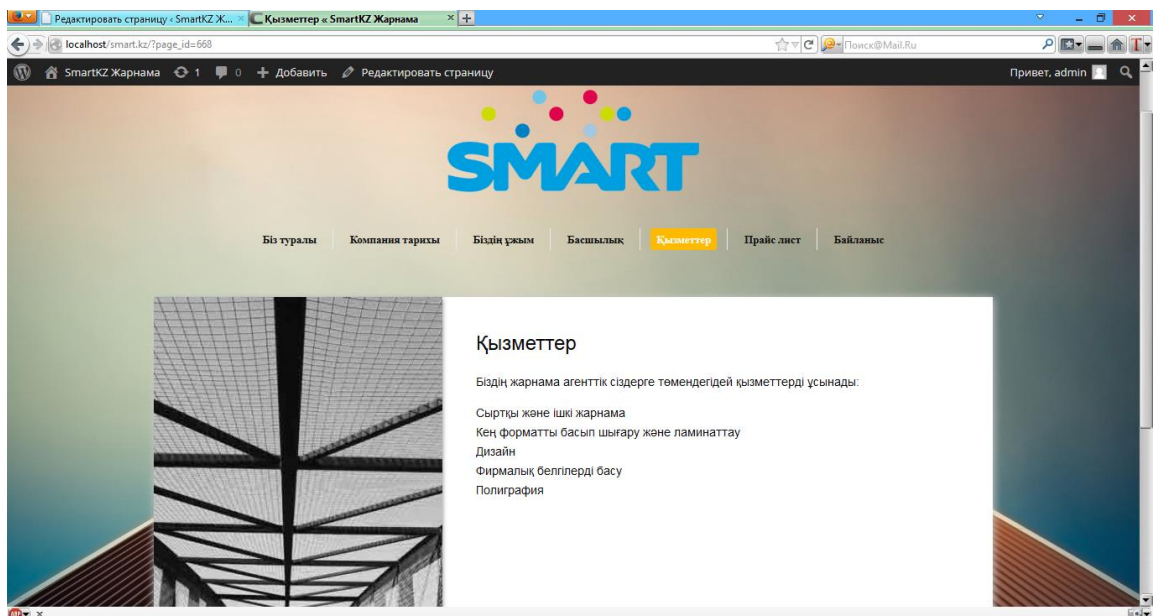
3.1.4 – сурет. Біздің ұжым

Бұл жерде сіз өзіңізге керекті өнім санымен өзіңізге ыңғайлы ақша төлеу формасын таңдай аласыз. Өз талғамына сай төмендегі ақпаратқа ие боласыз.



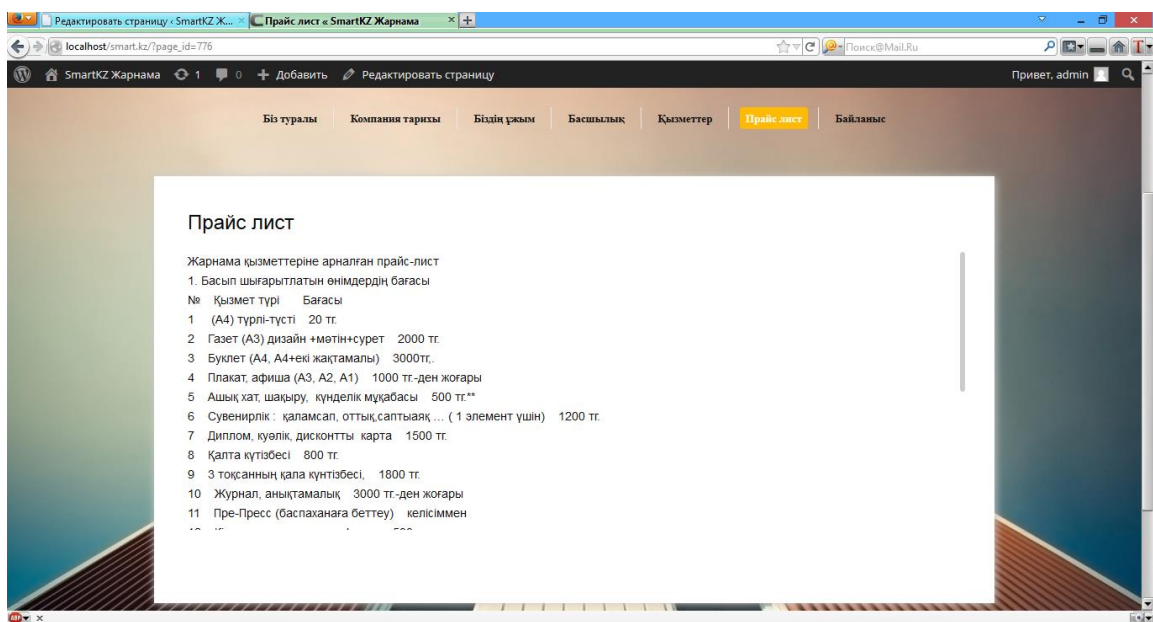
3.1.5 – сурет. Компания басшылығы

Бұл бөлімде сайттағы деректер қорына жаңа тауарлар қосуға, ондағы мәліметтерді өзгертуге, редакциялауға және оны жоюға болады:



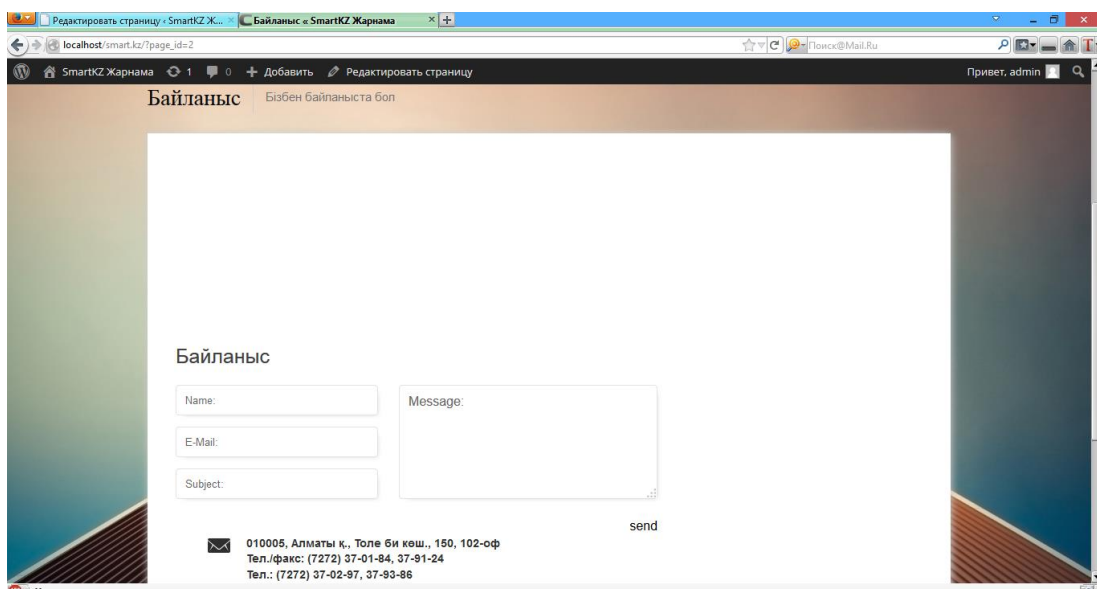
3.1.6– сурет. Қызметтер

Ал төменде деректер қорға жазылған тіркелген қолданушылар туралы мәліметтер.



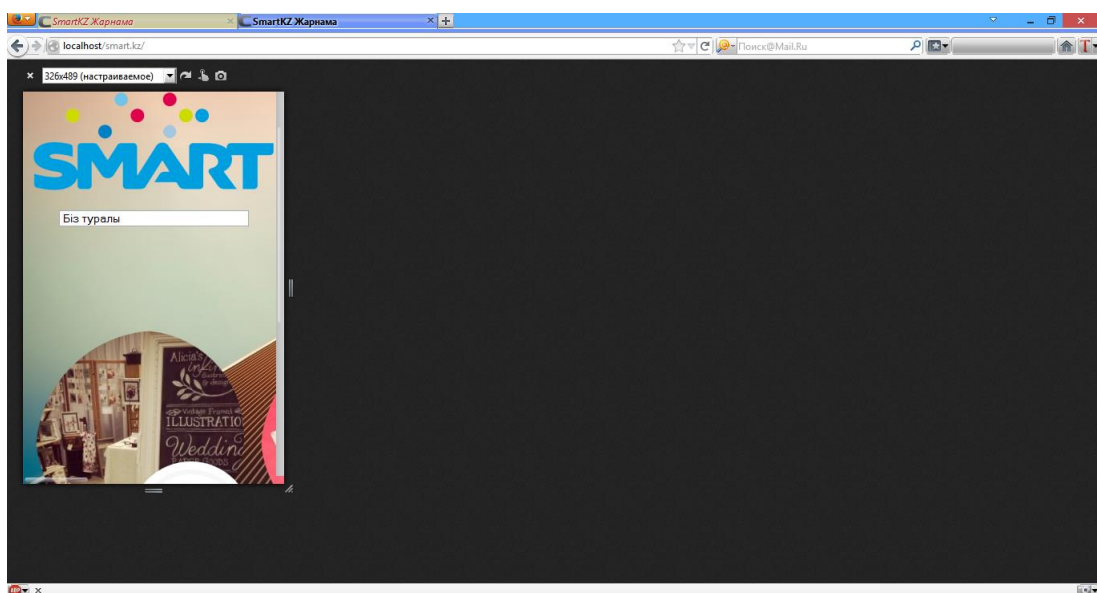
3.1.7– сурет. Прайс-лист

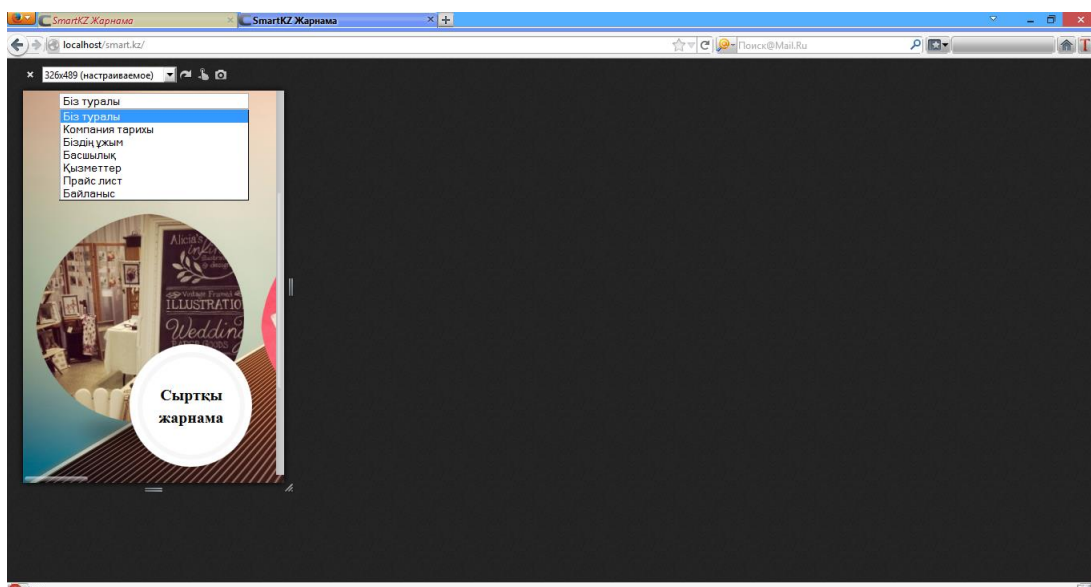
Ең соңғы бет бұл бізбен байланыс беті. Онда телефон нөмірлері, адрес орналасқан:



3.1.7– сурет. Байланыс

3.1.8– сурет. Сайтымыздың мобильді құрылғы арналған скриншоттары





3.1.9– сурет. Басты бетіміздің скриншоты

Бұл жерде біз өзімізге қажетті вкладканы таңдап алуымызға болады

4 Өмір тіршілік қауіпсіздігі

4.1 Еңбек шартының анализі

Эвакуация уақытын есептеу және кондиционерлеуді есептеу

Дизайнерлердің жұмыс орны деп іскердің өндірістік процессті жүргізу және бақылау мақсаты үшін жабдықталған кеңістік бөлігін айтады. Сонымен қатар, жұмыс орны болып, оның еңбек қызметін іске асыратын және басқару органы мен қосымша жабдықпен, ақпаратты көрсету құралдарымен жабдықталған жүйедегі орыны саналады. Дизайнерлердің жұмыс орынын құрастырғанда келесі шарттардың маңызы зор.

Мекеменің орналасқан орны

Дизайнерлер Алматы қаласында орналасқан «Smart» жарнамалық агенттігінде жұмыс жасау үстінде. Дизайн жасау барысында дербес компьютерді және электронды құрылғыларды қолданамыз.

Жұмысшылар саны

Қарастырылып отырған бөлмеде 3 дизайнер жұмыс атқарады, олардың әр қайсысының жеке жұмыс істеу орны бар.

Жұмыс уақыты

Дизайн жасау және оны басып шығару, басты рөлді жұмыс істеу орнына бөледі, себебі дизайнерлер аптасына 5 күн жұмыс істейді, 8 сағаттан, яғни олар компьютердің алдында аптасына 40 сағат жұмыс істейді, оған табиғи жарықты және жасанды жарықты түсу нормаларын ескеру қажет.

Жұмыстың ауырлық деңгейі

Бөлмеміз жұмыс үстеу шарты бойынша, жеңіл жұмыс категориясына жатады (физикалық жеңіл, категориясы Ia, жұмысты отырып істейді және физикалық ауыр жұмыс атқармайды), осы стандарт (ГОСТ 12.2.032-78);

Микроклимат

Жұмыс бөлмесін талдау барысында жұмыс аумағының қалыпты микроклиматтық шарттарын міндетті түрде қарастыру қажет.

Микроклиматтың күйін қадағалау дизайнерлер залында қолайлы жағдайды орнатуға көмектеседі. Ал жұмыс орнының қолайлы жағдайларында адамдардың жұмыс істеу қабілеттері жақсарады, жұмысшылардың ауруға шалдығуы азаяды сонымен қатар, еңбек өндірісі артады. Жұмыс орнындағы нормаланған климаттық параметрлер 4.1-кестеде келтірілген.

Жыл мезгілі	Жұмыс категориясы	Ауа температурасы, °С	Салыстырмалы ауа ылғалдылығы, %	Ауа қозғалысының жылдамдығы, м/с
-------------	-------------------	-----------------------	---------------------------------	----------------------------------

Суық	Жеңіл - 1а	22-24	40-60	0,1
		21-23	40-60	0,1
Жылы	Жеңіл - 1а	23-25	40-60	0,1
		22-24	40-60	0,2

Орта физикалық жұмыс үшін ССБТ. Бөлмеде қысқы кезеңдерде ауаның температурасы +18 до +20°С болады, ал жазғы ауқыт кезеңдерінде ауаның температурасы +25°С-ге дейін көтеріледі.

Жарықтандыру жүйесі

Бөлмеде күндіз табиғи жарық көзі ретінде екі терезе бар және жасанды жарықтандыру орнатылған, оның көмегімен түнде және күнде жұмыс істеуге болады. Жасанды жарықтандыру III дәрежелі өте үлкен дәлдікпен жұмыс істеу талабына сай болып келеді. Жасанды жарықтандыру 2 шамдал екі люминесцентты шамы арқылы жүзеге асады.

Өрт қауіпсіздігі

Электротехникалық жабдық, ең бірінші өрт көзі болып табылады және қауып тудырады. Жұмыс уақыты біткеннен кейін барлық электротехникалық жабдықтарды өшіруіміз қажет.

Бөлме өрт қауіпсіздігінің жөн-жоба талабына сай 100 м²-қа ОУ-5 көмір қышқылыды өрт сөндіргішпен жабдықталдануы керек. Бөлменің орта ауданы 24 м² болғандықтан, бір ғана көмір қышқылыды өрт сөндіргіш орнатылады.

Электрқауіпсіздігі

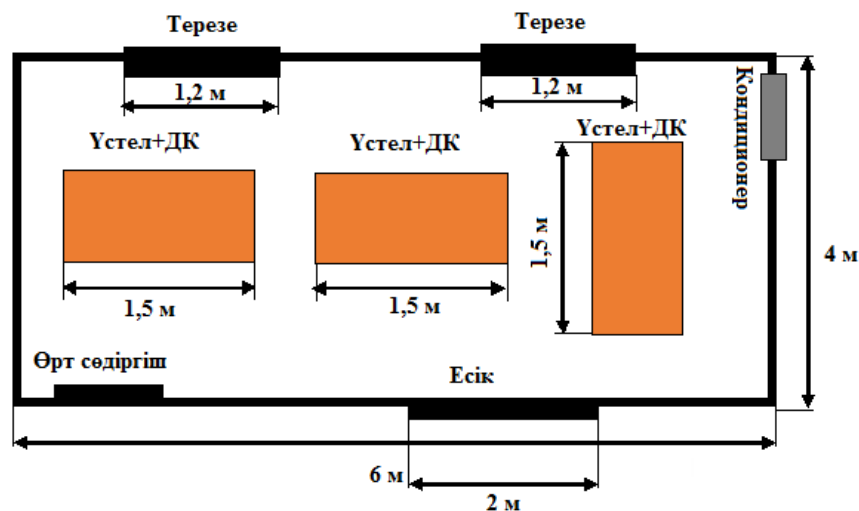
Берілген бөлмеміз барлық электр қауіпсіздігін қанағаттандырады, яғни бөлме ылғалдылығы 60% аспайды, тақтай еденнен жасалған. Жерге тұйықтау және нөлдеу сымдарының қимасы, олардың қосылыстары мен жалғанымдарының бүтіндігі және беріктігі тексерілген. Аппаратты жерге тұйықтау контурымен қосатын жерге тұйықтау сымдарда үзілу және көзге көрінетін ақаулар болмауы тиісі ескерілген. Дәнекерлеу жұмысының сенімділігі балғамен соғып тексерілген.

Шу көзі

Қарастырылып отырған жұмыс орындағы бөлмеміз, ғимаратта орналастырылған және ғимарат теміржол магистралінде немесе жүкті автомагистралінде, аэропорт және тағы басқа орындарда орналаспаған. Соған байланысты жұмыс істеу процессіне сыртқы шудың көзі жоқ.

Жұмыс орнының мінездемесі

Бөлменің жоспары 4.1 суретте көрсетілген.



4.1- Сурет – Бөлме жоспары

Бөлме келесі параметрлерге ие:

- бір қабатты ғимаратта орналасқан, бірінші қабатында;
- бөлменің өлшемдері (бөлме): ұзындығы 6 м, ені 4 м, биіктігі 3 м;
- жарық өткізетін материалдың түрі – шыны парағы, екіқабатты;
- тоқыманың түрі - екіқабатты стальден жасалған;
- күннен қорғанатың жабдықтар – жиналатын жазюли және шторлар;

- екі терезе 1,5*1,2;

- қабырғаның ішкі түрі – ақшыл; Жасанды жарық көзі – 2 шамдал екі люминесцентты шамы.

Жұмыс істеу кезіндегі қолданылатын құрылғының мінездемесі:

- Intel i3 core 2Duo CPU E8400 @ 3.4 GHz 3.4 GHz, 4 GB RAM / HDD 1000 Gb;

- монитор Asus VX279Q, диагональ 18”;

- қоректену: айнымалы кернеу 220-250 В, жиілігі 50 Гц. Қуат 400 Вт;

- 2 шамдал, 4 люминесцентты лампасы;

- қоректену: айнымалы кернеу 220-250 В, жиілігі 50 Гц, шамдардың қуаты 2x28 Вт.

Құралдарымыз аз шуылдайтын және зиянды ара сапалы шуы болмайды.

4.2 Есептеу бөлімі

4.2.1 Эвакуация уақытын есептеу

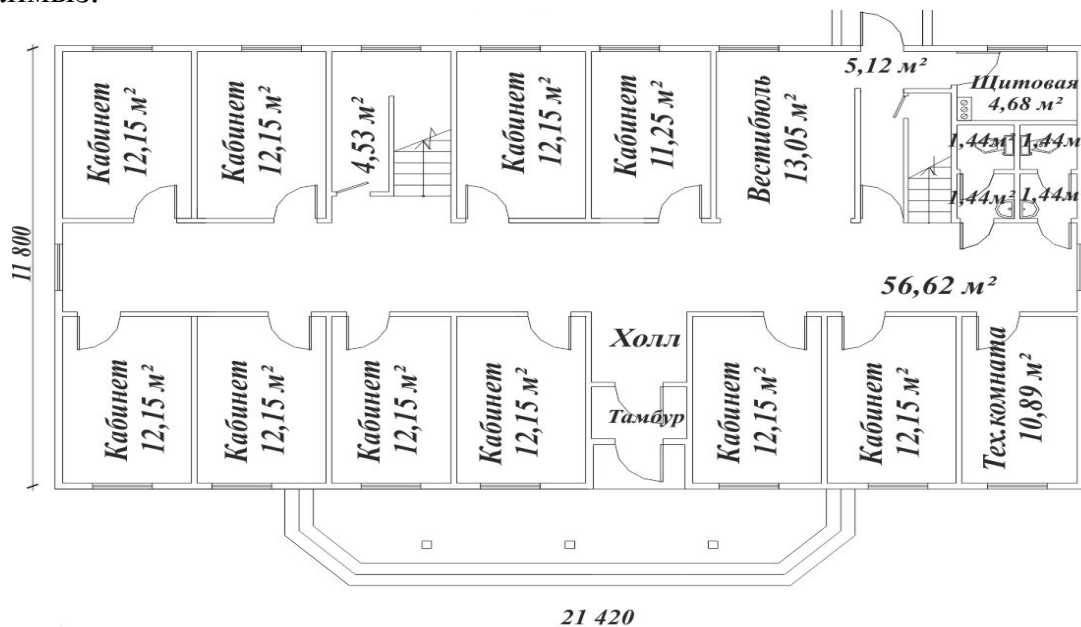
Өрттің пайда болу себептері:

- аппараттың элементтерінің жануы;

- жөнделмеген ажыратқыштардан, розеткалардан, бөлімдегі материалдармен тікелей қатынаста болатын желдету жүйелерінен бөлім материалдарының жануы;

– жабдықты эксплуатациялау режимдерін сақтамау, қызметшінің дұрыс емес әрекеті.

Өрт болған жағдайда тек бөлме ғана зиян шекпейді, сонымен бірге қымбат тұратын аппараттар, адам өлімі болуы мүмкін. Сондықтан өртті ертерек анықтап және сөндіру бойынша шара қолдану қажет. Бөлмені қорғау үшін дренчерлік қондырғы негізіндегі автоматтық өрт сөндіргішті таңдаймыз.



4.2 - Сурет - ғимарат жобасы

Қорғаудың негізгі әдістерінің қатты әсер ететін факторларынан бірі қауіпті аудандар және зілзала аймақтарының объекттердің қызметшілерін және тұрғындарын дер кезінде эвакуациялау болып табылады.

Эвакуация – қауіпті аймақтардан адамдарды алдын-ала шығаруға арналған шаралардың кешені.

Кәсіпорынның қызметкерлерін бөлме өртінің туу жағдайында ғимараттан адамдарды эвакуациялау уақыттын анықтау керек. «Smart» жарнамалық агенттігінің ғимараты 1-этажды және жобада 12x11м болып келеді. Бөлмелердің аудандарының қосындысы 110 шаршы метрді алып жатыр. Бұл мекемеде 15 адам жұмыс жасайды. Эвакуациялық шығуларға арналған есіктер ені-1метр және ғимараттан шығатын есіктерді саны – 1.

Эвакуациялық жолмен қозғалған адамның ағындары жол-жөнекейімен екі есік ойықтары арқылы өтеді. Олардың тиісті ені:

$$\beta = N/165$$

(4.1)

N – адамдардың жиынтық саны, адам;

165- I және II отқа төзімділік дәрежелері үшін ғимараттардың ені 1 м есіктің алдынан адамдарының жиналып қалмай, жіберілетін нормативтік адамдардың саны.

$$N=15/1=15\text{адам}$$

Бір эвакуациялық шығу үшін :

$$\beta = 15/165=0,09 \text{ м}$$

Эвакуациялық есіктердің нақты өлшемдері 1 м

Қорытынды: көрермен залынан эвакуациялық шығуды жобалаудың нормаларының талаптарына жауап береді.

Ғимараттан шығуға дейін адамдарды эвакуацияның ұзақтығы сыртқы есіктер және сатылар арқылы эвакуацияның созылымдық және өткізгіштік қабілеті арқылы анықтайды.

T - дәліз бойымен адам ағынының қозғалыс уақыты мына формула бойынша есептеледі:

$$t_1 = \frac{L_1}{V_1} = 5/47 = 0,11 \text{ мин}$$

(4.2)

L - дәліз ұзындығы;

V – көлбеу жол бойымен адам ағынының қозғалыс жылдамдығының мәні, салыстырмалы тығыздыққа D адам/м² тәуелді болып табылады.

D - адам ағынының тығыздығы, төмендегі формула бойынша есептеледі:

$$D_1 = \frac{N_1}{L_1 \cdot b_1} = 15/5 \cdot 1 = 3 \text{ адам/м}^2 \quad (4.3)$$

N₁ – адам саны

L₁ и b₁ - жолдың, дәліздің ұзындығы және жолдың бірінші бөлімінің ені

f - адамның көлбеу проекциясының орташа ауданы, ол 0,1 м²-қа тең.

13.4 кесте бойынша (Охрана труда на предприятиях связи и охрана окружающей среды, Баклашов Н.И., 1989 г., 288с.) 3 адам/м² адам ағынының тығыздығына горизонтальды жолда 47 м/мин адам ағынының жылдамдығы сәйкес келеді.

Ойықта қозғалыс уақыты төмендегі формуламен анықталады:

$$t_2 = \frac{N \cdot f}{v \cdot b} = \frac{15 \cdot 0,1}{47 \cdot 0,09} = 0,35 \text{ мин} \quad (4.4)$$

Жалпы ғимараттан эвакуациялау уақыты:

$$T = t_1 + t_2 = 0,11 + 0,35 = 0,5 \text{ мин} \quad (4.5)$$

Мысалы, белгілі бір өндіріс орнының өткетөзімділік дәрежесі IIIа, ондағы өндіріс категориясы В. Дәліз соңында орналасқан екі баспалдақтан 1 мин ішінде көлемі 150000 м³ болатын өндірістік мекемеден

эвакуацияланатын адамдар санын және баспалдақтағы есіктік өткелдің ең кіші рұқсат етілген енін анықтау керек болсын. Ол үшін баспалдаққа дейінгі арақашықтық $L_1 = 60\text{м}$, өткелдің ені $d = 1,5\text{м}$. Адамдардың керекті қозғалыс жылдамдығы:

$$V_1 = L_1/t = 60\text{м/мин}$$

Осы мәнге адам ағынының тығыздығы кесте бойынша сәйкес келеді $D_1 = 2 \text{ адам/м}^2$

$$D = N_1/L_1d_1$$

осыдан:

$$N_1 = D_1 * L_1 * d_1 = 2 * 60 * 1.5 = 180 \text{ адам}$$

Осыған сәйкес 180 адам эвакуацияланады. Адам ағынының тығыздығын D_1 ді және адам қозғалысының жылдамдығын біле отырып, кесте бойынша адам ағынының қозғалысының интенсивтілігін анықтай аламыз, ол $q_1 = 12 \text{ адам/м}^2$

Осыдан есіктік өткелдің қажетті ені:

$$d_2 = q_1 * d_1 / q_{\text{max}} = 12 * 1.5 / 19.6 = 0.92 \text{ м}$$

мұндағы $q_{\text{max}} = 19,6 \text{ адам/м}^2$ – кесте бойынша алынған адам ағынының максималды интенсивтілігі.

Егер дәлізден сыртқа шығатын есік болса немесе басқа горизонтальды дәліз болса, $q_{\text{max}} = 16,5 \text{ адам/м}^2$ ға тең болады. Онда есіктік өткелдің ені төмендегі мәнге тең болады:

$$d_2 = d_1 * q_1 / q_2 = 1.5 * 12 / 16.5 = 1.1 \text{ м}$$

Қосымшалар

Кесте 4.2-Өрт жарылыс қауіпі бойынша бөлмелердің классификациясы

Бөлменің категориясы	Заттардың сипаттамасы
А(өрт-жарылыс қауіпті)	Жарылыс қауіпті газдар, $< 28^\circ\text{C}$ тотығу температурасы бар жанғыш сұйықтықтар
Б (өрт-жарылыс қауіпті)	Жарылыс қауіпті шаңдар және $T \sim 28-61^\circ\text{C}$ жанғыш сұйықтықтар
В (өрт қауіпті)	Қатты , жанғыш, сұйықтықтар (ағаш өңдейтін цехтар)
Г	Қыздырылған немесе балқытылған түрдегі жанбайтын заттар
Д	Суық түрдегі жанбайтын заттар

4.2.2 Кондиционерлеу жүйесін есептеу

Жұмыс бөлмесінде белсенді вентиляция жүйесі бар. Жұмыс бөлмесін талдау барысында жұмыс аумағының қалыпты микроклиматтық шарттарын міндетті түрде қарастыру қажет. Мұндай әмбебап жүйе ретінде автономдық кондиционерлер болып табылады. Кондиционерді орнату алдында арнайы есептеулер жасау керек. Сол есептеулердің қорытындысымен сипаттамасы және талаптарына сай кондиционер таңдалады. Бөлмені кондиционерлеу төменде келтірілген.

Офис бөлмесін винтеляциялауға арнайы винтеляция каналдарын қолданады, бұл каналды ғимаратты салып жатқанда жасайды және жазғы уақытта терезе ашық болғанын ескереді. Жылдың жылы мезгілінде, офистағы температураның нормадан артуы 4.1-кестесінде көрсетілген, ал оптимальді микроклиматты ұстау үшін, бұл жағдайда кондиционерді қолданамыз. Әдеттегі офистағы микроклимат оператордың жұмыс істеу қабілетіне әсер етеді, егер офистағы микроклиматымыз жақсы болса, онда оператордың жұмыс істеу қабілеті арта түседі. Соған орай, микроклиматты жақсы дәрежеде ұстау үшін, біз кондиционер орнатуымыз қажет.

Төменде жұмыс бөлмесіндегі, кондиционерлеу жүйесінің есептеуі көрсетілген. Кондиционерлеу бөлмедегі лайықты климатты ұстап тұрады және берілген нормаларды қанағаттандырады.

Бөлмеге келетін ауаны L_k , $\frac{M^3}{c}$ формуласымен анықтаймыз:

$$L_{пр} = \frac{Q_{ашж}}{c \cdot \rho_{пр} \cdot (t_{выт} - t_{пр})} \quad (4.6)$$

Осындағы $Q_{ашж}$ артық шығындалатын жылу, $\frac{кДж}{c}$;

c - ауаның үлесті жылусиымдылығы, тұрақты түпкілікті қысымға тең $c=1 \frac{кДж}{кг \cdot ^\circ C}$;

$\rho_{пр}$ - бөлмеге келетін ауаның тығыздығы, тең $1,2 \frac{кг}{м^3}$;

$t_{выт}$ - бөлмеден ауа температурасының шығуы және жұмыс істеу зонасынан, $^\circ C$;

$t_{пр}$ - келетін ауа температурасы, $^\circ C$.

Бөлмеден шығатын ауаның температурсы $t_{шат}$, $^\circ C$, осы формуламен анықталады:

$$t_{шат} = t_{жз} + \Delta t \cdot (h_{ст} - z) \quad (4.7)$$

осында: $t_{жз}$ - жұмыс істеу зонасындағы температура, осы берілген нормадан аспауы ($t_{жз} \leq t_{доп}$), $^{\circ}\text{C}$;

h_{CT} – еденмен центрге дейінгі сорып алатын тесігі (кондиционер), м;

H - жұмыс зонасының биіктігі, м.

Осыған орай есептелу жылу мезгіліне арналған, сонда температура

$$t_{жз} = 22^{\circ}\text{C}$$

Кондиционердің үшкі жағы, осындай биіктікте $h_{CT} = 2,5$ м:

$$t_{\text{выт}} = 22 + 1,2 \cdot (2,5 - 3)^{\circ}\text{C}$$

Келетін ауаның температурасы $t_{пр}$ бар жылудың артықшылығына $5 - 7^{\circ}\text{C}$ төменде жұмыс істеу орнының:

$$t_{пр} = 22 - 7 = 15^{\circ}\text{C}$$

Артық бөлінетін және шығатын жылу $Q_{\text{АШЖ}}$ табамыз, бөлмедегі жылу балансына сүйене отырып осы:

$$Q_{\text{АШЖ}} = \sum Q - \sum Q_{\text{шы}} \quad (4.8)$$

осында: $\sum Q$ - бөлмеге келетін жылудың суммарлы саны;

$\sum Q_{\text{шы}}$ - бөлмеден шығатын суммарлы жылудың саны (есептеу арқылы жылудың жоғалуы, жылыңған жылудың бөлмеге келуі және т.б.).

Басты жылудың бөліну көзі болып, шамдар есептеледі және адамдар т.б. Тағыда, осыған орай есептеуде біз күн радиациясынан түсетін жылуды ескеруіміз қажет. Осы бөлмедегі дербес компьютерлерді және электр құрылғыларын ескермеуімізге болады. Осыдан біз бөлмеге бөлінетін жасанды жарықты, адамдардан, күн радиациясы арқылы келу жылу саннын есептейміз және терезеден келетін жылуды.

Жасанды жарықтан бөлінетін жылуымыз Q_2 , есептейміз, практикалық күйдегі шығатын энергияны, сонғы есепте төменде көрсетілген формуламен есептейміз:

$$Q_2 = 1000 \cdot N \quad (4.9)$$

Осында N – шамдардан шығатын қуат кВт .

$$Q_2 = 1000 \cdot 0,28 \cdot 4 = 1120 \text{ кВт}$$

Адамдардан шығатын жылу Q_3 төмендегі формуламен анықтаймыз:

$$Q_3 = n \cdot q_{\text{ч}} \quad (4.10)$$

осында: n – жұмыс атқаратын адамдар саны;

$q_{\text{ч}}$ - жылу саны, бір адамнан бөлінетін, 4.2- кестесінде көрсетілген.

Кесте 4.3 – Жылу саны, бір адамнан бөлінетін, жұмыс істеу категориясына байланысты және осы жұмыс зонадағы температураға байланысты.

Жұмыс категориясы	Жылу, Вт			
	Толық		Ашық	
	егер 100°C	егер 350°C	егер 100°C	егер 350°C
Жеңіл	180°C	145°C	150°C	5°C

$$Q_3 = 3 \cdot 145 = 435 \text{ Вт}$$

Жылу саны, күн радиациясынан бөлмеге келетін $Q_{\text{КУН.РАД}}$, осы формуламен анықталады:

$$Q_{\text{КУН.РАД}} = F_{\text{ост}} \cdot q_{\text{ост}} \cdot A_{\text{ост}} \quad (4.11)$$

Жабылған жағдай үшін:

$$Q_{\text{Ж.РАД}} = F_n \cdot q_n \cdot k_n \quad (4.12)$$

Осында $F_{\text{ост}}$ және F_n - жабылатын беттің ауданы, м^2 ;

$q_{\text{ост}}$ және q_n - жылу келетін 1 м^2 беткі шыныдан және беткі қабаттан, егер жылу өткізу коэффициент, тең болса $1 \frac{\text{Вт}}{\text{м}^2 \cdot ^{\circ}\text{C}}$;

$A_{\text{ост}}$ – шыналу коэффициенті;

k_n – қабатқа жылу берілу коэффициенті, $1 \frac{\text{Вт}}{\text{м}^2 \cdot ^{\circ}\text{C}}$.

Осы мән $q_{\text{ост}}$ тәуелділікке және географиялық қабаттың орнына терезенің және шамның 70–210 аралығындағы мәндерді қабылдайды, ал коэффициент $A_{\text{ост}}$ тәуекел шынының түріне және оның күнге қарсы қорғаныс құрлымына байланысты – осы шекте 0,25–1,25, жылу келудің орташа мәні, ол күннен келетін радиацияға, қандай қабаттардан өтетінін ескеріп отырылады 6 - 24.

$$F_{\text{ост}} = 1,5 \cdot 1,2 \cdot 2 = 3,6 \text{ м}^2$$

Жұмыс істеу орнының терезесі солтүстікке қарай бағытталған, соған орай оның мәні $q_{\text{ост}}$ тең $140 \frac{\text{Вт}}{\text{м}^2 \cdot ^\circ\text{C}}$. Осыдан $A_{\text{ост}} = 0,35$

$$Q_{\text{ост.рад}} = 3,6 \cdot 140 \cdot 0,35 = 176,4 \text{ Вт}$$

Жылу келудің орташа мәні географиялық ұзындығын ескеріп және қабатын қарастырсақ тең $Q_{\text{п.рад}} = 18 \text{ Вт}$.

Бөлмедегі жылудың жоғалуы Q_{yx} , кВт есіктен және қабырғадан, терезені формула арқылы бағалайды:

$$Q_{\text{yx}} = \frac{\lambda \cdot S \cdot (t_{\text{выт}} - t_{\text{пр}})}{\delta} \quad (4.13)$$

осында: λ - қабырғаның жылу өткіздігіштігі, $\frac{\text{Вт}}{\text{м} \cdot ^\circ\text{C}}$;

S – аудан, м^2 ;

δ - қабырғаның қалыңдығы, м.

Жұмыс істеу орнының қабырғасы ауыр бетоннан жасалған М600, жылу өткізгіштігі тең $12 \frac{\text{Вт}}{\text{м} \cdot ^\circ\text{C}}$. Қабырға қалыңдығы $\delta = 0,5 \text{ м}$

$$Q_{\text{yx}} = \frac{1,2 \cdot 24 \cdot (21,4 - 15)}{0,5} = 368,64 \text{ Вт}$$

Бөлмеге келетін жылудың суммарлы санын есептейік:

$$\sum Q = Q_2 + Q_3 + Q_{\text{ост.рад}} + Q_{\text{п.р.р}} \quad (4.14)$$

$$\sum Q = 1120 + 435 + 176,4 + 18 = 1749,4 \text{ кВт}$$

Сонда есептеуіміз жаздың жылу мезгіліне келеді және осыдан бөлінетін жазғы жылуымыз:

$$Q_{\text{изб}} = 1749,4 \text{ кВт}$$

Бөлмеге келетін ауаны есептейік:

$$L_{\text{пр}} = \frac{1749,4}{1 \cdot 1,2 \cdot (21,4 - 15)} = 227,8 \frac{\text{м}^3}{\text{сағ}}$$

Ауа шығының қамсыздандыру үшін $L=227,8 \text{ м}^3/\text{сағ}$, бізге тек бір кондиционер жеткілікті, фирмасы VESTEL Trend Plus W 9000 CH ауаның максималды шығыны $230 \text{ м}^3/\text{сағ}$, моделі R22



Модельдің қысқаша характеристикасы:

Түрі	индустриялық моноблукты кондиционер
Номиналды суық өндірілімдігі Вт	2350
Номиналды қажет ететін қуаты Вт	9400
Ауаның шығыны (min-max) м ³ /сағ	200-230

Бұл ақиқаттыққа сәйкес келеді және жайлы микроклиматтың қамсыздандыруы үшін жеткілікті болып табылады.

4.3 Тіршілік қауіпсіздігі бөлімі бойынша қорытынды

Өрт болған жағдайда тек бөлме ғана зиян шекпейді, сонымен бірге қымбат тұратын аппараттар, адам өлімі болуы мүмкін. Сондықтан өртті ертерек анықтап және сөндіру бойынша шара қолдану қажет. Бөлмені қорғау үшін дренчерлік қондырғы негізіндегі автоматтық өрт сөндіргішті таңдаймыз. Өрт кезінде адамдарды эвакуациялау мүмкіндігін тез арада қарастыру қажет.

Есептеулер бойынша жалпы ғимараттан адамдарды эвакуациялау уақыты 0,5 минутты құрады. Адамдарды белгілі бір мекемеден қауіпсіз эвакуациялау үшін есептелген эвакуациялау уақыты t_e қажетті эвакуациялау уақытынан t_k кіші болуы керек, яғни $t_e < t_k$. Қажетті эвакуациялау уақыты ғимараттағы қабат санына және ондағы адам санына байланысты арнайы белгіленген.

Мысал бойынша есептелген есіктік өткелдің қажетті ені – 0,92 м ге тең болды, негізінде есіктік өткелдің минималды ені 0,8 м – ді құрауы тиіс және эвакуацияланатын адам саны $N_1 = 180$ адамды құрады. Егер дәлізден сыртқа шығатын есік болса немесе басқа горизонтальды дәліз болса, $q_{\max} = 16,5 \text{ адам}/\text{м}^2$ ға тең болады, онда есіктік өткелдің ені - 1.1м ге тең болды.

Есептеу нәтижесі бойынша , бөлмеге қажетті ауа мөлшерін қамтамасыз ету үшін $L=227,8 \text{ м}^3/\text{сағ}$ байланысты біз тек 1 кондиционер қоямыз.

Модельдің қысқаша характеристикасы:

- Түпаты: индустриялық моноблокты кондиционер;
- Номиналды суық өндірілімдігі Вт 2350;
- Номиналды қажет ететін қуаты Вт 9400;
- Ауаның шығыны (min-max) м³/сағ 200-230;

5 Техничко-экономикалық негізделуі

Smart мобильді web-сайты

5.1 Жоба сипаттамасы

Мобильді құрылғыларға арналған жобаларды техникалық іске асырудың екі бағыты бар: мобильді web сайт және мобильді қосымша.

Мобильді қосымша – бұл белгілі бір мобильді операциялық жүйенің ерекшелігі арқылы жасалған дербес бағдарламалық жабдықтау.

Мобильді web-сайт - мобильді құрылғыда жұмыс істеу және мобильді браузер арқылы көру үшін арналған және бейімделіп құрастырылған сайт. Сайт JavaScript, HTML5, жаңа API браузерлерді қолданатын интерактивті компоненттерден тұрады. Бұл жағдайда осылай жүзеге асыруды мобильді web-сайт деп атайды.

Мобильді веб-сайттың артықшылықтары

Көптеген компаниялар мен ұйымдар мобильді веб-сайттарды кез-келген мақсатын мобильді құрылғыларда іске асыруды әдетке айналдырып келе жатқан тұтынушылардың сұраныстарын қанағаттандыру үшін арнайы жасады.

Мобильді веб-сайттар

- Әр түрлі мобильді құрылғыларда жұмыс істейді;
- Оңай жаңартылады;
- Іздеу мүмкіндігі бар;
- Веб-өңдеу дағдыларын қолдануға мүмкіндік береді;
- Мобильді тұтынушылардың қажеттілігіне арнайы жасалған ыңғайлы функционал және навигация;
- Бөлек басқару;
- Жылдам жүктеу.

1 Тұтас сегмент пен позициялау

Тұтас сегмент – бұл қазіргі техникалық прогресстің мүмкіндігін толық пайдаланып жұмыс істеуді қажет ететін телекоммуникация жүйесі.

«**Мобильді web-сайт**» – Клиенттің мүмкіндігін арттыратын мобильді бағдарламалық жабдықтау.

2 Қызметті SWOT-талдау

<i>Артықшылықтары</i>	<i>Кемшіліктері</i>
-----------------------	---------------------

<p>Қолдануға ыңғайлы:</p> <ul style="list-style-type: none"> • Үйде диванда отырып; • Серуендеп жүріп; • Кезекте тұрғанда; • Автобус, поезд немесе ұшақ күтіп тұрғанда; • Саяхаттау кезінде. 	<ul style="list-style-type: none"> • Бір уақытта бірнеше сайттың ашылу мүмкіндігі жоқтығы; • Сайттың негізгі нұсқасынан айырмашылық болуы.
<i>Мүмкіншілігі</i>	<i>Қайіні</i>
Қосымша табыс көзі, клиенттер санының артуы	Бағдарламалық жабдықтаудың рұқсатсыз таратылуы мүмкін

3 Маркетинг-микс қызметтері

Маркетинг-микс – Маркетингтік құралдардың жиынтығы, бұл арқылы компания тауарлар немесе қызметтер сұранысына ықпал етуге тырысады.

Маркетинг-микс – бұл белгілі бір тауарға анықталған баға бойынша белгілі бір орында қозғалыс арқылы сұранысты қалыптастыратын жиындардың (стратегия) жүйесі.

Айта кету керек, стратегия тек компанияның белгілі бір мақсатта нарықтың кез келген сегментінде қозғалуына арналған ұмтылысы және стратегия жалпы, бөлінбейтін сипатқа ие. Жағдайды басқару мен оны назардан тыс шығармау үшін тек нақты орындалу уақыты анықталған жиындарды іске асыру керек, осы жиындарға жауапты адамдарды, қажетті ресурстарды (қаржылық, адами және т.б.) басқару қажет. іске асыру қажет. Міне сондықтан таңдалған стратегия үшін маркетинг-микс деп аталатын арнайы жиындар іске асырылады. Осы маркетингтің тактикалық деңгейі болып табылады.

Ереже бойынша, стандартты маркетинг-микс жиындарды төрт құраушы тактикалық іс бойынша сипаттайды: Маркетинг жиынтығы (marketing-mix) – бұл фирма біртұтас нарықтың оң жауабын алу үшін қолданатын басқаруға мүмкіндік беретін маркетингтік факторлардың жиыны. Маркетинг-микстің міндеті – біртұтас нарықта потенциалды тұтынушылардың қажеттерін өтеумен ғана шектелмейтін, сонымен қоса ұйымның әсерін жоғарлатуға тырысатын жиынтықтарды (mix) құрастыру.

Продукт (Product) — бұл фирманың тұтас нарыққа ұсынатын «бұйымдар мен тауарлар» жиынтығы.

Баға (Price) — тұтынушылардың тауарды алу үшін төлеуі қажет ақша суммасы.

Дистрибуция (Place) — тауар барлық бірыңғай тұтынушыларға қолжетімді болу үшін жүзеге асырылатын мүмкіндігі көп іс.

Қозғалу, даму (Promotion) — фирманың өзінің тауарлары туралы ақпарат таратуы мен тұтынушыларды осы тауарларды сатып алуға үгіттейтін мүмкіндігі зор іс-шаралар.

Маркетинг-микс элементтері	Қажетті істер мен талдаулар
Product	
Мобильді құрылғыларға арналған web-сайт қызметі	<p>Жалпы тапсырманы шешу келесі бөлімдерден тұрады:</p> <p>1) Клиент тауарлар мен қызметтердің түрін, бағасын дербес компьютерге қосылу мүмкіншілігі жоқ болса да мобильді web-сайтты ашу арқылы көріп, таныса алады.</p> <p>2) Клиент арнайы web-сайтқа кіріп, сол жерден өзінің есептік шотын енгізу арқылы өзі таңдап алынған тауарды немесе қызмет түрін тікелей сатып алу мүмкіншілігіне ие болады. Егер web-сайтқа уақыт өте келе сұраныс артатын болса, онда web-қосымша жасалынады.</p>
Price	
Базалық бағасы	Жарнамалық агенттіктің тауарлар мен қызметтер бағасы мезгілінен бастап іске асатын «Прейскурант №1» тарифы арқылы жүргізіледі.
Promotion	
	Алғашқы уақытта әлеуметтік желілер арқылы жарнамалау, яғни әлеуметтік желілерде өз топтарымызды немесе беттерді ашу арқылы жарнамалау. Арнайы сайттарда хабарландыру беріп, акциялар жариялау.
Place	
Ұйым 1, ұйым 2, ұйым 3	Алматы қаласындағы жарнама қызметі мен тауарларын қажет ететін мекемелердегі арнайы қызметкерлерге хабарландыру, келіссөз жүргізу.

5.2 Бағдарламалық қамтаманың өнделуінің еңбек өнімділігінің есептелуі

Еңбекке кеткен шығынның базалық көрсеткіштері мына формула бойынша есептеледі:

$$Q=q*c$$

(1)

Мұндағы q - бағдарламалық өнімдегі (бастапқы команда) операторлар саны, менің жұмысымда операторлар саны 3000-ге тең, атап өткен q коэффициентінің мәнін 1-кестеден (бітіру жұмысының экономика бөлімін орындауға әдістемелік нұсқаулардан алынды Алматы: АЭЖБУ, 2014 – 27 б., 5-ші бет) тандап алуға болады.; c -бағдарлама күрделілігінің коэффициенті, ол менде 1,9 ге тең, себебі менің жобамның бағдарламалық тілі жоғарғы тілге жатады және күрделілік тобы үшінші топ, ал жаңалық дәрежесі бойынша В тобына,яғни типтік шешім қолданған бағдарламаларды дайындау тобына жатқандықтан осы мәндердің қиылысуынан арнайы кестеден алынды; Осы әдіспен базалық көрсеткішті табамыз:

$$Q = 3000 * 1,19 = 3570$$

Ары қарай бағдарламалық өнімді әзірлеуге кететін уақытты есептеу керек. Уақыт адам-сағатпен есептеледі, ал T_d нақты істелген уақытпен алынады, ал қалған кезеңнің уақыты Q командасының шартты санына байланысты есептік жолмен анықталады,

Бағдарламалық өнімін дайындауға кеткен әр кезеңнің уақытын анықтаймыз:

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

$$T_{ПО} = 24 \text{ адам / сағ.}$$

1) T_O (мақсат сипаттамасы уақыты) келесі формуламен анықталады:

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

(2)

$$T_O = 3570 * 1,35 / 50 * 0,8 = 120,48 \text{ адам/сағ}$$

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

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

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

$$T_A = Q / (50 \times K). \quad (3)$$

$$T_A = 3570 / (50 * 0,8) = 89 \text{ адам/сағ}$$

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

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

$$T_H = Q \times 1,35 / (50 \times K). \quad (4)$$

$$T_H = 3570 * 1,35 / (50 * 0,8) = 120,48 \text{ адам/сағ}$$

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

$$T_{II} = Q / 50 = 71 \text{ адам/сағ} \quad (5)$$

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

$$T_{OT} = Q \times 4,2 / 50 \times K = 3570 * 4,2 / 50 * 0,8 = 374,85 \text{ адам/сағ} \quad (6)$$

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

$$T_D = 24 \text{ адам / сағ.}$$

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

$$T = T_{II} + T_{TO} + T_A + T_{BC} + T_H + T_{II} + T_{OT} + T_D. \quad (7)$$

$$T = 24 + 120,48 + 89 + 89 + 120,48 + 71 + 374,85 + 24 = 912 \text{ адам/сағ}$$

5.3 Бағдарламалық қамтаманы өңдеуге кеткен шығынның есептелуі

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

Негізгі еңбек құны (8):

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

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

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

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

$$TC - \text{тарифтік қойылым} = 19966 / 8 = 2495,5 \text{ ке тең болды.}$$

t_{Σ} – (7) формуласы бойынша 912–ге тең.
Осылайша негізгі еңбек ақы:

$$Z_{\text{осн}}=912*2495,75/21=108386,8 \text{ теңге}$$

Қосымша еңбек ақы негізгі еңбек ақының 20%-ын құрайды. (9)
формуласы бойынша анықталады:

$$Z_{\text{доп}}=0,2*Z_{\text{осн}}=21677,3 \text{ теңге} \quad (9)$$

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

$$\text{ФОТ}=Z_{\text{доп}}+Z_{\text{осн}}=21677,3+108386,8=129998,1 \text{ теңге} \quad (10)$$

3 Қосымша шығындар статьясы

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

Қондырғының құны бағдарламалық қамтама өңделуінің өзіндік құнына кірмегенімен, қосымша шығындар статьясындағы есептеулерде қолданылады. ЭЕМ - на бағдарлама жазу кезінде құрылғы ретінде дербес компьютер қолданылады. Құны $C_{\text{обор}}=195000$ теңге (дербес компьютердің қазіргі уақыттағы нарықтағы теңгемен берілген құны алынды)

Материалдар мен көмекші бөлшектер шығыны, бағдарламалық өнімді жазу барысында қолданылды ($C_{\text{Мжк}}$), сонымен қатар техникалық қызмет көрсету шығыны ($C_{\text{ТО}}$), жабдықтың құнынан 1.5% және 2.5% (бұл мәндер бітіру жұмысының экономика бөлімін орындауға әдістемелік нұсқаулардан алынды Алматы: АЭЖБУ, 2014 – 27 б., 10-шы бет) құрайды және мына формулалар мен есептеледі (11):

$$\begin{aligned} C_{\text{Мжк}} &= 0,015 \times C_{\text{обор}} = 0,015 * 195000 = 2925 \text{ теңге} \\ C_{\text{ТО}} &= 0,025 \times C_{\text{обор}} = 0,025 * 195000 = 4875 \text{ теңге} \end{aligned} \quad (11)$$

Амортизациялық аударым өнімнің физикалық және моральдық жарамсыздығына байланысты сол өнімді толық қалпына келтіру үшін жиналатын ақша қоры болып есептеледі. Амортизациялық аударым амортизация нормасы бойынша орнатылады, (12) формуласы бойынша есептеледі:

$$A = \frac{C_{\text{обор}} \times H_A \times N}{100 \times 12 \times t}$$

(12)

Мұндағы $C_{обор}$ – компьютер құны; H_A – амортизация нормасы, N – жұмыс орындалуына кеткен күннің саны;
 t – дербес компьютерді қолдануға кеткен жалпы уақыт.

$$N = T/24 = 912/24 = 38 \text{ күн}$$

бұл жерде T еңбек шығынына тең, оның мәні – 912 адам/сағ тең болған.

Дербес компьютерде жалпы жұмыс істеу уақыты мына формуламен есептеледі:

$$t = T_A + T_{BC} + T_H + T_{II} + T_{OT} = 89 + 89 + 120,48 + 71 + 374,85 = 744,33 \text{ адам/ сағ}$$

Амортизация нормасы (13) формуласы бойынша анықталады:

$$H_A = \frac{C_{обор} - C_{ликв}}{T_{норм} \cdot C_{обор}} \cdot 100\%, \quad (13)$$

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

$$C_{ликв} = 0,05 \cdot C_{обор} = 0,05 \cdot 195000 = 9750 \text{ теңге} \quad (14)$$

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

$$H_A = \frac{C_{обор} - C_{ликв}}{T_{норм} \cdot C_{обор}} \cdot 100\% = \frac{195000 - 9750}{4 \cdot 195000} \cdot 100\% = 23,75\%$$

$$A = \frac{C_{обор} \times H_A \times N}{100 \times 12 \times t} = 195000 \cdot 23,75 \cdot 38 / 100 \cdot 12 \cdot 744,33 = 197,03 \text{ теңге}$$

Бағдарламалық қамтаманы жазу кезіндегі электроэнергияның шығынын қоса есептеу керек. Электроэнергияның құны мына формула бойынша есептеледі (15):

$$C_{ЭЭ} = M * k_3 * F_{эф} * C_{кВт-ч} \quad (15)$$

Мұндағы M – ЭВМ қуаты (450 Вт); k_3 – жүктеу коэффициенті (0.8); $C_{кВт.ч}$ – 1 кВт-сағ электроэнергияның құны, оның мәні ҚР – да 14,65 теңгеге тең;

$F_{эф}$ – жұмыс уақытының эффективті фонды, мына формула бойынша есептеледі (16):

$$F_{эф} = D_{ном} * d * (1 - f/100) = 258 * 8 * (1 - 2,6/100) = 2010 \quad (16)$$

Мұндағы $D_{ном} = 258$ – бір жылдағы жұмыс күнінің номиналды саны; $d = 8$ – жұмыс күнінің ұзақтығы [сағ]; $f = 2.6\%$ – ЭВМ –ды жөндеуге кеткен жоспарланған уақыт.

Берілген коэффициенттер мен параметрлер бойынша электроэнергияның құны:

$$C_{ЭЭ} = 0,45 * 0,8 * 2010 * 14,65 = 10601 \text{ теңге}$$

Дегенмен, алынған амортизациялық аударым мен электроэнергияға шығын – жылдық шығынның мәні, оны жалпы жылдық эксплуатациондық шығынмен анықталатын уақыт коэффициентімен коррекциялау (дұрыстау) керек. (17):

$$\mathcal{E}_3 = t_{\Sigma'} * C_{Э} / F_{эф}, \text{ теңге} \quad (17)$$

Мұндағы $C_{Э} = C_{ЭЭ} + C_{ТО} + A_{жыл}$ – ЭЕМ-ның жалпы жылдық эксплуатационды шығын құны:

$$C_{Э} = C_{ЭЭ} + C_{ТО} + A_{жыл} = 10601 + 4875 + 197,03 = 15673 \text{ теңге}$$

Мұндағы $F_{эф}$ – жұмыс уақытының эффективті фонды, (16) формуласы бойынша анықталды, – $t_{\Sigma'}$ – ЭВМ-нің (7) формуласы бойынша анықталған есепті шешу үшін жалпы қолданылған уақыты, бірақ бұл жерде тек компьютердегі жұмыс уақытын ескере отырып анықталған:

$$t_{\Sigma'} = t_{\text{пр}} + t_{\text{отл}} + t_{\text{д}} = 89 + 374,85 + 24 = 487 \text{ сағ}$$

Сәйкесінше, ЭЕМ-ның жалпы жылдық эксплуатационды шығын құны:

$$\mathcal{E}_3 = 487 * 15673 / 2010 = 3797,3 \text{ теңге}$$

ал уақыт коэффициенті мына формула бойынша есептеледі (18):

$$w = \frac{\mathcal{E}_3}{C_3} = \frac{3797,3}{15673} = 0,24 \quad (18)$$

Осылайша, жалпы эксплуатационды шығынды уақыт коэффициентін ескеріп коррекциялаймыз (дұрыстаймыз):

– Электроэнергияға шығын:

$$C_{\text{ЭЭ}}^* = C_{\text{ЭЭ}} \cdot w = 10601 * 0,24 = 2544,2 \text{ теңге};$$

– Амортизационды аударым:

$$A_{\text{жыл}}^* = A_{\text{жыл}} \cdot w = 197,03 * 0,24 = 47,28 \text{ теңге}$$

Сонымен қатар, формуласы бойынша есептелетін жұмыс жалақысына тәуелді шығындар бар. Осындай шығындар қатарына социалды сақтандыруға кететін аударымдар мен жүктеме аударымдар кіреді. Әлеуметтік салыққа кететін аударымдар бүкіл жалақының 11% құрайды [11%, ҚР ның Салық Кодексі], (19) формуласы бойынша анықталады:

$$CC = (\text{ФОТ} - 0,1 * \text{ФОТ}) * 0,11 = (129998,1 - 0,1 * 129998,1) * 0,11 = 12869,8 \text{ теңге} \quad (19)$$

Жүктеме шығындар қызмет көрсету мен басқаруға қатысты, құрылғының эксплуатациясы мен басқа да өндіріс процесін қамтамасыз ететін қосымша шығындар болып табылады, еңбек жалақысы фондының 50% құрайды, (20) формуласы бойынша анықталады:

$$C_{\text{накл}} = 0,5 * \text{ФОТ} = 0,5 * 129998,1 = 64999,05 \text{ теңге} \quad (20)$$

4 Өзіндік күн нәтижесінің кестесі

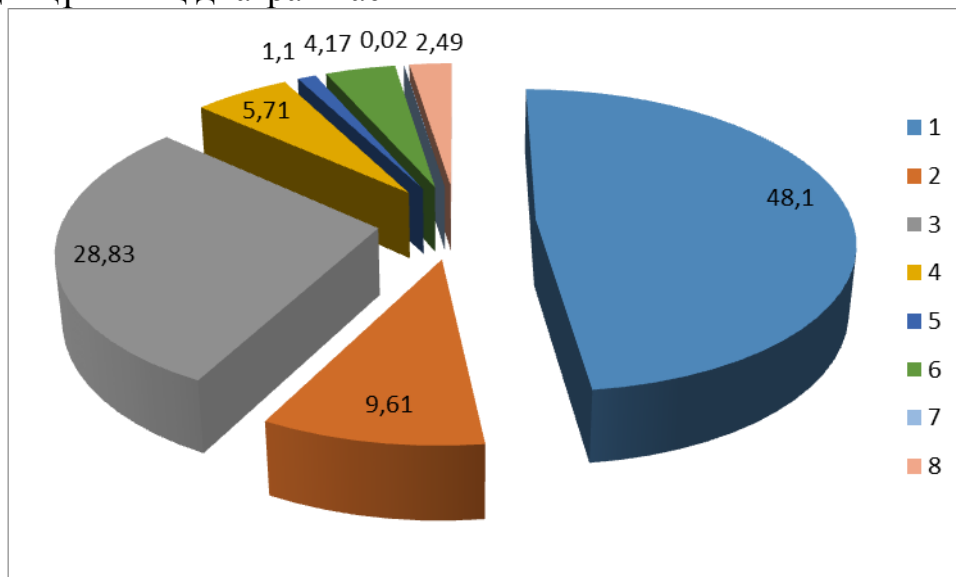
Бағдарламалық қамтаманы өңдеуге кеткен жалпы шығын еңбек жалақысы фондының, эксплуатационды шығынның, социалды сақтандыру, жүктеме шығындар мен материал және құрамалардың қосындысына тең болады.

Бағдарламалық қамтаманы өңдеудің қорытынды құны 1 кестеде көрсетілген.

Кесте 1 – Өзіндік құн нәтижесінің кестесі

Шығын статьялары		Сумма, теңге	Жалпы суммадан процентпен
ФЗП	Z_o	10838	48,1
	$Z_{он}$	6,8	
	$Z_{оп}$	21677,3	9,61
Жүктеме шығындар, $C_{накл}$		64999,05	28,83
Әлеуметтік сақтандыру, CC		12869,8	5,71
Эксплуатационды Шығындар	$C_{ээ}^*$	2544,2	1,1
	$C_{то}$	4875	4,17
	$A_{жыл}^*$	47,28	0,02
Материалдар мен комплектілер, $C_{мик}$		2925	2,49
Барлығы:		218352,15	100

Өзіндік құнының диаграммасы



Сурет 1 – Өнімнің өзіндік құнының диаграммасы

3.5 Бағдарламаның бағасына есептеу

Өнімнің өзіндік құны 218352,15 теңге (жоғарыда кесте 1) болды, онда бір енгізу объектісіне арналған жүйенің толық өзіндік құны ($C_{\text{пс}}$):

$$C_{\text{пс}} = 218352,15 / 4 = 54588,03 \text{ теңге}$$

3.6 Минималды бағаны, кірісті және пайданы есептеу

Өнімнің минималды бағасы келесі формуламен есептеледі:

$$C_{\text{min}} = (C_{\text{пс}} + C_{\text{мд}} + C_{\text{з}})(1 + r/100) = (54588,03 + 190 + 140) * 1,21 = 66450,7 \text{ теңге (21)}$$

Мұндағы $C_{\text{мд}}$ - магнитті диск бағасы, қазіргі нарықтық бағасын қарастырып, 190 теңге деп алынды;

$C_{\text{з}}$ – жазба бағасы;

r - Өзіндік құнға қатынасымен алынған тиімділіктің жоспарланған пайызы, ол $r = 21\%$;

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

$$C_{\text{min}_{\text{НДС}}} = C_{\text{min}} + N_{\text{ндс}} \quad C_{\text{min}} = 66450,7 + 0,12 * 66450,7 = 74424,78 \text{ теңге (22)}$$

Кіріс келесі формуламен есептеледі:

$$V_{\text{реал}} = C_{\text{min}_{\text{НДС}}} * V_{\text{объем}} = 74424,78 * 4 = 297699,9 \text{ теңге (23)}$$

Мұндағы $V_{\text{объем}}$ – бағдарламаны жүзеге асыру көлемі;

Жалпы пайда келесі формуламен есептеледі:

$$\Pi = V_{\text{реал}} - V_{\text{реал}} * N_{\text{ндс}} - C_{\text{разраб}} = 297699,9 - 35723,8 - 218352,15 = 43623,9 \text{ теңге (24)}$$

Мұндағы НДС – қосылған құн салығы (12%).

Таза пайда келесі формуламен есептеледі:

$$\Pi_{\text{чист}} = \Pi - N_{\text{прибыль}} = 43623,9 - 0,2 * 43623,9 = 34899,2 \text{ теңге (25)}$$

Мұндағы $H_{\text{прибыль}}$ – пайдаға салынатын салық (ҚР да $H_{\text{прибыль}}=20\%$)

3.6 Бағдарламалық өнімді иемденудегі кәсіпорындардың бірмезгілдегі шығындарын есептеу

Кәсіпорындардың бірмезгілдегі шығындары келесі бөлімдерден тұрады:

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

Жүйе құны 74424,78 теңге (22 формуласы).

Көлік шығындары жүйе құнының 20% құрайды:

$$74424,78 * 0,2 = 14884,9 \text{ теңге}$$

Жұмысшыларды жүйемен жұмыс істеуге оқыту шамамен екі сағатта консалтингті фирманың маманы көмегімен сағатына 2500-3000 теңгедей болып табылады (бұл мән бітіру жұмысының экономика бөлімін орындауға әдістемелік нұсқаулардан алынды, Алматы: АЭЖБУ, 2014 – 27 б., 11 бет).

Соған сәйкес оқу бағасы:

$$2500 * 2 = 5000 \text{ теңге}$$

Кесте 2 де кәсіпорынның бірмезгілдегі шығындарын есептеулерінің нәтижесі көрсетілген.

Кесте 2 – Ақпаратты жүйені енгізуге арналған бірмезгілдегі шығындардың есептелу нәтижесі

Шығындар түрлері	Сумма, теңге
Жүйе құны	74424,78
Көлік шығындары	14884,9
Жұмысшыларды оқыту бағасы	5000
Қорытынды:	94309,6

3.7 Қолдану аясындағы жылдық бірмезгілдегі шығындарды есептеу

Негізгі шығындар түрлері:

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

Есептеу жылына бір рет ПК операторымен 2500 теңге/сағ (жоғарыда берілген) сағаттық мөлшерде бір сағат көлемінде жүргізіледі.

$$2500 * 1 = 2500 \text{ теңге}$$

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

$$(2500-2500*0,1)*0,11=247,5 \text{ теңге}$$

Электроэнергияға кететін шығындар:

$$0,8*14,65= 11,72 \text{ теңге}$$

мұндағы 0,8 – жұмсалған қуат мөлшері, кВт-сағ;

14,65 – ҚР 1 кВт-с электроэнергияның орташа құны, теңге.

Есептеулерді 3 кестеге енгіземіз.

Кесте 3 – Кәсіпорынның жылдық бірмезгілдегі шығындарын есептеу нәтижесі

Шығындар түрлері	Сумма, теңге
Еңбекке төленетін қор	2500
Әлеуметтік салыққа аударымдар	247,5
Электроэнергияға төлем	11,72
Қорытынды:	2759,2

3.8 Ақпараттық өнімді енгізуден алынған үнемдеу мөлшері мен табысты есептеу

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

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

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

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

Қол еңбегімен келесі шығындар түрлері есептеледі:

- жұмысшы еңбегіне төленетін ақша қоры;
- әлеуметтік салық;

- басқа да шығындар.
- Қолмен есептеу барысында еңбек өнімділігін талдау үшін 1500 теңге/сағ (ұйымның қойып отырған талабы) мөлшерінде жұмыс істейтін мамандандырылған инженер қажет. Өткізу қабілетін есептеу үшін инженер 8 күн жұмсайды. Қолмен есептеу барысында еңбектің жылдық төлем қоры:

$$8 \cdot 8 \cdot 1500 = 96000 \text{ теңге}$$

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

$$(96000 - 96000 \cdot 0,1) \cdot 0,11 = 9504 \text{ теңге}$$

Кесте 4 – Ақпараттық жүйені қолданбай еңбек өнімділігін талдауға кететін шығындарды есептеу

Шығындар түрлері	Сумма, теңге
Еңбекке төленетін қор	96000
Әлеуметтік салыққа аударымдар	9504
Қорытынды:	105504

Ақпараттық жүйені қолданғандағы жылдық шығындар жоғарыда есептелді және 2759,2 теңге болды.

Осыған сәйкес, шығындардың шартты үнемделуі мынаған тең болады:

$$105504 - 2759,2 = 102744,8 \text{ теңге}$$

3.9 Пайда мен шығындарды есептеу

Пайда мен шығындарды есептеу кесте 5 те көрсетілген.

Кесте 5 - Бір жылдағы пайда мен шығындарды есептеу

Аты	Жыл
Шығындардың шартты үнемделуі, теңге	102744,8
Ағымдағы шығындар, теңге	2759,2
Шартты пайда, теңге	43623,9
Пайдаға салынатын салық, теңге	4820,28
Таза пайда, теңге	34899,2

3.10 Ақшалай құралдардың қозғалысы

Кесте 6 – Ақшалай тәсілдердің қозғалысы,
теңге

Аты	Жылдар					Барл ығы
	201 4	201 5	201 6	201 7	201 8	
Бірмезгіл дік шығындар	943 09,6					9430 9,6
Операция лы кәсіп		348 99,2	348 99,2	348 99,2	348 99,2	1395 96,8
Шығын дарды үнемдеу		102 744,8	102 744,8	102 744,8	102 744,8	4109 79,2
Ағымдағ ы шығындар		275 9,2	275 9,2	275 9,2	275 9,2	1103 6,8
Пайдаға салынатын салық		482 0,28	482 0,28	482 0,28	482 0,28	1928 1,12
Кіріс және шығыс сальдосы		348 99,2	348 99,2	348 99,2	348 99,2	Іске аса алатын жоба
Таза табыс	- 94309,6	348 99,2	348 99,2	348 99,2	348 99,2	4529 0,2
Дисконтт ау коэффициенті (10% мөлшерінде)	1	0,9 09	0,8 26	0,7 51	0,68 3	
Таза дисконтталған табыс (ТДТ)	- 94309,6	317 23,3	288 26,73	262 09,29	238 36	1628 5,72
ТДТ өспелі нәтижесімен	- 94309,6	- 35568,3	- 19642,1	- 5161,98	800 7,02	- 105460,2

$$K_{д2} = 1/(1+\alpha)^1 = 1/(1+0,1)^1 = 0.909$$

$$K_{д3} = 1/(1+\alpha)^2 = 1/(1+0,1)^2 = 0.826$$

$$K_{д4} = 1/(1+\alpha)^3 = 1/(1+0,1)^3 = 0.751$$

$$K_{д5} = 1/(1+\alpha)^4 = 1/(1+0,1)^4 = 0.683$$

Мұндағы α – инфляция коэффициенті, $K_{д}$ - дисконттау коэффициенті

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

Таза дисконтталған табыс (ТДТ) = 34899,2*0.909=31723,3 теңге

$$34899,2*0,826 = 28826,73 \text{ теңге}$$

$$34899,2*0,751 = 26209,29 \text{ теңге}$$

$$34899,2*0,683 = 23836 \text{ теңге}$$

ТДТ₂ (өспелі нәтижесімен) = ТДТ₁ + Таза дисконтталған табыс (ТДТ)₂

3.11 Экономикалық тиімділікті есептеу

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

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

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

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

NPV (10%) = $\left(\frac{34899,2}{(1+0,1)^1} + \frac{34899,2}{(1+0,1)^2} + \frac{34899,2}{(1+0,1)^3} + \frac{34899,2}{(1+0,1)^4} \right) - 53094,8 = 8007,05$ теңге > 0
 сондықтан ұсынылып отырған жоба табысты. Сол себептен жобаны қажет және жобаны талдау мен оған баға беруді жалғастыру керек.

3.11.2 Пайда индексін есептеу (Profitability index, PI)

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

$$PI = \frac{NPV}{C_0}$$

$$PI = (8007,05/218352,15)*100\% = 0.036*100\% = 3,6\%$$

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

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

$IRR = r$, мұндағы $NPV = 0$.

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

$r_b = 15\%$:

$$PV_1 = 34899,2 / (1 + 0,15) = 30347,1 \text{ теңге}$$

$$PV_2 = 34899,2 / (1 + 0,15)^2 = 26438,7 \text{ теңге}$$

$$PV_3 = 34899,2 / (1 + 0,15)^3 = 22960 \text{ теңге}$$

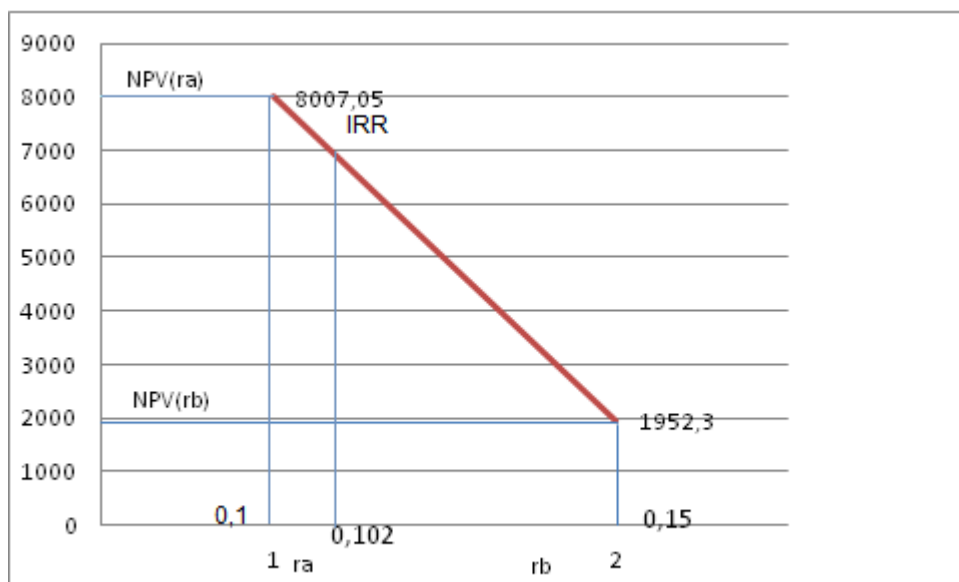
$$PV_4 = 34899,2 / (1 + 0,15)^4 = 19965,21 \text{ теңге}$$

$$NPV(15,0\%) = (30347,1 + 26438,7 + 22960 + 19965,21) - 94309,6 = 1952,3 \text{ теңге}$$

$$NPV(10\%) = 8007,05 \text{ теңге}$$

$$NPV(15,0\%) = 1952,3 \text{ теңге}$$

$$IRR = r_a + (r_b - r_a) * NPV_a / (NPV_a - NPV_b) = 10 + (15 - 10) * 8007,05 / (8007,05 - 1952,3) = 10,2\%$$



3.11.4 Өтімділік периодын есептеу (Payback period, PBP)

Өтімділік периоды мына формуламен есептеледі:

$$T_{\text{ок}} = t + \frac{C_0 - \sum_{i=1}^n B_i}{B_{i+1}}$$

Мұндағы:

C_0 – жоба басындағы салым;

B_i – i -ші жылдағы жобадан алынатын пайда;

Сонда біздің өтімділік периодымыз:

$$T_{\text{ок}} = 2 + \frac{94309,6 - 31723,3}{28826,73} = 4,1 = 4 \text{ жыл } 1 \text{ ай}$$

Бұл жобаның өтімділік периоды 4 жыл 1 айға тең, яғни жобамыз осы уақытта өз-өзін ақтайды.

А қосымшасы

А.1 к е с т е – q коэффициентінің мәні

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

Комплекстік тапсырма	5000 ден 5500
----------------------	---------------

А.2 к е с т е – Еңбек сыйымдылығын есептейтін коэффициент

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

А.3 к е с т е – Бағдарлама жасаушы білектілігін ескеретін коэффициент

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

NPV мәніне байланысты жоба бойынша есептеу нұсқалары

Көрсеткіш мәнінің нұсқалары	Жоба сипаттамасы	Шешімдер нұсқасы
NPV < 0	Құрдымға кететін жоба	Жобадан бас тарту керек немесе жобаның мәндерін өзгертіп, қайта есептеу жүргізу керек.
NPV = 0	Құрдымға кету үстіндегі жоба	Жобаны қабылдау қажет және жобаны талдау мен оған баға беруді жалғастыру керек.
NPV > 0	Табысты жоба	Жобаны қабылдау қажет және жобаны талдау мен оған баға беруді жалғастыру керек.

IRR және r мәндеріне сәйкес жоба бойынша шешім қабылдау нұсқалары

Көрсеткіш мәнінің нұсқалары	Жоба сипаттамасы	Шешім қабылдау нұсқалары
$IRR < r$	Құрдымға кететін жоба	Жобадан бас тарту керек немесе жобаның мәндерін өзгертіп, қайта есептеу жүргізу керек.
$IRR = r$	Құрдымға кету үстіндегі жоба	Жобаны қабылдау қажет және жобаны талдау мен оған баға беруді жалғастыру керек.
$IRR > r$	Табысты жоба	

ҚОРЫТЫНДЫ

Бүгінгі таңда дүниежүзілік желінің қарқынды дамуымен осы салада күн-сайын көптеген және таңғажайып жаңалықтар ашылуда.

Біз бұл жұмыста дипломдық жұмыстың тақырыбына сәйкес HTML, PHP, MySQL, JS, CSS бағдарламалық жабдықтау ортасы бойынша сайт жасау технологиясын зерттеп, технологияның құрылымдық элементтерімен танысып, қолдану аясына талдау жасадық. Сонымен қатар жоғарыда айтылған программалау тілдерін тереңдете үйрендік және “Smart” жарнама агенттігінің мобилді веб сайты табысты түрде жасап шықтық.

Біздің жасаған мобилді веб сайттың өзіндік ерекшелігі-ол қолданушылардың кез келген смартфон немесе планшеттерден қосыла алғандығында, сонымен қатар сайттан онлайн сатып алу, тапсырыс беру операцияларын орындай алуында.

«Тіршілік қауіпсіздігі» бөлімінде еңбек жағдайларына, жұмыс орнына талдау жасалынды. Жұмыс орнында жасанды жарықтандыру бойынша есептеулер жүргізілген.

Техника – экономикалық негізделуі бөлімінде барлық баптар бойынша жобаны жүзеге асырудағы шығындар анықталды және жобаға экономикалық пайдалылық бағасы берілді.

Қорыта келгенде, PHP тілі мен MySQL дерекқорын қолдану арқылы берілген жұмысты орындау ешбір қиындыққа әкеліп соқтырмады. Себебі сенімді дерекқор мен желідегі ең танымал программалау тілінің өзара қатынасқан жұмысы әрқашанда жақсы нәтижеге жеткізер жұмыстың кепілі.

Пайдаланған әдебиеттер тізімі:

- 1 Сайт www.wikipedia.com
- 2 Прохоров А.Н. Интернет: как это работает. СПб:БХВ –Петербург, 2004. – 280 с.
- 3 Комолова Н., Яковлева Е. HTML: Самоучитель. 2-е изд. СПб.: Питер, 2011. – 288 с.
- 4 Ташков П. Веб мастеринг на 100%: HTML, CSS, JavaScript, PHP, CMS, AJAX, раскрытка. – СПб.: Питер, 2010. – 512 с.
- 5 Котеров Д., Костарев А. PHP. В подлиннике. - СПб.: БХВ-Петербург, 2005. - 1120 с.
- 6 Шафер С. HTML, XHTML и CSS. Библия пользователя. 5 изд. - М.: Диалектика, 2010. - 656 с.
- 7 Шмитт К. Рецепты программирования CSS. Cookbook. - СПб.: БХВ-Петербург, 2007. - 656 с.
- 8 Мамаев Е. MySQL Server. Проектирование и реализация баз данных. Сертификационный экзамен. – СПб.: ВHV, 2004. – 416с.
- 9 Базылов Қ.Б., Алибаева С.А., Нурмагамбетова С.С. Бітіруші жұмысының экономикалық бөлімі үшін әдістемелік нұсқаулар. – Алматы: АИЭС, 2009. – 25 б.
- 10 Иванов И.Н. Экономика промышленного предприятия: Учебник. – М.: ИНФРА–М, 2011. – 395 с.
- 11 Рофе А.И. Экономика труда: Учебник. – М.: КноРус, 2010. – 400 с.
- 12 Остапенко Ю.М. Экономика труда: Учеб. пособие. – М.: ИЦ РИОР, 2010. – 160 с.
- 13 Санатова Т.С., Абдимуратов Ж.С. Өміртіршілігінің қауіпсіздігі. – Алматы: АЭЖБИ, 2008. – 16 б.
- 14 ГОСТ 12.2.032-78 ССБТ. Рабочее место при выполнении работ сидя.
- 15 ГОСТ 21.889-76 ССБТ. Система "человек-машина" Кресло человека – оператора. Общие эргонометрические требования.
- 16 Баклашов Н.И., Китаева Н.Ж., Терехов Б.Д. Охрана труда на предприятиях связи и охрана окружающей среды. – М.: Радио и связь, 1989. –288 с.
- 17 СНиП РК 2.04-05-2002. Естественное и искусственное освещение.

Қосышасы А

```
<!--
=====content=====
=====-->
<section id="content">
  <div class="container_12">
    <div class="pad_box">
      <div class="wrapper">
        <article class="grid_8">

          <h3>Қызмет <span>түрлері</span></h3>
          <ul class="ext_list latest_news">
            <?php foreach($news as $row):?>
              <li>
                <figure><a
href="<?=base_url();?>dornodmsut/view/<?=$row['url'];?>"
class="img_frame">"></a></figure>
                <div>
                  <time class="time" datetime="2013-02-
08"><?=$row['date'];?></time>
                  <h4>
                    <a
href="<?=base_url();?>dornodmsut/view/<?=$row['url'];?>"
                    <?=$row['title'];?>
                  </a>
                </h4>
                <p style="font-family:tahoma;">
                  <?=$row['descr'];?>
                </p>
                <a
href="<?=base_url();?>dornodmsut/view/<?=$row['url'];?>"
class="link4">[Толығырақ]</a>
              </div>
            </li>
          <?php endforeach;?>
        </ul>
```

</article>

А қосымшаның жалғасы

<!--

=====
=====>

<section id="content">

<div class="container_12">

<div class="pad_box">

<div class="wrapper">

<article class="grid_8">

<h3>Бізбен байланыс</h3>

<iframe

id="map_canvas"

src="http://maps.google.com/maps?f=q&source=s_q&hl=en&geocode=&q=Brooklyn,+New+York,+NY,+United+States&aq=0&ll=37.0625,-

95.677068&sspn=61.282355,146.513672&ie=UTF8&hq=&hnear=Brooklyn,+Kings,+New+York&ll=40.649974,-

73.950005&spn=0.01628,0.025663&z=14&iwloc=A&output=embed"></iframe>

<p>Біздің мекен жай </p>

<p>

010005, Алматы қ., Төле би көш., 150, 102-оф

</p>

<dl class="adress">

<dd>Тел./факс: (7272) 37-01-84, 37-91-24 </dd>

<dd>Тел.: (7272) 37-02-97</dd>

<dd>Факс: (7272) 37-91-24 </dd>

<dd>E-mail:

<a

href="#"

class="demo2">smart@mail.ru</dd>

</dl>

</article>

<article class="grid_4">

<div class="box1">

<h3 class="var1 ind">Байланыс</h3>

<form id="contact-form">

<div class="success">Contact form submitted!

We will be in touch soon.

</div>

<fieldset>

<label class="name">

<em class="title">Есіміңіз

```
        <input type="text">
        <span class="error">*This is not a valid name.</span> <span
class="empty">*This field is required.</span>
```

А ҚОСЫМШАНЫҢ ЖАЛҒАСЫ

```
</label>
    <label class="email">
    <em class="title">E-mail</em>
    <input type="text">
    <span class="error">*This is not a valid email
address.</span> <span class="empty">*This field is required.</span>
    </label>
    <label class="phone">
    <em class="title">Құпия сөз</em>
    <input type="text">
    <span class="error">*This is not a valid phone
number.</span> <span class="empty">*This field is required.</span>
    </label>
    <label class="message">
    <em class="title">Мәтін</em>
    <textarea></textarea>
    <span class="error">*The message is too short.</span>
<span class="empty">*This field is required.</span>
    </label>
    <div class="buttons2">
    <a href="#" data-type="reset" class="link5">Жіберу</a>
    <a href="#" data-type="submit" class="link5">Кепі</a>
    </div>
</fieldset>
</form>
</div>
</article>
</div>
</div>
</div>
</section>
<!--
=====footer=====
=====-->
<footer>
    <div class="container_12">
    <div class="pad_box">
    <div class="wrapper">
    <article class="grid_12">
    <div class="footer">
```



```

                                <script                                type="text/javascript"
src="<?=base_url();?>js/jquery-migrate-1.1.1.js"></script>
                                <script                                type="text/javascript"
src="<?=base_url();?>js/superfish.js"></script>

```

А қосымшаның жалғасы

```

                                <script                                type="text/javascript"
src="<?=base_url();?>js/jquery.mobilemenu.js"></script>
                                <script                                type="text/javascript"
src="<?=base_url();?>js/jquery.flexslider.js"></script>
                                <script                                type="text/javascript"
src="<?=base_url();?>js/jquery.easing.1.3.js"></script>
                                <script                                type="text/javascript"
src="<?=base_url();?>js/script.js"></script>
                                <script type="text/javascript">
                                    $(window).load(function() {
                                        $('.flexslider').flexslider({
                                            animation: "slide"
                                        });
                                    });
                                $('#accordion1>li>a').bind('click',function() {
                                    var _this = $(this);
                                    _this.toggleClass('active', 5);
                                    _this.next().toggleClass('closed', 500);
                                    $('#accordion1>li>a').not(_this).each(function() {
                                        $(this).next().addClass('closed', 500);
                                        $(this).removeClass('active', 5);
                                    });
                                });
                                });
                                });
                                </script>

```

```

                                <script type="text/javascript" src="<?=base_url();?>js/jquery.mousewheel-
3.0.4.pack.js"></script>
                                <script type="text/javascript" src="<?=base_url();?>js/jquery.fancybox-
1.3.4.pack.js"></script>
                                <link                                rel="stylesheet"                                type="text/css"
href="<?=base_url();?>css/jquery.fancybox-1.3.4.css" media="screen" />
                                <script type="text/javascript">

```

```

                                    $(document).ready(function() {
                                        $("#a#example8").fancybox({
                                            'titlePosition'      : 'over'
                                        });
                                    });

```

```

                                $("#a[rel=example_group]").fancybox({
                                    'transitionIn'          : 'none',

```

```

'none',
'transitionOut' :
'titlePosition' : 'over',
'titleFormat' :
function(title, currentArray, currentIndex, currentOpts) {
    А қосымшаның жалғасы
    return '<span id="fancybox-title-
over">' + title + '</span>';
}
});
});
</script>
</head>

```

```

<h3><span>жаңалықтар</span></h3>
<ul class="ext_list latest_news">
    <?php foreach($lastnews as
$row):?>
        <li>
            <figure><a
href="<?=base_url();?>dornodmsut/view/<?=$row['url'];?>"
class="img_frame">"></a></figure>
            <div>
                <time class="time" datetime="2013-02-
08"><?=$row['date'];?></time>
                <h4>
                    <a
href="<?=base_url();?>dornodmsut/view/<?=$row['url'];?>"
                        <?=$row['title'];?>
                    </a>
                </h4>
                <p style="font-family:tahoma;">
                    <?=$row['descr'];?>
                </p>
                <a
href="<?=base_url();?>dornodmsut/view/<?=$row['url'];?>"
class="link4">[Толығырақ]</a>
            </div>
        </li>
    <?php endforeach;?>
</ul>

```

```
                <a                                href="<?=base_url();?>dornodmsut/news"
class="link5">Қызмет түрлері</a>
            </article>
```

```
<body>
```

А қосымшаның жалғасы

```
<div class="main">
<!--
=====header=====
=====-->
    <header>
        <h1><span style="font-size:22px; font-weight:bold; text-transform:uppercase;
line-height:30px; margin-top:10px;">
            </span>
        </h1>
        <form id="search" action="search.php" method="GET" accept-
charset="utf-8">
            <input type="text" name="s" />
            <a onClick="document.getElementById('search').submit()">Іздеу</a>
        </form>
        <div class="header_links">
            <div class="twitter">
                <a href="#">Twitter</a>
            </div>
        </div>
    </header>
    <div class="white_bg">
        <nav>
            <ul class="sf-menu">
                <li class="current first"><a href="<?=base_url();?>">Басты
бет</a></li>
                <?php foreach($menu as $row):?>
                    <li><a
href="<?=base_url();?>dornodmsut/<?=$row['link'];?>"><?=$row['name'];?></a>
                    <?php
                                $query = $this->db->query("select *
from menus where subid=".$row['id']." order by sort");
                                if($query->num_rows() > 0)
                                {
                                    echo '<ul>';
                                    foreach ($query->result_array()
as $row1){
```

```

        echo ' <li><a
href="'.base_url().'dornodmsut/'.$row1['link'].'">'.$row1['name'].'</a></li>';
    }
    echo '</ul>';
}
?>

```

А қосымшаның жалғасы

```

<?php endforeach;?>
    </li>
</ul>
<div class="clear"></div>
</nav>

<!--
=====content=====
=====-->
<section id="content">
    <div class="container_12">
        <div class="pad_box">
            <div class="wrapper">
                <article class="grid_8">
                    <?php foreach($page as
$row):?>
                        <h3><?=$row['title'];?></h3>
                        <p>
                            <?=$row['body'];?>
                        </p>
                    <?php endforeach;?>
                </article>

                <article class="grid_4 last-col">
                    <div class="box m_bot1">
                        <h2 class="ind">Ереже</h2>
                        <ul class="ext_list feat_list">
                            <?php foreach($rulefirst as
$row):?>
                                <li>
                                    <figure>
                                        <a
href="<?=base_url();?>dornodmsut/rule/<?=$row['url'];?>" class="img_wrap1">
                                            "
                                        </a>
                                    </figure>
                                </li>
                            </ul>
                        </div>
                    </article>
                </div>
            </div>
        </div>
    </div>
</section>

```

```
<div>
    <time datetime="2012-02-
08"><?=$row['date'];?></time><br>
    <?=$row['title'];?> <a
href="<?=base_url();?>dornodmsut/rule/<?=$row['url'];?>"
class="link2">[Толығырақ]</a>
</div>
```

А ҚОСЫМШАНЫҢ ЖАЛҒАСЫ

```
</li>
<?php endforeach;?>
<?php foreach($rulelast as
$row):?>
```

```
<li class="last">
<figure>
    <a
href="<?=base_url();?>dornodmsut/rule/<?=$row['url'];?>" class="img_wrap1">
    "
    </a>
</figure>
```

```
<div>
    <time datetime="2012-02-
08"><?=$row['date'];?></time><br>
    <?=$row['title'];?> <a
href="<?=base_url();?>dornodmsut/rule/<?=$row['url'];?>"
class="link2">[Толығырақ]</a>
</div>
</li>
```

```
<?php endforeach;?>
```

```
</ul>
<a href="#" class="link3">Барлығы</a>
</div>
<div class="box1">
<h3 class="ind">Біз жайлы</h3>
<ul id="accordion1" class="accordion">
    <?php foreach($content as
$row):?>
    <li>
    <a href="javascript:;"><i></i><?=$row['title'];?> </a>
    <div class="accordion_content closed">
    <?=$row['descr'];?> <a
href="<?=base_url();?>dornodmsut/page/<?=$row['url'];?>"
class="link1">[Толығырақ]</a>
    </div>
```


</div>