

ҚАЗАҚСТАН РЕСПУБЛИКАСЫ БІЛІМ ЖӘНЕ ҒЫЛЫМ МИНИСТРЛІГІ

Коммерциялық емес акционерлік қоғамы
АЛМАТЫ ЭНЕРГЕТИКА ЖӘНЕ БАЙЛАНЫС УНИВЕРСИТЕТІ

«Компьютерлік технологиялар» кафедрасы

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

(қолы)

« _____ » _____ 2014 ж.

ДИПЛОМДЫҚ ЖОБА

Тақырыбы: ««Motoshop» ЖШС ақпараттық жүйесін автоматтандыру»
5B070400 – «Есептеу техникасы және бағдарламалық қамтамасыз ету»
мамандығы бойынша

Орындаған Калимбетов Нурдаулет Бахытбекулы тобы: ВТк-10-2

Жетекші п.ғ.к., аға оқытушы Нугманова С.Э.

Кеңесшілер :

Экономикалық бөлім бойынша :

доцент Боканова Г.Ш.

« 29 » 05 2014ж.

(қолы)

Өмір тіршілігі қауіпсіздігі бойынша:

т.ғ.к., аға оқытушы Муташева Г.С.

« 20 » 05 2014 ж.

(қолы)

Есептеу техникасын қолдану бойынша :

п.ғ.к., аға оқытушы Нугманова С.Э.

« 20 » 05 2014 ж.

(қолы)

Мөлшер бақылаушы:

аға оқытушы Рахимжанова З.М.

« 29 » 05 2014 ж.

(қолы)

Пікір жазушы :

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

« _____ » _____ 2014 ж.

(қолы)

Алматы 2014

ҚАЗАҚСТАН РЕСПУБЛИКАСЫ БІЛІМ ЖӘНЕ ҒЫЛЫМ МИНИСТРЛІГІ

Коммерциялық емес акционерлік қоғамы
АЛМАТЫ ЭНЕРГЕТИКА ЖӘНЕ БАЙЛАНЫС УНИВЕРСИТЕТІ

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

жобаны орындауға берілген

ТАПСЫРМА

Студент Калимбетов Нурдаулет Бахытбекулы

Жоба тақырыбы «Motoshop» ЖШС ақпараттық жүйесін автоматтандыру
ректордың « 24 » Қыркүйек № 115 бұйрығы бойынша бекітілген.

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

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

MySQL SERVER дерекқорларды басқару жүйесін
не РНР тілін пайдалану арқылы Motoshop
ЖШС ақпараттық жүйесін автоматтандыру
қарастырылды.

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

Motoshop компаниясының ифрормациялық
жүйесін автоматтандыруға арналған Web-ресми
нұру. жобата қойылған талаптар орындалды:
бағдарламашаға қажетті функцияларды анықтау,
деректер қорын құру.

Сызба материалдарының (міндетті түрде дайындалатын сызуларды көрсету) тізімі:

дерек қордың ЕК-диаграмма ел.
 тасаяған бағдарламаның қосымшаның
 құрылымы істеу қабілеттілігін, функциясын -
 дұрыс көрсететін графикалық материалдар

Негізгі ұсынылатын әдебиеттер:

1. Маммаев Е. MySQL SERVER. Проектирование и реализация баз данных. - СПб.: БХВ, 2004. - 416 с.
2. Полякова Л.И. Основы SQL. Курс лекций: учеб. пособие - Москва, 2004. - 368 с.
3. Плужников С.В., Ломотова Д.В. База данных: Учебный курс. - М.: ООО "Издательство АСТ", 2000 - 504 с.
4. Котеров Ф., Костарев А. РНР. В подпиткине - СПб.: БХВ - Петербург, 2007. - 656 с.
5. Голуцкий О.Л., Основы алгоритмизации, учеб. пособие, 2002. - 432 с.

Жоба тараулары бойынша кеңес берушілер және оның мерзімі:

Бөлім	Кеңесші	Мерзімі	Қолы
Негізгі бөлім	Нугманова С.Э.	20.05.2014 ж.	
Тіршілік қауіпсіздігі	Муташева Г.С.	20.05.2014 ж.	
Экономикалық бөлім	Боканова Г.Ш.	29.05.2014 ж.	
Норма бақылаушы	Рахимжанова З.М.	29.05.2014 ж.	
Есептеу техникасын қолдану	Нугманова С.Э.	20.05.2014 ж.	

Аңдатпа

Жобаның мақсаты «Motoshop» ЖШС ақпараттық жүйесін автоматтандыру болып келеді.

Дипломдық жоба PHP, CSS, HTML және JavaScript тілдерін қолдана отырып құрылған, ал деректер қоры MySQL Server негізінде жүргізілген.

«Тіршілік қауіпсіздігі» бөлімінде еңбек жағдайларына, жұмыс орнына талдау жасалынған. Инженердің жұмыс орнындағы жасанды жарықтандыру бойынша есептеулер жүргізілген.

Ал техника-экономикалық негізделуі бөлімінде барлық баптар бойынша жобаны жүзеге асырудағы шығындар анықталған және жобаның өзіндік құны мен зияткерлік еңбектің бағасы есептелген.

Аннотация

Целью проекта является – автоматизация информационных систем компании ТОО «Motoshop».

Дипломный проект разработан с использованием языков PHP, CSS, HTML и JavaScript, а база данных проведена на основании MySQL Server.

В главе «Безопасности жизнедеятельности» анализируются трудовые условия и рабочее место. На основании полученных данных сделан расчет по искусственному освещению.

В главе «Технико-экономического обоснования проекта» по всем статьям определены расходы по осуществлению проекта, себестоимость и цена на реализацию.

Annotations

This thesis aims to project is - automation of information system of the company LPP «Motoshop».

The diploma project is developed using language PHP, CSS, HTML and JavaScript, and the database is made on the basis MySQL Server.

In connection with life's dangers were analyzed elections fire safety and the establishment of secure systems.

In the economic part of the costs have been calculated on the design of the system, and work to make. It also describes the benefits of working with the completion of the economic side.

Мазмұны

Кіріспе.....	13
1 Автоматтандырылған жұмыс орнының (АЖО) арналуы мен құрылымы, интернеттің рөлі	14
1.1 АЖО құрылымы	15
1.2 АЖО-ға қойылатын жалпы талаптар.....	16
1.3 АЖО-ды жобалау сатылары	17
1.4 Интернет ұғымы	17
1.5 Интернет жетістігі	18
1.6 Интернеттің негізгі принциптері	19
1.7 Қазақстан интернеті	20
1.8 DNS server ұғымы.....	21
2 Web қосымша құру құралдары.....	25
2.1 Интернеттегі бағдарламалау тілдері.....	27
2.2 PHP бағдарламалау тілі.....	28
2.2.1 PHP тілінің артықшылықтары.....	31
2.3 HTML бағдарламалау тілі.....	32
2.4 JavaScript бағдарламалау тілі	34
2.5 CSS бағдарламалау тілі	36
2.6 MySQL дерек қоры.....	38
3 Web - қосымша құруда жобалау және оны іске асыру	41
3.1 Web - қосымша құру кезеңдері	41
3.2 Жобаны іске асыру мерзімі	41
3.3 Реляционды қатынастарды құрастыру	42
3.4 Бағдарлама скриншоттары.....	42
4 Тіршілік қауіпсіздігі	50
4.1 Программалық өнімді жасау кезінде зиянды, потенциалдық қауіпті факторларды талдау.....	50
4.2 Кондиционерлеу жүйесін есептеу.....	52
4.3 Табиғи жарықты есептеу	56
4.4 Тіршілік қауіпсіздігі бөлімі бойынша қорытынды	59
5 Техникалық экономикалық негізделуі	60
5.1 Бизнес жоспар	60
5.2 Бағдарламалық қамтаманың өңделуінің еңбек өнімділігінің есептелуі	62
5.3 Бағдарламаның бағасын есептеу.....	69
5.4 Бағдарлама өнімін сатып алуға кеткен бір жолғы шығындар есебі.	70
5.5 Игеру саласындағы жылдық бір жолғы шығындар есебі.	71
5.6 Ақпараттық жүйе енгізуден үнем мен табыс мөлшерінің есебі.	72
5.7 Ақшалай құралдардың қозғалысы	73
5.8 Экономикалық тиімділікті есептеу.....	74
Қорытынды.....	78
Әдебиеттер тізімі	79
Қосымша А.....	80

Кіріспе

Бұл дипломдық жұмыста “Motoshop” компаниясының администраторлары және жұмысшылары үшін құрылған дерекқор негізінде клиенттік бағдарлама жасалынды. Бұл бағдарламалық шешім компания администраторлары жұмысына көмек көрсету үшін арналған.

Бүгінгі таңда автокөліктер Қазақстан Республикасының өндірістік инфрақұрылымының маңызды бөлшегі болып табылады. Қазақстанның географиялық орналасуы (теңізге шығу мүмкіндігі болмауы, су кемелері жүретін өзендер болмауы), территориясының кеңдігі, өндірістің шикізаттық құрылымы, автокөлік саласының маңыздылығын арттыра түседі.

Веб технологиясы бәріне жетімді технология болғаннан кейін, жер бетінде Интернет пайдаланушылар саны күрт өсті. Бұл, өз алдына, жаңа ресурстарға сұранысты көбейтті. Аз уақыт ішінде Веб-серфингті сүйшілер арасында аудио, видео және анимация ең әйгілі ресурстарға айналды. Сондықтан, бұл жұмысты жасау үшін веб технологиялары қолданылды.

Дерекқор құру үшін алдымен осы саланы талдау қажет. Одан кейін жұмыс процесіндегі деректерді сақтайтын логикалық байланысқан кестелер құрылып, олардың негізінде жүйенің ақпараттық моделі алынады. Жүйенің ақпараттық модель MySQL-жүйесінде деректер базасы жасалады. Клиенттік бағдарламаны жасау PHP(ағылш. PHP: Hypertext Preprocessor — «PHP: еренмәтінді алдын-ала үдірістегіш») тілі арқылы іске асырылады.

Бағдарлама жасау нәтижесінде, компания администраторларының деректермен жұмыс жасаудың тиімділігі, жылдамдығы және ыңғайлылығы артады.

1 Автоматтандырылған жұмыс орнының (АЖО) арналуы мен құрылымы, интернеттің рөлі

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

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

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

АЖО-да шешілетін мәселелерді шартты түрде ақпараттық және есептеуіш деп бөлуге болады.

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

Есептеу мәселелер формальдандырылатын және жартылай формальдандырылатын болады. Формальдандырылатын мәселелер формальды алгоритм негізінде шешіледі және екі топқа бөлінеді: тура есептеу мәселелер мен математикалық модельдер негізіндегі мәселелер. Тура есептеу мәселелер қарапайым алгоритмдер көмегімен шешіледі. Одан күрделі мәселелерді шешу үшін әртүрлі математикалық модельдерді қолдану талап етіледі. Соңғы кезде жартылай формальданатын мәселелерді шешуге арналған сематикалық деп аталатын амалдарды жетілдіруге аса назар аударылады. Мұндай мәселелер жиі

түрде экономикалық объектілерді оперативті басқару барысында, әсіресе толық емес ақпарат жағдайында шешімді қабылдау кезінде туады.

1.1 АЖО құрылымы

АЖО бұл ұйымдастыру, функционалды және қамтамасыз ету бөліктерінен тұратын кешенді жүйе.

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

АЖО-дың функционалды бөлігі бұл оперативті және перспективалық жоспарлау мәселелердің шешімін, техника-экономикалық көрсеткіштердің тіркеуін мен сараптауын қамтамасыз ететін экономика-математикалық тәсілдер кешені. Кешен құрылымы АЖО ның анықталған функцияларын жүзеге асыратын жүйенің біршама дербес ішжүйеден тұрытыны болжамдалады.

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

АЖО-ның қамтамасыз ету бөлігі:

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

АҚЕту АЖО бұл құжаттардың, ақпарат массивтерінің (ДҚ) және кодалаудың стандартталған және бәрыңғай формалар жиынтығы.

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

АЖО-ның техникалық құралдар кешеніне (ТҚК) ақпаратты жинау, жеткізу, өңдеу және сақтау процестердің жүзеге асырылуын қамтамасыз ететін құрылғылар жиынтығы кіреді.

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

- АЖО-ның функционалды мәселелерін шешу;
- АЖО қолданушылар жағынан мәселелердің шешімін ыңғайлы және түсінікті басқару – мәселелер интерфейстері.

1.2 АЖО-ға қойылатын жалпы талаптар

АЖО-ға қойылатын жалпы талаптар келесідей:

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

Жүйелілік. АЖО жұмыс орынның функционалды арналуымен анықталатын құрылымы бар жүйе деп қарастырғаны жөн.

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

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

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

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

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

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

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

Мұндай жүйеде келесі қамтамасыз етілу керек:

- мамандардың ақпаратты өңдеу машина құралдарына максималды жақындығы;
- диалогты тәртіпте жұмыс жасау;
- эргономика талаптарына сәйкестілік;
- компьютердің жоғары икемділігі;

- ескілік процестердің максималды автоматтандырылуы;
- жекелік жағдайда мамандардың творчестволық белсенділігін арттыратын, ал кейін жүйенің дамуына әкелетін мамандардың еңбек жағдайларына моральды түрде қанағат болуы;
- мамандардың өзін-өзі оқыту мүмкіндігі.

1.3 АЖО-ды жобалау сатылары

АЖО-ды дұрыс ұйымдастыру, құрастыру және енгізу мақсатында оның жобалау процесін келесі сатыларға жіктейді:

- Бар басқару жүйесін зерттеу мен сараптама жасау;
- АЖО құруының техникалық тапсырманы құрастыру;
- АЖО-ды техникалық жобалау;
- АЖО-ды жұмыстық жобалау.

1.4 Интернет ұғымы

Интернет (оқылуы [интэрнэт]; 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[1].

Қазіргі кезде Интернет сөзін пайдаланғанда, физикалық желінің өзін емес, Дүниежүзілік желі және ондағы ақпаратты айтамыз. Егер бұл терминді енгізген ағылшын тіліндегі RFC құжатына сүйенсек, онда бұл термин екі түрде жазылып, сәйкесінше екі мағынаға ие болады.

Егер Интернет сөзі кішкентай әріптен басталса, онда бұл термин мәліметтер пакетін маршрутизациялау арқылы желілерді байланыстыру ұғымын білдіреді. Бұл кезде ауқымды ақпараттық кеңістік туралы айтылмайды. Көбінесе, бұл екі түсінікті бір-бірінен ажыратып жатпайды.

Интернет – кез-келген компьютерлермен бүкіл әлем бойынша ақпарат алмасу мен беру мүмкіндігі, желілер жүйесі. Интернет – байланыс араларын өзара біріктіретін, тораптардың жиынтығы.

Әрбір топтарда көбінесе UNIX операциялық жүйесін басқару арқылы жұмыс істейтін бір немесе бірнеше қуатты компьютер-сервер болады. Мұндай торапты кейде хост деп атайды. Торапты оның иесі – провайдер деп аталатын ұйым немесе Интернет қызметін жабдықтаушы басқарады.

Интернет әр түрлі ережемен жұмыс істейтін желілерді біріктіреді. Бұл ережелерді үйлестіру үшін шлюз құрылғысы қызмет етеді. Шлюз – басқаша тәсілмен үйлеспейтін желілерді қосатын құрылғы. Шлюз әр түрлі желілердің бірлескен жұмысын қамтамасыз етуге арналған мәліметтерді өзгертпейді. Интернет желісіне әр түрлі операциялық жүйелерді басқару арқылы жұмыс

істейтін компьютерлер кіреді . Алайда, ақпарат алмасу кезінде барлық ЭЕМ хабар беру тәсілдері туралы бірыңғай келісімдер қолданылуы тиіс. Сонда ЭЕМ-ның қай-қайсысыда басқа кез келген ЭЕМ-нан алынған ақпарат түсінуге қабілетті болады.

Электронды почта немесе e-mail (electronic mail – электронды почта), адамдар арасындағы байланыс тәсілдерінің бірі болып табылады. Электронды почтаның Интернеттегі негізгі функциясы – планетаның қай нүктесінде болса да, Интернеттің кез келген екі пайдаланушысы арасында электронды хаттармен – мәтіндік хабарламамен оперативті және өте тез алмасуды жүзеге асыру. Электронды почтаның қосымша мүмкіндіктеріне мыналарды жатқызуға болады: дыбыстық хабар, құжаттарды, сызуларды, фотосуреттерді, бейнематериалдарды беру; ғылыми журналға, сирек кітаптарға, жарнамаға жету және әр түрлі тауарларды жолдау немесе сату; сонымен қатар, ұжымдық іс-әрекеттерді программалық қамтамасыз ету; мекемелер мен ұйымдарда құжат қолдану; ұжымдық жұмысты жоспарлау. EFT хаттамалары бойынша электронды почтаның жаңаша пакеттері (Electronic Funds Transfer – ақшалай қаражатты электронды аудару) және EDI (Electronic Funds Data Interchange – мәліметтермен электронды алмасу) желі бойынша іс жүзінде қамтамасыз етілетін ақша, шоттар және басқа қаржылық құжаттарды аудару.

Қазіргі уақытта кез-келген информациялық технологияның жаппай техникалық компоненті компьютер болып табылатыны белгілі.

Компьютерлік телекоммуникацияны пайдалану – алыстағы компьютермен диалогтық режимде немесе электрондық почта режимде жұмыс істеуді ұйымдастыруды қамтамасыз етеді. Сондай-ақ, электрондық почта өте қолайлы және аса қымбат емес.

1.5 Интернет жетістігі

Интернет технологисы жылдам өзгеріп отырады. Интернетпен жұмыс істеу оңайлағандықтан, 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.

Адамдарды іздеу. Егер қажетті адамыңыздың қайда екенің білмесеңіз, оны қайда болмаса да сіз қазіргі уақытта тауып алу мүмкіндігіңіз бар. Ол үшін сіз қызмет каталогын немесе телефон кітапшасын қолдана аласыз.

Компанияларды, өнімдерді немесе басқа да қызметтерді іздеу. Сары парақ (Yellow Page) атты жаңа каталог қызметі сізді қызықтыратын мамандықтары бар компанияларды іздеуге мүмкіндік береді. Оның адресін тез табу үшін сіз сол облысқа сәйкес кодты бере аласыз. Осылайша өзіңіз қолыңыз жетпей жүрген затыңызды да табуға болады.

Зерттеу. Заң кенселері бұрын қажетті информациялар үшін сағатына \$8600 төлесе, қазір олар оны Интернеттен өте аз бағаға ала алады. Жылжымайтын заттармен айналысатын фирмалар, олардың бағасын бағалау үшін Интернетке сай келетін демографиялық мәліметтерді пайдаланады. Ғалымдар өздерінің соңғы зерттеу нәтижелерімен ауысады. Интернет көмегімен бизнесмендер болашақтағы нарықтарды үйренеді.

Білім. Мектеп мұғалімдері бүкіл әлемдегі оқыту программаларын бақылап отыра алады. Колледж оқытушылары өз шаңырақтарымен электрондық почта арқылы хабарласып, сонымен қатар телефонмен сөйлесу ақысын үнемдейді. Студенттер компьютерде курстық жұмыстарын жасайды. Сонымен қатар, диалогтық режимде энциклопедияның соңғы басылымын және басқа да қажетті материалдарды алуға болады.

Сапар. Үлкен, кіші қалалар, штаттар және бүкіл мемлекеттерді Web-те туристік және басқа да қажетті информациялардан табуға болады. Желіде сапар шегушілер ауа райы туралы мәліметтерді, транспорт қозғалысының уақыт кестесін немесе мұражайдың жұмыс уақыттары туралы мәліметтер ала -алады.

Маркетинг және сауда. Мұнда программалық жабдықтарды шығарушы компаниялар өз өнімдерін сатып, желіде жаңа нұсқаларын ұсынады. Желі көмегімен басқа да түрлі өнімдер сатылады. Электронды кітап және грампластинка дүкендерді оперативті режимде істейді. Клиент өзіне қажетті мәліметті каталогтардан көре алады.

Денсаулық сақтау. Науқас адамдар мен дәрігерлер денсаулық сақтау облысындағы соңғы ашылған жаңалықтарды біліп отырады. Өз білімдерімен алмасып және медициналық мәселелерді шешуде бір-біріне көмектесіп отырады.

Инвестиция. Адамдар акция сатып алып ақшаларын пайдалы айналымға жібереді. Кейбір компаниялар өздерінің акцияларын оперативті режимде ұсынады. Осылайша инвесторлар жаңа өнеркәсіптерді, ал өнеркәсіптер капитал табады. Конференция және аукциондарды ұйымдастырушылар хабарлама жасау, өтініш жинау немесе қатысушыларды тіркеу, т.б. жұмыстарды Web-те жасайды. Мұнда информация барлық уақытта жаңарып отырады, мұнда қағазды және транспорт шығынын әлдеқайда үнемдеуге болады.

Дін. Дін немесе басқада қоғамдық ұжымдар web-те өздері туралы айтып, басқа адамдарды ұжымдарына шақыратын өз парақтары бар.

1.6 Интернеттің негізгі принциптері

Интернет мыңдаған корпоративті, үкіметтік, ғылыми және үй желілерінен құралған. Өртүрлі архитектуралы және топологиялы желілерді біріктіруге IP (Интернет Protocol) протоколын және мәліметтер пакеттерін маршрутизациялауды қолдану арқылы қол жеткізілді. IP протоколы әдейі физикалық байланыс арналарына тәуелсіз етіп жасалды. Яғни цифрлық

мәліметерді тасымалдауға арналған кез-келген жүйе Интернетпен де байланыса алады. Желілердің байланысқан түйіндерінде арнайы маршрутизаторлар (бағдарламалық немесе аппараттық) пакеттердің қабылдаушылардың IP-адрестерін қарай отырып, мәліметтер пакеттерін сұрыптаумен және бағыттаумен айналысады. IP протоколы бүкіл әлем көлемінде біртұтас адрес кеңістігін құрады, бірақ әрбір жеке желіде өзіндік адрес кеңістігі болуы мүмкін. IP-адрестерді осылайша ұйымдастыру маршрутизаторларға әрбір мәлімет пакетінің бағытын анықтауға мүмкіндік береді. Осылайша, Интернет құрамындағы жекелеген желілер арасында конфликттер болмайды, ал мәліметтер бүкіл әлем көлемінде дәл жеткізіледі.[2]

IP протоколын IETF (Интернет Engineering Task Force) ұйымы ойлап тапқан болатын. IETF және оның жұмыс топтары қазіргі күні де Бүкіләлемдік желінің протоколдарын дамытумен айналысады. IETF қызметіне қарапайым пайдаланушылар қатыса алады. Бұл ұйым комитеттері RFC құжаттарын жариялайды. Бұл құжаттарда көптеген сұрақтардың техникалық спецификациялары және дәл түсініктемелері беріледі.

Қазіргі заманғы кең тараған Интернет қызметтері:

- бүкіләлемдік тор;
- веб-форумдар;
- блогтар;
- вики-проекттер;
- интернет-дүкен;
- электронды почта;
- жаңалықтар топтамасы (негізінен, Usenet);
- файл-алмасу желілері;
- интернет-радио;
- интернет-теледидар;
- IP-телефония;
- мессенджерлер;
- FTP-серверлер;
- IRC (веб-чаттар);
- іздеу жүйелері;
- интернет-жарнама;
- өшірілген терминал;
- төлеу жүйелері.

1.7 Қазақстан интернеті

1998 жылы наурыз айының басында Республика телекоммуникациялық компанияларының басқаруымен телекоммуникация ассоциациясы деген жаңа ұйым пайда болды .

Коммерциялық емес құрылымның шығу мақсаты – кәсіпорынның іскерлігін байланыстыру, желілер телекоммуникациясының дамуын, ақпараттық технологияны пайдалану және байланыс қызметін ұсынады.

Ассоцианың негізгі сан есебінде : отандық және шетелдік операторлар арасындағы құжат қатынастарын құруға практикалық көмек көрсету көзделген . Осыған орай - тұтынушылар мен қолданушылар арасындағы қызметті де қамтиды. Мұнда – электрондық World Wide Web-тің өсуімен қоса, көптеген қызықты және арнайы web-қосымшалар пайда болуда. Соңғы уақытта, әртүрлі анықтамалар, каталогтар және мәліметтер қоры (солардың ішінде неше түрлі керекті тақырыптар ақпараты бар) кіре бастады. Мысалы, web-қосымшада --- өзіңізге керекті Голливудта түсірілген фильмді таңдап алып, сол фильм туралы толық анықтама алуға болады. Оның жанрын, актерлік бөлімін, режиссерлік және де фильмнен бірнеше кадрларын көрсетіп шығарады. Сол жерден сізді қызықтыратын кино актерлері, режиссерлері және де дауыс операторлары жайында толық мәлімет ала аласыз.

World Wide Web - қазіргі заманғы Интернеттің ең танымал қызметі . Оны көбіне Интернетпен теңдестіреді, бірақ шын мәнінде бұл оның көптеген қызметтерінің бірі. Бір Web-серверде бірнеше web-қосымшалар болуы мүмкін. Электронды Wed - құжаттар принтерде басу үшін емес , компьютер экранында қарап шығуға арналған және сонымен бірге оның қайсысында екені алдын ала белгісіз .Web – парақтарды қарап шығуға арналған программалар браузерлер деп аталады .World Wide Web түп- түгел гипермәтіндерден тұрады .Гипермәтін- мәтінің дара блоктарымен байланысты ассоциациялық түрде ұсынылған мәтін. Бізге кенеттен бір файл керек болып қалды делік және оның қай жерде екенін бізге белгілі болсын. Енді сол файлды өз компьютерімізге қалай әкелу жолын қарастырайық. Мұндай мақсат үшін файлды тасымалдау протоколы деп аталатын жүйе қолданылады(File Transfer Protocol – FTP). FTP-серверінің мәліметтерімен қатынас жасау үшін әркім стандартты кіру сұбатын орындау керек. Оның бір мысалы мынадай болу керек: Open named password-FTP - серверге кірердегі сұраныс тізбегі, мұнда:

- Open FTP-сервермен қатынас жасауды сұрау;
- қажетті информациямен толықтырылған аты немесе anonymous;
- Password- сұраушы адамның паролі немесе оның почталық адресі.

1.8 DNS server ұғымы

DNS-сервер – тиісті хаттама бойынша сұрау салулар DNSқа жауаптары үшін қолайлы қосымша. DNS сонымен бірге - сервермен қосымша іске қосылған хост деп атай алады.

DNS-тың атқарылатын функциялары бойынша – серверлер нақты сервердің кескініне байланысты бірнеше түрлерге жата алады топ-топтарға

жіктеледі; авторитативті DNS-сервер – аймақты ар жағында қандай болмасын сәйкес келетін сервер.

Ұста немесе (BINDтің терминологиясында) алғашқы сервер - өзгерістердің енгізуіне осы аймақта құқыққа ие болған сервер. Аймақтар үшін әдетте сервер бір-ақ ұстада болады. Microsoft DNS-тің жағдайында – ұстаның Active Directoryмен сервер және оның кірігуі – серверлер емес, (өзгерістердің репликациясы өйткені DNS құралдар емес іске асады - сервер, Active Directoryнің құралдарымен, серверлердің тең құқықтылығы және мәліметтердің өзектілігі арқасында не қамтамасыз етіледі) бірнеше бола алады.

Слейв немесе екінші сервер, өзгерістердің ұста өзгерістері туралы хабар үйрететін рұқсатсыз енгізу құқығы осы аймақта - сервер. Ұстаға қарағанда - олар сервер шексіз сан бола алады. Дәл осылай Слейв (және қолданушы ұста және слейв, айырмашылық айырып тани алмайды аймақтың күйге келтіруіндегі өзгерістерінің конфигурирования/внесениясы тек қана кезеңде көрініп қалады) сервермен авторитативным болып табылады.

Кэш жасайтын DNS-тан сервер – клиенттердің сұрау салулары қызмет көрсететін сервер (рекурсия сұрау салуы алады, серверлерге авторитативнымға рекурсивті емес сұрау салулары көмегімен оны орындайды немесе DNS – жоғарғы серверге рекурсия сұрау салуын алып береді)

Жергілікті DNS-тер – сервер; DNS-тың қызмет көрсетуі үшін қолданылады - жергілікті машина атқарушы клиенттер. Бұл DNS кэш жасайтын сервердің бір түрлері іс жүзінде жергілікті қосымшалардың қызмет көрсетуі үшін кескінделген.

Қайта бағыттайтын DNS – сервер; рекурсия сұрау салуларын түрде жоғарғы кэш жасайтын серверге алған рекурсия сұрау салу қайта бағыттайтын сервер (forwarder, ішкі DNSтер ағылшынша - сервер). Кэш жасайтын DNS-қа жүктемесінің төмендетуі үшін көбінесе қолданылады – сервер.

Түбірлік DNS-сервер – түбірлік аймағына авторитативным болатын сервер. Жиі қолданылатын түбірлік серверлерінің жинағы 13 даналардың әлемінде, олардың домен аттары root-servers.netтың аймақтарында болады және a.root-servers.net, m.root-servers.netтың b.root-servers.net деп аталады, m.root-servers.net. Нақтылы кескіндерде болуы мүмкін жергілікті түбірлік серверлерінің күйге келтіруін ахуал жергілікті жүйеді.

Есепке алатын DNS-тар – сервер. Қолданушыларданғы динамикалық жаңарту қабылдаушы сервер. DHCP мен жиі сәйкес келеді - сервермен. Microsoft DNS-сервері доменінің контроллерінде жұмыс істегенде сервері DNS - есепке алатын сервердің тәртібінде ат және IP компьютердің сәйкестігі туралы мәліметтің доменнің компьютерлерінен қабылдай және доменнің осы аймақтары ол сәйкес жаңарта жұмыс істейді.

DNSBL - (мекенжайлар және аттардың қара тізімдері бар сервері) сервер. үстірт, мұндай сервер DNS-тың иерархиясына кірмейді, дегенмен жұмыс, DNS-тың несі үшін тетік және хаттама сол қолданады - сервер.

Кейбірі серверлер желінің әртүрлі сегменттері үшін әртүрлі тәртіптердегі жұмыс істеу мүмкіндіктерін қолдайды. Виндтерге бұл тәртіп view деп аталады. Мысалы, мысалы, жергілікті мекенжайлар үшін сервер 10.0.0.0/8) серверлердің жергілікті мекенжайлары бере алады, сыртқы желінің қолданушылары үшін - сыртқы мекенжайлар. Мысалы, сонымен бірге, internalдің аймағынан атының сұрау салуына сыртқы мекенжайлар үшін internalдің аймағына авторитативным өзін 10.0.0.0/8 сервердің желісіндегі мекенжайларды көрсетілген диапазоны үшін тап қалған аймақ үшін авторитативным сервер дәл осылай тек қана жауапқа жауап теуіп шығауға жариялайтын бола алады белгісіз.

Барлық DNS-тар – стандарт бойынша сервер RFC 1035 TCP және UDP-нің 53 порттарына жауап береді. BIND-тің ерте болжамының сұрау салуларын жіберудің жанында 53 порттарды қолданды, еркін тіркелмеген мекенжайлар қолдана жаңалау өзін-өзі DNS-тар сияқты ұстайды – клиенттер.

DNS (Домен атау жүйесі).

DNS домен аттарын аударуға жауапты, мысалы www.microsoft.com доменін IP мекенжайына мысалы 192.168.164.123. Бұл IP мекенжайлар интернеттегі ресурстарға қолжеткізу үшін қажет.

Домен ақпараты интернеттегі DNS серверлерінде сақталады. DNS серверлері домен үшін орнатылатын ресурстық жазбаларды іздеу үшін пайдаланылады. Ресурстық жазбалар веб серверлері немесе хабар алмасу сервері сияқты ресурстар қолжетімді болу үшін кейін оны IP мекенжайға сілтейді.

Интернетте домен атын пайдалану алдында домен ақпаратын басқаратын екі жеке аттарды пайдалануыңыз қажет: домен тіркеушісі және DNS орналастыру қызметі. Сондай-ақ, домен тіркеушісі жиі DNS иелік ету қызметі болып табылады. Алайда, бұл әрдайым дұрыс бола бермейді.

Домен тіркеушілері.

Домен тіркеушісі — домен аттарын тіркейтін компания. Әркім, жеке адамдан бастап халықаралық корпорацияға дейін интернетте оны пайдалану алдында өздерінің домен аттарын тіркеу үшін домен тіркеушісін пайдалану қажет. Барлық домен тіркеушілер Internet Corporation for Assigned Names and Numbers (ICANN) ұйымы мен куәландірілуі қажет. Белгілі бір домен тіркеушісінде қолжетімді домен атын іздегеніңізде сіз шынында осы домен атының қолжетімділігін дүниежүзіндегі барлық домен тіркеушілерінен тексерген боласыз.

Негізінде тіркелетін домендер саны жылсайын көбейеді. Домен тіркелулері бір домен тіркеушісінен басқасына көшіп жүру мүмкін. Егер домен тіркелуі жаңарған болса, домен атын жалпыға бірдей қолжетімді болып қалады.

DNS иелік ету қызметтері

DNS иелік ету қызметтері — домен үшін DNS жазбасын қамтитын DNS серверлерге ие компания. Кейбір домен тіркеушілері DNS иелік ету қызметтерін өздерінің домен тіркелу ретінде қамтамасыз етеді: басқа домен

тіркеушілері DNS иелік ету қызметтері мен қамтамасыз етпейді. Аккредитацияланған домен тіркеушілерінде тіркелу қажеттілетін домен аттарының айырмашылығы кез-келген тұлға немесе тіркелген домен аты және ортақ IP мекен-жайы бар компания DNS серверді жасай алады және DNS жазбаларын домендердің кез-келген санына орналастыра алады. Доменнің DNS жазбалары DNS иелік ету қызметімен орналастырылғанда сіз және барлық интернет доменді пайдалана алады.

Кейбір DNS иелік ету компаниялары доменіңіз үшін DNS жазбаларын жасауға және өзгертуге мүмкіндік береді. Басқа DNS иелік ету компаниялары доменіңіз үшін DNS жазбаларын тікелей өзгертуге мүмкіндік бермейді. Сондай-ақ, барлық DNS иелік ету серверлері барлық DNS жазбаларын қолдамайды. Мысалы үшін, кейбір DNS иелік ету қызметтері TXT жазбаларын немесе SRV жазбаларын қолдамайды.

2 Web қосымша құру құралдары

Интернеттің дамуына орай жасалынған web-беттердің бет-бейнесі өзгере түсуде. Бастапқыда HTML беттер тек белгілі бір ақпаратты бере отыра, ешқандай қосымша қызметтерді атқармайтын. Бірте-бірте сұраныстар күрделене, енді web қосымшаны құруда тек HTML тілін білу жеткіліксіз. Түрлі технологиялар икемді web қосымша жасауға ықпалын тигізеді. Қазіргі заманауи web қосымшаларды құру үшін: HTML, CSS, JavaScript (немесе VBScript), Flash, PHP (немесе Perl) және MySQL реляциялық мәліметтер қоры пайдаланылады. Технологиялардың қолданысқа енуіне байланысты web-қосымшаның оңтайлы өзгеруін байқауға болады. Технологиялардың қолданысқа енуіне байланысты web қосымша оңтайлы өзгеруін 1.1 – суреттен көруге болады[3].

HTML. HTML – бұл бағдарламалау да, құжаттарды басқару тілі де емес. Бұл ең алдымен гипермәтінді белгілеу тілі. Оның басты мақсаты – мәтінді орналастыру, құрылымын тегтер арқылы сипаттау: параграф, цитата, тізім, бөлім.

HTML тілінде көптеген құрамдас элементтері бар. Олар құжатты кез келген қалау бойынша безендіруге көмектеседі. Браузердің үнсіздік кейпіндегі қаріп стильдерінің көрінісі ұнамаса, қосымша тегтер арқылы оңай өзгертуге болады. Егер бұл да көңілден шықпаса CSS стильдер кестесін пайдалануға болады.

XHTML. XHTML тілі XML және HTML тілдерінің қоспасы, біртіндеп HTML-ды ығыстырып келеді.

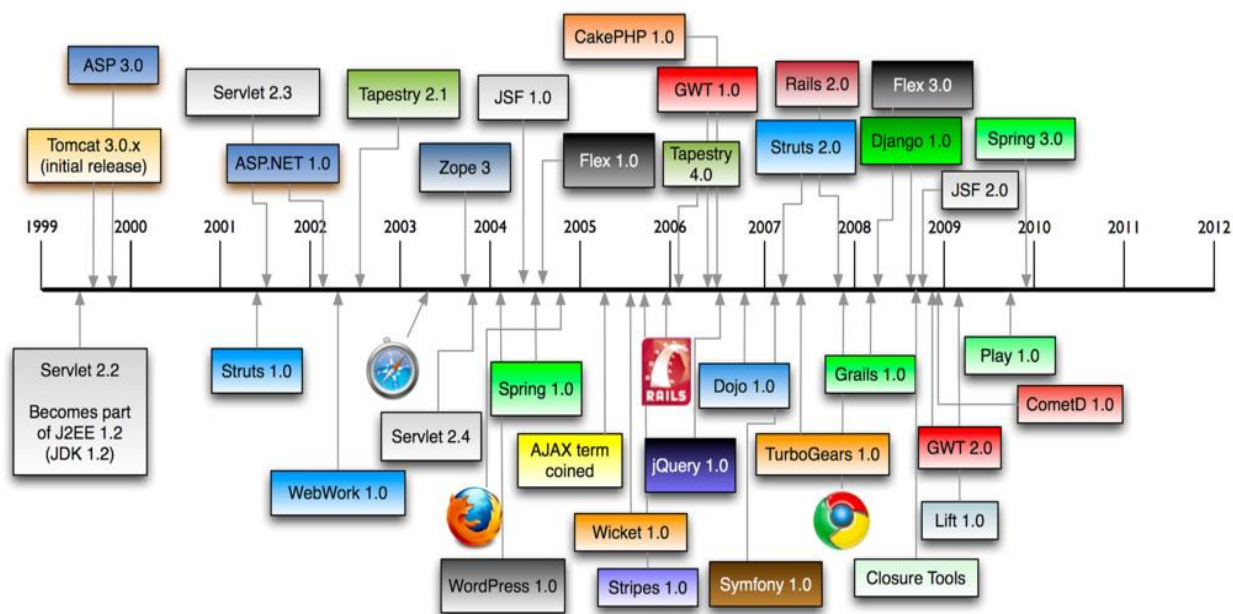
CSS. CSS (Cascading Style Sheets) – каскадты стильдер кестесінің тілі. Ол web-беттердің мүмкіндіктерін арттыру үшін құрастырылған.

Web-құрастырушыларға беттің сырт келбетін жақсартуға мүмкіндік береді (қаріптер, түстер, шегіністер, орналасуы, т.б.). CSS құжаттың негізгі ішкі құрылымын (мысалы, HTML тілінде жазылған) және оны безендіруінен (CSS-те жазылғанынан) айыру үшін жасалған. Мұндай әрекет құжаттың сырты көрінісі мен қайталанатын элементтерін безендіруге септігін тигізеді.

HTML мен CSS-ті пайдаланғанда келесілерді ескеру қажет:

– HTML-код мәтінді логикалық тұрғыда жазады, яғни web-беттің құрылымын береді: абзацтардың, графикалық суреттердің, жолдар мен ұяшықтардың орналасуы мен кезегі.

– CSS стильдер кестесі мәтіндерді физикалық тұрғыда орналастырады, яғни web-беттің көрінісін береді: абзац мәтіндері қандай қаріппен терілетіндігі, атаулары қандай түспен көрінетіндігі, кестелердің қалыңдығын және т.б.



2.1 - сурет – Web-технологиялардың қолданысқа енуі

Web-беттің көрінісі оның құрылымынан бөлек жасалуы тиіс. Сондықтан, мамандандырылған web-құрастырушылар CSS стильдерін бөлек файлға жазады.

Сонымен қатар, стильдерімен толтырылмаған HTML бет оқылуға ыңғайлы. Себебі, HTML бетте бір уақытта web-беттерді (форматтаудың) өңдеудің логикалық және физикалық тегтері болады. Қаріпті қоюлатып ерекшелейтін `` және `` тегтері, сонымен қатар курсивті `<i>` және `` тегтері бар. Бірақ `` және `<i>` физикалық өңдеу тегтері мәтінді тек ерекшелеп қояды (сондықтан олар ескірген және қолданылмайтын тегтер қатарына жатқызылады), ал `` және `` логикалық өңдеу тегтері тек ерекшелеп қана қоймай, мәтінге маңыздылық дәрежесін береді. Егер web-бетті мүмкіндігі шектеулі адамдар арнайы браузер арқылы оған дауыстап оқып берсе, `` және `` тегтер арасындағы сөздерге өзге интонация беріледі[5].

Каскадты стильдер өзінің мәні бойынша динамикалық емес. Олар құжаттың жүктелу барысында көріну күйін анықтауға мүмкіндік береді. CSS көмегімен web-беттердің қасиеттерін JavaScript тілін пайдаланып динамикалық өзгертуге болады.

DOM. DOM (Document Object Model) - құжаттың нысандық моделі. Сценарийлер тілін қолданып web-бетті динамикалық өзгертеді. DOM әр элементіне ID атрибуты арқылы JavaScript функциясын қояды. Осындай функциялармен CSS арқылы берілген нысан атрибуттарының қасиеттерін басқаруға болады.

ID (объект идентификаторы) барлық HTML - тегтерімен қолдайды. Оның мәні уникалды, яғни бетте бір id- мен бірнеше элемент орналаса алмайды, ал class элементі бірнеше элементтерге қолданыла береді. ID - динамикалық HTML (DHTML)-дың негізгі түсінігі оның көмегімен JavaScript уникалды объектіні анықтап басқарады.

DHTML. DHTML тілі жоқ. Динамикалық HTML– HTML- дың шектеулерін айналып өтіп, web- браузерлердің 4 версиясына ендірілген технологияларды сипаттау үшін ойлап табылған коммерциялық термин.

DHTML web-стандарттардың комбинациясы:

CSS + JavaScript + DOM + XHTML = DHTML

мұнда CSS - объектілер атрибутын анықтайды;

JavaScript - объектілерді өзгертеді;

DOM - объектілерді табады;

XHTML - объектілерді құрады (мәтінді белгілейді).

DHTML негізінде құрылған web қосымша серверге жүгінбей өзгере алады, яғни web-беттің интерактивтілігін арттыратын клиенттік код (client-side-code) ретінде ұсынылады.

DHTML - дың мүмкіндіктері:

– барлық браузерлер қолдайды;

– стандарты технологияларды қолданылады;

– web-бетті жүктегеннен кейін өзгертулерді енгізуге болады;

– DHTML мәтіндік файлдары Flash пен Java-ға қарағанда жылдам жүктеледі;

– кеңейту модульдерін қажет етпейді;

– меңгеруде қарапайым;

– web-беттерді жасау жылдамдығы жоғары;

Кемшіліктері:

– JavaScript пен CSS синтаксисте жіберілген қателерге сезімтал;

– браузерлердегі қателерге байланысты жұмысы тұрақсыз;

– браузерлер мен операциялық жүйенің үйлесімсіздігіне байланысты web-беттердің көрінісі әр түрліше.

2.1 Интернеттегі бағдарламалау тілдері

Web-қосымша құрастыруда қолданылатын программалау тілдерін 2 топқа бөліп қарастыруға болады.

Біріншісіне, web-қосымшаға кірушінің компьютерінде, браузерінде, орындалатын код жатқызылады. Бұлар бәріне таныс JavaScript, VBScript, Java-апплеттері. Осы тілдерде жазылған бағдарламалар, web-беттің кодына ендіріледі, немесе бөлек файлға жазылып web-беттен шақырылады.

Екінші топқа, web-сервер орналасқан компьютерде орындалатын бағдарламалар жатқызылады. Бұл топқа PHP жатады, және басқа тілдерге

қарағанда өзінің қарапайымдылығымен ерекшеленеді. Бұл тілде бағдарламаны құра отырып С++ немесе Perl тілдеріндегі сияқты файл сценарийлеріне қатынау құқығы туралы, түрлі модульдерге қатынау жолдарын көрсету және т.б. әрекеттерді ескермеуге болады. PHP бағдарламасын тестілеу үшін хостинг провайдерде орнатылған операциялық жүйені орнату қажет емес. Сәйкес web-серверді жүктеп PHP модулін қосса жеткілікті.

2.2 PHP бағдарламалау тілі

PHP-бұл Web-серверге жіберілетін скриптердің көмегімен Web-беттерінің динамикалық генерациясына қажетті бағдарламалау тілі. Сіз бетті PHP және HTML-дің көмегімен ашасыз. Web-қосымшаны пайдаланушы бетті ашқанда, сервер html-код қосылған PHP операторларын орындайды және нәтижені пайдаланушының браузеріне жібереді. Бұл әрекет дәл осылай ASP және Cold Fusion-ның көмегімен жасалады. Дегенмен ASP және Cold Fusionға қарағанда, PHP ашық бастапқы кодты өнім болып табылады және платформалы тәуелсіз. PHP Windows NT және Unix-тің көптеген версияларын да жұмыс жасайды. Ол Apache-дегі модуль ретінде жіберіле береді. Егер жіберілу Apache модулі түрінде болса PHP оңай және жылдам жұмыс жасайды. Бұл кезде процессті жасауға байланысты туындайтын қосымша шығындар болмайды[4].

Сіздің құжат бетіңіздегі әртүрлі операциялардан басқасын PHP көмегімен HTTP – тақырыптарды қалыптастырып HTTP орната аласыз. Сонымен қатар аутентификацияны басқарып пайдаланушыны басқа бетке бағыттай аласыз. PHP берілген дерек қорына еруге үлкен мүмкіндіктер ашады. Бұл сізге PDF құжаттарды инерациялаудан, XML-дегі грамматикалық талдауды жасауға мүмкіндік береді.

PHP операторлары сіздің бетіңізге Web-беттерді қойып береді, сондықтан арнайы ортада жүзеге асырудың қажеттілігі туындамайды. Сіз PHP-кодының блогын `<?php` тегінен бастап, оны `?>` тегімен аяқтайсыз. Бұл тегтердің арасындағылардың барлығы PHP код ретінде интерпритацияланады. PHP тілінің синтаксисі Си және Perl синтаксисіне ұқсас келеді сіз ауспалыларды оларды пайдаланудан бұрын жарияламауыңыз керек. Массивтермен Хэмтер оңай жүзеге асырылады. Дегенмен PHP Apache құрылған жағдайда бәрінен жылдам жұмыс жасайды. PHP web-қосымшада оны Microsoft IIS және Netscape Enterprise Server орналастырудың реті бар. Егер сізде PHP-ді орнатуға бағдарламалық қамтамасыз етудің көшірмесі жоқ болса, оны сізге ресми web-қосымша ала аласыз. Ол жерден сіз PHP-дің барлық ерекшелігімен қызмет бейнеленген түсіндірілген жетекшілікті таба аласыз.

Соңғы он бес жылдары интернеттің қиял-ғажайып дамуы және адамдардың арасындағы қарым-қатынастың жаңа әдістерін белгіледі. Бұл құбылыстың алдыңғы өлкелеріне (WWW) World Wide Web-те болады.

PHP тарихы 1995 жылдан тәуелсіз бағдарламашы (Rasmus Lerdorf) Расмус Лердорф сайта оның онлайндық резюмесін оқып шыққан келушілердің санын есептеуге арналған Perl/CGI-нің сценариын жазудан басталады. Оның сценариы екі есепті шешті : келушінің деректерінің тіркелуі және келушілердің санын - web бетке шығару. WWW дамыту әлі тек қана басталды, бұл есептерді шешуге арналған ешқандайда арнайы құралдар болмады, сондықтан авторға көп сұрақтар лап берді. Лердорф (PHP) Personal Home Page немесе (гипер мәтін процессоры) Hypertext Processor аталған өз құрал-сайманын тарата бастады.

Лердорфаның PHP құрал-сайманының шулы жетістігі PHP-дың кеңейтулерін өңдеуге әкеп соқты. Кеңейтулердің бірі HTML формада енгізілген мәліметтерді символдық айнымалыларға түрлендірді, осыдан оларды басқа жүйелерге шығаруға мүмкіндік берген. Қойылған мақсатқа қол жеткізу үшін Лердорф өңдеулерді жасауды ендігәрі Perl ден C-ге өтуге шешті. Сол кездегі PHP-дың құрал-сайманының кеңейтуі 2.0-ші PHP пайда болуына, немесе (Personal Home Page Form Interpretator) PHPFI-і келтірді. Дүниежүзінен бағдарламашылар 2.0 версияны жетілдіруге қатысты. Жаңа PHP версиясы ерекше мәлімділікпен пайдаланды, және өңдеушілердің негізгі командасы жақында іріктелді. Олар HTML-ға программалық кодтың енгізуінің бастапқы тұжырымдамасын тікелей сақтады және PHP-дің 3.0-ші версиясының пайда болуына келтірген лексикалық талдаудың механизмін қайтадан жазды.

PHP 4 бұл қозғалыста жұмыс жасаушы ретінде 2000 жылы жарыққа шықты. Өндірісті жоғарылатуда қосымша ретінде PHP 4-тің мүмкіндіктері зор еді. PHP 4-тің шығуымен ол интернет доменінің 20 пайыздан көбінде қолданыла бастады.

2000-2004 жылдар арасында 4 версияны жұмсарту үшін белсенді жұмыстар жалғасты. Шамамен сол кезден бастап-ақ PHP Group жаңа версияның мүмкіндігін ойластыруға кірісті. Бірінші кезекте тілдің объектілік мүмкіндіктерін күшейту шешілді. Бұл оның жүзеге асыру үшін ірі жобаларды пайдалануға мүмкіндік берді. 5-ші версияны жарыққа шығару үшін қосымша уақыт бөлінді. Оған көптеген мамандар қатысты олардың ішінен Стерлинг Хьюза және Маркус Бергерді атап өткен жөн болар.

Ақыры 2004 жылдың шілдесінде PHP 5 ресми түрде жарыққа шығады. Бірінші кезекте жоспарланғандай өңдеуде объектілермен жұмысты барлық механизм жұмылдырылды. Егер алдыңғы версияларда PHP-дегі объектілік-жобалаушылық бағдарламалау минимальды деңгейде мүмкін болатын болса, ал PHP 5 объектілік бағдарламалауды жүзеге асыруда көптеген потенциалдарға ие. Оныменқоса PHP XML мен жұмыс жасауда бағалы кеңейтулермен, берілгендердің әртүрлі қайнар көзіне байыды.

PHP 4-дегі XML мен жұмыс жасауға арналған барлық негізгі кітапханалар күрделі өңдеуге ұшырады. Мынадай танымал кеңейтілулер SAX, DOM, және XSLT қазір libxml2-нің құрал-сайманын пайдаланады. Сонымен қатар тағыда жаңа екі кеңейтулер қосылды – SimpleXML және SOAP.

SimpleXML XML – берілгендермен жұмысты айтарлықтай оңайлатады. Бұданалдын PHP-да XML мен жұмыс бұншалықты қарапайым болмаған еді.

SOAP–тың кеңеюі PHPда XML–хабарламаның көмегімен басқа қосымшалар мен ақпараттауысатын сценарий құруға мүмкіндік береді. Мысалы HTTP. PHP қосымшалы қазіргі таңдағы танымал веб-сервистер мен интеграциялауға мүмкіндік туындайды.

PHP 5 MySQL (MySQL Improved) жаңа кеңейтулер MySQL – сервер версиясымен 4.1.2 және жоғарылар мен жұмыс жасауға арналған. MySQL – ға тек қана процедуралық емес сонымен қатар объекті жобалаушылық интерфейс жүзеге асады. Бұл модульдің қосымша мүмкіндіктері SSL-ді, транзакцияны бақылауды қосады.

SQLite–тің кеңейтілуі мәліметтерді кәдімгі файлдарда сақтайтын қосымшаны құрастыруға мүмкіндік береді. Ол SQL интерфейсін пайдалануға мүмкіндік тудырады. SQLite – нің маңызды ыңғайлылығы – пайдаланылу қарапайымдылығы. SQLite процедуралық пен қатар мәліметтермен жұмыс жасау үшін мықты объектілік – жобалаушылық мүмкіндіктері бар. SQLite-нің басқа ерекшеліктері – жоғарыжылдамдық, басқарудағы күрделі механизмдердің жоқ болуы және оңай тізгіштік. Көптеген эксперттер PHP –нің бұл кеңейюінің үлкен танымалдылыққа ие болатындығын жоспарлап отыр. PHP 5-тің басқа кеңейтілуі өзіне төмендегілерді қосады:

Tidy –HTML-құжаттарды сараптау және түзету үшін.

Perl – Perl сценарийлерді пайдалану.

SPL – кластармен интерфейстердің стандартты жиынтығын анықтайтын ZE2–нің кеңеюі.

PDO -PHP-нің мәліметтеріне әмбебап интерфейс ұсынады.

PIMP-GD модульінің орнына келген, графикамен жұмыс жасауға арналған жаңа кітапхана.

Бұдан басқа PHP 5-те есте сақтаудың әлдеқайда жылдам диспетчері пайдаланылады.

PHP 5 REAR құрылымын нақты жүзеге асыруға керекті қасиеттермен толықтырылған.

PHP-дың мінездемесі. Сіздер түсініп отырғандай, PHP тілін бағдарламалауда ең бас фактор болып іскерлік табылады. PHP бағдарламасы бағдарламашыға есептердің тез және тиімді шешетін құралын беруі керек. PHP жаттығу сипаты бес маңызды мінездемелер мен шартталған:

- дәстүрімен;
- қарапайымдығымен;
- тиімділікпен;
- қауіпсіздікпен;
- иілгіштікпен.

Дәстүрлілігі. PHP тілі әртүрлі облыста жұмыс істейтін бағдарламашы үшін таныс болып көрінеді. Көптеген тіл конструкциялары Си Perl-ден

пайдаланылған, алкейде PHP коды C және PASCAL типті бағдарламаларында кездеседі. Бұлбастапқы күштерді PHP зерттеуде елеулі төмендетеді.

Қарапайымдылығы. PHP сценарийі 10 000 жолдар немесе бір жолдан тұра алады – барлығы сіздің есебіңіздің ерекшелігінен тәуелді болады. PHP сценарийі 10 000 жолдар немесе бір жолдан тұра алады – барлығы сіздің есебіңіздің ерекшелігінен тәуелді болады. Сізге кітапхана жазып алуға, компиляцияның арнайы параметрлерін көрсетуге немесе осы тектесістер жасауға тура келмейді. PHP механизмі экрандалған тізбектен кейін код жасай алады және ол өзіне қос экрандалған тізбек іздегенге дейін жүре береді. Егер код дұрыс синтаксисті болса, ол бағдарламашы айтқандай бағдарлама орындалады.

Тиімділігі. Тиімділік бағдарлама жасауда маңызды фактор болып табылады, олардың бірі WWW. PHP 4.0-ші версиясында ресурстарды бөлу ресурсы іске асырылды және объективті-бағдарланған программалауды қолдау жақсартылды, сонымен бірге сеансты басқару құралы қамтамасыз етілді.

Қауіпсізділігі. PHP-ды өңдеушілердің қарамағында болады және администратордың қауіпсізділігінің ыңғайлы және тиімді құралдары болады және ол екі категорияға бөлінеді: жүйелік деңгей және қосымшалық деңгейінің құралдары.

Иілгіштігі. PHP (embedded) орнатылатын тіл болғандықтан, оны өңдеушінің қажеттіктеріне қарағанда ерекше иілгіштік айырмашылығы болады. PHP әдетте HTML бағдарламасымен қатар қолдануға ұсынылады, ол осы жетістікпен JavaScript, WML-ге, XML және тағыда басқа тілдермен интеграцияланады. Дегенмен, бұдан басқа, PHP жақсы жіктелген қосымшалары бойынша оңай кеңейе алады.

2.2.1 PHP тілінің артықшылықтары

Бүгінгі таңда PHP — әлемдегі ең танымал бағдарламалау тілдердің бірі және миллиондаған веб-құрастырушылар осы тілді өздерінің негізгі құралы ретінде таңдады. Соңғы статистикалық мәліметтер бойынша бұл тіл 22 миллионнан аса web-қосымшаларды және бүкіл жер шарындағы веб-серверлердің үштен бір бөлігінде қолданылады.

PHP-дің мұндай танымалдылығының себебін түсіну қиын емес. Ол масштабты, қол жетерлік және басқа бағдарламалармен оңай әрекеттеседі. Ол анық және қарапайым синтаксисты пайдаланады және оны тез оқып үйренуге болатын және бағдарламаларды тез құрастыруға көмектесетін өзінің түсінікті кодымен қуантады. Коммерциялық құралдардың алдында олардың артықшылығы бар, себебі ол тегін және көптеген платформалар мен архитектураларда, оның ішінде UNIX, Microsoft Windows и Mac OS-та қолданылады және ашық лицензия пайдаланады.

2009 жылдың тамызында Evans Data Corporations жасаған зерттеуде ондаған тілдердің ішінде пайдаланушылар PHP тіліне (Ruby мен Python сәл

артта қалып отыр) риза болды. Жекелеп қарастырғанда, PHP тілі келесі көрсеткіштер бойынша бірінші орында – кроссплатформалы үйлесімділік, қол жетімділік, бағдарламалық құралдардың сапасы және өнімділік, ал қарапайымдылық пен кодтың ыңғайлылығы, кеңейгіштік, қолдану қарапайымдылығы мен қауіпсіздік бойынша екінші орында.

Организациялар мен тәуелсіз құрастырушылар үшін бұл фактілердің барлығы келесіні білдіреді – PHP-ді қолдану уақыт пен ақшаны үнемдейді. PHP-де бағдарламаны жасау ақшаның шығынын азайтады, себебі бұл тілді лицензиялық аударым жасамай-ақ және қымбат құрал-саймандар мен бағдарламалық қамтамасыздандыруларға ақша салмай-ақ әр түрлі мақсатпен пайдалануға болады. Дайын сенімді және тексерілген басқару мен кеңейгіштік элементтерінің қол жетімділігі және осы элементтерді пайдалана отырып құрастырушылар тілге жаңа функцияларды енгізе алуына байланысты PHP-ді пайдалану сапасына әсер етпей бағдарламаны іске асыру уақытын азайтады.

Бұл бірден түсінікті болмағанымен, мұндай мақтаулы қолданудың қарапайымдылығы PHP тілінің артықшылығымен қатар кемшілігі де болып отыр. Артықшылығы – C++ немесе Java тілдеріне қарағанда PHP тілінде бағдарламаларды салыстырмалы түрде оңай оқуға және түсінуге болады. Осындай факт жаңа үйреніп жүрген құрастырушыларды азаптанып оқып үйренуден босатып, эксперимент жасау мен тілді тез меңгеруге итермелейді. Кемшілігі – “қаталдығы” болмағандықтан, құрастырушыларға жалған қауіпсіздік сезімін тудырады және оларды қажетті сапа стандарты, қауіпсіздік және кодты қайта пайдалану мүмкіндігін түсінбей, бәріне қол жетімді бағдарлама жасауға итермелейді.

Осыны ескере отырып, соңғы жылдары PHP қатарында “барлығын жасауға болады” стиліндегі бағдарламалаудан стандартталған фреймворктарды пайдалануды бағытталған әдіске көшу үшін қызу және жемісті жұмыс жүргізілді. Бұл әдіс тек қана PHP-бағдарламаны жазған кездегі алғашқы дайындықты жеңілдетпейді, сонымен қатар анық, ретті және қауіпсіз код жазуға мүмкіндік береді.

2.3 HTML бағдарламалау тілі

HTML тілін ағылшын ғалымы Тим Бернерс-Ли 1989—1991жж Женевадағы (Швейцария) Жарылыс зерттеулері бойынша Еуропалық кеңес қабырғаларында өңдеді. HTML тілі алғашында беттеу аймағында маман болып табылмайтын адамдарға ғылыми және техникалық құжаттармен алмасу үшін құрылды. HTML тілі құрылымдық және семантикалық элементтер— дескрипторлар жиынтығын анықтау жолы арқылы SGML мәселелерін жақсы шешті. Дескрипторлар өте жиі «тегтер» деп аталады. HTML тілі көмегімен салыстырмалы түрде қарапайым, бірақ әдемі рәсімделген құжатты оңай құруға

болады. HTML тілінде құжаттың қарапайым құрылымынан басқа гипермәтіндік қолдау бар. Мультимедиялық мүмкіндіктер кейін қосылды.

Алғашында HTML тілі құжаттарды ойнату (көрсету) құралдарына байланыстырмай құрылымдау және пішімдеу құралы ретінде ойластырып табылды. Идеалды түрде HTML белгісі бар мәтін құрылғыда әр түрлі техникалық жабдықталулармен стилистикалық және құрылымдық бұрмалаусыз өндірілуі керек (дербес компьютердің түрлі түсті экраны, орғанайзердің монохромды экраны, өлшемі бойынша шектеулі ұялы телефонның немесе мәтіндерді дыбыстық ойнату бағдарламасының және құрылғының экрандары). Бірақ қазіргі кездегі HTML тілінің қолданылуы өзінің бастапқы қолданысынан әлдеқайда алысырақ. Мысалы, бірнеше рет беттерді пішімдеуге қолданған <TABLE> тегі құжаттарда қарапайым кесеттер құру үшін арналған. Уақыт өте келе HTML тілінің платформаға тәуелсіздіктің негізгі идеясы мультимедиялық және графикалық рәсімдеу қажеттіліктеріне берілген.

Браузерлер. HTML белгісі бар мәтіндік құжаттар (бұл құжаттардың кеңейтілуі .html немесе .htm), пішімделген түрде көрсетілетін арнайы қосымшалармен өңделеді. «Браузерлер» немесе «интернет-шолушылар», деп аталатын мұндай қосымшалар әдетте қолданушыларға веб-беттер сұранысы, оларды қарауға және (басқа сыртқы құрылғыларға шығару үшін) және қажеттілік жағдайында қолданушылармен енгізілген мәліметтерді серверге жіберу үшін ыңғайлы интерфейсті ұсынады. Қазіргі кезде ең танымал браузерлер Internet Explorer, Mozilla Firefox, Apple Safari, Google Chrome және Opera болып табылады.

Нұсқалары. HTML 1.0 тілінің ресми түрдегі спецификациясы жоқ. 1995 жылға дейін HTML тілінің көптеген бейресми түрдегі стандарттары болды. Ресми түрдегі стандарттарды олардан айыру үшін оған екінші нөмірді меншіктеді.

1995 ж наурызында 3 нұсқасы бүкіләлемдік тордың Консорциумымен ұсынылды (W3C) және көптеген кестелерді құру, мәтіндердің суреттерді «айналып өтуі», күрделі математикалық формулаларды көрсету сияқты жаңа мүмкіндіктерді қамтамасыз етті. Және бұл стандарт екінші нұсқамен сәйкес болған кезде де сол кездегі браузерлер үшін оны жүзеге асыру қиын болды. 3.1 нұсқасы ресми түрде ешқашан ұсынылған жоқ және HTML стандартының нұсқасы 3.2 болды. Бұл нұсқаға Netscape Navigator және Mosaic браузерлері қолдайтын стандартты емес элементтер қосылған.

HTML 4.0 нұсқасында стандартта кейбір «тазартылулар» болды. Көптеген элементтер ескірген және ұсынылмаған (ағ. deprecated) болып белгіленді. Әдетте шрифт қасиетін өзгертуге қолданылатын font элементі ескірген деп белгіленді (оның орнына CSS каскадты кестелер стилі қолдану ұсынылды).

1998 ж. бүкіләлемдік тордың Консорциумы HTML 4 тіліне негізделген, бірақ XML синтаксисіне сәйкес белгісі бар жаңа тілмен жұмыс істей бастады. Соңында атауы XHTML болды.

WHATWG бірлестігімен (ағ. Web Hypertext Application Technology Working Group) 2004 ж. бастап, HTML тілін кеңейтетін, жиі бейресми «HTML 5» деп аталатын және әр түрлі типтік беттердің семантикасын жақсы көрсету үшін, мысалы форумдар, аукциондар web-қосымшасы, іздеу жүйелері, онлайн-дүкендер және т.б. үшін Web Applications 1.0 спецификациясы өңделіп жатыр.

HTML-құжаттың құрылымы. HTML тілі өз алдына элементтер жиынтығын келтіреді және онда әр элементтің басы мен соңы арнайы белгілермен — тегтермен белгіленеді. Элементтер бос болуы мүмкін, яғни ешқандай мәтін мен басқа да мәліметтер болмауы мүмкін (мысалы, жолды ауыстыру тегі
). Мұндай жағдайда әдетте жабушы тег көрсетілмейді. Одан басқа, элементтерде қандай-да бір қасиеттерді анықтайтын атрибуттар болуы мүмкін, (мысалы, font элементі үшін шрифт өлшемі). Атрибуттар ашылушы тегте көрсетіледі. HTML-құжат фрагментінің мысалы:

 Екі тег арасындағы мәтін — ашылушы және жабылушы.

 мұнда элементтің href. атрибуты бар.

2.4 JavaScript бағдарламалау тілі

JavaScript – прототипті-бағытталған скриптті бағдарламалау тілі.

ECMAScript тілінің диалектісі болып табылады. JavaScript әдетте қосымшалар объектісіне бағдарламалық рұқсатнама ретіндегі құрылған тіл болып пайдаланылады. Браузерлерде веб-беттерге интерактивтілік беру тілі болып кең қолданыс табады.

Негізгі құрылымдық сипаттамалары: динамикалық типтеу, әлсіз типтеу, жадымен автоматты түрде басқару, прототипті бағдарламалау, бірінші класс объектісі сияқты функциялар.

JavaScript тіліне көптеген тілдер әсер етті. JavaScript тілі қандай-да бір компанияға жатпайды. «JavaScript» атауы Oracle Corporation компаниясының тауарлық белгісімен тіркелген.

JavaScript тілінің танымалдылығы web-бет элементтерін бір-бірімен байланысу және қайта жүктеудің кең мүмкіндіктеріне байланысты. Дизайн бөліктерін көрсетуге, жасыруға, көшіруге және безендіруге көмектеседі. Осындай әрекеттер презентациялық эффектiлердi жасауға, форма мәліметтерін өңдеуге және құрамын басқаруға септігін тигізеді.

Бірінің үстіне бірін орналастыруға болатын HTML бөліктері (пиксельге дейінгі дәлдікпен):

– пішіндермен жұмыс. Кез келген мәліметтерді скрипттер арқылы өңдеу ыңғайлы, ол ақпаратты серверге жіберу алдында дұрысытығына тексереді. Егер мәтіндік алаңға қандай да бір санды енгізуін сұраса, әріптік символдардың енгізілу мүмкіндігін ескеру қажет;

– суреттермен жұмыс. Скрипттер арқылы суреттің алдын ала жүктелуін жасауға болады. Суреттерді тікелей көру алдында браузер өзінің жадысында сақтайды да, қажет болған жағдайда жылдамырақ бейнелейді. Ені, биіктігі, графикалық файлдың адресі сияқты суреттің негізгі параметрлерін динамикалық өзгертуге болады. Ол тінтуірдің әрекетіне байланысты жауап береді, ерекшелегенде сурет өлшемі өзгереді. Ұқсас әдіспен суреттер галереясы жасалады суреттің өзгеруі бір бет шекарасында жүргізіледі;

– уақыт пен күнді айқындау. Көптеген дизайнерлер өз жобаларында ағымдағы күн мен уақытты көп қолданады. Арнайы күнмен жұмыс істеуге арналған Date объектісі пайдаланылады;

– cookies файлдарымен жұмыс. JavaScript cookie файлдарымен толық жұмысты қамтамасыз етеді. Cookie- техникалық ақпарат сақталатын локальды компьютердегі мәтіндік файл. Бұл файлды оқырманның соңғы кіру күнін, парольдерді, сонымен қатар web-қосымшадағы басқа да әрекеттері туралы ақпарат сақталады. Аталған әрекет web-қосымшаны жекешелеп кірушілерге ыңғайлы етіп жасайды;

– оқиғаларды бақылау. Оқиға ретінде пайдаланушының белгілі бір әрекеті немесе құжат күйінің өзгеруі айтылады. JavaScript көптеген оқиғаларды бақылап отырады да оған жасалған әрекеттерді анықтауға мүмкіндік береді. Мысалы, web-бетті жүктегенде onLoad оқиғасы жүреді. Егер скриптті құжат жүктелгеннен кейін жүзеге асуы керек болса, онда осы оқиға басталғандағы болатын функцияны анықтау қажет;

– жаңа терезелерді құру. Скрипттер жаңа терезелерді құруға, оларға сыртқы пішінін және ішкі құрамын өңдеуге мүмкіндік береді. Жасалған терезелердің адрестік жолын алып тастауға, айналдыру жолағын жасыруға және экранда орналасу орнын беруге болады;

– браузерді тексеру. Түрлі браузерлерде объектілік модельдерді бейнелеу стандартталмаған, соңына дейін барлығы қолдамайды. Сол себепті универсалды құжатты жасау үшін, ағымдағы уақытта қандай браузер қолданыста екендігіне байланысты сәйкес кодты беру керек;

– математикалық функциялар. JavaScript қажетті арифметикалық операцияларды қамтиды, барлық стандартты математикалық функцияларды, бүтін мен үтірі жылжымалы сандарды, қолдайды;

– құжат құрамын басқару. Write() әдісімен кез келген ақпаратты жасалған құжатқа ендіруге болады, мысалы, бүгінгі күнді. Жаңа құжатты динамикалық құрастыруға болады. Бұл түрлі браузерлердің, операциялық жүйелердің ерекшеліктерін ескере, жеке мәтіндерді шығаруға мүмкіндік береді.

– әрине, мұнда JavaScript тілінің барлық мүмкіншіліктері жазылмаған. Дегенмен, айтылған мысалдардан web-беттің мүмкіндіктерін арттыратын универсалды және ыңғайлы құрал екендігін көруге болады.

Тіл мүмкіндіктері. JavaScript тілі объектті-бағытталған тілдің қасиеттерін иеленеді, бірақ прототипті тілде жүзеге асырылғандықтан басқа объектті-бағытталған тілдерден айырмашылықтары болады.

JavaScript тілінің Си тілінен кейбір айырмашылықтары:

- интроспекция мүмкіндігі бар объекттер;
- бірінші класс объектісі сияқты функциялар;
- типті автоматты түрде келтіру;
- қоқыстарды автоматты түрде жинау;
- анонимді функциялар.

Семантика және синтаксисі. JavaScript тілінің синтаксисі Си және Java тілдерінің синтаксисіне өте ұқсас. JavaScript тілі семантикалық жағынан Self, Smalltalk және Лисп тілдеріне жақын болып келеді.

JavaScript тілінде:

- барлық идентификаторлар тіркелуге тәуелді;
- айнымалылар атауларында әріптер, сызықшалар, доллар белгісін, араб цифрларын қолдануға болады;
- айнымалылар атаулары цифрлардан басталмайды;
- біржолдық түсініктеме жазу үшін // қолданылады, көп жолды және жол ішіндегі түсініктемелер /* басталып және */ аяқталады.

2.5 CSS бағдарламалау тілі

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 web-қосымшада 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[7].

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.

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.

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 немесе 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 (контекстік 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 дизайнерларының болуы.

Бөлек CSS-файлға мәліметтердің көрсету ережесі есебінен web-қосымша 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.

2.6 MySQL дерек қоры

MySQL – үлестірілген мәліметтер қорының жылдам және мықты басқару жүйесі. Ол ақпаратты тиімді түрде сақтауға, іздеуге, сұрыптауға және таңдауға мүмкіндік береді[9].

MySQL-ді ойлап тапқан ТсХ швед компаниясынан Михаил Видениус. Ол 1979 жылда UNIREG деп аталған деректер қорын басқару құралын жасады.

Бұдан әрі UNIREG үлкен деректер қорлары қолдауы үшін кеңейтілген және бірнеше тілде қайта жазылған. 1994 жылы ТсХ компаниясы www үшін UNIREG-ті пайдаланатын қосымша өңдей бастады. Алайда үлкен қосымша шығындар үшін UNIREG Web-беттер динамикалық генерациясына пайдаланыла алмады. Сондықтан Видениус MySQL-ді UNIREG-тегі B+ ISAM-ге қосу үшін, MySQL авторы Хьюзбен байланысуға бел буды. Алайда Хьюз MySQL 2 жолында алға басты және компания өз мұқтаждықтарына деректер базасының серверін жасауға шешті. ТсХ-та UNIREG-ті негізге алды және MySQL үшін бөтен өндеушілер утилитасын пайдаланды, өзжүйесі үшін API жазды, оның бастапқыда MySQL үшін API дәлкелетін. Алайда бұл Тсх деректер базасының серверіне өткісі келген MySQL-дың кез-келген қолданушысына, өз кодына өзгерістер енгізуге мүмкіндік беруші еді. Жаңа деректер қорының бастапқы коды толық біртума болды. Осыдан, 1995 жылының мамыр айында компанияда MySQL 1.0 деректер қоры қажеттік толық жеткілікті болды. Ал енді атауына келсек, онда Видениус бұл туралы осылай айтады: «MySQL-ның атауы қайдан шыққаны әлі күнге дейін белгісіз. ТсХ негізді тізбегінде, кейбір кітапханларда және утилиталарда он жыл шамасында “my,, префиксі күйінде белгілі болды. Сонымен қоса, менің қызымның да аты “My”. Сондықтан енді “MySQL,, деген сөз қайдан шыққаны белгісіз құпия болып қалады. MySQL ОС UNIX-тің көп БЖі, Win32 және OS/2 арқылы интернетке шығарылымның моментінен бастап деректер қоры жылдам дамитын платформадан тұрады, бұл осының дамуына қызыққан бағдарламашылар арқылы болады.

MySQL – тығыз көп тасқынды деректер базасының сервері. MySQL қолдануда үлкен жылдамдықпен, орнықтылықпен және жеңілдікпен бейнеленеді. MySQL керемет үлкен деректер қорын тез өндеуге болатын ішкі мұқтаждықтар үшін тсх компаниясымен жасалды. Компания MySQL-ді 1996-жылдан бастап қолданатындығын бекітеді, ол серверде 10,000 кестеден тұратын 40 шақты деректер қорын, оның 500 шамасындағысы 7 миллион жолдан тұрады. MySQL шағын және орта қосымшалар үшін тамаша шешім болып табылады. Сервердің түп нұсқалары платформалардың жиындарынан құрастырылады. Сервердің мүмкіндіктері көптасқындылықты қолдау және өнімділіктің түбегейлі өсуі бар, өте толық Unix серверлерде айқындалады. Қазіргі кезде MySQL әліде даму үстінде, алайда 3.22 версиясы толық іскер. MySQL – сервер коммерциялық емес қолдану үшін тегін болып табылады. Болмаса құны 190 EUR құрайтын лицензияның алу керек.

MySQL мүмкіндіктері. MySQL ANSI 92-ші стандарттағы SQL-дің сұрау салулар тілін қолдайды және бұдан басқа осыған ДҚБЖ бірде-біреуінде де жоқ болатын стандарттық кеңейтулерден тұрады.

MySQL-дің мүмкіндіктерінің қысқаша тізімі:

- деректер қорымен бір уақытта жұмыс істейтін алатын қолданушылар шексіз болу мүмкіндігі;
- кестелердегі жолдарының саны 50 миллионға жете алады;

– командалардың тез орындауы. Қазіргі уақытта бар барлық серверлер ішінен MySQL ең тез болуы мүмкін;

– қауіпсіздіктің оңай және тиімді жүйесі.

MySQL кемшіліктері. MySQL шындығында тез жұмыс істейтін сервер, бірақ бұл үшін өңдеушілерге табыс үшін реляциялық ДҚБЖға кейбір талаптарымен садақа етуге тура келді.

MySQL-де болмайды:

SELECT * FROM table1 WHERE id IN (SELECT id FROM table2) типіндегі қабаттасқан сұрау салуларды қолдайды. 3.23 версиясында осындай мүмкіндігі болатындығы бекиді;

транзакциялардың қолдауы іске асырылмаған. Оның орнына TABLE LOCK/UNLOCK-тарды пайдалануға ұсынылған;

(foreign) сыртқы кілттердің қолдаулары жоқ;

триггерлер және сақталатын процедуралардың қолдаулары жоқ;

(VIEW) ұсыныстардың қолдаулары жоқ. 3.23 версиясында ұсынысты құру мүмкіндігі жоспарланады.

Жасаушылар пікірі бойынша биік жылдамдыққа жету үшін мүмкіндік берді. Оларды іске асыру сервердің жылдамдығын айтарлықтай төмендетеді. Бұл мүмкіндіктер биік жылдамдығымен және және арзан бағамен серверге үлкен атақтылық бермеді, сын көзімен Web-қосымша жасауда критикалық болып есептелмейді.

3 Web - қосымша құруда жобалау және оны іске асыру

3.1 Web - қосымша құру кезеңдері

Web - қосымша концепциясы – жобаның мақсаты мен міндеттері тапсырыс берушімен бірге ұйғарылады, керек болса техникалық тапсырма құрастырылады. Бұл кезеңде баға мен дайындау мерзімі белгілі болады.

Web - қосымша дизайны – бұл кезең маңызды болып келеді, өйткені тапсырыс беруші компанияның фирмалық стилі ескеріледі, егер фирмалық стиль болмасы біз фирмалық стиль жасау қызметін де көрсетеміз. Web - қосымша басты парағының дизайнын мақұлдаған соң, қалған парағының суреті салынады, олар: каталог, байланысу парағы т.б.

Беттеу – беттеуден кейін web-қосымшаны барлық браузерлерде тексеруден өткіземіз.

Мағлұмат толтыру – Web қосымша мәтіндер және суреттер мағлұматтарымен толтырылады, осымен қатар web-қосымша модулдері тексетуден өтеді. Дайын жоба тексеруге арналған доменге (домен.invision.kz) бекітіліп тапсырыс берушіге көрсетіледі.

Жобаны тапсыру – web-қосымша тапсырыс беруші таңдаған доменге бекітіліп, web-қосымшаны басқару жүйсін қолдануды таспырушыға үйретілген соң актқа қол қойылады.

3.2 Жобаны іске асыру мерзімі

Жобаны іске асыру мерзімі келесі қадамдардан тұрады:

1-қадам. Бизнес жоспар даярлау– сауалнама жүргізу, техникалық тапсырма жазу, қадамдарды құру, жұмыстың орындалу мерзімі мен бағасы, келісім шарт жасасу;

2-қадам. Web-қосымшаның концепциясын жасау – техникалық тапсырмаға байланысты web-қосымшаның концепция жасалады – ондаған әртүрлі идеялар қарастырылады, сонын ішінде ең жақсысы таңдалып алынады. Web-қосымшаның архитектурасы жасалады;

3-қадам. Дизайнның өңделуі– таңдалып алынған web-қосымша концепциясына эксклюзивті дизайн жасалады;

4-қадам. Анимацияны программалау – Беттеу, анимацияны программалау, басқару жүйесін баптау (настройка), web-қосымшаны ақпаратпен толтыру, доменді атты тіркеу, хостингке орналастыру;

5-қадам. Web қосымшаны интернетте жариялау – барі дайын болған кезде тесттен өткізу, жетілдіреміз және web-қосымшаны интернетте жариялаймыз;

Вход в систему

Email address

Password

Вход

3.2 - сурет – Web-қосымшаға кіру беті

Email address-ті дұрыс енгізбеген жағдайда – 3.3 – сурет көрсетілген.

Вход в систему

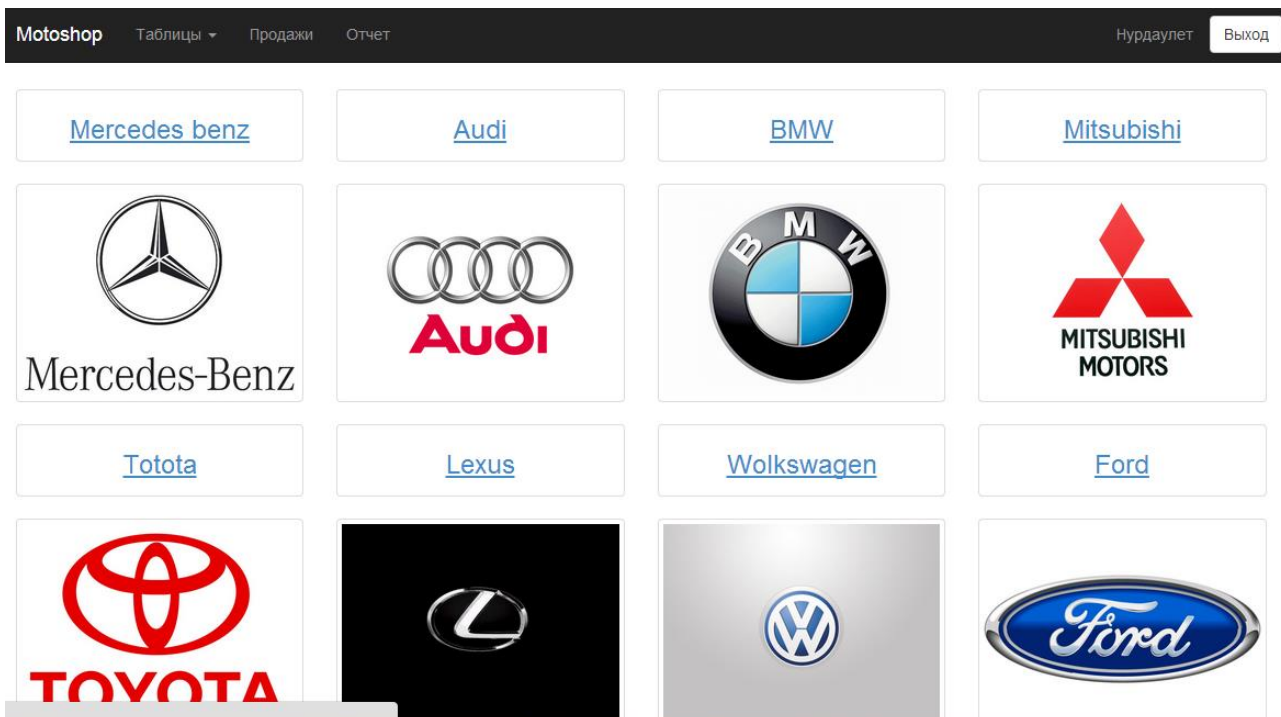
nk

.....

Адрес электронной почты должен содержать символ "@". В адресе "nk" отсутствует символ "@".

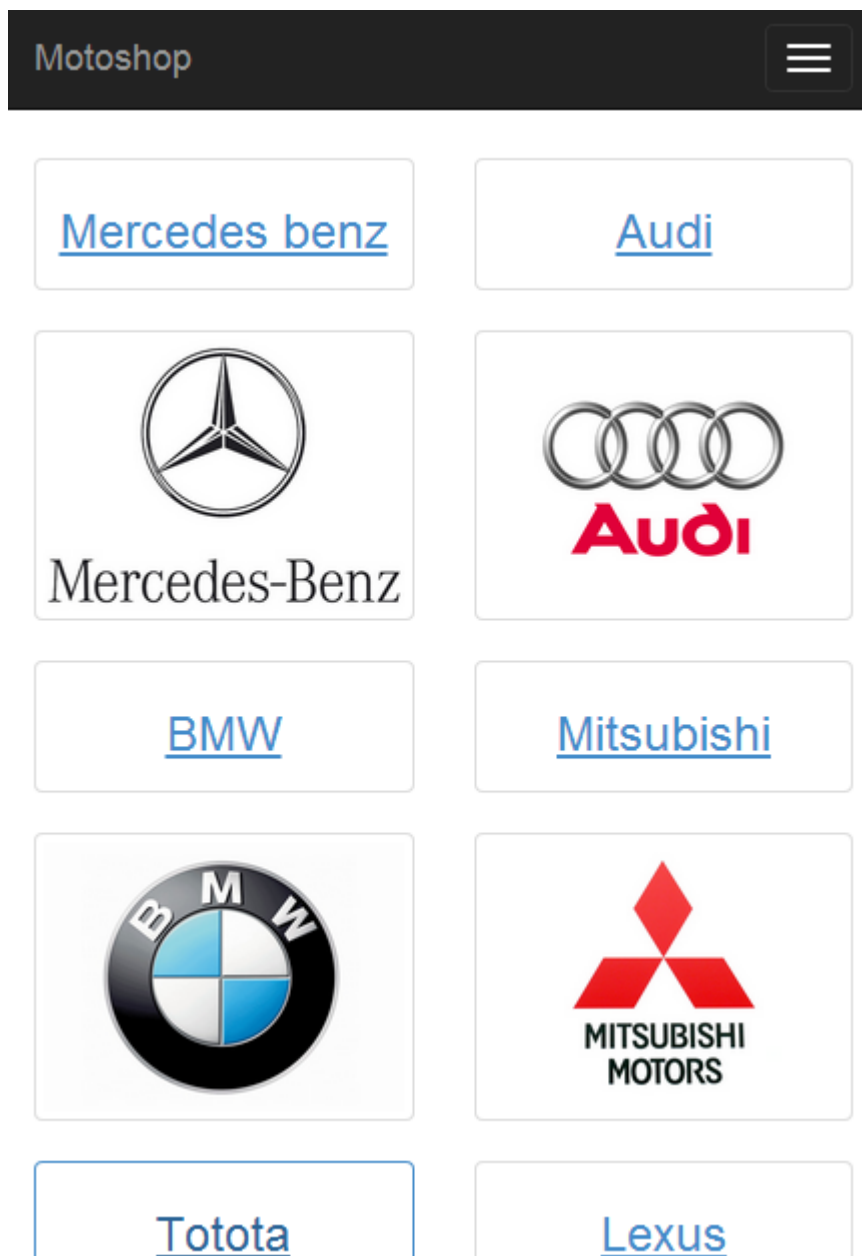
3.3 - сурет – Web-қосымшаға кіру бетіне информацияны қате енгізген жағдайда

Web-қосымшаның басты беті 3.4 – суретте көрсетілген. Бұл web-қосымшаға сіз тұрақты admin ретінде кіруіңізге болады. Басты бетті автокөлік маркаларын көре аласыз. Бұл жерде базада бар автокөлік маркалары ғана көрсетіледі. Тағы да айта кететін болсақ мұнда бүкіл кестелерге сілтемелер бар және де кірген адамның есімі көрсетіледі.



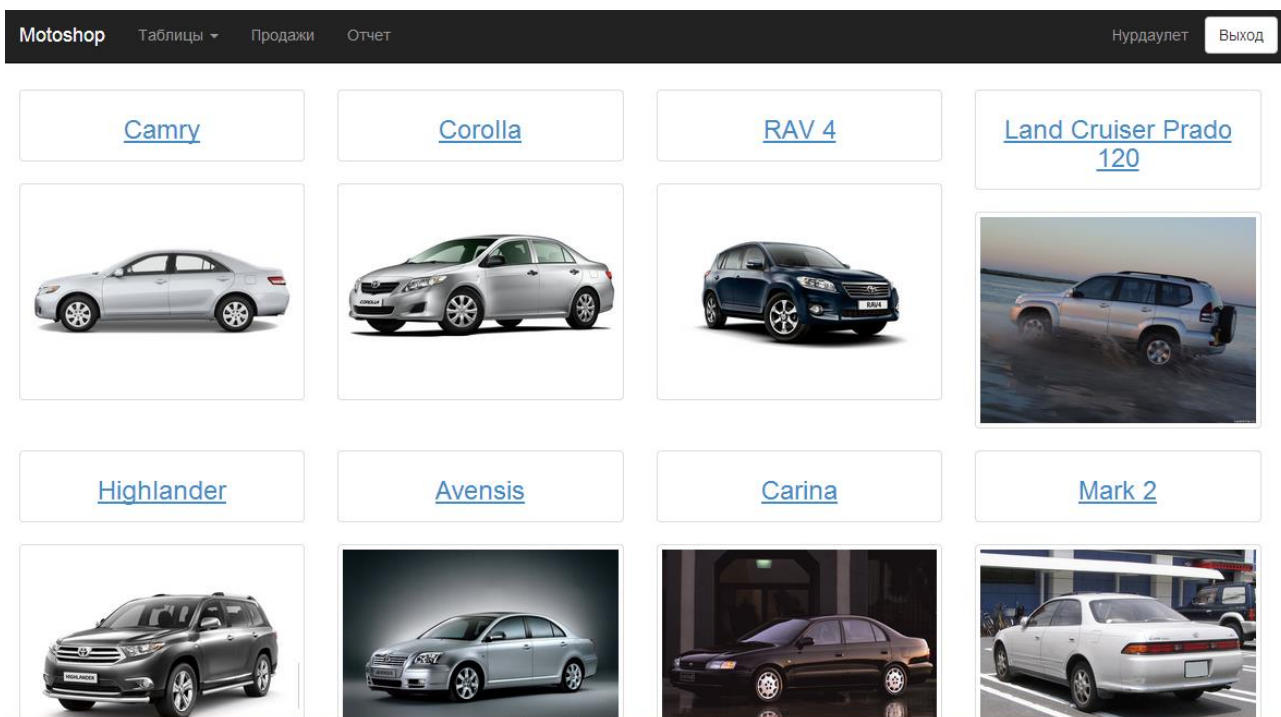
3.4 - сурет – Басты бет

Web-қосымшамыздың басты беті басқа да операциялық жүйелер мен платформаларға келістіріп жасалған. Қазіргі кезде кеңінен дамып жатқан мобильдік платформаларды 3.5- суреттен көре аласыз.



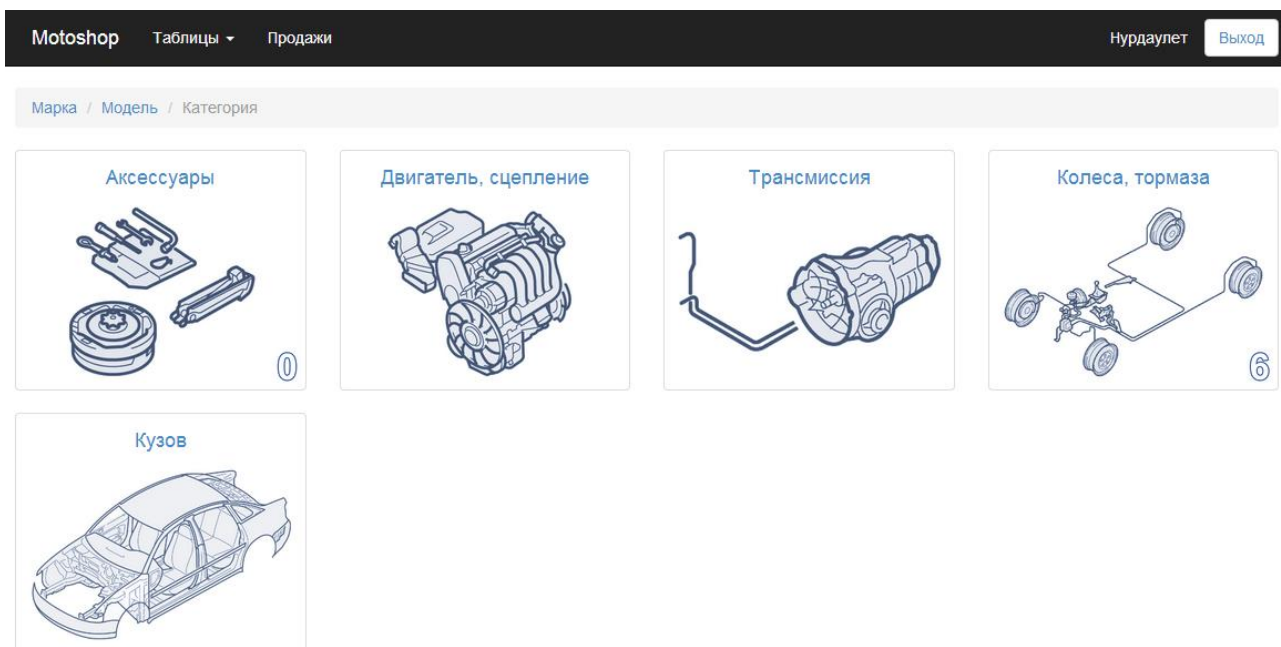
3.5 - сурет – Басты бет мобильдік платформаларда көрінуі.

Автокөлік маркасын таңдаған соң, оның модельдер қатары – 3.6 суретте көрсетілген.



3.6 - сурет – Модель

Кез келген автокөлік автобөлшектер жиынтығынан тұрады. Сол автобөлшектерді бір ретке келтіру үшін категорияларға жинайды. Ол 3.7 суретте көрсетілген.




3.7 - сурет – Автобөлшектер категориялары

Автобөлшектер қатары 3.8-суретте көрсетілген. Автобөлшектер қатарында керекті барлық ақпарат бар.

Motoshop Таблицы ▾ Продажи Нурдаулет Выход


Марка / Модель / Категория / Запчасть

Болт головки блока к-т 06B 103 385



Код запчасти: 2
Категория: Двигатель, сцепление
Каталожный №: 101200
Модель: Audi Q7
Примечания: 4,2 л.
Поставщик: Тулубаев Аяз
Количество: 1
Цена покупки, тенге: 150000
Цена продажи, тенге: 180000

Болт головки цилиндра CYLINDER HEAD SCREW



Код запчасти: 3
Категория: Двигатель, сцепление
Каталожный №: 101200
Модель: Audi Q7
Примечания: 5,2 л.
Поставщик: Тулубаев Аяз
Количество: 15
Цена покупки, тенге: 21000
Цена продажи, тенге: 26000

3.8 - сурет – автобөлшектер қатары

Сатылған автобөлшектер тізімін 3.9 - суреттен көре аласыз.

Motoshop Таблицы ▾ Продажи Нурдаулет Выход

Таблица - Запчасти Добавить

Код запчасти (kod_zapchasty)	Наименование (zapchast_name)	ID Категорий (id_kategory)	ID Каталога (katalog_id)	ID Модели (model_id)	Примечание (primecheniya)	ID Поставщика (supplier_id)	Количество, шт (kol)	Цена покупки, тенге (purchase)	Цена продажи, тенге (sale)	Действие
2	Болт головки блока к-т 06B 103 385	12	101200	1	4,2 л.	1	1	150000	180000	Редактировать Удалить
3	Болт головки цилиндра CYLINDER HEAD SCREW	12	101200	1	5,2 л.	1	15	21000	26000	Редактировать Удалить
4	Тормозные колодки	16	101203	1	Колодки в комплекте	1	2	3000	5000	Редактировать Удалить
6	Фары	11	1651	1	фывф	1	1231	15615	15	Редактировать Удалить
7	Шины R15	13	612	1	вфы	1	61	135	161	Редактировать Удалить
8	Шины R16	14	101204	1	5,2 л.	1	15	150000	25000	Редактировать Удалить
9	Цилиндр блока	15	101203	1	5,2 л.	1	16	65	1465	Редактировать Удалить

3.9 - сурет - сатылған автобөлшектер

Біздің деректер базасына жаңа ақпарат қосу терезесін 3.10 – суреттен көре аласыз.

Motoshop Таблицы ▾ Продажи

Добавить новую запчасть:

kod_zapchasty

zapchast_name

zapchast_image_name Файл не выбран

id_kategory ▾

katalog_id

model_id ▾

primecheniya

supplier_id ▾

kol

purchase

sale

3.10 - сурет –жаңа автобөлшек қосу беті

Деректер базасында бар ақпаратты өңдеу терезесін 3.11 - суреттен көре аласыз.

Motoshop Таблицы ▾ Продажи

Редактирования записи:

kod_zapchasty

zapchast_name

zapchast_image_name Файл не выбран

id_kategory

katalog_id

model_id

primecheniya

supplier_id

kol

purchase

sale

3.11 – сурет – Автобөлшек өңдеу беті

Отчет шығару беті. Мұнда компанияның сатылымы бойынша отчеттар шығаруға болады және де бұл отчеттарды каталог, категория, модель бойынша шығаруға болады. Оны 3.12 - суреттен көруге болады.

Motoshop Таблицы ▾ Продажи Отчет Нурдаулет

Поиск ID каталога ID категорий ID модели

Отчет								
Наименование	Код запчасти	ID каталога	ID категорий	Примечание	ID модели	ID поставщика	Количество, шт	Цена покупки, тенге
Болт головки блока к-т 068 103 3865	12	101200	1	4.2 л.	1	1	150000	180000
Тормозные колодки	12	101200	1	5.2 л.	1	15	1500	1800
Фары передние	12	101200	1	Колодки в комплекте	1	2	3000	5000

3.12- сурет – Отчет

4 Тіршілік қауіпсіздігі

4.1 Программалық өнімді жасау кезінде зиянды, потенциалдық қауіпті факторларды талдау.

Берілген дипломдық жобаның тақырыбы – «Motoshop» ЖШС ақпараттық жүйесін автоматтандыру. Жасау компьютерлік техникаларды қолданылуымен, ал аңығырақ ноутбукпен жүзеге асырылады. Қарастырылатын жайларда өзінің жұмыс орны бар 1 қызметкер жұмыс істейді.

Программалық қамтамасыздандыруды жасау кезінде үлкен рөлін жұмыс жайындағы еңбек жағдайларын дұрыс ұйымдастыруы алады. Еңбек жағдайларын талдай отырып, компьютермен жұмыс істеу кезінде адам бірқатар қауіпті және зиянды өндірістік факторлар – электромагниттік өрістердің, инфрақызыл және иондайтын сәулелердің, шу мен дірілдің, статикалық электр қуатының және тағы басқалардың әсерлеріне шалдығатынын естен шығармағаны жөн.

Аз шулайтын құрылғылардың – жоғарылатылған шу сапасында зияндылығы болмайды. Электромагниттік сәулеленудің көтерілген деңгейі жоқ болады. Ғимарат отқа төзімділіктің (СНиП ҚР 2.02-05-2002) I дәрежесіне жатады (жанбайтын жапырақ материалдарын қолдануымен бетон немесе темірбетон, табиғи немесе жасанды материалдардан алып жүретін және қорғайтын құрылымдарымен ғимараттар). Өрт қауіпсіздігі бойынша жұмыс жайлары “Д” тап қатарына жатады. Өрт қауіпсіздігінің түр ережелерімен сәйкестігінше әкімшілік ғимараттар мен бөлек жайлар және технологиялық қондырғылар қалып мөлшерлеріне сәйкес өрт сөндіргіштің алғашқы заттарымен қамтамасыз етіледі.

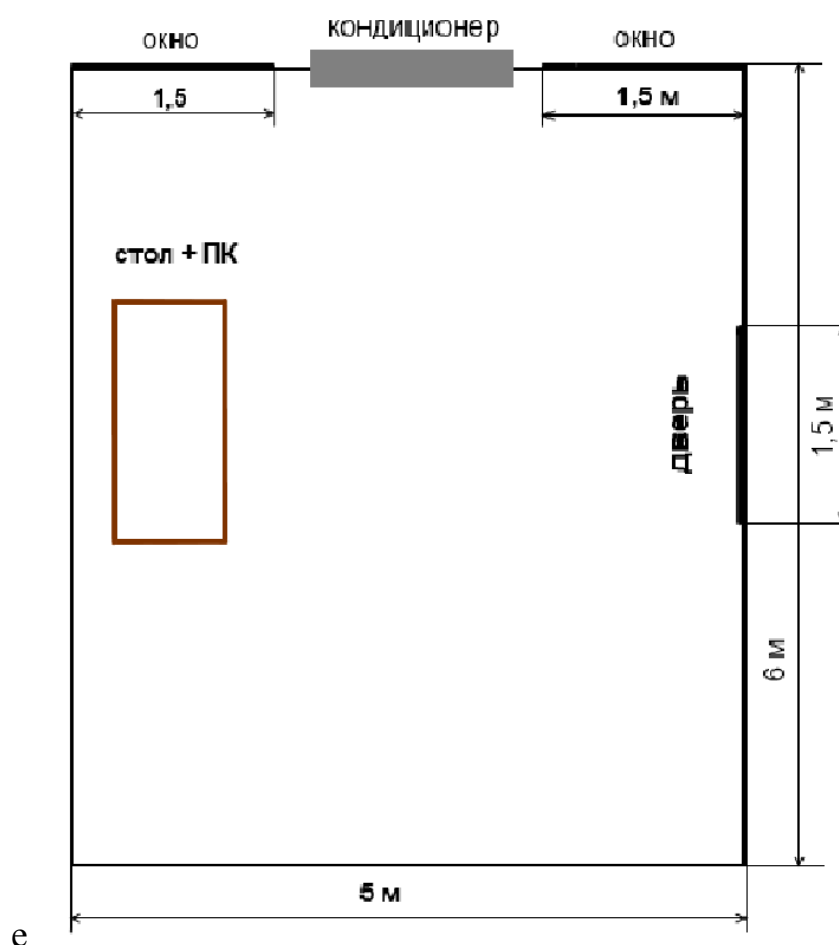
Бағдарламаны қамтамасыз ету және жасау кезінде, басты рөлді жұмыс істеу орнына бөледі, себебі оператор аптасына 5 күн жұмыс істейді, 8 сағаттан, яғни ол компьютердің алдында аптасына 40 сағат жұмыс істейді, оған табиғи жарықты және жасанды жарықты түсу нормаларын ескеру қажет.

Жайда екі терезе орналасуы және табиғи жарық қалыптарға сәйкес келеді. Табиғи жарық барлық жұмыс уақытының ауқымында, әсіресе қысқы уақытта (жұмыс тәртібі: 9:00 ден 18:00 дейін) қажетті жарықтандыруды қамтамасыз ете алмайды, сондықтан жұмыс жайында люминесценттік шамдармен шырақшамдардан тұратын жасанды жалпы жарықтандырудың жүйесі қарастырылған.

Жұмыс орынының мінездемесі.

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 жайларының жоспары 4.1- суретте көрсетілген:



4.1 - сурет – Жайдың жоспары

Жайлар келесі параметрлеріне ие болады:

- бес қабатты ғимараттың бірінші қабатында болу;
- жайлардың (бөлмелердің) өлшемдері: ұзындығы 5 м, ені 6 м, биіктігі 3 м;
- жарық өткізетін материалдардың түрлері – жапырақты, қосарлы әйнек;
- есік – ашылатын ағаш;
- күннен қорғайтын құрылғы – жиыстырыла реттелетін тор қақпақ (жалюзи) және перделер;
- 1,5*2 өлшемімен терезе;
- Жай 1 Samsung ауабаптағышына ие;
- қабырғаның ішкі әрлеуі – ашық;
- жұмыстың көрермендік жағдайлары бойынша жай жеңіл жұмыс (жеңіл физикалық, 1а санаты) санаттарына;
- жасанды жарық – төрт люминесценттік шамдар.

- Жұмыс істеу кезіндегі қолданылатын құрылғының мінездемесі:
- Intel(R) core 2Duo CPU E8400 @ 3.0 GHz 3.0 GHz, 4 ГБ RAM / HDD 1000 Gb;
- монитор Samsung SyncMaster 932 BF, диагональ 19”;
- 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, шамдалдың қуатты 2x28 Вт.

Электротехникалық жабдық, ең бірінші өрт көзі болып табылады және қауып тудырады. Жұмыс уақыты біткеннен кейін барлық электротехникалық жабдықтарды өшіруіміз қажет. Құралдарымыз аз шуылдайтын және зиянды ара сапалы шуы болмайды.

Берілген бөлmemіз барлық электр қауіпсіздігі қанағаттандырады, яғни бөлме ылғал емес 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.

Жұмыс орындарын ұйымдастыруына ерекше талаптар ұсынылады:

- бір адамға жұмыс орын 6 м² кем емес, ал көлемі –24 м³ кем емес болуы керек;
- терезеге компьютер экранымен орналастыруына рұқсат етілмейді, өйткені бұл монитор экранына дақтар мен көлеңкені туғызады.

Алынған мәліметтерге негізделе орындалатын жұмыстардың түрлерін саралауға болады.

Жұмыс орындарын бағалау жүргізейік:

- орындалатын жұмыс жеңіл физикалық санаттарына жатады, I а санаттағы жұмыс отыру жағдайында жүргізіледі және физикалық күштенуді (ГОСТ 12.2.032-78) керек етпейді;
- жұмыс орындары 30 м² және көлемі– 90 м³ құрайды;
- жұмыскерлердің саны – 1.

4.2 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.

Микроклиматтың күйін қадағалау оператор 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

К е с т е 4.1 – Жұмыс орнындағы нормаланған климаттық параметрлер

Жыл мезгілі	Ауа температурасы, °С	Ауа ылғалдылығы, %, аса емес	Ауа қозғалысының жылдамдығы, аса емес м/с
Жылы	20-22	60-40	0,1
Суық	20-22	45-30	0,1

Жайда жайлы еңбек жағдайларын жасау үшін ауабаптағыштардың қажетті сандарын анықтаймыз. Жылдың жылы мезгілдері үшін әдістемелер бойынша есептеуді жүргіземіз. Жайда жылу бөлу арқасында өндірістік құрылғылар айтарлықтай артық жылу (жайда жылу бөлінуімен және қабырға, терезе, есік пен т.б. арқылы жылу беру арасындағы айырмашылық) орнына ие болуы мүмкін, бәрінен бұрын осыларды жоюын желдету жүйелері қамтамасыздандыруы тиіс.

Біздің жайымызда артық жылуды сіңіру үшін қажетті м³/сағ ауа ағынының мөлшерін елеулі жылу бөлінуімен есептейміз.

Ең алдымен аппаратурадан жылу бөлінуін анықтаймыз:

$$Q_1 = \eta_1 \cdot \eta_2 \cdot \eta_3 \cdot \eta_4 \cdot N_{\text{НОМ}} \quad (5.1)$$

$$Q_1 = 0,8 \cdot 0,6 \cdot 0,6 \cdot 0,75 \cdot 1500 = 540 \text{ кДж/сағ}$$

мұндағы $\eta_1 = 0,8$ – номиналды қуатты пайдалану коэффициенті;

$\eta_2 = 0,6$ – жүктеу коэффициенті, яғни орташа тұтынылатын қуаттың барынша қажеттіге қатынасы;

$\eta_3 = 0,6$ – аппаратураның бір уақыттағы жұмыс коэффициенті;

$\eta_4 = 0,75$ – жылуға айналып кеткен механикалық энергияның үлесін сипаттайтын коэффициент;

$N_{\text{НОМ}} = 1500$ кДж/сағ – барлық аппаратуралардың номиналды қуаттары.

Жасанды жарық көздерінен жылу бөлінуін іс жүзінде барлық шығындалатын қуаттың (энергияның) ақырғы нәтижесі формула бойынша жылуға айналатынын болжай есептеп шығарады:

$$Q_2 = \varphi \cdot N_{\text{ОСВ}} \quad (5.2)$$

$$Q_2 = 0,8 \cdot 4 \cdot 60 = 192 \text{ Дж/сағ}$$

мұндағы $\varphi = 0,8$ – жылуға өтетін энергия мөлшерін ескеретін коэффициент;
 $N_{\text{ОСВ}} = 4 \cdot 60 = 300 \text{ Вт}$ – жайға жарық беруші қондырғылардың қуаттылығы (әрбіреуі 60 Вт бойынша 4 шам).

Адамдардан жылу бөлінуді формула бойынша анықтайды:

$$Q_3 = n \cdot q_n \quad (5.3)$$

$$Q_3 = 1 \cdot 120 = 120 \text{ Дж/сағ}$$

мұндағы $n=1$ – жұмыскерлердің саны;

$q_n = 120 \text{ Вт}$ – бір адамнан бөлінетін жылу мөлшері.

Терезе арқылы жайға келіп түсетін күн радиациясынан бөлінетін жылу мөлшері мына формула бойынша анықтайды:

$$Q_4 = F_{\text{ОСТ}} \cdot q \cdot m \cdot k \quad (5.4)$$

$$Q_4 = 3 \cdot 224 \cdot 2 \cdot 1,25 = 1680 \text{ Дж/сағ}$$

мұндағы $F_{\text{ОСТ}} = 1,5 \cdot 2 = 3 \text{ м}^2$ – терезе ауданы;

$q=224 \text{ Вт/м}^2$ – 1 м^2 терезе (терезе оңтүстікке қаратылған) арқылы жылу түсімі;

$m=2$ – терезе саны;

$k=1,25$ – металдық мұқаба үшін түзету көбейткіші;

Шамадан тыс жылу бөлінуін формула бойынша анықталады:

$$Q_{\text{ИЗБ}} = Q_1 + Q_2 + Q_3 + Q_4 \quad (5.5)$$

$$Q_{\text{ИЗБ}} = 540 + 192 + 120 + 1680 = 2532 \text{ Дж/сағ}$$

мұндағы Q_1 – аппаратурадан жылу бөлінуі;

Q_2 – жарық көздерінен жылу бөлінуі;

Q_3 – адамдардан жылу бөлінуі;

Q_4 – терезе арқылы күн радиацияларынан жылу түсімі.

Артық жылуды жұту үшін қажетті ағындық ауа мөлшерін анықтайық:

$$L = \frac{\Sigma Q_{\text{ИЗБ}}}{c \cdot (t_{\text{уд}} - t_{\text{пр}}) \cdot \gamma_{\text{пр}}} \quad (5.6)$$

$$L = \frac{2532}{1 \cdot (27 - 20) \cdot 1,2} = 301,4 \text{ м}^3/\text{сағ}$$

мұндағы $Q_{изб} = 2532 \text{ Дж/сағ}$ – Айқын жылудың шамадан тыс бөлінуі;

$C=1 \text{ Дж/(кг*град)}$ – ауаның жылу сыйымдылығы;

$t_{уд} = 27 \text{ }^\circ\text{C}$ – жайлардан жойылатын ауаның температурасы;

$t_{пр} = 20 \text{ }^\circ\text{C}$ – ағындық ауаның температурасы;

$\gamma_{пр} = 1,2 \text{ кг/м}^3$ – ағындық ауаның тығыздығы, $1,2 \text{ кг/ м}^3$.

Алынған есептерімен сәйкестігінше ауабаптағышты таңдаймыз. Ауа ағынын қамтамасыздандыру және осы мақсаттарды орындау үшін қабырғаға ілетін ауабаптағышын Samsung MH18ZC2 (5.2-сурет) қолданамыз.



4.2 – сурет – Samsung MH18ZC2 ауабаптағышы

Samsung MH18ZC2 ауабаптағышының топтамасы жайдың 60 м^2 дейінгі ауданы үшін арналған. Бұл ауабаптағыштары ауаның берілген температуралық параметрлерін шапшаң қамтамасыздандыра салқындату және жылытудың жоғарылатылған тиімділігімен ерекшеленеді, әлдеқайда зиянды қосылыстарды – аммиак, формальдегид, күкіртсутек ауадан алып тастайды.

Белсенді көмір сүзгілері шаңдардың өте ұсақ түйіршіктерін ұстап және жағымсыз иістерді жояды. Ауабаптағыштың техникалық сипаттамалары 5.2-кестесінде көрсетілген.

К е с т е 5.2 – Ауабаптағыштың (кондиционердің) сипаттамасы

Атауы	Сипаттамасы
Ауданы аса емес, ш. М	60
Д/б тізбесандықшасы	Бар
Салқындату/қыздыру қуаты, кВт	5,26/5,56
Қоректенуі (В/Гц)	220/50
Салқындату/қыздырудың тұтынылатын қуаты, кВт	1,78
Суық/жылудың жұмыс тоғы, А	9
Шудың деңгейі, дБ	37
Ауа шығыны аса емес, м ³ /мин (м ³ /сағ)	10 (600)

Ауа шығының қамсыздандыру үшін $L=301,4 \text{ м}^3/\text{сағ}$, бізге тек1 кондиционер фирмасы Samsung сериясы НА ауаның максималды шығыны $600 \text{ м}^3/\text{сағ}$, моделі MN18ZC2.

4.3 Табиғи жарықты есептеу

Табиғи жарық барлық жұмыс уақытының барысында қажетті жарықты қамтамыз ете алмайды, өйткені ауа райы өзгеріп кетуі мүмкін немесе қараңғылап жатқанда және табиғи жарық жеткіліксіз мүмкін кеш уақытында жұмыстар болуы мүмкін, сондықтан да жұмыс орнында люминесценттік шамдармен жабдықталған шырақшамдардан тұратын жалпы жарықтандырудың жасанды жүйесі қарастырылған. Жарық көздерінің қалып мөлшерлері 5.3 кестесінде көрсетілген. Берілген жайында қалыпты жұмыстың орындалуы үшін қажетті жарықтылық: 400 лк.

Кесте 5.3 – Ортақ жарықтандыру жүйесіндегі ұсынылған жарық көздері

Түс айырмашылығына талап ету бойынша көрермендік жұмысының сипаттамасы	Жарықтылығы, лк	Жарық көзіндегі ауқым (диапазон) түстерінің температуралары $T_c, ^\circ K$	Жарық көзінің қолданылатын түрлері
Түс айырмашылығына көп емес талаптары кезінде түсті нысандардың айырмашылықтары	300, 400	3500 – 5500	ЛД, ЛДЦ, ЛБ,
	150, 200	3000 – 4500	ЛБ, (ЛХБ), НЛВД+МТЛ, ДРЛ

Жұмыс орнында мөлшерленетін жарықтылықты жасау үшін қажетті жайдың бүйірлік жарығы арқылы ойықтардың ауданын есептейміз.

Жай мынадай өлшемдеріне ие: ұзындығы $L = 6 \text{ м}$, ені $B = 5 \text{ м}$, биіктігі $H = 6 \text{ м}$. Еден деңгейінен жұмыс беткейінің биікті h_{pp} , $h_{pp} = 0,725 \text{ м}$, терезе $h_{но}, h_{но} = 0,8 \text{ м}$ биіктігінен басталады, терезе биіктігі h_o , $h_o = 2 \text{ м}$. Жұмыс жайы IV сағаттық белдеуде – Алматы қаласында табылады (жарық климатының белдеуі – IV 50° солтүстік ендігі және оңтүстік (Алматы, Қарағанды)).

Жұмыс орны терезелік ойықтарды жобалайтын жайының сыртқы қабырғасынан l_{pt} , $l_{pt} = 1,5 \text{ м}$ орналасқан. Терезелік ойықтан 4 м қашықтықта тұратын нүктеде ең төменгі жарықтылық болады.

Терезелердің жалпы ауданын $S_0, \text{ м}^2$, формула бойынша анықтаймыз

$$100 \cdot \frac{S_0}{S_n} = \frac{e_n \cdot \eta_0}{\tau_0 \cdot r_1} \cdot k_{зд} \cdot k_3 \quad (5.7)$$

$$S_0 = \frac{S_n \cdot e_n \cdot \eta_0}{100 \cdot \tau_0 \cdot r_1} \cdot k_{зд} \cdot k_3 \quad (5.8)$$

мұндағы S_n – жайдың ауданы m^2 ;

$$S_n = 30 m^2;$$

e_n – КЕО мөлшерленген 5.4 кесте бойынша таңдалатын мағынасы.

Кесте 5.4 – Тұрғын үй жайларындағы (СНиП ҚР 2.04.-05-2002) жасанды, табиғи және біріктірілген жарықтардың мөлшерленетін көрсеткіштері

Жайлар	Шет жағынан жарықтандыруы кезінде табиғи жарығы КЕО e_n , %
Әкімшілік ғимараттары: Кабинеттер, жұмыс бөлмелері, кеңселер, өкілдік	1,2

Көрсеткіштік жұмыстың жоғары дәлдігі үшін $e_n = 1,2$ қабылдаймыз.

m_N – 5.5 кесте бойынша көрсетілетін жарық климатының коэффициенті.

Кесте 5.5 – Климат коэффициенттерінің мағыналары

Көкжиек жағымен бойынша жарық ойықтарының бағдарлануы	Жарық климатының коэффициент, m
С	0,9

Берілген жарық белбеуін ескере, Солтүстіктегі жарық ойықтарының бағдарын қабылдап анықтаймыз: $m_N=0,9$.

$$e_n^{IV} = e_n \cdot m \cdot c \quad (5.9)$$

мұндағы $e_n^{IV} = 1,2 \cdot 0,9 \cdot 0,75 = 0,78$;

$c = 0,75$ (ғимараттардың сыртқы қабырғалары);

Жай түрлерін ескере, қор коэффициенті $k_3 = 1,4$ тең болады (кабинеттер және жұмыс орындары, зертхана, конструкторлық бюро);

τ_0 - жарық өткізудің жалпы коэффициенті бірдей

$$\tau_0 = \tau_1 \cdot \tau_2 \cdot \tau_3 \cdot \tau_4 \quad (5.10)$$

мұндағы

- $\tau_1 = 0,5$ (қуыс денелі шыны блоктар);
- $\tau_2 = 0,6$ (ағашты бөлек қосарлы мұқабалар);
- $\tau_3 = 0,8$ (темірбетонды фермалар және аркалар);
- $\tau_4 = 1$ (жиналатын реттелетін жалюздер және перделер);
- $\tau_0 = 0,5 \cdot 0,6 \cdot 0,8 \cdot 1 = 0,24$;
- η_0 - терезенің жарықтық сипаттамасы.

Анықтаймыз η_0 :

$$l = B - 1,$$

$$l = 5 - 1 = 4\text{м},$$

$$\frac{L}{l} = \frac{L}{B-1} = \frac{6}{4} = 1,5.$$

$$h_{\text{расч}} = h_{\text{НО}} + h_0 - h_{\text{рп}} \quad (5.11)$$

$$h_{\text{расч}} = 0,8 + 2 - 0,72 = 2,08,$$

$$\frac{B}{h_{\text{расч}}} = \frac{5}{2,03} = 2,5.$$

Ені жайдың тереңдігіне қатынасы бойынша $\eta_0 = 10$ аламыз.

τ_1 – ғимаратқа тірілетін төсемдік қабаттардың және жайдың беткейінен шағылған жарық арқасында бүйірден жарық түсіру кезінде КЕО жоғарылауын ескеретін коэффициент.

Жайдағы шағылысудың коэффициенті $\rho_{\text{СР}} = 0,5$, бір жақты бүйір жарығын қабылдаймыз.

$$\frac{\rho_{\text{ст}}}{B} = \frac{0,5}{5} = 0,1$$

онда $\tau_1 = 1,05$;

$k_{\text{зд}}$ – қарсы алдында тұратын ғимараттарымен терезелердің көлеңкелеуін ескеретін коэффициент.

Дегенімен жакын арада көлеңке туғызатын ғимараттар жоқ, онда $k_{\text{зд}} = 1$

Терезенің жалпы ауданыны есептеп шығарайық:

$$S_0 = \frac{30 \cdot 1,4 \cdot 10 \cdot 1 \cdot 0,78}{100 \cdot 0,24 \cdot 1,05} = 13$$

Жарықтық ойықтардың ауданы $S_{\text{от}} = 13 \text{ м}^2$

Осындай жағдаймен, жарықтық ойықтардың ауданы ($1,5 \cdot 2 \cdot 2 = 7 < 13$) жұмыс орнында қажетті еңбек жағдайларын қамтамасыз ете алмайды. Көрермендік жұмыстың III сипаттамасы кезінде КЕО, $e_N = 0,84$ мөлшерленген мағынасын қамтамасыздандыру үшін $5 \times 6 \times 3$ параметрлерімен жайдағы әлдеқайда жағымды еңбек жағдайларын жасау мақсатында табиғи жарықпен бірлесе жасанды жарық пайдаланылады.

4.4 Тіршілік қауіпсіздігі бөлімі бойынша қорытынды

Бұл бөлімде офистік компаниядағы жұмыс шарты және микроклимат параметрін есептеуді орындау, жағдайын тексеру мақсатында анализ жасалды. Еңбек шарты деңгейі дұрыс және есептеу бойынша алынған есептеулер, тіршілік қауіпсіздік бойынша стандарттарды қанағаттандырады.

Нәтиже бойынша, екі терезе $1,5 \times 1,2 \text{ м}$ өлшемді жұмыс орнындағы табиғи жарығы нормаларды қанағатандырмайды. Жұмыс орнында жарықты қамтамасыз ету үшін жасанды жарықты қолданамыз.

Есептеу нәтижесі бойынша, бөлмеге қажетті ауа мөлшерін қамтамасыз ету үшін $L = 301,4 \text{ м}^3/\text{сағ}$ байланысты біз тек 1 кондиционер қойамыз, ол Samsung сериясы МН максимальды ауаның шығыны $600 \text{ м}^3/\text{сағ}$, модель МН18ZС2.

5 Техникалық экономикалық негізделуі

5.1 Бизнес жоспар

Бұл бағдарламада “Motoshop” ЖШС ақпараттық жүйесін автоматтандыруға арналған бағдарлама. Информациялық технологиялар нарығының дамуы Қазақстан Республикасында басқа салаларға қарағанда динамилалық түрде жүруде. Қазіргі кездегі информациялық-анықтама, коммуникация, іздеу, үйрету және ойын сауық ситамалары информациялық технологиялар базасында құрылуда.

Бұл бағдарламалық жобаның экономикалық маркетингтік стратегиясын құруда көптеген мәліметтерді ескере отырып жүзеге асырамыз. Қазіргі заманғы экономикалық қажеттіліктерді, экономикалық талаптарды, нарық қажеттіліктерін ескереміз. Бұл бағдарлама бір компанияға арналып жасалғанына қарамастан, келешекте тираждауға мүмкін болады, себебі мұнда құрылған деректер базасы әмбебап болып келеді.

Бағдарламаның комплекстік шешімдері:

- бүкіл тауарларды категориялар бойынша бөлу;
- қызметкерлер жайлы ақпаратты бір жүйегі келтіру;
- web-қосымша кез келген мобильдік платформаларда ашылады: Android, iPhone, iPad;
- айлық есептемені ыңғайлы түрде шығару.

Стратегиялық деңгейі: өнімнің даму картасы.

Маркетингтік теорияға байланысты бұл деңгейде бағдарламаның жалпы, конкуренттік және функциональдық даму жолы анықталады. Практика жүзінде бұл әр бағдарламаның нарығын таңдау және осы нарықта жоғары конкурентті бағдарламаны ұсыну.

Осы деңгейде компания қандай технологиялармен(Java, .Net, PHP немесе басқа) жұмыс жасайтыны анықталады.

Тактикалық деңгейі: нарық анализі және алға жылжуы

Маркетингтің тактикалық дейгейінде бағдарламаны нарықта жылжытуы және компанияның осы нарықтағы жылжуы қарастырылады. Бұл деңгейде потенциалды сатып алушылар базасы жиналады. Келешекте электрондық коммерциямен айналысатын компания болғандықтан тек тәжірибелі қызметкерлерді жұмысқа қабылданады. Қызметкер тұтынушыларға қызмет көрсету мен олардың Интернетпен жұмыс барысында туындаған мәселелерін шешумен айналысады.

Маркетингтік рольдер.

Нарық талдампазы – бағдарламаның нарықтағы сұранысын жинау және талдау.

Бағдарлама менеджері – бағдарламаның жұмысын және өндіруіне жауапты.

Бағалау маманы – бағдарламаның жалпы соммасын шығару жауапты.

Операциялық деңгейі: сатылым және сатылған өнімді бақылау

Біз өзіміздің тұрақты тұтынушыларымыз жоғалтып алмауымыз керек. Веб-сайт түсінікті және қарапайым болуы тиіс. Біздің басты міндетіміз қолданушыларды қанағаттандырып отыруымыз тиіс.

Жоба сәтті болып табылады, егер ол жоғарғы деңгейдегі трафикті көрсетсе яғни тұтынушылар санының сеансы жоғарғы деңгейде болуы бұл түсетін пайданың көп болуын білдіреді.

Тұтынушылардың сұранысын қамтамасыз ету үшін қарапайым Интернет-дүкен жасау қажет

Ұсынылып отырған комерциялық жоба компанияның осы бағыттағы нарықта алдыңғы орынға шығаруы тиіс және жобаланған трафиктік деңгейді ғана емес асырып орындауы керек.

Қызметті SWOT-талдау 5.1-кестеде, маркетинг-микс қызметтері 5.2-кестеде көрсетілген.

К е с т е 5.1 – Қызметті SWOT-талдау

Мықты жақтары	Әлсіз жақтары
Бағдарлама мүмкіншіліктерінің көптігі; Қолжетімділігі; Қызмет бірақ рет орнатылады қосымша жаңартуларды жыл сайын қажет етпейді;	Автобөлшектер саласындағы адамдар үшін пайдалануға арналған. Басқа қызмет иелері үшін тиімсіз болып табылады.
Мүмкіндіктері	Қауіп
Қосымша табыс көзіне қол жеткізу.	Ғаламтор арқылы арқылы ақпарат алғанда компьютерде антивирус қызметі болуы тиіс.

К е с т е 5.2 – Маркетинг-микс қызметтері

Маркетинг-микс элементтері	Қажетті істер мен талдаулар
Product	
Автобөлшектер жүйесіне арналған web-қосымша қызметі	Жалпы тапсырманы шешу келесі бөлімдерден тұрады: – Тауар алмасу процесін жеңілдету. – Мекеме дәрежесіндегі қолданушыларға бағдарламаны бейімдеу

Кесте 5.2-нің соңы

Маркетинг-микс элементтері	Қажетті істер мен талдаулар
Price	
Базалық бағасы	Мекенің қажеттілік ерекшеліктерін ескере отырып ақысы аланады.
Promotion	
	Бағдарлама Motoshop компаниясында қолданылады және осы салада жұмыс істейтін компанияларға ұсынылады.
Place	
Фирма 1, фирма 2, фирма 3	Алматы қаласындағы осы мекеме қызметімен айналысатын мекеме админстраторларын хабарландыру, келіссөз жүргізу. Программа толығымен орнатылмағанша қызметімізді жалғастыру.

5.2 Бағдарламалық қамтаманың өңделуінің еңбек өнімділігінің есептелуі

Еңбекке кеткен шығынның базалық көрсеткіштері мына формула бойынша есептеледі:

$$Q=q*c \quad (5.1)$$

Мұндағы q - бағдарламалық өнімдегі (бастапқы команда) операторлар саны, менің жұмысымда операторлар саны 3500-ге тең, c -бағдарлама күрделілігінің коэффициенті, ол менде 1.26 ге тең [16]. Осы әдіспен базалық көрсеткішті табамыз:

$$Q=3500*1.26=4410$$

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

Бағдарламалық өнімін дайындауға кеткен әр кезеңнің уақытын анықтаймыз:

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

$$T_{ПО} = 24 \text{ адам / сағ.}$$

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

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

$$T_O = 3500 \times 1,5 / 50 \times 0,8 = 132,3 \text{ адам/сағ}$$

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

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

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

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

$$T_A = 4410 / (50 \times 0,8) = 110,25 \text{ адам/сағ}$$

4) T_{BC} (блок – сұлба құруға кеткен уақыт) T_A сияқты 5.3-формуламен есептеленеді.

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

$$T_H = Q \times 1,5 / (50 \times K). \quad (5.4)$$

$$T_H = 4410 \times 1,5 / (50 \times 0,8) = 163,37 \text{ адам/сағ}$$

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

$$T_{II} = Q / 50 = 88,2 \text{ адам/сағ}. \quad (5.5)$$

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

$$T_{OT} = Q \times 4,2 / 50 \times K = 4410 \times 4,2 / 50 \times 0,8 = 463,05 \text{ адам/сағ}. \quad (5.6)$$

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

$$T_D = 24 \text{ адам / сағ}.$$

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

$$T = T_{\text{ПО}} + T_{\text{ТО}} + T_{\text{А}} + T_{\text{БС}} + T_{\text{Н}} + T_{\text{П}} + T_{\text{ОТ}} + T_{\text{Д}}, \quad (5.7)$$

$$T = 24 + 132,3 + 110,25 + 110,25 + 163,37 + 88,2 + 463,05 + 24 = 1115,42 \text{ адам/сағ.}$$

5.2.2 Еңбекақыға жұмсалатын шығындар

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

Негізгі еңбек құны:

$$З_{\text{осн}} = t_{\Sigma} * TC / (t_{\text{cp}} * 8), \text{ теңге.} \quad (5.8)$$

мұндағы t_{Σ} – (5.7) формуласы бойынша есептелетін жалпы еңбек шығыны;

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

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

TC – тарифтік қойылым = 19966 -ке тең болды.

t_{Σ} – (7) формуласы бойынша 1115,42 –ге тең.

Осылайша негізгі еңбек ақы:

$$З_{\text{осн}} = 1115,42 * 19966 / (21 * 8) = 132562,35 \text{ теңге.}$$

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

$$З_{\text{доп}} = 0,2 * З_{\text{осн}} = 26512,47 \text{ теңге.} \quad (5.9)$$

Жалпы еңбек ақысы (еңбекті төлеу фонды) негізгі және қосымша еңбек ақының қосындысы ретінде анықталады (5.10):

$$\text{ФОТ} = З_{\text{доп}} + З_{\text{осн}} = 26512,47 + 132562,35 = 159074,82 \text{ теңге.} \quad (5.10)$$

5.2.3 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 $C_{обор} = 150000$ теңге (дербес компьютердің қазіргі уақыттағы нарықтағы теңгемен берілген құны алынды)

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

$$C_{МжК} = 0,015 \times C_{обор} = 0,015 * 150000 = 2250 \text{ теңге,} \quad (5.11)$$

$$C_{ТО} = 0,025 \times C_{обор} = 0,025 * 150000 = 3750 \text{ теңге.}$$

Амортизациялық аударым өнімнің физикалық және моральдық жарамсыздығына байланысты сол өнімді толық қалпына келтіру үшін жиналатын ақша қоры болып есептеледі. Амортизациялық аударым амортизация нормасы бойынша орнатылады, (5.12)-формуласы бойынша есептеледі:

$$A = \frac{C_{обор} \times H_A \times N}{100 \times 12 \times t} \quad (5.12)$$

мұндағы $C_{обор}$ – компьютер құны;

H_A – амортизация нормасы;

N – жұмыс орындалуына кеткен күннің саны;

t – дербес компьютерді қолдануға кеткен жалпы уақыт.

$$N = T/24 = 1115,42/24 = 46,47 \text{ күн.}$$

бұл жерде T еңбек шығынына тең, оның мәні – 1115,42 адам/сағ тең болған.

Дербес компьютерде жалпы жұмыс істеу уақыты мына формуламен есептеледі:

$$t = T_A + T_{БС} + T_H + T_{П} + T_{ОТ} = 110,25 + 110,25 + 163,37 + 88,2 + 463,05 = 935,12 \text{ адам/сағ}$$

Амортизация нормасы (5.13)-формуласы бойынша анықталады:

$$H_A = \frac{C_{обор} - C_{ликв}}{T_{норм} \cdot C_{обор}} \cdot 100\% \quad (5.13)$$

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

$$C_{ликв} = 0,05 * C_{обор} = 0,05 * 150000 = 7500 \text{ теңге.} \quad (5.14)$$

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

$$H_A = \frac{C_{обор} - C_{ликв}}{T_{норм} \cdot C_{обор}} \cdot 100\% = \frac{150000 - 7500}{4 * 150000} * 100\% = 23,75\%,$$

$$A = \frac{C_{обор} \times H_A \times N}{100 \times 12 \times t} = 150000 * 23,75 * 46,47 / 100 * 12 * 935,12 = 147,5 \text{ теңге.}$$

Бағдарламалық қамтаманы жазу кезіндегі электроэнергияның шығынын қоса есептеу керек. Электроэнергияның құны мына формула бойынша есептеледі (5.15):

$$C_{ЭЭ} = M * k_3 * F_{эф} * C_{кВт-ч}. \quad (5.15)$$

мұндағы M – ЭВМ қуаты (450 Вт);

k_3 – жүктеу коэффициенті (0.8);

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

$F_{эф}$ – жұмыс уақытының эффективті фонды, мына формула бойынша есептеледі (5.16):

$$F_{эф} = D_{ном} * d * (1 - f / 100) = 258 * 8 * (1 - 2,2 / 100) = 2019 \text{ сағ.} \quad (5.16)$$

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

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

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

Берілген коэффициенттер мен параметрлер бойынша электроэнергияның құны:

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

Дегенмен, алынған амортизациялық аударым мен электроэнергияға шығын – жылдық шығынның мәні, оны жалпы жылдық эксплуатациондық шығынмен анықталатын уақыт коэффициентімен дұрыстау керек (5.17):

$$\mathcal{E}_3 = t_{\Sigma} * C_3 / F_{эф}, \text{ теңге} \quad (5.17)$$

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

$$C_{\Sigma} = C_{\Sigma\Sigma} + C_{TO} + A_{жыл} = 10437 + 3750 + 147,5 = 14334,5 \text{ теңге.}$$

мұндағы $F_{эф}$ – жұмыс уақытының эффективті фонды, (5.16) формуласы бойынша анықталды, $t_{\Sigma'}$ - ЭЕМ-нің (5.7) формуласы бойынша анықталған есепті шешу үшін жалпы қолданылған уақыты, бірақ бұл жерде тек компьютердегі жұмыс уақытын ескере отырып анықталған:

$$t_{\Sigma'} = t_{пр} + t_{отл} + t_{д} = 110,25 + 463,05 + 24 = 597,3 \text{ сағ.}$$

Сәйкесінше, ЭЕМ-ның жалпы жылдық эксплуатационды шығын құны:

$$\Sigma_3 = 597,3 * 14334,5 / 2019 = 4240,71 \text{ теңге}$$

Ал уақыт коэффициенті мына формула бойынша есептеледі (5.18):

$$w = \frac{\Sigma_3}{C_{\Sigma}} = \frac{4213,84}{14334,5} = 0,29 \quad (5.18)$$

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

– Электроэнергияға шығын:

$$C_{\Sigma\Sigma}^* = C_{\Sigma\Sigma} \cdot w = 10437 * 0,29 = 3026,73 \text{ теңге;}$$

– Амортизационды аударым:

$$A_{жыл}^* = A_{жыл} \cdot w = 147,5 * 0,29 = 42,77 \text{ теңге}$$

Әлеуметтік салыққа кететін аударымдар бүкіл жалақының 11% құрайды [11%, ҚР ның Салық Кодексі], (5.19) формуласы бойынша анықталады:

$$CC = (\text{ФОТ} - 0,1 * \text{ФОТ}) * 0,11 = (159074,82 - 0,1 * 159074,82) * 0,11 = 15748,4 \text{ теңге} \quad (5.19)$$

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, (5.20) формуласы бойынша анықталады:

$$C_{\text{накл}} = 0,5 * \Phi OT = 0,5 * 159074,82 = 79537,41 \text{ теңге} \quad (5.20)$$

5.2.4 Өзіндік құн нәтижесінің кестесі мен диаграммасы

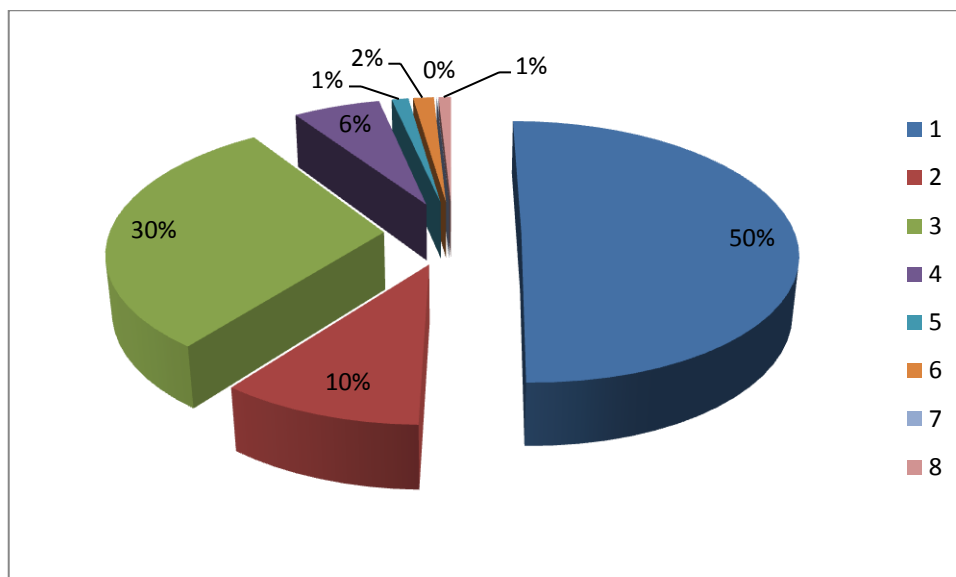
Бағдарламалық қамтаманы өңдеуге кеткен жалпы шығын еңбек жалақысы фондының, эксплуатационды шығынның, социалды сақтандыру, жүктеме шығындар мен материал және құрамалардың қосындысына тең болады.

Бағдарламалық қамтаманы өңдеудің қорытынды құны 5.3 кестеде көрсетілген. 5.1 - сурет – Өнімнің өзіндік құнының диаграммасы көрсетілген.

К е с т е 5.3 – Өзіндік құн нәтижесінің кестесі

Шығын статьялары		Сомма, теңге	Жалпы соммадан, %
ФЗП	$Z_{\text{осн}}$	132562,35	50,32
	$Z_{\text{доп}}$	26512,47	10,06
Жүктеме шығындар, $C_{\text{накл}}$		79537,41	30,19
Әлеуметтік сақтандыру, CC		15748,4	5,98
Эксплуатационды Шығындар	$C_{\text{ээ}}^*$	3026,73	1,15
	$C_{\text{ТО}}$	3750	1,42
	$A_{\text{жыл}}^*$	42,77	0,02
Материалдар мен комплектілер, $C_{\text{МК}}$		2250	0,86
Барлығы:		263430,1	100

Өзіндік құнының диаграммасын 5.1- суреттен көре аласыз.



5.1 - сурет – Өнімнің өзіндік құнының диаграммасы

5.3 Бағдарламаның бағасын есептеу

Өнімнің өзіндік құны 263430,1 теңге (жоғарыда кесте 5.3) болды.
 Өнімнің минималды бағасы келесі формуламен есептеледі:

$$Ц_{\min} = (C_{\text{пс}} + Ц_{\text{мд}} + Ц_{\text{з}})(1 + r/100) = (263430,1 + 200 + 150) * 1,21 = 319173,921 \text{ теңге} (5.21)$$

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

$Ц_{\text{з}}$ – жазба бағасы;

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

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

$$Ц_{\min_{\text{НДС}}} = Ц_{\min} + Н_{\text{ндс}} Ц_{\min} = 319173,921 + 0,12 * 319173,921 = 357474,79 \text{ теңге} (5.22)$$

Сонда бағдарламаның құны 360000 теңге.

5.3.2 Бағдарламалық өнімді иемденудегі кәсіпорындардың бірмезгілдегі шығындарын есептеу

Кәсіпорындардың бірмезгілдегі шығындары келесі бөлімдерден тұрады:

- жүйе құны;
- көлік шығындары;

– оқуға кететін төлемдер;
 Жүйе құны 360000 теңге (5.22 формула).
 Көлік шығындары жүйе құнының 20% құрайды:

$$C_{\text{тp}}=360000 *0,2=72000 \text{ теңге}$$

Жұмысшыларды жүйемен жұмыс істеуге оқыту шамамен екі сағатта консалтингті фирманың маманы көмегімен сағатына 2000-3000 теңгедей болып табылады [16]. Соған сәйкес оқу бағасы:

$$C_{\text{обуч}}=2000*8=16000 \text{ теңге}$$

Кәсіпорынның бірізгідегі шығындарын есептеулерінің нәтижесі 2-кестеде көрсетілген.

К е с т е 5.4–Ақпаратты жүйені енгізуге арналған бірізгідегі шығындардың есептелу нәтижесі

Шығындар түрлері	Сума, теңге
Жүйе құны	360000
Көлік шығындары	72000
Жұмысшыларды оқыту бағасы	16000
Қорытынды:	448000

5.4 Бағдарлама өнімін сатып алуға кеткен бір жолғы шығындар есебі.

Бағдарламаның бағасы: $C_C = 357474,79 \text{ тг}$

$C_{\text{тp}}$ – көлік шығыны, жүйе құнынан - 25 %, мың теңге;

$$C_{\text{тp}} = 0,25 * C_C = 357474,79 * 0,25 = 89368,69 \text{ тг}$$

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

$$C_O = T \times C_{\text{оп}}, \quad (5.23)$$

$$C_O = T \times C_{\text{оп}} = 4 * 8 * 2800 = 89600 \text{ тг}$$

мұндағы T – оқытуға кеткен уақыт, сағ.;

$C_{\text{оп}} = 2800 \text{ тг}$. Себебі қазіргі таңда сағатына 2500-3000 тг аралығында.

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

$$\Sigma Z = C_C + C_{TP} + C_O, \quad (5.24)$$

$$\Sigma Z = C_C + C_{TP} + C_O = 357474,79 + 89368,69 + 89600 = 536443,49 \text{ тг.}$$

Бағдарлама өнімін сатып алуға кеткен бір жолғы шығындар есебін 5.5-кестеге келтіру керек.

К е с т е 5.5 – Ақпарат жүйелерін енгізуге керекті бір жолымғы шығындар есебінің жиынтығы

Шығын бабы атауы	Сомасы, мың тенге
Жүйенің құны	357474,79
Көлік шығыны	89368,69
Жүйені оқуға кеткен шығыны	89600
Барлығы:	536443,49

5.5 Игеру саласындағы жылдық бір жолғы шығындар есебі.

Ақпараттық технологияларын қолданған кездегі жылдық шығындары келесі баптардан тұрады:

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

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

$$ЗП = (O_C \times t \times K_P) \times Ч \times (1 + K_D), \quad (5.25)$$

мұндағы O_C – маманның сағаттық ақысы, 1100 мың теңге/сағ.;

t – жұмыс күнінің ұзақтығы, 8 сағ.;

K_P – жылдағы жұмыс күндер саны, 245 күн;

$Ч$ – үдеріске қатысы бар мамандар саны, 5 адам;

K_D – қосымша еңбек ақы коэффициенті, 20 %.

$$ЗП = (O_C \times t \times K_P) \times Ч \times (1 + K_D) = (1100 * 8 * 245) * 3 * (1 + 0,2) = 7761600 \text{ тг.}$$

$$ЗА = ЕАҚ * 10\% = 7761600 * 0,1 = 776160 \text{ тг.}$$

$$O_{CH} = (\Phi OT - ЗА) \times 11\% = (7761600 - 776160) * 0,11 = 768398 \text{ тг.}$$

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

$$Pr = 3П \times 0,3, \quad (5.26)$$

$$Pr = 3П \times 0,3 = 7761600 \times 0,3 = 2328480 \text{ тг.}$$

Ақпараттық технологияларын қолданған кездегі жылдық бір жолғы шығындарды келесі кестеге сомасын және әр баптарын көрсетіп толтырыңыз (5.6 кестені қара).

К е с т е 5.6 – Ақпараттық технологияларын қолданған кездегі жылдық бір жолғы шығындар

Шығын баптары атауы	Сомасы, мың теңге
Жылдық ЕАҚ	7761600
Әлеуметтік салық аударым	768398
Басқа да шығындар	2328480
Барлығы:	10858478

5.6 Ақпараттық жүйе енгізуден үнем мен табыс мөлшерінің есебі.

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

- еңбек ақы қоры;
- әлеуметтік салық;
- басқада шығындар.

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

$$3П = (O_c \times t \times K_p) \times Ч \times (1 + K_d), \quad (5.27)$$

$$3П = (O_c \times t \times K_p) \times Ч \times (1 + K_d) = (1100 \times 8 \times 245) \times 4 \times (1 + 0,2) = 10348800 \text{ тг.}$$

$$3A = EАҚ \times 10\% = 10348800 \times 0,1 = 1034880 \text{ тг.}$$

$$O_{CH} = (ФОТ - 3A) \times 11\% = (10348800 - 1034880) \times 0,11 = 1024531.$$

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

$$Pr = 3П \times 0,3, \quad (5.28)$$

$$Pr = 3П \times 0,3 = 10348800 \times 0,3 = 3104640 \text{ тг.}$$

Осының барлығысын енді формула арқылы жазайық:

$$Cэ = 3П + O_{CH} + Pr \quad (5.29)$$

$$Cэ = 3П + O_{CH} + Pr = 10348800 + 1024531 + 3104640 = 14477971 \text{ тг}$$

К е с т е 5.7 – Ұйымда жүйе енгізілмей тұрғанда жылдық бір жолғы шығындар

Шығын баптары атауы	Сомасы, мың теңге
Жылдық ЕАҚ	10348800
Әлеуметтік салық аударым	1024531
Басқа да шығындар	3104640
Барлығы:	14477971

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

$$\Delta_{ye} = C_o - C_1, \quad (5.30)$$

$$\Delta_{ye} = C_o - C_1 = 14477971 - 10858478 = 3619493 \text{ тг}$$

5.7 Ақшалай құралдардың қозғалысы

К е с т е 5.8 – Ақшалай тәсілдердің қозғалысы, теңге

Аты	Жылдар		
	2014	2015	2016
Бірмезгілдік шығындар	448000		

Кесте 5.8 - соңы

Аты	Жылдар		
	2014	2015	2016
Үнемділіктен шыққан шартты пайда.		3619493	3619493
Дисконттау коэффициенті (21% мөлшерінде)	1	0,826	0,683
Таза дисконтталған табыс (ТДТ)	-448000	2989701,2	2472113,7
ТДТ өспелі нәтижесімен	-448000	2024113,7	4496227,4

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

$$ТДТ_1 = 3619493 * 0.826 = 2989701,2 \text{ теңге}$$

$$ТДТ_2 = 3619493 * 0,683 = 2472113,7 \text{ теңге}$$

5.8 Экономикалық тиімділікті есептеу

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

Шығындары бізмезгілде тек жобаның басында күрделі салымдар (C_0) ретінде іске асатын жобалар үшін NPV (5.31) формуламен есептеледі:

$$NPV = \sum_{i=1}^n \frac{B_i}{(1+r)^i} - C_0. \quad (5.31)$$

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

$$NPV(21\%) = \left(\frac{3619493}{(1+0.21)^1} \right) - 448000 = 2024113,7 \text{ тг} > 0,$$

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

5.8.2 Пайда индексін есептеу (Profitability index, PI)

PI жобаның салыстырмалы пайдасын көрсетеді. Бұл көрсеткіш келесі (5.32) формуламен есептеледі:

$$PI = \frac{NPV}{C_0} \quad (5.32)$$

$$PI = (2989701,2 / 448000) = 6,67\%$$

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

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

$$IRR = r_1 + \frac{f(r_1)}{f(r_1) - f(r_2)} * (r_2 - r_1) \quad (5.33)$$

$$r_b = 33\%:$$

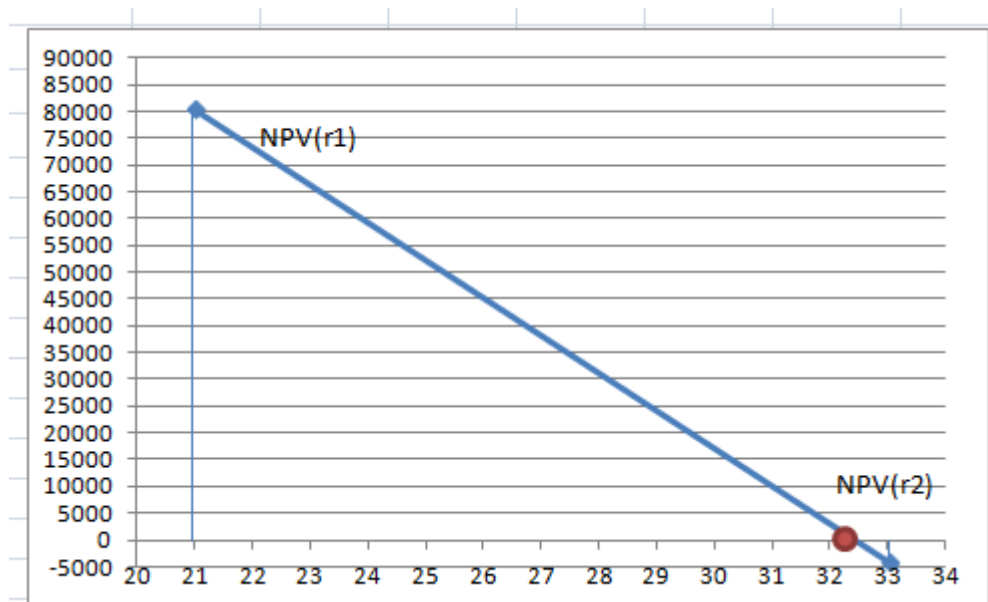
$$NPV(33\%) = \left(\frac{361949}{(1+0.33)^1} \right) - 448000 = -175857,89 > 0$$

$$NPV(21\%) = 2024113,7 \text{ теңге,}$$

$$NPV(33\%) = -175857,89 \text{ теңге,}$$

$$IRR = r_1 + \frac{NPV(r_1)}{NPV(r_1) - NPV(r_2)} \times (r_2 - r_1) = 21 + \frac{2024113,7}{2024113,7 - (-175857,89)} * (33 - 21) = 32\%$$

Табыстың ішкі нормасы 5.2-суретте көрсетілген:



5.2 - сурет - Іріктеу әдісі арқылы IRR табу

5.8.4 Өтімділік периодын есептеу (Payback period, PBP)

Өтімділік периоды мына формуламен есептеледі:

$$T_{ок} = t + \frac{C_0 - \sum_{i=1}^n B_i}{B_{i+1}} \quad (5.34)$$

мұндағы C_0 – жоба басындағы салым;

B_i – i -ші жылдағы жобадан алынатын пайда;

Сонда біздің өтімділік периодымыз:

$$T_{ок} = 2 + \frac{448000 - 2689701,2}{2472113,7} = 1,19 = 1 \text{ жыл } 3 \text{ ай.}$$

Бұл жобаның өтімділік периоды 1 жыл 3 айға тең, яғни жобамыз осы уақытта өз-өзін ақтайды.

Қорытынды

Бұл дипломдық жобада барлық қойылған мақсаттар іске асырылды. «Motoshop» ЖШС ақпараттық жүйесі автоматтандырылды. Web-қосымшаны өңдеу барысында барлық нюанстар ескерілді.

Ақпараттық жүйе PHP, CSS, HTML тілдерін және MySQL Server деректер қорын қолдана отырып құрылды. Жаңа түскен ақпаратқа байланысты деректерді тек администратор ғана енгізе алады. Сонымен қатар, авторизация қолдану арқылы деректердің қауіпсіздік пен құпиялылық мәселелері қарастырылды.

Бұл бағдарламалық өнім сәйкесінше «Нәзік» кондитерлік фабрикасында қолданылады.

«Тіршілік қауіпсіздігі» бөлімінде офистік компаниядағы жұмыс шарты және микроклимат параметрін есептеуді орындау, жағдайын тексеру мақсатында анализ жасалды. Еңбек шарты деңгейі дұрыс және есептеу бойынша алынған есептеулер, тіршілік қауіпсіздік бойынша стандарттарды қанағаттандырады.

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

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, раскрутка. – 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.

Қосымша А

Кетелерді құруға арналған скрипт

```
create table marka (  
  marka_id int AUTO_INCREMENT primary key,  
  marka_name varchar(20)  
);
```

```
create table model (  
  model_id int AUTO_INCREMENT primary key,  
  model_name varchar(20),  
  katalog_id int,  
  kod_proizvod varchar(15)  
);
```

```
create table katalog (  
  katalog_id int AUTO_INCREMENT primary key,  
  proizvo_date date,  
  id_kategory int  
);
```

```
create table kategory (  
  id_kategory int AUTO_INCREMENT primary key,  
  category varchar(30),  
  kod_zapchasty int  
);
```

```
create table zapchasty (  
  kod_zapchasty int AUTO_INCREMENT primary key,  
  zapchast_name varchar(128),  
  nomer_dvigatel int,  
  primecheniya varchar(30)  
);
```

```
ALTER TABLE model ADD CONSTRAINT FOREIGN KEY (marka_id) REFERENCES  
marka(marka_id)
```

```
ALTER TABLE katalog ADD CONSTRAINT FOREIGN KEY (katalog_id) REFERENCES  
model(katalog_id)
```

```
ALTER TABLE katalog ADD CONSTRAINT FOREIGN KEY (id_kategory) REFERENCES  
kategory(id_kategory)
```

```
ALTER TABLE kategory ADD CONSTRAINT FOREIGN KEY (kod_zapchasty) REFERENCES  
zapchasty(kod_zapchasty)
```

```
create table zakazhik(  
  zakazhik_id int AUTO_INCREMENT primary key,  
  zakazhik_lastname varchar(20),
```

А қосымшасының жалғасы

```
zakazhik_firstname varchar(20),  
country_id int,  
zakazhik_address varchar(20),  
zakazhik_phone_number int  
);
```

```
create table city(  
city_id int auto_increment primary key,  
city_name varchar(20),  
country_id int  
);
```

```
create table country(  
country_id int auto_increment primary key,  
country_name varchar(20)  
);
```

```
alter table city add constraint foreign key (country_id) references country(country_id)
```

```
alter table zakazhik add constraint foreign key (country_id) references country(country_id)
```

```
create table prodaji(  
kod_zapchasty int not null primary key,  
kol_prodej int not null,  
data_prodaji date  
);
```

```
create table nakladnoi (  
zakazhik_id int not null primary key,  
kod_zapchasty int not null,  
nom_nakladnoi int not null,  
kol_prodaji int not null,  
data_prodaji date,  
oformil int not null  
);
```

```
alter table nakladnoi add constraint foreign key (zakazhik_id) references zakazhik(zakazhik_id)
```

```
alter table nakladnoi add constraint foreign key (kod_zapchasty) references  
zapchasty(kod_zapchasty)
```

```
create table sotrudniki(  
sotrudnik_id int auto_increment primary key,  
sotrudnik_lastname varchar(20) not null,  
sotrudnik_firstname varchar(20) not null,  
sotrudnik_data_rojdenia date not null,  
sotrudnik_mesto_rojdenia int not null,
```

А қосымшасының жалғасы

```
sotrudnik_address_projivania varchar(30),  
sotrudnik_inn varchar(9),  
sotrudnik_number_phone varchar(11),  
sotrudnik_doljnost int not null,  
sotrudnik_zarabotok int  
);
```

```
alter table nakladnoi add constraint foreign key (ofornil) references sotrudniki(sotrudnik_id)
```

```
alter table sotrudniki add constraint foreign key (sotrudnik_mesto_rojdenia) references city(city_id)
```

```
alter table zakazhik add constraint foreign key (city_id) references city(city_id)
```

```
create table doljnost (  
doljnost_id int auto_increment primary key,  
doljnost varchar(20) not null  
);
```

```
alter table sotrudniki add constraint foreign key (sotrudnik_doljnost) references  
doljnost(doljnost_id)
```

```
create table supplier (  
supplier_id int auto_increment primary key,  
supplier_lastname varchar(20) not null,  
supplier_firstname varchar(20) not null,  
supplier_phone_number varchar(11),  
supplier_email varchar(20),  
supplier_address varchar(30),  
supplier_city_id int  
);
```

```
alter table zapchasty add constraint foreign key (supplier_id) references supplier(supplier_id)
```

```
alter table supplier add constraint foreign key (supplier_city_id) references city(city_id)
```

```
alter table zapchasty add constraint foreign key (id_kategory) references kategory(id_kategory)
```

```
SELECT zap.kod_zapchasty, zap.zapchast_name, zap.id_kategory  
FROM zapchasty zap  
INNER JOIN kategory kat ON zap.id_kategory = kat.id_kategory  
where zap.kod_zapchasty=2
```

```
SELECT zap.kod_zapchasty, zap.zapchast_name, kat.kategory, ktl.proizvo_date,  
model.model_name, primecheniya, sup.supplier_firstname, sup.supplier_lastname, kol, purchase,  
sale  
FROM zapchasty zap
```

А қосымшасының жалғасы

```
INNER JOIN category kat ON zap.id_category = kat.id_category
INNER JOIN katalog ktl ON zap.katalog_id= ktl.katalog_id
INNER JOIN model ON zap.model_id= model.model_id
INNER JOIN supplier sup ON zap.supplier_id= sup.supplier_id
```

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" lang="<?php print $language->language ?>"
xml:lang="<?php print $language->language ?>" dir="<?php print $language->dir ?>">

<head>
  <?php print $head ?>
  <title><?php print $head_title ?></title>
  <?php print $styles ?>
  <?php print $scripts ?>
  <script type="text/javascript"><?php /* Needed to avoid Flash of Unstyle Content in IE */ ?>
</script>
</head>

<body>

<table border="0" cellpadding="0" cellspacing="0" id="header">
  <tr>
    <td id="logo">
      <?php if ($logo) { ?><a href="<?php print $front_page ?>" title="<?php print t('Home')
?>">" /></a><?php } ?>
      <?php if ($site_name) { ?><h1 class='site-name'><a href="<?php print $front_page ?>"
title="<?php print t('Home') ?>"><?php print $site_name ?></a></h1><?php } ?>
      <?php if ($site_slogan) { ?><div class='site-slogan'><?php print $site_slogan ?></div><?php }
?>
    </td>
    <td id="menu">
      <?php if (isset($secondary_links)) { ?><?php print theme('links', $secondary_links, array('class'
=> 'links', 'id' => 'subnavlist')) ?><?php } ?>
      <?php if (isset($primary_links)) { ?><?php print theme('links', $primary_links, array('class' =>
'links', 'id' => 'navlist')) ?><?php } ?>
      <?php print $search_box ?>
    </td>
  </tr>
  <tr>
    <td colspan="2"><div><?php print $header ?></div></td>
  </tr>
</table>

<table border="0" cellpadding="0" cellspacing="0" id="content">
  <tr>
    <?php if ($left) { ?><td id="sidebar-left">
```

```
<?php print $left ?>
</td><?php } ?>
<td valign="top">
  <?php if ($mission) { ?><div id="mission"><?php print $mission ?></div><?php } ?>
  <div id="main">
    <?php print $breadcrumb ?>
    <h1 class="title"><?php print $title ?></h1>
    <div class="tabs"><?php print $tabs ?></div>
    <?php if ($show_messages) { print $messages; } ?>
    <?php print $help ?>
    <?php print $content; ?>
    <?php print $feed_icons; ?>
  </div>
</td>
<?php if ($right) { ?><td id="sidebar-right">
  <?php print $right ?>
</td><?php } ?>
</tr>
</table>

<div id="footer">
  <?php print $footer_message ?>
  <?php print $footer ?>
</div>
<?php print $closure ?>
</body>
</html>
<div class="node"><?php if ($sticky) { print " sticky"; } ?><?php if (!$status) { print " node-
unpublished"; } ?>>
  <?php if ($picture) {
    print $picture;
  } ?>
  <?php if ($page == 0) { ?><h2 class="title"><a href="<?php print $node_url?>"><?php print
$title?></a></h2><?php } ; ?>
  <span class="submitted"><?php print $submitted?></span>
  <div class="taxonomy"><?php print $terms?></div>
  <div class="content"><?php print $content?></div>
  <?php if ($links) { ?><div class="links">&raquo; <?php print $links?></div><?php } ; ?>
</div>
```

Javascript

```
Drupal.behaviors.fusionEqualheights = function (context) {
  if (jQuery().equalHeights) {
    $("#header-top-wrapper div.equal-heights div.content").equalHeights();
    $("#header-group-wrapper div.equal-heights div.content").equalHeights();
    $("#preface-top-wrapper div.equal-heights div.content").equalHeights();
  }
}
```

```
$("#preface-bottom div.equal-heights div.content").equalHeights();
$("#sidebar-first div.equal-heights div.content").equalHeights();
$("#content-top div.equal-heights div.content").equalHeights();
$("#content-region div.equal-heights div.content").equalHeights();
$("#content-bottom div.equal-heights div.content").equalHeights();
$("#node-top div.equal-heights div.content").equalHeights();
$("#node-bottom div.equal-heights div.content").equalHeights();
$("#sidebar-last div.equal-heights div.content").equalHeights();
$("#postscript-top div.equal-heights div.content").equalHeights();
$("#postscript-bottom-wrapper div.equal-heights div.content").equalHeights();
$("#footer-wrapper div.equal-heights div.content").equalHeights();
}
};
```

```
Drupal.behaviors.fusionIE6fixes = function (context) {
  // IE6 & less-specific functions
  // Add hover class to primary menu li elements on hover
  if ($.browser.msie && ($.browser.version < 7)) {
    $('form input.form-submit').hover(function() {
      $(this).addClass('hover');
    }, function() {
      $(this).removeClass('hover');
    });
    $('#primary-menu ul.sf-menu li.expanded').hover(function() {
      $(this).addClass('hover');
    }, function() {
      $(this).removeClass('hover');
    });
    $('.sf-menu li').hover(function() {
      $(this).addClass('hover');
    }, function() {
      $(this).removeClass('hover');
    });
    $('#search input#search_header').hover(function() {
      $(this).addClass('hover');
    }, function() {
      $(this).removeClass('hover');
    });
  };
};
```

```
Drupal.behaviors.fusionOverlabel = function (context) {
  if (jQuery().overlabel) {
    $("div.fusion-horiz-login label").overlabel();
  }
};
Drupal.behaviors.fusionSuperfish = function (context) {
```

```
$("#primary-menu ul.sf-menu").superfish({
  hoverClass: 'sfHover',
  delay: 250,
  animation: {opacity:'show',height:'show'},
  speed: 'fast',
  autoArrows: false,
  dropShadows: false,
  disableHI: true
}).supposition();
};
```

```
Drupal.behaviors.fusionSuperfishBlocks = function (context) {
  $("#div.block ul.sf-menu").superfish({
    hoverClass: 'sfHover',
    delay: 250,
    animation: {opacity:'show',height:'show'},
    speed: 'fast',
    autoArrows: false,
    dropShadows: false,
    disableHI: true
  }).supposition();
};
```

```
Drupal.behaviors.fusionGridMask = function (context) {
  if ($('#body.grid-mask-enabled').size() > 0) {
    var grid_width_pos = parseInt($('body').attr('class').indexOf('grid-width-')) + 11;
    var grid_width = $('body').attr('class').substring(grid_width_pos, grid_width_pos + 2);
    var grid = '<div id="grid-mask-overlay" class="full-width"><div class="row">';
    for (i = 1; i <= grid_width; i++) {
      grid += '<div class="block grid' + grid_width + '-1"><div class="inner"></div></div>';
    }
    grid += '</div></div>';
    $('body.grid-mask-enabled').prepend(grid);
    $('#grid-mask-overlay .row').addClass('grid' + grid_width + '-' + grid_width);
    $('#grid-mask-overlay .block .inner').height($('body').height());
  }
};
```

```
Drupal.behaviors.fusionGridMaskToggle = function (context) {
  if ($('#body.grid-mask-enabled').size() > 0) {
    $('body.grid-mask-enabled').prepend('<div id="grid-mask-toggle">grid</div>');
    $('#div#grid-mask-toggle')
      .toggle( function () {
        $(this).toggleClass('grid-on');
        $('body').toggleClass('grid-mask');
      },
      function() {
```



```
$(this).toggleClass('grid-on');
$('body').toggleClass('grid-mask');
});
}
};

Drupal.behaviors.fusionPanelsShowEdit = function (context) {
  // Sets parent row and block elements to have "overflow: visible" if editing Panel page
  if ($("#panels-edit-display-form").size() > 0 || $("#page-manager-edit").size() > 0) {
    $("#panels-edit-display-form").parents('.row, .block').css("overflow", "visible");
    $("#page-manager-edit").parents('.row, .block').css("overflow", "visible");
  }
};

Drupal.behaviors.fusionScreenshotpreview = function (context) {
  // Displays Skinr previews
  if ($('#span.preview-icon').size() > 0) {
    // configure distance of preview from the cursor
    var xOffset = 20;
    var yOffset = 0;

    $('#span.preview-icon').hover(function(e){
      var img_class = this.id;
      var caption = $(this).parent().text();
      // add preview markup
      $('body').append('<div id="screenshot">' +
        '<div class="screenshot-preview ' + img_class + '" alt="preview"></div>' +
        '<div class="screenshot-caption">' + caption + '</div>' +
        '</div>');
      $('#screenshot').hide(); // hide preview until dimensions are set
      $('#screenshot').css("left", (e.pageX + xOffset) + "px").css("top", (e.pageY + yOffset) + "px");
    // set initial preview position
    // load image in order to set preview dimensions
    var img = new Image();
    img.onload = function() {
      var caption_height = parseFloat($("#screenshot .screenshot-caption").css("height"));
      $("#screenshot").css("height", img.height + caption_height);
      $("#screenshot").css("width", img.width);
      $("#screenshot ." + img_class).css("height", img.height);
      $("#screenshot ." + img_class).css("width", img.width);
      $("#screenshot .screenshot-caption").css("width", img.width - 10);
      $("#screenshot").fadeIn("fast"); // now show preview
    }
    img.src = $(". " + img_class).css("background-image").replace(/^url\[\(\)\]/g, "");
  },
  function(){
    $('#screenshot').remove();
  }
};
```

```
});  
// adjust preview position with cursor movement  
$("#span.preview-icon").mousemove(function(e){  
    $("#screenshot").css("left", (e.pageX + xOffset) + "px").css("top", (e.pageY + yOffset) + "px");  
});  
}  
};
```

CSS3

```
/*  
** HTML elements  
*/  
body {  
    margin: 0;  
    padding: 0;  
    color: #000;  
  
    background-color: #fff;  
    font: 76% Verdana, Arial, Helvetica, sans-serif;  
}  
tr.odd td, tr.even td {  
    padding: 0.3em;  
}  
h1, h2, h3, h4, h5, h6 {  
    margin-bottom: 0.5em;  
}  
h1 {  
    font-size: 1.3em;  
}  
h2 {  
    font-size: 1.2em;  
}  
h3, h4, h5, h6 {  
    font-size: 1.1em;  
}  
p {  
    margin-top: 0.5em;  
    margin-bottom: 0.9em;  
}  
a {  
    text-decoration: none;  
    font-weight: bold;  
}  
a:link {  
    color: #39c;  
}
```

```
a:visited {
  color: #369;
}
a:hover {
  color: #39c;
  text-decoration: underline;
}
fieldset {
  border: 1px solid #ccc;
}
pre {
  background-color: #eee;
  padding: 0.75em 1.5em;
  font-size: 12px;
  border: 1px solid #ddd;
}
table {
  /* make <td> sizes relative to body size! */
  font-size: 2 em;

  font-size: 1em;
}
.form-item label {
  font-size: 1em;
  color: #222;
}
.item-list .title {
  font-size: 1em;
  color: #222;
}
.links {
  margin-bottom: 0;
}
.comment .links {
  margin-bottom: 0;
}

/*
** Page layout blocks / IDs
*/
#header, #content {
  width: 100%;
}
#header {
  background-color: #69c;
}
#logo {
```

```
vertical-align: middle;
border: 0;
}
#logo img {
float: left; /* LTR */
padding: 0 1em;

border: 0;
}
#menu {
padding: 0.5em 0.5em 0 0.5em; /* LTR */
text-align: right; /* LTR */
vertical-align: middle;
}
#navlist {
font-size: 1.0em;
padding: 0 0.8em 1.2em 0; /* LTR */
color: #9cf;
}
#navlist a {
font-weight: bold;
color: #fff;
}
#subnavlist {
padding: 0.5em 1.2em 0.4em 0; /* LTR */
font-size: 0.8em;
color: #9cf;
}
#subnavlist a {
font-weight: bold;
color: #9cf;
}
ul.links li {
border-left: 1px solid #9cf; /* LTR */
}
ul.links li.first {
border: none;
}
#search .form-text, #search .form-submit {
border: 1px solid #369;
font-size: 1.1em;
height: 1.5em;
vertical-align: middle;
}
#search .form-text {
width: 8em;
padding: 0 0.5em;
```

```
}
#mission {
  background-color: #369;
  padding: 1.5em 2em;
  color: #fff;
}
#mission a, #mission a:visited {
  color: #9cf;
  font-weight: bold;
}
.site-name {
  margin: 0.6em 0 0 ;
  padding: 0;
  font-size: 2em;
}
.site-name a:link, .site-name a:visited {
  color: #fff;
}
.site-name a:hover {
  color: #369;
  text-decoration: none;
}
.site-slogan {
  font-size: 1em;
  color: #eee;
  display: block;
  margin: 0;
  font-style: italic;
  font-weight: bold;
}
#main {
  /* padding in px not ex because IE messes up 100% width tables otherwise */
  padding: 10px;
}
#mission, .node .content, .comment .content {
  line-height: 1.4em;
}
#help {
  font-size: 0.9em;
  margin-bottom: 1em;
}
.breadcrumb {
  margin-bottom: .5em;
}
.messages {

  background-color: #eee;
```

А қосымшасының соңы

```
border: 1px solid #ccc;
padding: 0.3em;
margin-bottom: 1em;
}
.error {
border-color: red;
}
#sidebar-left, #sidebar-right {
background-color: #ddd;
width: 16em;
/* padding in px not ex because IE messes up 100% width tables otherwise */
padding: 10px;
vertical-align: top;
}
#footer {
background-color: #eee;
padding: 1em;
font-size: 0.8em;
}
```