

ҚАЗАҚСТАН РЕСПУБЛИКАСЫ БІЛІМ ЖӘНЕ ҒЫЛЫМ МИНИСТРЛІГІ
«ҒҰМАРБЕК ДАУКЕЕВ АТЫНДАҒЫ АЛМАТЫ ЭНЕРГЕТИКА ЖӘНЕ
БАЙЛАНЫС УНИВЕРСИТЕТІ»

Коммерциялық емес акционерлік қоғамы
IT-инжиниринг кафедрасы

ҚОРҒАУҒА ЖІБЕРІЛДІ

Кафедра меңгерушісі

_____ А.А. Досжанова
(ученая степень, звание, Ф.И.О.)

_____ « ____ » _____ 2020 ж.
(подпись)

ДИПЛОМДЫҚ ЖОБА

Тақырыбы: Android платформасында сауда орталығының мобильді қосымшасын құру

Мамандығы 5B060200 - Информатика

Орындаған Тлеген С.Б. Тобы ИНФ(ИТ)к-16-1

Ғылыми жетекші ф-м.ғ.к, доцент Калижанова А.У.

Кеңесшілер:

Экономикалық бөлім: э.ғ.к., профессор _____ К.Р. Габелашвили
« ____ » _____ 2020 ж.

Өміртіршілігі қауіпсіздігі: аға оқытушы _____ Ж.К. Мусаева
« ____ » _____ 2020 ж.

Есептеу техникасын қолдану: аға оқытушы _____ Ж.С. Айтқулов
« ____ » _____ 2020 ж.

Норма бақылаушы: аға оқытушы _____ Б.Р. Абсатарова
« ____ » _____ 2020 ж.

Сын-пікір беруші: _____
« ____ » _____ 2020 ж.

Алматы 2020

ҚАЗАҚСТАН РЕСПУБЛИКАСЫ БІЛІМ ЖӘНЕ ҒЫЛЫМ
МИНИСТРЛІГІ
«ҒҰМАРБЕК ДАУКЕЕВ АТЫНДАҒЫ АЛМАТЫ ЭНЕРГЕТИКА ЖӘНЕ
БАЙЛАНЫС УНИВЕРСИТЕТІ»
Коммерциялық емес акционерлік қоғамы

Басқару жүйелері мен ақпараттық технологиялар институты

IT-инжиниринг кафедрасы

Мамандығы 5В060200 - Информатика
Дипломдық жобаны орындауға берілген

ТАПСЫРМА

Білім алушы Тлеген Сабина Бауржанқызы

Жобаның тақырыбы: Android платформасында сауда орталығының
мобильді қосымшасын құру

2020 жылғы «__» _____ № _____ университет бұйрығымен бекітілген.

Аяқталған жобаны тапсыру мерзімі: «__» _____ 2020 ж.

Дипломдық жобаның бастапқы мәліметтері(зерттеу(жоба)нәтежиелерінің талап етілген параметрлері мен объектінің бастапқы мәліметтері):Java бағдарламалау тілін қолданып, Android платформасында сауда орталығының мобильді қосымшасын құру.

Дипломдық жобаның

Дипломдық жобада қарастырылған мәселелер тізімі немесе дипломдық жобаның қысқаша мазмұны:

- а) мобильді құрылғыларға қосымшалар құрудың негіздері;
- ә) қолданылған бағдарламалық қамтамалар;
- б) программалық қамтаманы құру;
- в) өміртіршілігі қауіпсіздігі бөлімі;
- г) экономикалық бөлім;
- ғ) пайдаланылған әдебиеттер тізімі;
- д) А қосымшасы. Программа листингі;
- е) ендіру актісі;

Графикалық материалдар тізімі(міндетті сызбалар дәл көрсетілуі тиіс):

Ұсынылатын негізгі әдебиеттер:

1 Ибраева Л.К., Хисаров Б.Д. Басқару объектілерін модельдеу және идентификациялау. Оқу құралы. - Алматы: АИЭС, 2009. – 207 б.

2 Сыздықов Д.Ж., Сақабаев Н.К. Басқару объектілерін моделдеу және идентификациялау. – Шымкент, 2008. – 151 б.

3 Королев А.Л. Компьютерное моделирование – М.: БИНОМ. Лаборатория знаний, 2010.-230 с.

4 Черноруцкий И.Г. Методы оптимизации. М.: ВИНТ, 2011.

5 Куралбаев З.К. Решение задач по математическому программированию. Учебное пособие.-Алматы: РИК МОиК РК, 1997.-168 с.

Дипломдық жобаның бөлімдеріне қатысты белгіленген кеңестер

Бөлімдер	Кеңесшілер	Мерзімі	Қолы
Экономикалық бөлім	Габелашвили К.Р.		
Өміртіршілік қауіпсіздігі	Мусаева Ж.К.		
Программалық қамтама	Айтқулов Ж.С.	14.05.2020 ж. – 15.05.2020 ж.	
Норма бақылау	Абсатарова Б.Р.	13.05.2020 ж. – 18.05.2020 ж.	

Дипломдық жобаны дайындау

КЕСТЕСІ

Бөлімдер атауы, қарастырылатын мәселелер тізімі	Ғылыми жетекшілерге ұсыну мерзімі	Ескерту
Мобильді құрылғыларға қосымшалар құрудың негіздері;		
Қолданылған бағдарламалық қамтамалар;		
программалық қамтаманы құру;		

Тапсырманың берілген уақыты « ___ » _____ 20 ___ ж.

Кафедра меңгерушісі _____ Досжанова А.А.

Жоба жетекшісі _____ Табултаев С.С.

Орындалатын тапсырманы
қабылдаған студент _____ Аяпов К.А.

Аңдатпа

Бұл дипломдық жобада “Uniprice.kz” ЖШС сұранысы бойынша жасалған Android платформасында мобильді қосымшасын құру қарастырылған. Осы қосымшада мекемедегі кезектердің санын қысқарту, кәсіпкерлер мен тұтынушылардың жұмысын жеңілдету және жеделдету болып табылады. “Уақыт–алтынан да қымбат” деп бекер айтылмаған, әсіресе қазіргі уақытта, бұл бағдарлама кез келген адамның уақытын үнемдеуге мүмкіндік береді.

Жүйені дамытубарысында келесі құралдар пайдаланылды: Java программалау тілі, MySQL, Open Server.

Техникалық экономикалық негіздеу бөлігінде жобаның экономикалық тиімділігі есептелді. Сондай-ақ өміртіршілік қауіпсіздігі мәселелері қарастырылды және қажетті есептеулер жүргізілді.

Аннотация

В этом дипломном проекте “Uniprice.kz” по запросу ТОО предусмотрено создание мобильного приложения на платформе Android. В данном приложении является сокращение количества очередей в учреждении, упрощение и ускорение работы предпринимателей и потребителей. Не зря говорится, что” время дороже шести”, особенно в настоящее время, эта программа позволяет сэкономить время любого человека.

В процессе разработки системы используются следующие средства: язык программирования Java, MySQL, Open Server.

В части технико-экономического обоснования была рассчитана экономическая эффективность проекта. Также были рассмотрены вопросы безопасности жизнедеятельности и проведены необходимые расчеты.

Annotation

In this diploma project at the request of LLP " Uniprice.kz " provides for the creation of a mobile application on the Android platform. This application aims to reduce the number of queues at the institution, simplify and speed up the work of entrepreneurs and consumers. No wonder he says that "time is more expensive than six", especially at the present time, this program allows you to save time for any person.

The system development process uses the following tools: Java programming language, MySQL, Open Server.

As part of the feasibility study, the economic efficiency of the project was calculated. The issues of life safety were also considered and the necessary calculations were made.

Мазмұны

Кіріспе	9
1 Мобильді құрылғыларға қосымшалар құрудың негіздері.....	10
1.1 Мобильді	қосымшаға
анализ.....	10
2 Қолданылған бағдарламалық қамтамалар.....	12
2.1 Android платформасы туралы.....	12
2.2 Android SDK құралы.....	15
2.3 Java бағдарламалау тілі.....	15
2.4 Бағдарламаның деректер қорын басқару жүйесі.....	22
2.5 Деректер қорын құру жолдары	25
3 Программалық қамтаманы құру	28
3.1 Прецеденттер диаграммасы	30
3.2 Тізбектер диаграммасы.....	31
3.3 Кооперация диаграммасы.....	32
3.4 Класстар диаграммасы	33
3.5 Функционалдық жұмыс істеуі	33
4 Экономикалық бөлім	40
4.1 Жобаны іске асыру кезеңдері мен мерзімдері.....	41
4.2 Мобильді қосымшаны әзірлеуге арналған шығындарды есептеу.....	41
4.2.1 Материалдық шығындарды есептеу.....	42
4.2.2 Электр энергиясына жұмсалған сомаларды есептеу.....	42
4.2.3 Еңбекақы төлеу шығындарын есептеу.....	45
4.2.4 Есеп шығару әлеуметтік қажеттіліктері.....	47
4.2.5 Амортизациялық аударымдарды есептеу.....	48
4.2.6 Өзге де шығындарды есептеу.....	48
4.3 Мобильді қосымшаның ықтимал бағасын анықтау.....	49
4.4 Мобильді қосымшаның нәтижелілігі мен әлеуметтік тиімділігін бағалау	49
.....	49

5 Өміртіршілік қауіпсіздігі бөлімі	49
5.1 Кәсіби қызметкердің жұмыс орнында еңбек талдауын сипаттау.....	50
5.2 Төтенше жағдайда сауда орталығының эвакуациялау уақытын есептеу.....	58
Қорытынды	58
Әдебиеттер тізімі	59
А қосымшасы	79

Кіріспе

XXI ғасыр - бұл технология ғасыры екенін білеміз. Технология жағынан біз басқа дамыған елдерден бір сатыға төменбіз. Сондықтан біздің мақсатымыз технология дамыған елдерді қуып жету және бәсекелестерді тең дәрежеде сәйкестендіру керекпіз.

2003 жылдан бастап елімізде индустриялық-инновациялық жобалар көп екенін барлығымыз білеміз. Дегенмен, Ақпараттық технологиялар саласындағы үлес өте аз, сондықтан бұл саланың дамуы бәсеңдейтін болады. Әрине, бұл жағдай осы салада жұмыс істейтін адами капиталмен тікелей байланысты. Сондықтан елдегі экономикалық жағдай адами капиталмен тығыз байланысты. Осыған байланысты біз елімізде әлі дамымаған ақпараттық технологиялардың негізгі саласына өз үлесімізді қосуымыз керек.

Біз ұсынатын сала виртуалды сатып алулар деп аталады. Бұл бағдарламаның негізгі мақсаты-супермаркеттегі кезектердің санын қысқарту, кәсіпкерлер мен тұтынушылардың жұмысын жеңілдету және жеделдету. Бұл бағдарлама барлық мәмілелерге қолданылуы мүмкін. Оны үлкен супермаркеттегі қарапайым сатып алу үшін де қолдануға болады. Мобильді қосымша смартфондарда, планшеттерде және басқа да портативті құрылғыларда пайдалануға арналған. Мобильді қосымшалар осы ғасырда вирус болды. Ұялы байланыс біздің өмірімізге интеграцияланған, егер біз үйде ұялы телефондарды қалдырсақ, біз толыққанды сезінбейміз. Бастапқыда сіздің бағдарламалық қамтамасыз етудегі негізгі бағдарламалар болып табылатын мобильді қосымшалар жасалды: калькулятор, Күнтізбе және сағаттар. Бірінші ұялы телефон қарапайым операциялық жүйе болған Motorola Brick компаниясымен шығарылды, тек қоңырау шалу үшін ғана пайдаланылды және контактілер тізімі болды. 90-шы жылдары ұялы

телефондар сұр реңктері бір түсті болды және рұқсат өте төмен болды. Экран кішкентай болды және аз жады немесе есептеу қуаты болды. Біздің бұл проектті тандаудағы мақсатымыз, еліміздің техникалық тұрғыда дамуына үлесімізді қосу. Әзірге E-Market-ке ұқсас бағдарлама елімізде өкінішке орай жоқ. Жобамыз алға жүріп, қолданысқа еніп кетеді деген үміттеміз.

Енді бағдарлама жайында қысқаша айта кетсек. Бағдарлама негізінде мобильді бағдарламасы бар. Мобильді бағдарлама ANDROID платформасына арнап жасалған және Java тілінде жазылған. Бұл жерде мобильді телефонның басты мақсаты, уақыт үнемдеу.

1 Мобильді құрылғыларға қосымшалар құрудың негіздері

1.1 Мобильді қосымшаға анализ

Мобильді қосымша - бұл телефон, планшет немесе сағат сияқты мобильді құрылғыда жұмыс істеуге арналған компьютерлік бағдарлама немесе бағдарламалық жасақтама. Бастапқыда бағдарламалар электрондық пошта, күнтізбе және байланыс дерекқорлары сияқты өнімділікке көмектесу үшін жасалынған, бірақ қолданушылардың сұранысы мобильді ойындар, зауыттық автоматика, GPS және орынға негізделген қызметтер, тапсырысқа бақылау және билет сату сияқты басқа салаларда көп болуы, осы қосымшалардың тез пайда болуына себеп болды. Миллиондаған қосымшалар қол жетімді болуы үшін сатып алуды жүзеге асыру керек болды. Бағдарламалар, әдетте, App Store (iOS) немесе Google Play Store сияқты мобильді амалдық жүйенің иесі басқаратын қолданбаларды тарату платформаларынан жүктеледі. Кейбір қосымшалар ақысыз, ал басқаларында бағасы қосымшаны жасаушы мен тарату платформасы арасында бөлінеді. Мобильді қосымшалар көбінесе жұмыс үстелі компьютерлерінде жұмыс істеуге арналған жұмыс үстелі бағдарламаларына және мобильді құрылғыдай емес, мобильді веб-браузерлерде жұмыс істейтін веб-қосымшалардан ерекшеленеді. Android операциялық жүйелердің арасында жаңалық болып табылады. Бұл операциялық жүйе мобильді құрылғылардың кең шеңберіне арналған. Android операциялық жүйесі коммуникаторларға, планшетті компьютерлерге, смартфондарға және нәтбуктерге орнатылады.

"Android Inc." – осы жүйені жасап шығарған компания. Сосын "Android Inc." компаниясың кейін "Гугл" (Google) компаниясы сатып алған болатын. Қазіргі уақытта Android негізіндегі жүйелерді даму және өндіруді

"OpenHandsetAlliance" компаниясы қолға алған. "OpenHandsetAlliance" арқасында Гугл ғана емес, Моторола, НТС, Интэл, Самсунг және көптеген басқа да техникалық өндіріс алыптарын біріктіреді.

Мобильді қосымшалар жоғары деңгейдегі бағдарламалау тілдерінде жазылады, содан кейін максималды өнімділік алу үшін операциялық жүйенің машиналық кодына компиляцияланады. Қосымшаларды әзірлеу өз ерекшеліктері бар: мобильді құрылғылар батареядан жұмыс істейді және дербес компьютерлерге қарағанда аз өндірістік процессорлармен жиынтықталады. Сонымен қатар, заманауи смартфондар мен планшеттер барлық жерде гироскоптар, акселерометрлер мен фотокамералар сияқты қосымша құрылғылармен жабдықталған.

Әдетте мобильді құрылғылар кейбір алдын ала орнатылған қосымшалармен сатылады. Қалғандарын пайдаланушының қалауы бойынша (ақылы және тегін) Apple AppStore, Google Play, Windows Phone Store және т.б. арнайы сервистерде жүктеуге болады. Apple AppStore және Android Market сияқты қосымшалардың алғашқы дүкендері кейіннен Google Play болған 2008 жылы пайда болды. Екі жылдан кейін американдық диалектикалық қоғам "қосымшаны" терминін жылдың сөзімен атады.

Android операциялық жүйесі Linux операциялық жүйесі негізінде жасалған. Бірақ Linux-тің барлық ерекшеліктерін қамтымайды. Бұл виртуалды машинасы "Дэлвик" пайдаланумен байланысты. Виртуалды машина барлық программалық жабдықтың жұмысын атқарады. Өндірушілер бір орында тұрмайды. Android жүйесі үнемі жетілдіріліп және техниканың жаңа түрлеріне ендірілуде. Android операциялық жүйесі 2008 жылдың қыркүйегінде пайда болды. Бастапқыда ол T-Mobайл G1 (T MobileG1) коммуникаторы үшін жасалды. Жарты жылдан кейін жаңа, жетілдірілген әрі өте ыңғайлы 1.0 нұсқасы пайда болды.

2009 жылы операциялық жүйенің төрт жаңартуы таныстырылған болатын. Осылай, ақпанда әр түрлі қателерді түзетумен 1.1 нұсқасы шықты. Сәуірде және қыркүйекте тағы екі жаңарту жарыққа шықты – 1.5 «Cupcake» және 1.6 «Donut». «Cupcake» жаңартуы маңызды өзгерістерді алып келді: виртуалды пернетақта, видео жазу және ойнату, браузер және т.б. «Donut» жаңартуында алғаш рет әр түрлі экранның тығыздығы мен кеңейтілімдерін және CDMA желілерін қолдау пайда болды. Сол жылдың қазанында Google-дың бірнеше аккаунттарын қолдануға мүмкіндік беретін, HTML5 тілін және басқа да жаңалықтарды қолдайтын браузері бар Android 2.0 «Eclair» операциялық жүйесі жасап шығарылды.

2010 жылдың ортасында Google «Froyo» атауымен Android 2.2 нұсқасын, ал 2010 жылдың соңында – Android 2.3 «Gingerbread» нұсқасын таныстырды. «Froyo» жаңартуынан кейін смартфонды қолжетімділік нүктесі ретінде пайдаланып, смартфонда дәстүрлі цифрлық немесе әріптік – цифрлы кілттерді қолдануға мүмкіндік пайда болды, ал «Gingerbread» жаңартуы «көшіру және қою» функцияларына толық бақылау жүргізуге, қоректі

басқару және қосымшаларды бақылауды жақсартға, бір құрылғыда бірнеше камераларды пайдалануға және т.б. мүмкіндік берді.

2011 жылдың 22 ақпанында планшеттарға бағытталған Android 3.0 «Honeycomb» нұсқасы ресми таныстырылған болатын. 2011 жылдың 19 қазанында шыққан Android 4.0 «IceCreamSandwich» – планшеттерде және смартфондарда қолдануға арналаған алғаш әмбебап нұсқа. 2012 жыл операциялық жүйенің «JellyBean» нұсқасына дейін жаңартуын әкелді. Ол маусым айында 4.1–ші нұсқа ретінде таныстырылды, ал қазанныңсоңында шағын жаңартудан кейін өз нөмір 4.2–ге ауыстырды. 2008 жылдың 22 қазанында GoogleAndroid операциялық жүйесі үшін онлайн қосымшалардүкені – AndroidMarket–тің ашылуы туралы жариялады. Келісімге бойынша өндірушілер – 70 %, мобильді байланыс операторлары 30 % пайдаға ие болады. 2009 жылдың ақпанда АҚШ және Ұлыбритания өндірушілері AndroidMarket–гі өз қосымшалары үшін төлем алу мүмкіндігін алды. SonyEricsson – AndroidMarket қосымшалар дүкенінде өз онлайн арнасын іске қосқан бірінші компания. Арнада компания қолдануға кеңес беретін ойындар және қосымшалар таныстырылған. 2011 жылдың желтоқсанына дейін AndroidMarket–тен 10 миллиард қосымша жүктеп алынған болатын.

Google компаниясы 2012 жылдың наурызында «Kitap», «Android Market», «Музыка» және т.б. мультимедиа сервистерін Google Play біртұтас сервисіне топтастыды. Google Play интернет дүкені 190 елде жұмыс істейді, 700 мыңнан астам қосымшаны қамтиды, ал сервистің жұмыс уақытында 25 миллиард шамасында жүктеу жасалған. Windows Phone – 2010 жылдың 11 қазанында жарық көрген Microsoft компаниясымен құрастырылған мобильді операциялық жүйе. 21 қазаннан бастап жаңа платформаға құралған алғашқы құрылғылар сатылуға түсе бастады. Windows Phone сәйкес болмаса да Windows Mobile операциялық жүйесінің толығымен жаңартылған интерфейсі бар және алғашқы рет Microsoft–тың Xbox Live және Zune медиаплеері қызметі интеграцияланған ізбасары саналады. Жүйенің тұсаукесері Барселонада өткен Mobile World Congress 2010 шарасына байланыстырылды. IOS, Windows, Android және Blackberry операциялық жүйесі келесі деңгейге өтті. Жаңа мыңжылдықта ұялы байланыс қосымшаларға және телефон мазмұнына жаңа эволюцияны ұсынды. Әдетте жасалған әрбір смартфон әр тұтынушыға, оның үлкендер болсын, ересектер болсын, кәсіпкерлер және т.б. болуына бағытталған. Зерттеулер көрсеткендей, жас ересектер смартфонға иелік ететін 80% смартфонға ие болу үшін ең жоғары болып табылады. 65 жастан асқан ересектер әдеттегі ұялы телефондарға қарағанда смартфондарды таңдай бастады. Blackberry, Apple және Samsung сияқты компаниялар өз өнімдерін жас ересектерге, ересектерге және кәсіпкерлерге бағыттайды. Бұл компаниялар мұны смартфонға тұтынушыларға қажет және қалауы мүмкін функцияларды қосу арқылы жасайды.

2 Қолданылған бағдарламалық қамтамалар

2.1 Android платформасы туралы

Мобильді қосымша қосымшаларды әзірлеу кезінде таңдау үшін жүздеген платформалары бар агрессивті нарық болып табылады. Қазіргі уақытта басқа платформалардан асып түсетін бірнеше платформалар бар. Android платформасы өте танымал, бұл қосымшаларды жасау үшін ең ыңғайлы платформалардың бірі. Оның көптеген қолданбалары JAVA бағдарламалық жасақтамасын пайдалану арқылы жасалады, бірақ басқа да бағдарламалар бірдей немесе одан да көп. Java бағдарламалық жасақтамасы тегін және бағдарламалық жасақтаманы әзірлеудің тегін нұсқасын ұсынады. Жиынтыққа қосымшалар мысалдары, бастапқы код, өңдеу құралдары және эмуляторлар кіреді.

Android-ұялы құрылғыны (телефонды, планшетті, смартфонды) басқара алатын операциялық жүйе. Android платформасы Linux ядросына негізделген. Жоғарыда айтылғандай, 2005 жылы Google Android Inc сатып алды., оны өзінің еншілес компаниясымен құрып, мобильді құрылғыларға арналған аттас платформалар шығара бастады. Содан бері платформа дамып, Android жаңа нұсқалары жиі шығарылады. Әрбір жаңа нұсқаның бірінші әріптері латын әліпбиінің әріптеріне сәйкес келеді. Бүгінде Android iPhone үшін әзірленген iOS операциялық жүйесін ғана пайдалануға мүмкіндік беретін әлемдегі ең танымал екінші құрылғы болып табылады.

Неге сізге Android қажет:

- Операциялық жүйе-бұл барлық адам командаларын орындау үшін қажетті кез келген "Электрондық құрылғының" миы".

- Тиісінше, Android-бұл ұялы құрылғының ішінде орналасқан виртуалды робот.

- Бұл платформаның артықшылығы Android ыңғайлы және өте түсінікті интерфейс, сондай-ақ икемді және көпжүйелік жүйесі бар.

Сонымен қатар, Android платформасы үшін арнайы әзірленген көптеген қосымшалар осы жүйенің бақытты пайдаланушыларына шын мәнінде шексіз мүмкіндіктер береді.

- Осы қолданбалар арқылы сіз сатып алуға, фотосуреттер жасауға, фильмдер көруге немесе кітаптар оқуға болады.

- Бұл платформа шығармашылық адамдарға арналған деген қорытынды жасауға болады, себебі екі бірдей ұялы құрылғылар мүлдем басқаша көрінуі мүмкін.

- Android өзіңізбен бірге фитнес бойынша жеке жаттықтырушы, дәрігер, ойыншық немесе теледидар алып жүруге мүмкіндік береді.

Android жүйесінің архитектурасының ерекшеліктері. Android үшін қосымшаны жасамас бұрын жүйенің архитектурасымен және оның негізгі функцияларымен танысу қажет. Android-операциялық жүйеден, аралық

бағдарламалық жасақтамадан, сондай-ақ базалық қосымшалардан тұратын мобильді құрылғыларға арналған бағдарламалық стек.

Android архитектурасын төрт деңгейге бөлу қабылданды:

- негізгі деңгей;
- орта кітапханасының деңгейі;
- қосымшаның құрылымы деңгейі;
- қосымшалар деңгейі.

Android платформасы Құрылғыларды, жабдықтарды, жадыны және үдерістерді басқару үшін Linux негізіндегі бағдарламалық стек болып табылады. Бұл стектің әр түрлі компоненттері иерархияның бірнеше деңгейлері болып табылады және жалпы, ұялы құрылғының жұмысын қамтамасыз етеді. Бұл компоненттердің бірі құрылғымен тікелей жұмыс істеу үшін қажет, басқалары - телефония, позициялау, мультимедиа және т. б. бар. Басқаша айтқанда, Android платформасы тікелей операциялық жүйені, орта деңгейдегі бағдарламалық жасақтаманы және қосымшалар жиынтығын (кіріктірілген және бөгде) қамтиды.

Біздің білуімізше, Android-бұл бағдарламалық стек, яғни әр түрлі бағдарламалық компоненттердің тұтас жиынтығы (біз бұл туралы мақала бойы бірнеше рет айтатын боламыз). Екінші жағынан, Android платформасындағы құрылғыда экран, пернелер, батарея, процессор, жады, әртүрлі датчиктер мен датчиктер және т.б. бар.

Мұның бәрі Android платформасы туралы келесі құрылыммен бірге бірыңғай бағдарламалық-аппараттық кешен ретінде айтуға мүмкіндік береді:

- Жабдықтың жұмысын қамтамасыз ететін элементтер жиынтығы.
- Linux ядросы (+драйверлер) - Linux операциялық жүйесінің ядросы және жабдықты басқару және жадыны бөлу, міндеттерді басқару, қауіпсіздік және т.б. негізгі механизмдерді қамтамасыз ететін драйверлер жиынтығы.
- операциялық жүйенің әртүрлі компоненттерімен қолданылатын кітапханалар (C / C ++).
- Dalvik Virtual Machine (DVM) - бұл Android қосымшаларын және операциялық жүйе компоненттерін іске қосу үшін ортаны қамтамасыз ететін Dalvik виртуалды машинасы.

Android ядросы 2.6 Linux операциялық жүйесінің нұсқасына негізделген, бірақ Android Linux таза жүйесі болып табылмайды, оның бірқатар функциялары мен өз жады бөлу механизмдері, процестер арасындағы қарым-қатынас және т.б. бар.

Ядро-бұл жабдық пен бағдарламалық стектің қалған бөлігі арасындағы абстракция деңгейі. Бұл деңгейде процестерді басқару, жад таңдау және файл жүйесін басқару сияқты қызметтер бар.

Бұл деңгейдің үстінде Application Framework деңгейі, кейде application құрылымы деп аталады. Қосымшалардың инфрақұрылымы деңгейінде өндірушілер жүйенің төменгі деңгейлі компоненттерімен ұсынылатын API-интерфейстерге қол жеткізеді. Сонымен қатар, фреймворк архитектурасының арқасында кез келген қолданба басқа қолданбалардың рұқсат етілген

мүмкіндіктеріне қол жеткізе алады. Әрбір қосымшаның негізі және платформаның бөлігі болып табылатын қызметтер мен жүйелердің базалық жиынтығы:

- визуалды қосымшаларды жасау үшін пайдаланылатын көріністердің бай және кеңейтілген тізімі (мысалы, тізімдер, мәтіндік өрістер, кестелер, түймелер немесе тіпті енгізілген веб-шолғыш);

- басқа қосымшаны пайдалану үшін бір қосымшамен ашылған деректерді басқаратын контент-провайдерлер;

- Resource Manager, ол функционалды емес ресурстарға (деректер жол, кестелер, файлдар және т. б.) қол жеткізуді қамтамасыз етеді.);

- Әрбір қолданба күй жолағында өз үзінділерін көрсетуге мүмкіндік беретін хабарлама диспетчері;

- Қосымшалардың өмірлік циклын басқаратын Activity Manager әрекеттермен жұмыс істеу туралы ақпаратты сақтайды, сондай-ақ әрекеттер арасындағы навигация жүйесін қамтамасыз етеді;

- құрылғының ағымдағы географиялық орналасуы туралы

- Жаңартылған ақпаратты уақтылы алуға мүмкіндік беретін Location Manager.

Қорытынды, Application Framework арқасында, Android қосымшасы көмекші функцияларды пайдалануға мүмкіндігі бар. Яғни операциялық жүйенің қосымшалары мен компоненттерін бірнеше рет пайдалану қағидаты іске асырылды. Android бағдарламасының жоғарғы деңгейінде қолданбалы деңгей бар. Бұл Android ОЖ алдын ала орнатылған негізгі бағдарламалар жиынтығын қамтиды. Мысалы, ол SMS, карталар, күнтізбе, браузер, пошта клиенті және т.б. жіберу қосымшасын қамтиды. Біріктірілген бағдарламалар тізімі құрылғы түріне және Android нұсқасына байланысты әр түрлі болуы мүмкін. Және бұл бағдарлама негізгі жиынтықтан Басқа, Android үшін барлық қосымшаларды қамтиды.

2.2 Android SDK құралы

Көп ұзамай мобильді құрылғыларға арналған Android операциялық жүйесін алғаш рет шығарғаннан кейін, танымалдылықтың айтарлықтай артуынан кейін әзірлеушілерге арналған жабдық шығарылды.

Android SDK-Android операциялық жүйесі үшін қосымшаларды әзірлеу ортасы. Сонымен қатар, мультимедиалық контентпен жұмыс (аудио, бейне, MPEG4, h.264, MP3, AAC, AMR, JPG, PNG және GIF форматтағы суреттер), SQLite деректер қоры, WebKit қозғалысындағы кіріктірілген браузер, Dalvik виртуалды машинасы, GSM телефониясы және т. б.). тіректер. Сонымен қатар, Android SDK пайдаланушылары өз қолданбаларын кірістірілген эмулятор арқылы сынап алады.

Android SDK ортасы Linux негізінде виртуалды құрылғыны қолданбаларды іске қосу үшін пайдаланады және 3G, WiFi, GPS, сенсорлық

экран және қазіргі ұялы телефондарда қол жетімді басқа да стандартты функцияларды қолдайды.

2.3 Java бағдарламалау тілі

Java тілі Java-бұл объектілі-бағытталған, платформаға тәуелді емес, Интернетте жұмыс істейтін тармақталатын қосымшаларды әзірлеу үшін қолданылатын бағдарламалау тілі. Java жобасын 1995 жылы Sun Microsystems компаниясы іске қосты. Java бағдарламалау жүйесі шағын интерактивті қосымшалар - апплеттерді тарату үшін Интернетті пайдалануға мүмкіндік береді. Олар интернет серверлерінде орналастырылады және таратылады, клиентке желі арқылы беріледі, автоматты түрде орнатылады және www құжатының бөлігі ретінде орындалады. Апплет клиенттің компьютер ресурстарына шектеулі қолжетімділікке ие, сондықтан ол жанама мультимедиалық интерфейс ұсына алады және дискіде сақталатын деректермен қандай да бір қиындықтарсыз күрделі есептеулерді орындай алады. Java-қосымшаларды бағдарламаның басқа түрі ретінде қарастыруға болады. Бұл портативті бағдарламалар, олар іске қосуға болады кез келген компьютер, қарамастан, оның архитектурасы. Құрастырылған байт-код Java (JVM) виртуалды машинасында виртуалды код интерпретаторында орындалатын нұсқаулықтар жиынтығы болып табылады. JSP (Java Server Pages) және серверлер серверлік қосымшалар мен деректер қорына қол жеткізуге мүмкіндік беретін сервлеттер өте кең таралған.

Java C ++ синтаксисін пайдаланады, бірақ объектілі модель Smalltalk туындысы болып табылады. Осыдан біз Java C ++ тек сыртқы түріне ұқсас екенін көре аламыз. Java көрсеткіштерді қолдамайды (C ++, Pascal және басқа тілдердегі ең қауіпті құрал), себебі, жад қорғанысын елемейді. Сондықтан да, бұл жағдайда, бір-бірімен байланысу үшін, бір-бірімен байланысу үшін, бір-бірімен байланысу үшін, бір-бірімен байланысу үшін, бір-бірімен байланысу үшін, бір-бірімен байланысу үшін, бір-бірімен байланысу үшін, бір-бірімен байланысу үшін, бір-бірімен байланысу үшін, бір-бірімен байланысу үшін. Бұл компиляторды көрсетеді, арифметикалық операциялар айнымалы үтірлі сандарға арналған кез келген платформада бірдей орындалуы тиіс.

Тіл сыныптарының жүйелік кітапханасы тілдің түрлі базалық функцияларын жүзеге асыратын сыныптар мен пакеттерден тұрады. Осы кітапханаларға қосылған сыныптар әдістері Java-бағдарламасын түсіндіру кезінде JVM-ден шақыртылады. Java-да барлық бағдарламалық Нысандар динамикалық жадыда (шоғырда) орналасқан, ал дестелерде сақталатын объектілерге кіру сілтемелер арқылы жүзеге асырылады. Бұл шешім жадқа тікелей кіруге рұқсат берді, бірақ массив элементтерімен жұмысты қиындатты. объектілер сілтемелері емес, дескрипторлар. Дескрипторлардың болуы JVM кодтың интерпретациясы кезеңінде үлгілердің сәйкестігін тексеруге мүмкіндік береді. Ява, сондай-ақ динамикалық жады бөлу тұжырымдамасын қайта қарады: динамикалық таратылған жадты босатудың

жолы жоқ. Оның орнына жаңа жады жүйесі (қоқыс жинаушы) жаңа оператордың көмегімен іске асырылды.

Java бағдарламаларында класс спецификациясы және оны жүзеге асыру әрқашан бір файлда болады. Java көпше мұра жоқ, тек конструктор, бірақ деструктор емес (автоматты қоқыс жинау қолданылады), тілде резервтік сөздер бар, goto және const операторы сөз жоқ.

Java тілінің маңызды ерекшеліктері интерфейстер және көпмәртептілік (бағдарлама бөлімдерін бір уақытта іске қосу мүмкіндігі) болып табылады. Лексикалық негіздер. Интерпретация бағдарламаны орындау кезінде ешқандай нәтиже бермесе де, оны пайдалану программист жұмысын жеңілдетеді.

Сақталған кілт сөздер. Сақталған кілт сөздер-Java-ға енгізілген басқару типтерін, модификаторлары мен бағдарламалық элементтерін идентификациялау үшін қажетті арнайы сөз идентификаторлары. Қазіргі уақытта Java-да сақталған 59 кілт сөз бар. Айнымалылар, сыныптар немесе әдістер идентификаторлар ретінде пайдаланыла алмайды.

1.1 кесте – Java тілінде резервтелінген сөздер

Резерв сөздер				
abstract	boolean	Break	Byte	byvalue
case	cast	Catch	Char	class
const	continue	Default	Do	double
else	extends	False	Final	finally
float	for	Future	Generic	goto
if	implements	Import	Inner	instanceof
int	interface	Long	Native	new
null	operator	outer	Package	private
protected	Public	rest	Return	short
static	Super	switch	synchronized	this
throw	throws	transient	True	try
var	Void	volatile	While	

Айталық byvalue, cast, const, future, generic, goto, inner, operator, outer, rest, var осы сөздер Java-да резервтелінген, бірақ олар қолданылмайды. Java да әдістердің резервтелінген аттары бар. Оларды 1.2-кестеде көре аласыз.

1.2 кесте – Java-да әдістердің резервтелінген аттары

Резерв аттары				
clone	equals	finalize	getClass	hashCode

				e
notify	notifyAll	toString	wait	
	1			

Идентификаторлар әртүрлі зерттеу түрлерін, әдістерді ұйымдастыру үшін қолданылады. Идентификатор қандай да бір әрекеттер немесе сандар, таңбалар (\$, _) болуы мүмкін. Java тілі тіркелімге байланысты болады.

Java константалары литерлік түрде беріледі. Бүтін сандарды, жылжымалы нүктелердің санын, логикалық нәтижелерді, символдар мен таңбаларды әр жерде орналастыруға болады. Тұтас литерлер. Бүтін сандар - тұтас Литер. Бұл бағдарламаларда жиі қолданылады. Әрбір Сан-1,2,3,4,5-тұтас Литер. Ондық әдебиеттен басқа компьютерде екілік, сегіздік, таңбалар қолданылады.

Жылжымалы литералдар. Жылжымалы нүктелі литерлер оларстың нақты түрі туралы айтады.

Логикалық литерлер. Логикалық әдебиетте тек екі мән болады-true (шындық) және false (жалған). True және false туындылары ешқандай сандық нәтижелерге өтпейді. Java сондай-ақ 0 және 1 болмаған кезде true және false мәні жоқ. Символдық литерлер. Java сондай-ақ Unicode кестесіндегі индекстер символдары. Олар бүтін санға ауыстыруға болатын 16 биттік мәнді береді. 1.3 кестеде. 1.3 кесте – Басқару символдар тізбегі

Басқару тізбегі	Сипаттама
\\ddd	Сегіздік символ (ddd)
\\uxxxx	Оналтылық символ символ UNICODE (xxxx)
\\'	Апостроф
\\"	Тырнақша
\\	Кері қиғаш сызықша
\\r	Карретканың қайтаруы (carriage return)
\\n	Жолды түсіру (line feed, new line)
\\f	Бетті түсіру (form feed)
\\t	Горизонтальді табуляция (tab)

Оператор – олқандайда бір әрекетті екі немесе бір аргументтерге орындап нәтижені шығарады. Синтаксистикалық тұрғыда операторлар идентификатор және литерал арасында орналасады. 1.4–кестеде операторлар түрлері көрсетілген.

1.4 кесте – Java тілінің операторлары

Операторлар			
+	+=	-	-=
*	*=	/	/=
	=	^	^=
&	&=	%	%=

>	>=	<	<=
!	!=	++	--
==	=	~	?:
	instanceof	[]	

Бөлгіштер. Тек қана бірнеше символдардың тобы Java–программада қолданылады және олар аталған жоқ. Олар программаның сыртқы бетіне, функционалдығына қарапайым бөлгіштер жауап береді. 1.5–кестеде бөлгіштерді көре аласыз.

1.5 кесте – Java тіліндегі бөлгіштер

Символдар	Аты	Не үшін қажет
()	Дөңгелек жақша	Жариялауда және әдісте параметрлердің тізімін белгілейді. Сонымен тапсырманың операция шығарылуын приоритетін есепте белгілейді.
{ }	Фигурлық жақша	Массивтің инициализациялау кезінде оның мәнін бергенде және класста әдісте блокты шектеу.
[]	Квадратты жақша	Массивті жасау кезінде және оның жеке элементтерін пайдалану кезінде қолданылады.
,	үтір	Айнымалыларды жариялау кезінде идентификаторларды бөледі, for циклда операторларды бөлу кезінде қолданылады.

Айнымалылар. Айнымалы бұл java программада ақпаратты сақтау негізгі элемент болып саналады. Айнымалы идентификатордың комбинациямен, типі мен және орындалу аймағымен сипатталады. Айнымалыларды сипаттау. Айнымалыларды сипаттау формасы осындай:

–идентификатортипі[=мәні] [, идентификатор [= мәні 7...];

–тип – бұл кірістірілген типтердің бірі byte, short, int, long, char, float, double, boolean, немесе класс интерфейс аты. Айнымалылар жазу үлгілері 1.6 кестеде көрсетілген.

1.6 кесте – Айнымалыларды сипаттау

Мәні	Айнымалылар
Int a, b, c;	a, b, c үш бүтін айнымалыны сипаттаймыз.
Int d = 3, e, f	Тағыда үш айнымалыны сипаттаймыз d және f

= 5;	мәнің инициализациялаймыз.
byte z = 22;	Z инициализациялау.

Келтірілген мысалда үш айнымалы жасалынады. Тік бұрышты үшбұрышқа сай, содан кейін Пифагор теоремасы бойынша гипотенузаның ұзындығы табылады ол 5 ке тең:

```
class Variables {
public static void main (String args []) {
double a = 3;double b = 4;
double c;
c = Math.sqrt (a * a + b * b);
System.out.println ("c = "+ c);
} }
```

Қарапайым типтер. Қарапайым типтер java тілінде объектілі бағытталған болып анықталмайды. Олар алғашқы тілдердегі типтеріне ұқсайды. Java–да сегіз қарапайым типтер бар:– byte, short, int, long, char, float, double және boolean. Оларды төрт бөлікке бөлуге болады:

–оларға byte, short, int и long типтер жатады. Осы типтер бүтін белгісандар үшін қолданылады;

–жылжымалы нүктемен типтер – float и double. Олар бөлшегі бар сан үшін қолданылады;

–char символдық тип. Символдар таблицадан элементтер үшін қолданылады, мысалы әріптер мен сандар;

–логикалық boolean типі. Бұларның тип, логикалық өлшемдер үшін. Java тілінде кейбір тілдерден айырмашылығы автоматты типтердің бейімделенуі жоқ болады. Типтердің сәйкес келмеуі программада ескертуге емес қатені туғызады. Әрбір тип үшін мәндердің мүмкін жиыны және операциялардың рұқсат түрлері болады.

Бүтін сандар byte – бұл 8–битті тип. Оның диапазоны – 128 ден 127–дейін byteb.

Егер биттермен манипуляция емес, онда осы типті қолдануға шектеу керек, оның орнына int типін қолдану артық болады.

Short – 16–битті тип. Оның диапазоны – 32768 ден 32767 – дейін. Java–да ең аз қолданыстағы тип оның басты байты бірінші тұрады.

Int типі 32–битті бүтін сандарды ұсыну үшін қолданылады. Оның диапазоны 2147483648–дан 2147483647 дейін. Көбінесе ол жай бүтін екі миллиардқа дейін типтер үшін қолданылады. Бұл тип массивтер мен счетчитер үшін қолдану аясында жақсы келеді.

Long типі 64–битті сандар ұсыну үшін қолданылады. Оның диапазоны кеңістіктегі барлық атомдарды санап шығуға болады. Астыда 1.7 кестеде бүтін типтердің дәрежелілігі және олардың диапазоны берілді.

1.7 кесте – Бүтін типтердің диапазоны

Аты	Дәрежелілігі	Диапазоны
long	64	-9, 223, 372, 036, 854, 775, 808.. 9, 223, 372, 036, 854, 775, 807
Int	32	-2, 147, 483, 648.. 2, 147, 483, 647
Short	16	-32, 768.. 32, 767
byte	8	-128.. 127

Жылжымалы нүктесі барсандар. Жылжымалы нүктелі сандарды жиінақты сандар айтады. Оны бөлшек бар амалдарды шешу үшін қолданылады. Java-да келесі нақты типтер – float және double. Оларды 1.8 кестеде көре аласыз.

1.8 кесте – Нақты тип

Аты	Дәрежелілік	Диапазоны
double	64	1. 7e-308.. 1. 7e+308
float	32	3. 4e-038.. 3. 4e+038

Егер қарапайым немесе біртекті дәлдікпен айнымалы қажет болса, онда float түрін пайдалану қажет. Ол 32 битті алады.

Double. Егер екі дәлдік қажет болса, біз double кілт сөзін қолданамыз. Ол 64 биттік болады. Бұл жағдайда 4.4-кестеде баяндалған талаптарға сәйкес 4.4-кестеде алынған деректер келтірілген.

Типтерді келтіру – type casting) - Бұл C++ және нашар бір, сондай-ақ Java тілінде типті келтіру сақталған. Кейде бізде қандай да бір түрі бар, бірақ оны басқа түрге әкелуі керек. Кейбір түрлерді типтік реттеу ақталмаған ісінулерге әкелуі мүмкін. Java-да түрлердің өлшемі сәйкес келсе, типтерді автоматты түрде теңшейді. Бұл byte немесе short түріндегі int түріне әкелуі мүмкін. Оның кеңеюі (widening) дейді немесе арттырады (promotion), шағын түрдегі айнымалы үлкен типтегі айнымалы. Егер айнымалыны үлкен түрден кішіге ауыстырса, онда азайтыңыз (narrowing) :

```
int a = 100;
```

```
byte b = (byte) a;
```

Түрлерін автоматты түрде реттеу. Мәннің нәтижесін шығарғанда, аралық мәндердің дәлдігі соңғы мәндерден жоғары болуы тиіс жағдай туындайды:

```
byte a = 40;
```

```
byte b = 50;
```

```
byte c = 100;  
int d = a B / c мәні;
```

Аралық жұмыстың мәні ($a * b$) сондай-ақ byte ауқымынан шыға алады. Сондықтан Java әр өрнектің бөлігінде int түрінің мәнін келтіреді. Типтерді автоматты түрде реттеу кейде кездейсоқ қателіктерден хабар шығуы мүмкін. Мысалы, біз келтірген код бірінші көзқарасқа адал, бірақ хабар сатысындағы қатені тудырады. Біз $50 * 2$ мәнін byte айнымалысына жазамыз. Дегенмен, автоматты түрде int түрін орнату нәтижесінде біз жағдайды естиміз, ол тронаслятордан қате тудырды. Int түрін byte түріне келтіргенде дәлдік жоғалуы мүмкін:

```
byte b = 50;  
b = b * 2;
```

Incompatible type for =. Explicit cast needed to convert int to byte (үйлесімді емес түрі =. Int түрі byte түріне сәйкес болуы керек). Түзетілген мәтін:

```
byte b = 50;  
b = (byte) (b * 2);
```

Егер түйінде byte, short және int түрі жазылған болса, онда түрі шектен аспау үшін, ол int-те өседі. Егер long түріндегі айнымалы мәнде нәтиже long түріне ауысады.

Java JDK платформасында бағдарламаларды жасау. Java тілінде бастапқы файл бір типті файлға немесе бір типті файлға беріледі. Java трансляциясы бағдарламаның соңғы коды java кеңейту файлдарында сақталатынын хабарлайды. Трансляция кезіндегі Код әрбір сыныпқа жеке Шығыс файлына жазылады, сынып атауымен сәйкес келеді және сынып кеңейтуде болады.

Алдымен біз "Hello World" бағдарламасын жазамыз. Содан кейін барлық негізгі лексикалық элементтерді қарастырайық. Олар сынақтан, сілтемелермен, кілт сөздермен, сәйкестендіргіштермен, операторлармен сыналуы мүмкін. Java дабірінші бағдарламамыз:

```
class HelloWorld {  
    public static void main (String args []) {  
        System.out. println ("Hello World"); } } }
```

Java тілі барлық программалық код аталған класстардың ішінде болғаның қажет етеді. Келтірілген мысал HelloWorld.java файлында жазылғаның қажет етеді. Кодты трансляциялау үшін Java – javac трансляторы қажет. Транслятор HelloWorld.class процессордан әмбебап байт кодты файлың құрады. Файлды іске асыру үшін бізге java іске қостыру ортасын қостыру керек.

Осы жолда class резервтелген сөз қолданылған. Ол трансляторға жаңа классты сипаттайтымыз деп айтады. Класстың толығымен сипаттауымыз фигурлық жақшалар ішінде сипатталады.

Осындай бірінші көзқарасқа қиын жол java да құрастыру кезінде өте керек міндет болады. Өйткені java да глобальді функциялар жоқ болады. Осы жол көп кездесетіндіктен біз оны қарастырайық.

Public. Бұл – бағдарламашыға кез келген әдістің көрінетіндігімен басқаруға мүмкіндік берген және кез келген айнымалы рұқсаттың түрлендіргіші. Осы мысалда public рұқсат модификаторы main әдісі әр бір классқа көрінеді және ашық болады.

Static. Келесі кілтті сөз ол – static. Осы класстың әдістері локальді не статикалық айнымалылармен ғана істей алады.

Void. Сізде әдістерді қолдану барысында нәтижені қандай да бір типте шығаруын қажет етеді. Мысалы int бүтін, float нақты. Біздің мысалда экранға жай ғана жолды шығару керек.

2.4 Бағдарламаның деректер қорын басқару жүйесі

Деректер қорын басқару жүйесі (ДББЖ) бағдарламалық қамтамасыз етуді ұсынады, барлық деректер қорын басқарады және авторизациялайды.

Деректер қорын басқару жүйесінің сыни сәті-бұл сенімділік пен өнімділік. Деректер базасында үлкен деректер сақталуы мүмкін болғандықтан, көптеген пайдаланушылар деректер базасын манипуляциялық түрде қолданатынын ескеру қажет.

ДББЖ-ға қойылатын негізгі талаптар:

- сыртқы жадтағы деректерді басқару (дискіде);
- дискілік кәшті пайдалану арқылы жедел жад деректерін басқару;
- өзгерістер журналы, деректер базасын қалпына келтіру;
- реляциялық модельдік деректер базасын сақтау;
- клиент пен сервер арасындағы өзара әрекеттесуді қолдау;
- Сыртқы утилитамен басқару үшін ДҚБЖ;
- құны төмен;

Заманауи ДҚБЖ шолу.

Әлемде көптеген ДҚБЖ бар, мұның бәрі дипломдық жұмыстың қоршауына кірмейді. Тек ең танымал ДББЖ қарастырайық және ерте уақытта олардың тарихын жазамыз.

Oracle Database 11g ДҚБЖ бірден-бір болып табылады, есептеуіштерді орнатуға немесе бөлуге арналған (Grid). Oracle Database 11g шығара, Oracle компаниясы қолданушыларға басқару мүмкіншілігін жеңіл жолын көрсетті, соған байланысты клиентер өздерінің бизнесін іске асырып инновацияларды еңгізді. Oracle Database 11g өндірімділіктің ең асқақ деңгейлерін қамсыздандырады, масштабтандырады, қолжетімділік, қауіпсіздік және басқарманың желідегі жайлылығын стандартты сервер және сақтау жүйесімен танымалы. Oracle Database 11g ДҚБЖ тиімді ұңғылау үшін

арналған және әртүрлі типтегі базалармен және құрылғылармен жұмыс істеуге ыңғайлы. Бұл өнім автоматизацияланғандағы басқару функциясын ұсынады және эффективті эксплуатациялауды ұсынады.

Firebird — FirebirdSQL)-ықшам, кросс-платформа, еркін басқару жүйесінде деректер қоры (ДББЖ), GNU/Linux, Microsoft Windows және Unix түрлі платформаларында.

Firebird ерекшелігі ретінде көп мақсатты, жедел және аналитикалық сұранысты қамтамасыз ету қарастырылады. Компакт (5МВ дистрибутиві) жоғары тиімді және күшті сөйлеу қолдау процедуралары мен триггерлерге арналған.

MsSQL.

Microsoft SQL Server - деректер қорын басқарудың реляциялық жүйесі (ДББЖ). Реляциялық деректер базасындағы деректер кестелерде сақталады. Сонымен қатар, кестеде көрсетілген талаптарға сәйкес, кестелер арасындағы қосылым орнатылуы мүмкін. Осыдан реляциялық атаудың негізі, ағылшын сөзі relational (ұқсас, байланысты). Пайдаланушылар бағдарлама арқылы деректерге қатынайды, ал әкімшілер серверге тікелей қатынайды.

MySQL Enterprise жылдық лицензиясының құны-600 ш. б., немесе 90000 теңге.

MySQL Community edition лицензиясының құны-тегін.

Талаптарға сәйкес ДҚБЖ таңдау.

Oracle деректер қорын таңдау шешілмеген, өйткені лицензия бағасы өте жоғары және ұялы телефонда деректер қорының түрі енгізілмеген.

Осыған байланысты, деректерді беру жылдамдығы өте төмен болуына байланысты деректерді беру жылдамдығы азаятынына байланысты, деректерді беру жылдамдығы азаяды. Сондай-ақ FireBird туралы деректер аз, осыған байланысты Интернет желісінде деректер аз. Ол біздің жұмысымызды тездетіп шешуге кедергі келтіреді және шешім іздеуде қиындық тудыруы мүмкін.

Қазіргі уақытта осыған байланысты қазіргі уақытта осыған байланысты қазіргі уақытта интернет желісіне қол жеткізуді қамтамасыз етуге байланысты бірқатар проблемалар бар. x тармақтарын ұстайды, ал ең тұрақты-бұл көрсетілген сәт. MySQL екі лицензиясы бар. GPL MySQL лицензиясының талаптары. GPL шарттарына байланысты, егер бір бағдарлама бастапқы кодты қосса, ол сондай-ақ GPL лицензиясымен таратылуы тиіс. Егер ол өз бағдарламасының ішкі кодына шыққысы келмесе, осы әзірлеуші бағдарламаны әзірлеушілермен жоспарлау қажет, нарықты сервистік қолдаумен қамтамасыз ететін байланысты коммерциялық лицензиялар көзделген.

MySQL де API, Delphi тіліне арналған C, C++, Эйфель, Java, Лисп, Perl, PHP, Python, Ruby, Smalltalk және Tcl, кітапханалар платформа тілдері үшін. NET, тағыда ODBC арқылы ODBC-драйвері MyODBC.

SQL-StructuredQuery Language (құрылымдық сұрау тілі). SQL реляциялық деректер базасымен жұмыс істеуге арналған. Ол пайдаланушыға деректер қорымен жұмыс істеуге мүмкіндік береді (ақпаратты қарау, іздеу, қосу және басқару). MySQL SQL 92 ANSI ерекшелігі-нормативтерге сәйкес келеді.

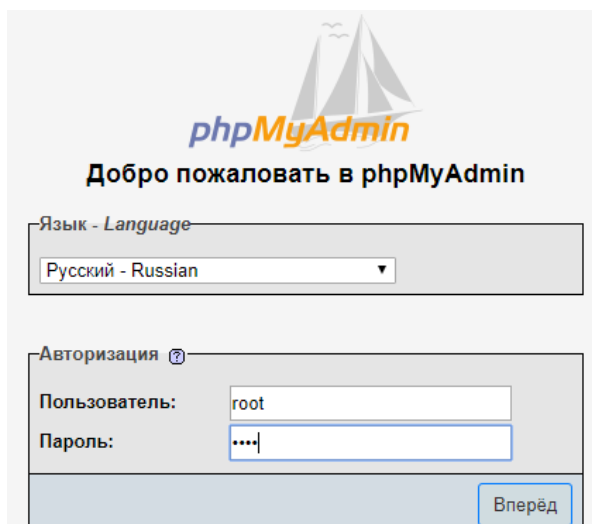
MySQL-интернеттегі ең танымал және ең танымал деректер қорын басқару жүйесі. Ол үлкен көлемді ақпаратпен жұмыс істеуге арналмаған, бірақ Үлкен, Шағын интернет-сайттар үшін өте ыңғайлы жүйе.

MySQL үлкен және орта бағдарламалармен жұмыс істеу үшін өте ыңғайлы. Сервер негіздері көптеген платформаларда компиляция жасайды. Сервердің толық мүмкіндіктері өнімділікті арттыратын көп дәлдікті қолдайтын Unix серверлерде байқалады.

Жоғары MySQL жылдамдығымен жұмыс, сенімділік, ыңғайлылығымен ерекшеленіледі. Онымен жұмыс үлкен қиындық тудырмайды. Бостандықта үлкен рөл атқарады. MySQL GNU (GPL, GNU Public License) жалпы лицензиясы негізінде таратылады. Ақпаратты ұзақ уақыт сақтау үшін, ең алдымен, файлдармен жұмыс жүргізіледі: бірнеше файлдарды енгізу, сондай-ақ келесі жұмыстар. Ақпаратты ұзақ уақыт сақтау мәселесі Web-бағдарламаларды бағдарламалауда жиі кездеседі: пайдаланушыларды бақылау, форумдағы хабарламаларды сақтау, сайтта ақпаратты қашықтықтан басқару.

2.5 Деректер қорын құру жолдары

Деректер қоры (database) - белгілі бір құрылымдық бірізділікпен орналастырылған деректер жиналған файлдар немесе файлдар жиынтығы. Деректерді басқару үшін деректерді басқару бағдарламалары қолданылады. Қазіргі уақытта олардың түрлі түрлерін кездестіруге болады. Мысалы, кең қолданылатын деректер базасымен жұмыс істейтін Microsoft Access бағдарламасын ерекшелеуге болады. Деректер базасы жинақталған ақпаратты қалыптастыру құрылымына байланысты бірнеше түрге бөлінеді. Атап айтқанда, ең көп таралған реляциялық деректер базасы болып табылады. 2.4 сурет-жергілікті желіге қосылу сұлбасы (2.4 сурет).

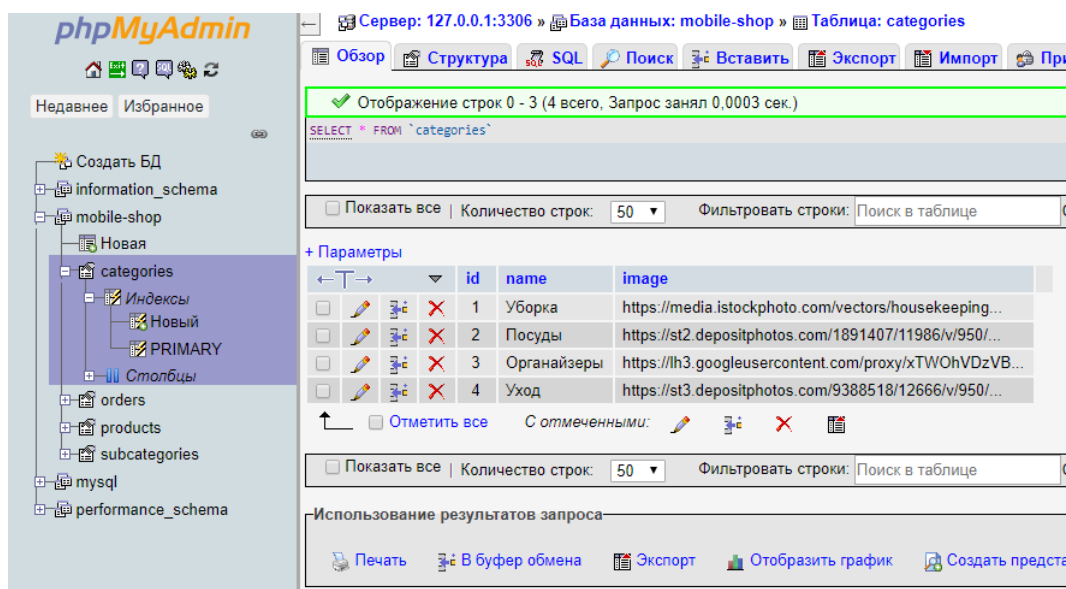


2.4.1 сурет – Мәліметтер базасына кіру

Реляциялық деректер қоры-бұл құрылым-кесте түріндегі өзара байланысты деректер жиынтығы. Бұл әртүрлі тізімдер, журналдар, каталогтар, бухгалтерлік кітаптар, анықтамалықтар және т.б. болуы мүмкін. Сонымен қатар, бұл МБ келісілген 35 бағдарламаларда тез өңделеді. Мысалы, Microsoft Access, Corel Paradox, Borland dBase, Microsoft FoxPro. Барлық аталғандар қарапайым пайдаланушыларға арналған. Олар өз бетінше сақтайды және пайдаланады Компьютердің қатты дискісі немесе сервер дискі файл.

Көптеген пайдаланушылар бар ақпаратқа қол жеткізуді қамтамасыз ету үшін, атап айтқанда, BorlandInterBase, MySQL, FireBird, Microsoft SQL Server, Informix, Sybase, Oracle сияқты бағдарламалар.

Қазіргі уақытта, қазіргі уақытта Қазақстанда бірде-бір жоба, бірде-бір жоба, бірде-бір жоба жоқ. Әрбір ДББЖ (дерекқорларды басқару жүйесі) ақпаратты тек өз форматында сақтайды. Бұл дипломдық жобада осы дипломдық жобада қолданылатын SQL (Structured Query Language, сұранымның құрамдас тілі) тілін қарастырамыз. Бұл сұрау тілі барлық бағдарламалармен жұмыс істей алады. Бұдан басқа, ДБ-да ақпаратпен сенімді, тез, қарапайым жұмыс істейді. Осы тілдің көмегімен клиент серверге сұрау жіберу арқылы өзіне жауап ала алады. SQL тілінде сұрау, ақпаратты қосу, өзгертуге және жоюға болады (2.4 сурет).



2.4.2 сурет – Мәліметтер базасымен жұмыс жасау

Деректер қорлар басқару жүйесі-(ДҚБЖ) - деректер қорын ұйымдастыруға арналған арнайы бағдарлама (көбінесе бағдарламалар кешені). Алгоритмдік тілде бағдарлама жасау үшін ДББЖ ақпараттық жүйені құру және басқару үшін де маңызды болуы қажет.

Көптеген мекемелер жұмыс процестерін қолдау үшін электрондық деректер базасын пайдаланады. Қазіргі уақытта осыған байланысты қазіргі уақытта осыған байланысты қазіргі уақытта осыған байланысты қазіргі уақытта осыған байланысты қазіргі уақытта осыған байланысты әдетте бірқатар өзара тәуелсіз базалар қолданылатындығына байланысты. Егер деректер базасындағы ақпарат тек пайдаланушыларға ғана емес, сонымен қатар басқа тұлғаларға да қызықты болса, онда ақпаратты одан әрі тарату үшін деректер базаларымен жасалған қағаз есептер мен анықтамалар қолданылады.

Жергілікті желілердің пайда болуы, интернетке қосылу, сондай-ақ торлар, корпоративтік желілерді құру кез келген желіде аралық жұмыстың арқасында ақпараттық базаларға қол жеткізу мүмкін болады. Бірақ, қолданыстағы деректер қорын пайдалану кезінде файлдық серверді жүктеу және аралықпен жұмыс істеу мүмкін емес (мысалы, жұмыс іссапарлары) сияқты желіде үлкен трафикке байланысты жұмыс орнының біртектілігін талап ететін проблемалар туындауы мүмкін ("туған" интерфейсті қосу кезінде). Осы қолайсыздықтардың шешімі ретінде WWW пайдаланушы интерфейсі ұйым ресурстарына қол жетімділікті пайдаланады деп айтуға болады.

ДҚБЖ негізгі функциялары:

- сыртқы жадта (дискіде));
- дискілік кәшті пайдалану арқылы жедел жад деректерін басқару;

- өзгерістерді тіркеу, резервтік көшіру және қате жұмыстан кейін деректер базасын қалпына келтіру;

- МБ тілдерін пайдалану (деректерді іздеу тілі, деректерді басқару тілі).

- МБ қатынау тәсілі бойынша: файл-серверлік.

ДҚБЖ деректерінің Файл-серверлік файлы орталықтандырылған түрде файл - серверде орналастырылады. ДББЖ әрбір клиенттік компьютерде орналастырылады. Осы ДБЖ-ға қатынау жергілікті желі арқылы жүзеге асырылады. Оқу мен жаңартуды синхрондау файлдық бұғаттау арқылы жүзеге асырылады. Серверге аз жүктеме, орталық портқа жүктеме, бұл архитектураның артықшылығы.

Кемшіліктері: ықтимал жоғары жүктеме жергілікті желі; жоғары сенімділігі, қауіпсіздігі және қол жетімділігі маңызды сипаттамаларын ретінде орын - күрделілігі далуының. Деректер қорын басқару функциялары пайдаланылатын жергілікті қосымшаларда деректер қорын басқару функциялары қолданылады. Қазіргі уақытта серверлік СУБД файлдар ескірген деп саналады. Мысалы: Microsoft Access, Paradox, dBase, FoxPro, Visual FoxPro.

Клиент-серверлік. Клиент-серверлік ДҚБЖ деректер базасымен бірге серверде орналасады және монополиялық режимде МБ жетеді. Барлық деректерді өңдеуге байланысты клиенттік сұрау салулар клиенттік-серверлік ДБЖ-да орталықтандырылған өңделеді. Клиенттік-серверлік ДҚБЖ кемшілігі серверге жоғары талап болып табылады. Артықшылықтары: әлеуетті жергілікті желінің салыстырмалы төмен жүктемесі; орталықтандырылған басқарудың ыңғайлылығы; қол жетімділіктің осындай маңызды сипаттамаларын орындаудың жоғары сенімділігі, қауіпсіздігі және ыңғайлылығы. Мысалы: Oracle, Firebird, Interbase, IBM DB2, MS SQL Server, Sybase, PostgreSQL, MySQL, ЛИНТЕР, Caché, MDBS.

Қондырғылар. Орнатылған ДҚБЖ-жүйеленген түрде жергілікті машинада үлкен деректер базасын сақтауға мүмкіндік беретін кітапхана. Деректерге қатынауды SQL арқылы немесе кез келген ерекше ДБЖ функциясы арқылы алуға болады. Осыған байланысты қазіргі уақытта мысалдар: OpenEdge, SQLite, BerkeleyDB, Firebird, MySQL, Sav Zigzag, Microsoft SQL Server, Firebird, MySQL, Sav Zigzag, Microsoft SQL Server, Firebird, MySQL, Sav Zigzag, Microsoft SQL Server, Firebird, MySQL, Sav Zigzag, Microsoft SQL Server және т.б. Compact, Линтер.

Web-портал функционалдық-ақпараттық сайт негізінде бизнестің дамуына елеулі әсер етеді. Интерактивті web-қосымшаның көмегімен компанияға өз функцияларын жүзеге асыруға мүмкіндік беретін маркетинг мәселелері шешіледі. Функционалдық бағдарламалар маңызды мәселелерді қамтиды, мысалы, клиенттермен жұмыс істеуге және компаниялармен басқаруға көмектеседі. Қазіргі уақытта әр фирма, әдетте, интернет әлемінде өз өкілдігін жариялайды. Егер бұған дейін сайтта тек статистикалық ақпараттық контент көрсетілсе, қазір web-қосымша кез келген мәселелер бойынша әлеуетті клиенттермен хабарласа алады. Қарастырылатын қосымшаға сәйкес,

мысалы, егер қарастырсақ, онда әлеуетті клиент ретінде интернет арқылы турларға, билеттерге, қонақүйлерге және т.б. тапсырыс беру мүмкіндігі ұсынылуы мүмкін

3 Программалық қамтаманы құру

3.1 Прецеденттер диаграммасы

Программалық жүйенің келесі сатысы прецеденттер диаграммасы болып табылады. Прецеденттер диаграммасы – бұл шамаланған жүйені жүргізудің құжаттық моделі.

Прецеденттер диаграммасында прецедент элементтері, актерлері мен олардың қатынастар жиынын көрсетеді. Прецеденттер диаграммалары көмегімен жүйе үшін прецеденттің статикалық көрінісі құрылады.

Прецеденттер диаграммасы қолданушы көзқарасы бойынша жүйе тәртібін анықтайды. Прецедент диаграммасы жүйе динамикасын алғашқы модельдеуі үшін басты құрал ретінде қарастырылады, өндірілетін жүйелерге талаптарды анықтау үшін қолданылады, кейінгі өндірулерді жүргізуге мүмкіндік беретін формаға бұл талаптарды тіркейді. Басқа да диаграммалар сияқты прецедент диаграммалары ескертулер мен шектеулерді қосып отырады. Одан басқа прецедент диаграммалары ірі фрагменттерге модель элементтерін топтастыру үшін қолданылатын пакеттерден тұрады.

Актер – бұл нақты элемент пен тікелей өзара әрекеттесетін жүйе сыртындағы объект рөлі.

Прецедент элементі – бұл жүйемен орындалатын және бөлек актер үшін көрінетін нәтижені өндіретін әрекеттер кезектілігінің сипаты.

Актерлер жүйе жұмысын қажет ететін сыртқы дүниені көрсетеді, ал прецедент элементтері актер мүддесінде жүйемен орындалатын әрекеттерді көрсетеді.

Активті субъектілер және жүйенің қызмет түрлеріне байланысты олардың арасында байланыс орнату керек. Ол байланыс ассоциациялық байланыс деп аталады.

Прецеденттер 2 типті тәуелділік байланысын көрсете алады:

- include - қосылатын байланыстар;
- extend - кеңейтілген байланыстар.

Қосылатын байланыс – сол функцияның орындалғанынан басқа прецеденттің анықталғандығын және осы прецедентке басқа да бір прецеденттердің біріккенін көрсететін байланыс.

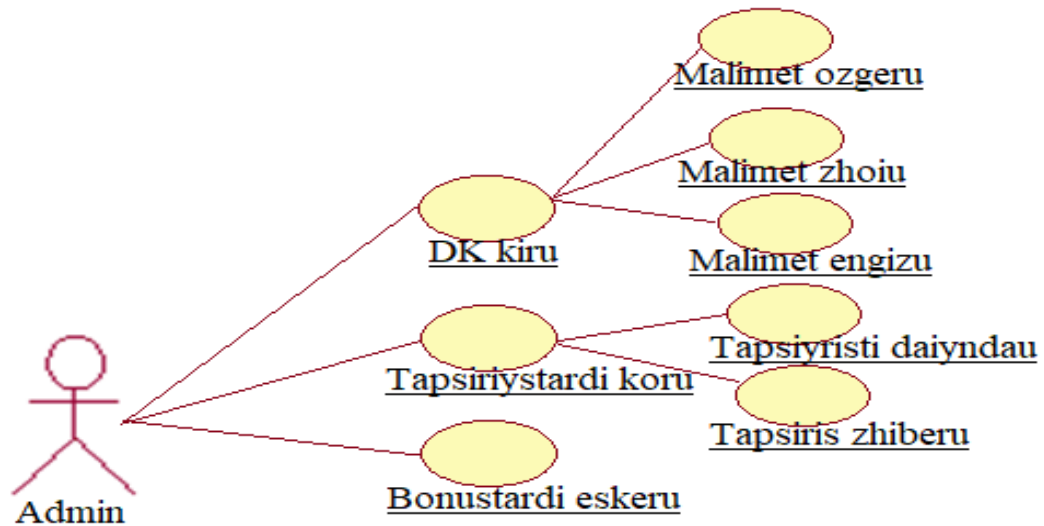
Кеңейтілген байланыс – шартты емес жағдайда туындайтын міндетті емес функцияның жиынтығынан туындайтын байланыс.

Use Case элементі (прецеденті) әрбір актердің мүддесіне қатысты жүйемен орындалатын және актер үшін нақты нәтиже беретін әрекеттер тізбектілігін (немесе бірнеше тізбектіліктерді) сипаттау. Қолдану вариантының диаграммасы (Use Case Diagram) – бұл активті субъектілердің көптеген графикалық көрсетімі оны қолданудың, сол немесе басқа

варианттардың амалдарымен өзара әрекеттеседі. Басқа да диаграммалар сияқты прецедент диаграммалары ескертулер мен шектеулерді қосып отырады. Одан басқа прецедент диаграммалары ірі фрагменттерге модель элементтерін топтастыру үшін қолданылатын пакеттерден тұрады.

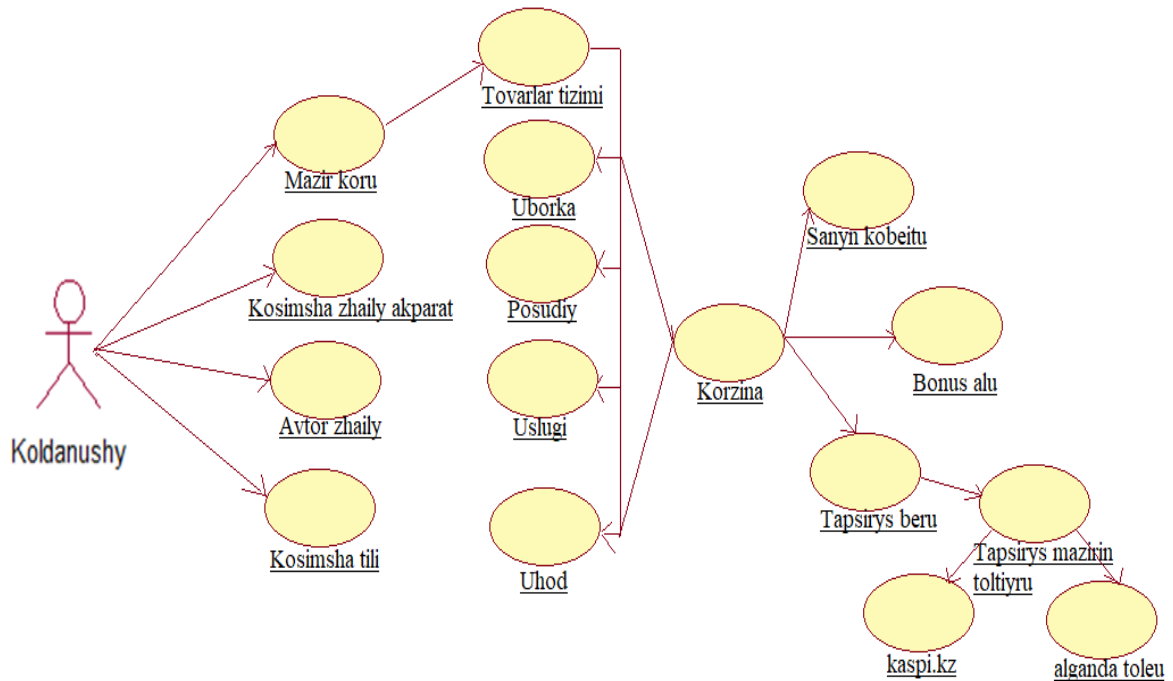
Жүйені жоспарлағанда жүйенің кілттік функциясын және көптеген қолданушыларды ұсынатын негізгі диаграмма (Main Use Case Diagram) конструкцияланады.

3.1-суретте пайдаланушы (администратор) үшін прецеденттер диаграммасы көрсетілген.



3.1-сурет – Прецеденттер диаграммасы, пайдаланушы (администратор)

3.2-суретте пайдаланушы (қолданушы) үшін прецеденттер диаграммасы көрсетілген.



3.2-сурет – Прецеденттер диаграммасы, пайдаланушы (қолданушы)

3.2 Тізбектер диаграммасы

Өзара әрекеттесу диаграммасының екі түрі бар – тізбек диаграммасы мен кооперация диаграммасы. Тізбек диаграммасы – бұл хабарламаларды уақыт бойынша тәртіпке келтіруді айқындайтын өзара әрекеттесу диаграммасы, яғни бір уақыт мезетіндегі іс-әрекетті жобалау. Кооперация диаграммасы – бұл хабарламаларды жіберетін және қабылдайтын объекттерді құрылымдық ұымдасуын айқындайтын өзара әрекеттесу диаграммасы. Өзара әрекеттесу диаграммаларының элементтері – объекттер, байланыстар, хабарламалар болып табылады.

Өзара әрекеттесу диаграммасы объектілер мен олардың арасындағы қатынастар, сондай-ақ объектілер арасындағы бір-біріне жіберілетін хабарламалар жиынын қосатын өзара әрекеттесулер. Өзара әрекеттесу диаграммалары жүйенің динамикалық көрінісін беруді қамтамасыз етеді.

Тізбек диаграммалары жүйеде тәртіп сценарийін бере отырып, бұл диаграмма хабарларды беру тәртібінің анағұрлым айқын көрінісін қамтамасыз етеді. Бірақ ол кооперация диаграммасында көрінетін бөлшектерді көрсетуге мүмкіндік береді.

Тізбектер диаграммасы төрт негізгі элементтерден тұрады:

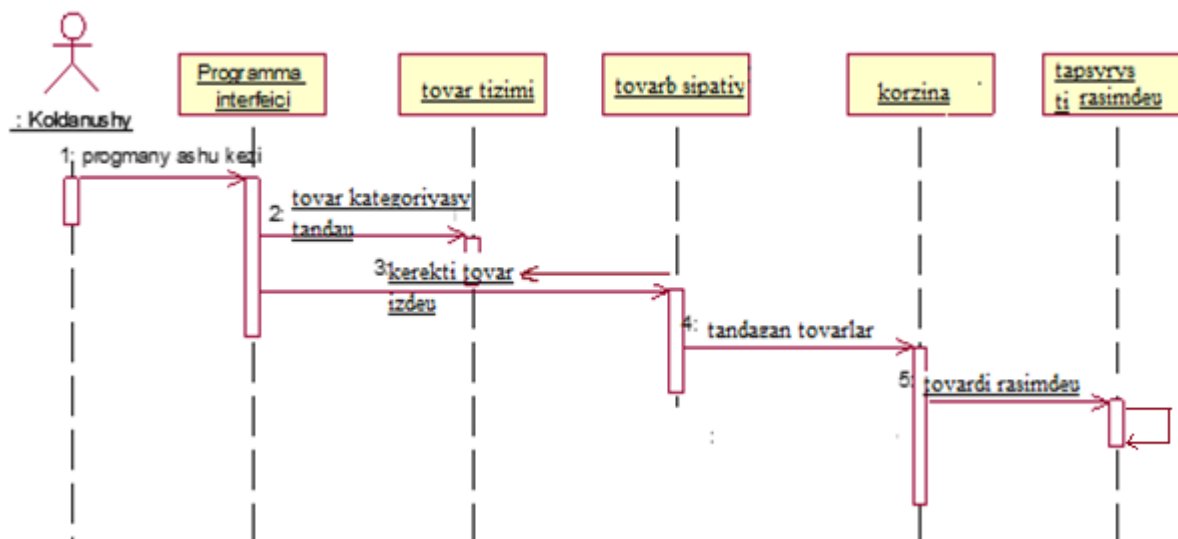
- прецеденттегі ізбасар мәтінінің іс-әрекеті. Ол сол жақтан жоғарыдан төменге жазылады. Сол терезеде іс-әрекет сипатталуы болып, жұмыс уақытындағы орындалатын ақпараттар қызмет етеді;

- объекттер "объект-класс" форматында аты немесе объект данасының

номері және класс объектісінің аты жазылады;

– хабарландыру – бағытпен көрсетілген бір объектіден келесіге бағытталған іс-әрекет туралы ақпарат жолдамасынан тұрады. Белгілі бір уақытта орындалуы және осы іс-әрекеттегі жүйенің жауапты реакциясы болуы мүмкін.

3.3-суретте тізбектер диаграммасы көрсетілген.

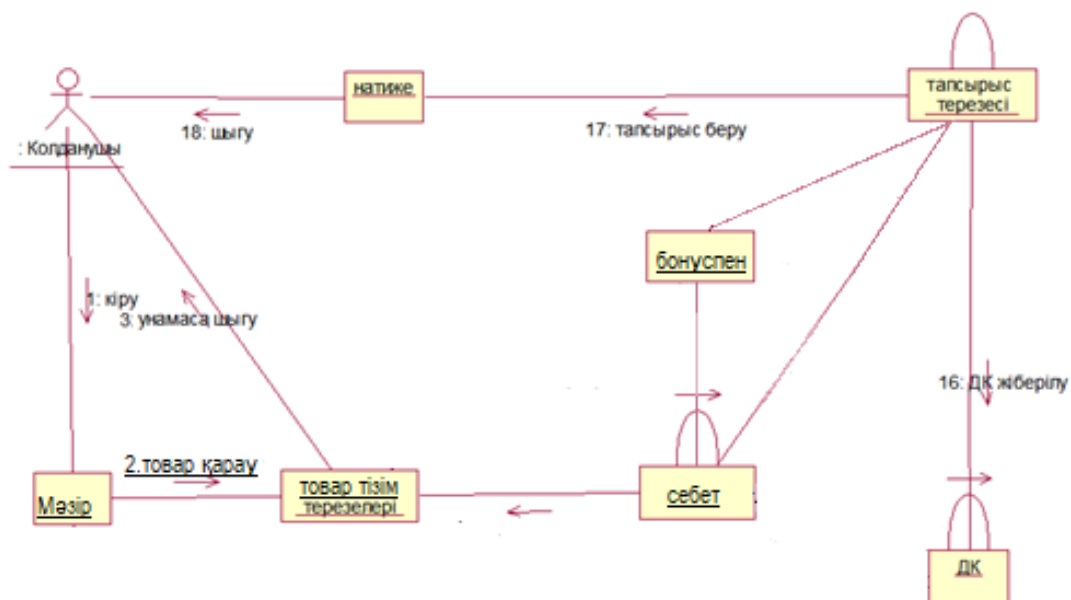


3.3-сурет – Тізбек диаграммасы

3.3 Кооперация диаграммасы

Диаграмманың бұл үлгісі хабарды беру тізбегінен дерексіздендірілу объекттердің өзара әрекеттерін сипаттауға мүмкіндік береді. Диаграмманың бұл үлгісінде тығыз түрінде нақты объекттің барлық қабылданатын және берілетін хабарлары және осы хабарлардың үлгілері көрсетіледі.

Sequence және Collaboration диаграммалары бір процесске әртүрлі көзқарас болғанымен, Rational Rose Sequence диаграммасынан Collaboration диаграммасын құруға мүмкіндік береді және керісінше, сонымен қатар, осы диаграммалардың автоматты синхронизациясын жүргізеді. 3.4-суретте кооперация диаграммасы көрсетілген.



3.4-сурет – Кооперация диаграммасы

IBM Rational Rose 2003 ортасында кооперация диаграммасының жұмыс терезесін іске қосуды бірнеше тәсілмен жүргізуге болады:

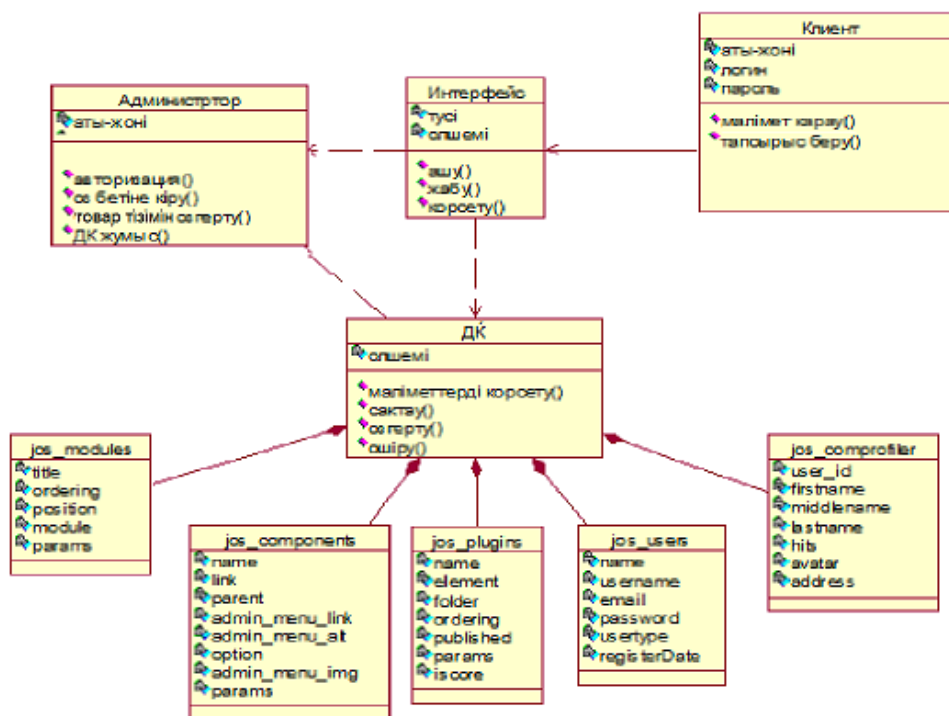
– стандартты аспаптар панеліндегі өзара әрекеттесу диаграммасының суреті бар батырманы шерту және жаңа кооперация диаграммасын құру үшін таңдау;

– бас мәзірдің операциясын орындау: Browse → Interaction Diagram (Браузер → Өзара әрекеттесу диаграммасы) және жаңа кооперация диаграммасын құру үшін таңдау;

– контексті мәзірдің операциясын орындау: логикалық ұсыну және жобаның браузерінде қолдану варианттарын ұсыну үшін New → Collaboration Diagram (Жаңа → Кооперация диаграммасы).

3.4 Класстар диаграммасы

Класстар диаграммалары класстар, интерфейстер, кооперация және олардың қатынастарының жиынын көрсетеді. Объектілі-бағытталған жүйелерді модельдеу кезінде кластар диаграммалары өте жиі қолданылады. Кластар диаграммалары жүйенің статикалық жобалық көрінісін қамтамасыз етеді. Объект пен класс түсініктері тығыз байланысты. Класс – бұл объектілердің маңызды сипаттамаларының абстракциясы. Класс диаграммасы модельдің негізгі логикалық көрінісі болып табылады, кластар диаграммасы 3.5-суретте көрсетілген.

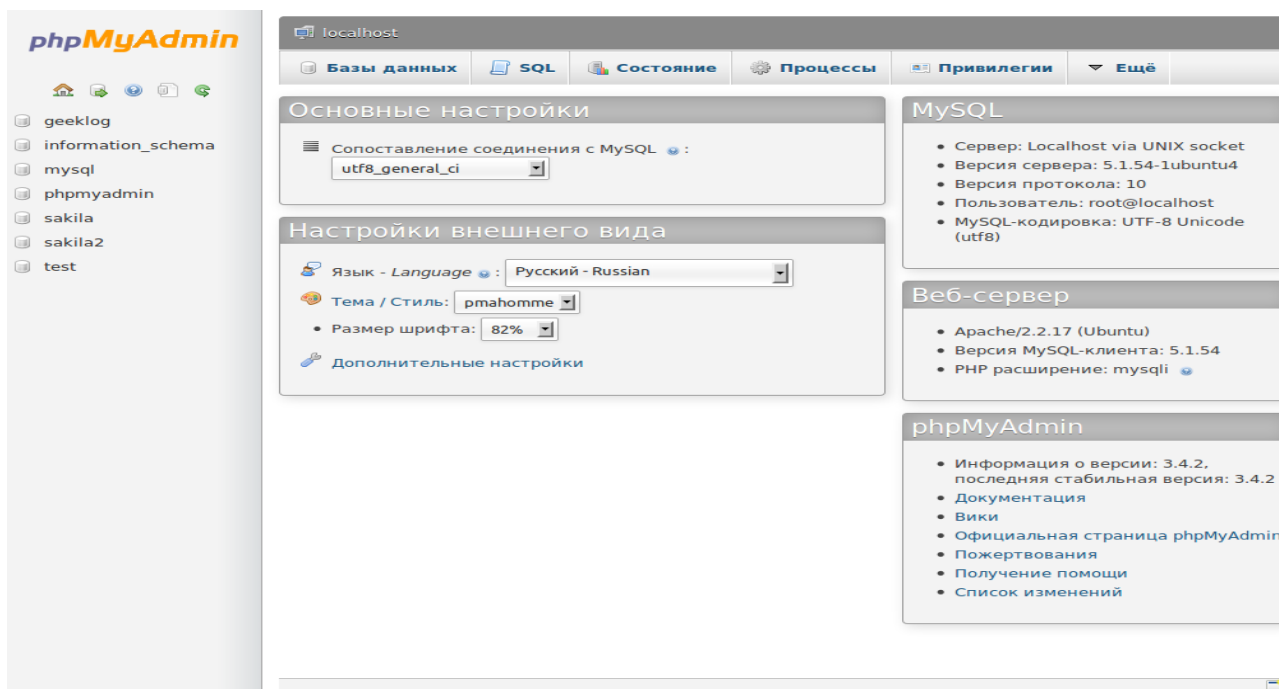


3.5-сурет – Класстар диаграммасы

Класс диаграммасы – қосымша кодын енгізу үшін негізгі диаграмма класы диаграмма көмегімен жүйенің ішкі жүйесі шығарылады, яғни мұралауды сипаттауын және қосымша кластар бір-бірімен байланысты. Осында жүйенің логикалық көрінісі сипатталады.

Кластар диаграммасы - енгізу үшін, қосымша кодты негізгі диаграмма шығарылады кіші жүйесін пайдалана отырып, диаграммалар сынып, яғни сипаттау мұрагерлік және қосымша сынып оқушылары бір-бірімен байланысты. Міне, жүйе туралы логикалық түсінік.

3.1



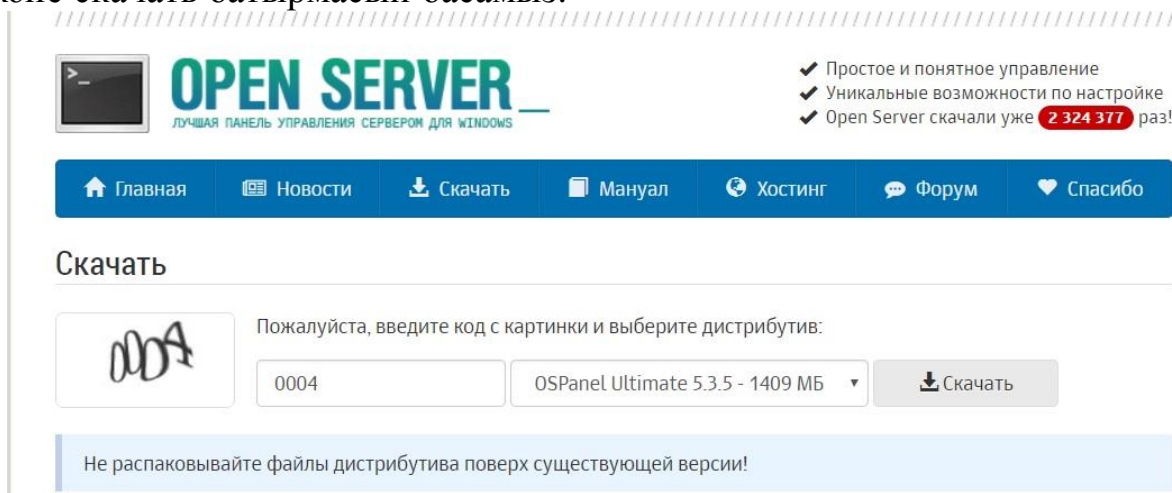
3.1 сурет - phpMyAdmin – нін жалпы түрі

Функционалдык тағайындалуы

Open Server веб-серверін орнату

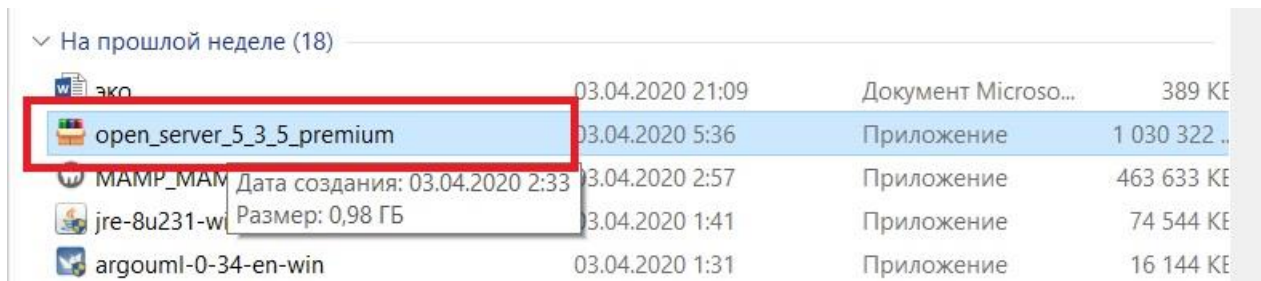
Open Server Panel - бұл ұсыныстар мен тілектерді ескере отырып, веб-әзірлеушілер үшін арнайы құрылған портативті серверлік платформа және бағдарламалық орта.

Open Server веб-серверін орнату үшін алдымен ресми сайтқа кіреміз және скачать батырмасын басамыз:



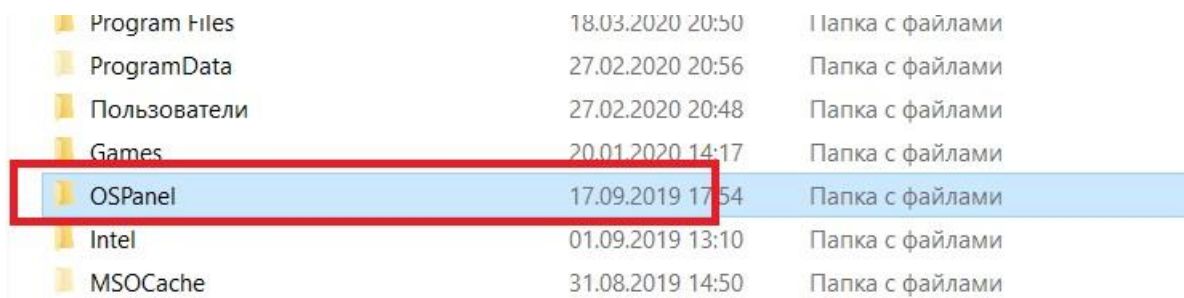
3.2 сурет – OpenServer – ді жүктеу

Одан соң жүктеліп алынған файлды ашамыз:



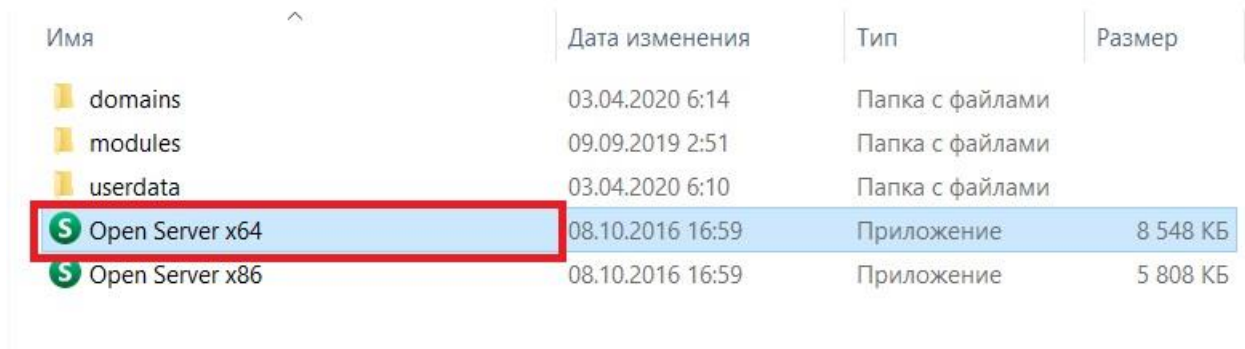
3.3 сурет – Файлды ашу

Файл ашып С дискісіне тарқатамыз (C://OSPanel):



3.4 сурет – С дискісіне тарқату

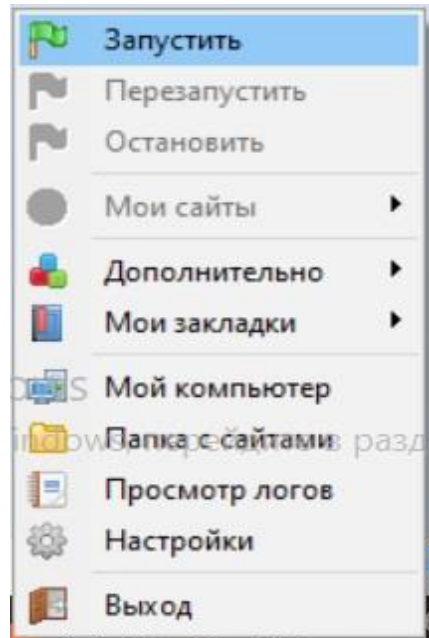
Тарқату процесі біткен соң, папкаға кіріп *Open Server x64* программасын іске қосамыз:



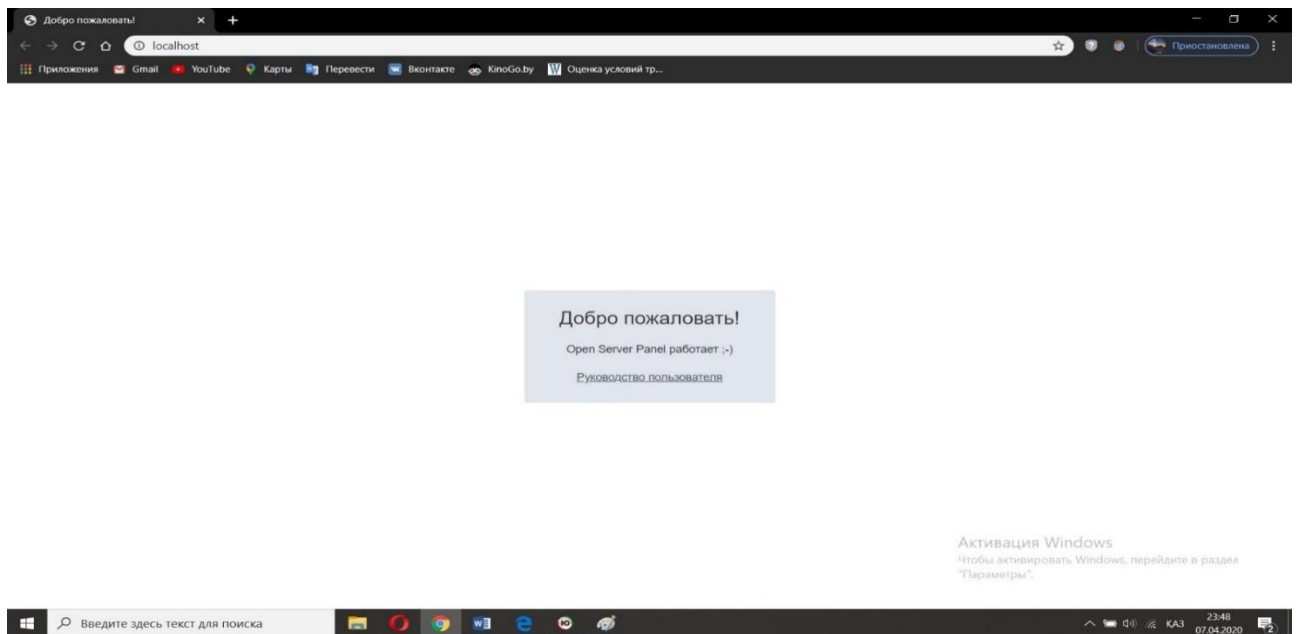
3.5 сурет – Open Server x64 программасын іске жіберу

Іске қосылу процесі аяқталғаннан кейін жұмыс үстелінің оң жақ астыңғы бұрышынан тудың белгісі шығу қажет, одан кейін тудың үстін басып Запустить батырмасын басамыз:





3.6 сурет – Open Server x64 программасын іске қосу



3.7 сурет - Open Server веб-сервері іске қосылды

Open Server веб-серверін орнату аяқталды.

Келесі қадам Open Server –ді администратор көмегімен іске қосамыз. Сосын Командная строка көмегі арқылы мобильді қосымшамызды шақырамыз.

```
Microsoft Windows [Version 10.0.17763.1217]
(c) Корпорация Майкрософт (Microsoft Corporation), 2018. Все права защищены.

C:\Users\Sabina>cd/sabina-diplom/shop-loopback

C:\sabina-diplom\shop-loopback>npm start

> shop-api@1.0.0 prestart C:\sabina-diplom\shop-loopback
> npm run build

> shop-api@1.0.0 build C:\sabina-diplom\shop-loopback
> lb-tsc

> shop-api@1.0.0 start C:\sabina-diplom\shop-loopback
> node -r source-map-support/register .

Server is running at http://[::1]:8500
Try http://[::1]:8500/ping
```

```
Windows PowerShell

[ng] chunk {runtime} runtime.js, runtime.js.map (runtime) 9.61 kB [entry] [rendered]
[ng] chunk {shadow-css-d7d058ec-d59cb009-js} shadow-css-d7d058ec-d59cb009-js.js, shadow-css-d7d058ec-d59cb009-js.js.map (shadow-css-d7d058ec-d59cb009-js) 15.9 kB [rendered]
[ng] chunk {status-tap-2ec46489-js} status-tap-2ec46489-js.js, status-tap-2ec46489-js.js.map (status-tap-2ec46489-js) 1.79 kB [rendered]
[ng] chunk {stencil-polyfills-css-shim} stencil-polyfills-css-shim.js, stencil-polyfills-css-shim.js.map (stencil-polyfills-css-shim) 22 kB [rendered]
[ng] chunk {stencil-polyfills-dom} stencil-polyfills-dom.js, stencil-polyfills-dom.js.map (stencil-polyfills-dom) 19.8 kB [rendered]
[ng] chunk {styles} styles.js, styles.js.map (styles) 119 kB [initial] [rendered]
[ng] chunk {swipe-back-d2cdbf9a-js} swipe-back-d2cdbf9a-js.js, swipe-back-d2cdbf9a-js.js.map (swipe-back-d2cdbf9a-js) 2.88 kB [rendered]
[ng] chunk {swiper-bundle-ccdaac54-js} swiper-bundle-ccdaac54-js.js, swiper-bundle-ccdaac54-js.js.map (swiper-bundle-ccdaac54-js) 176 kB [rendered]
[ng] chunk {tap-click-606f325e-js} tap-click-606f325e-js.js, tap-click-606f325e-js.js.map (tap-click-606f325e-js) 6.38 kB [rendered]
[ng] chunk {vendor} vendor.js, vendor.js.map (vendor) 5 MB [initial] [rendered]
[ng] Date: 2020-05-23T19:30:58.170Z - Hash: 35810a0c4f428435fd88 - Time: 8294ms

[INFO] Development server running!

Local: http://localhost:8100

Use Ctrl+C to quit this process

[INFO] Browser window opened to http://localhost:8100!

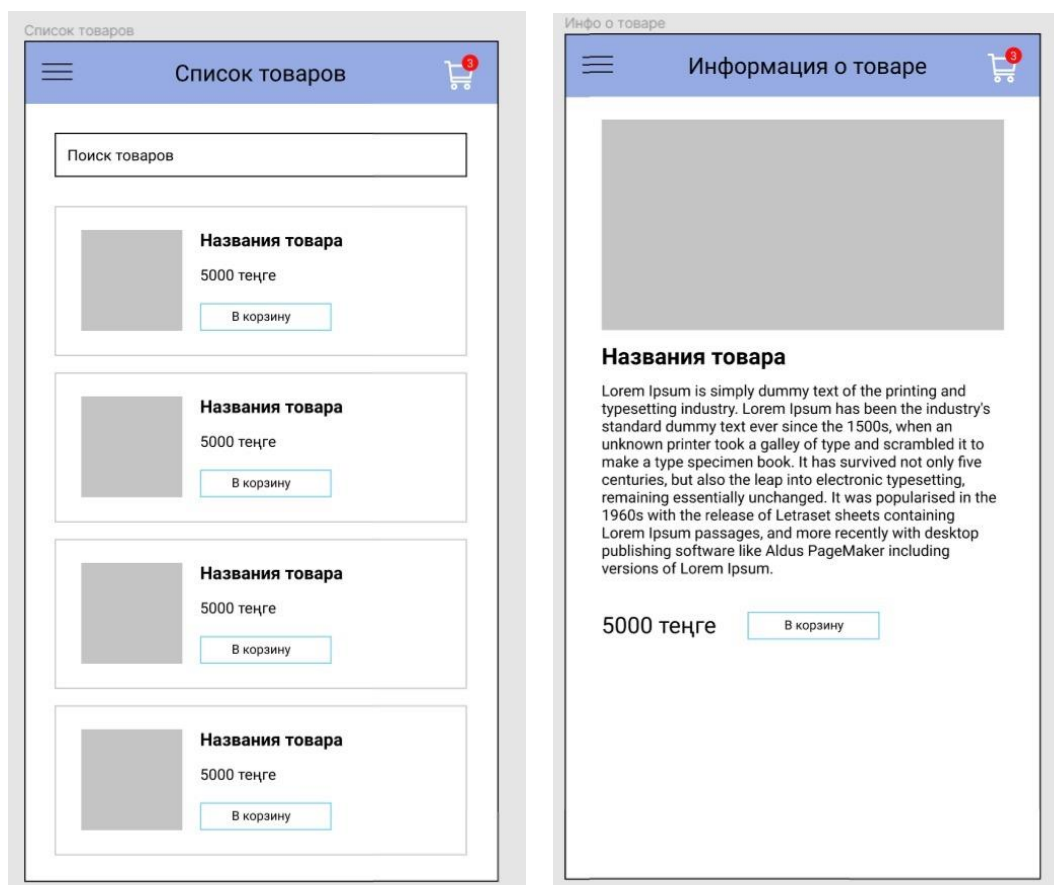
[INFO] ... and 77 additional chunks
[ng] i @wdm@: Compiled successfully.
```

3.8 сурет – Мобильді қосымшаны Командная строка арқылы шақыру
Ары қарай оның функционалдық қызметін көреміз.



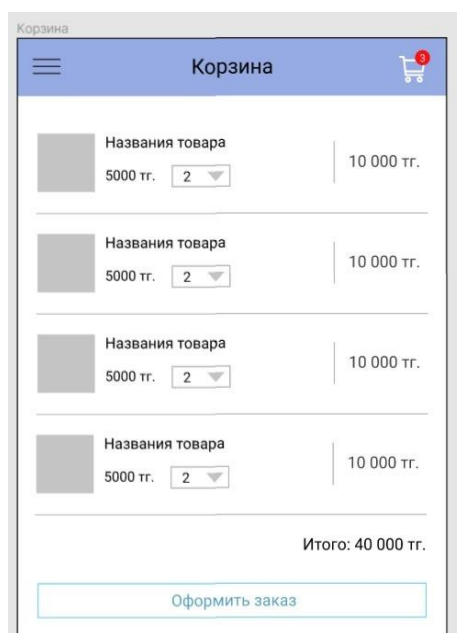
1.5 сурет . Негізгі беті(товар тізімі)

Мобильді қосымшаның негізгі беті болады(1.5 суретте). Бірінші бетінде категория товары болады(Хоз.товар, Тұрмыстық товар).



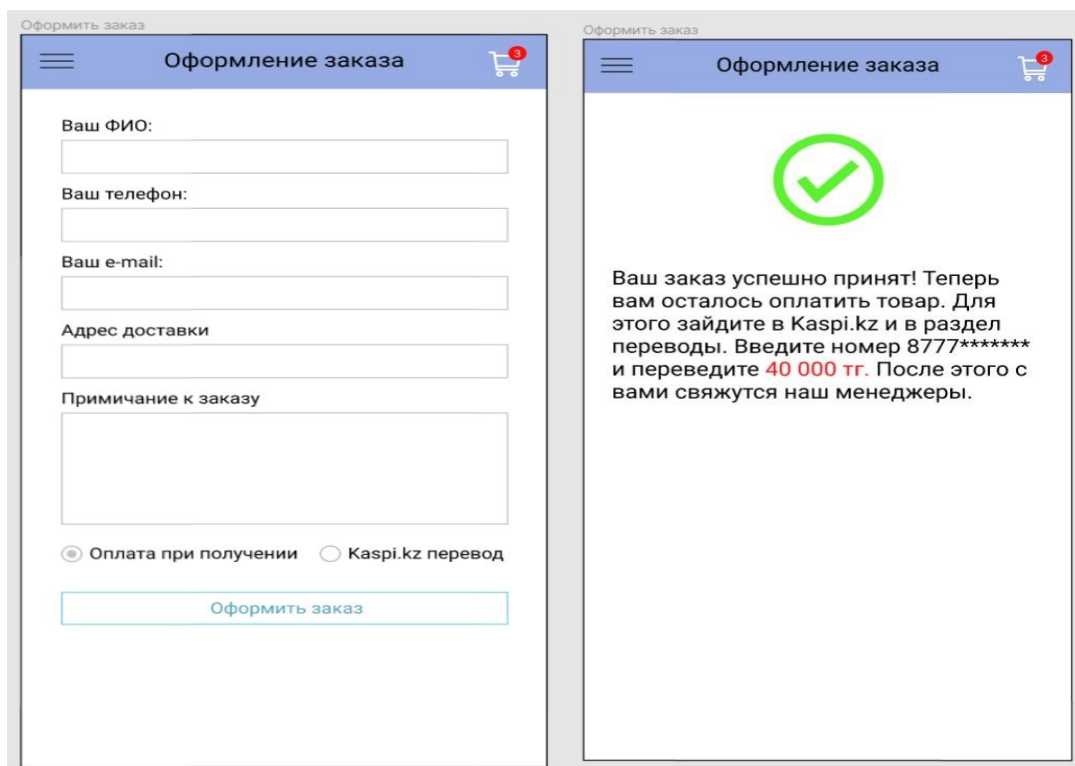
1.6 сурет. Келесі беттері

Сол категорияға товардың ішіне кіргенде товарлар болады. Ол жерде товардың бағасы беріледі. Егер сол товарға бассаныз, сол товар туралы ақпарат шығады. (1.6 сурет)



1.7 сурет Товарға оформлення жасау

Егерде товар ұнап жатса, онда оны корзина салады. Корзинада товар санын өзгерте алады, оформить деген батырманы басып оформление бетіне өтеді. Оформление бетінде аты-жөнінді, ұялы телефон, почтаныңызды, адрес деген бағандарды толтырып, төлеу ақысын тандайды.



1.8 сурет - Тапсырыс беру беті

Егер осы қадамдарды бәрін істеп болған соң, тапсырыс чегі шығады.

4 Экономикалық бөлім

4.1 Жобаны іске асыру кезеңдері мен мерзімдері

Бұл дипломдық жұмыста үйге қажетті заттар сатуға арналған мобильдік қосымша әзірленуде. Мақсаты компания үшін Android платформасында мобильді қосымшаны әзірлеу болып табылады. Бағдарлама қажетті өнімді есікке жеткізумен ыңғайлы түрде тапсырыс беруге көмектеседі. Бағдарлама интерфейсі өте қарапайым және ыңғайлы. Бұл бөлім қосымша әзірлеу үшін еңбек ақы мен ақша қаражатын төлеуге арналған шығыстарды есептейді және талдайды. 1-кестеде жұмыстардың кезеңдері мен түрлері бойынша бөлінуі және олардың еңбек сыйымдылығын бағалау берілген.

4.1 кесте-жұмыстарды кезеңдер мен түрлер бойынша бөлу және олардың еңбек сыйымдылығын бағалау

Қосымшаны әзірлеу кезеңдері	Кезеңдердің атауы	Орындаушы	Орындалу еңбек сыйымдылығы, адам*с.	Орындалу еңбек сыйымдылығы, күндер
1	Идеяларды әзірлеу және тұжырымдау	Басшы	16	2
2	Техникалық	Басшы	35	4

	тапсырманы әзірлеу (ТТ)			
3	ПҚ әзірлеу үшін қажетті материалдарды зерттеу	Программист	210	27
4	Нарықты талдау	Маркетолог	12	2
5	ПП бірінші прототипі	Программист	190	24
6	Тестілеу және түзету	Программист	60	8
7	Атқарылған жұмыс қорытындысы	Басшы	80	10
Орындалған жұмыстың сыйымдылығының жиынтығы			еңбек 603	77 (2,5 ай)

4.2 Мобильді қосымшаны әзірлеуге арналған шығындарды есептеу

Мобильді қосымшаны әзірлеуге арналған шығындарды есептеу тиісті сметаны жасау жолымен жүргізіледі, ол мынадай баптарды қамтиды:

- а) Материалдық шығындар;
- б) еңбекақы төлеу шығындары;
- в) Әлеуметтік салық;
- г) Негізгі қорлардың амортизациясы;
- д) өзге де шығындар.

4.2.1 Материалдық шығындарды есептеу

Материалдық шығындарға мобильді қосымшаны әзірлеу кезінде қолданылатын негізгі және қосымша материалдарға арналған шығыстар кіреді.

Бағдарламалық қамтамасыз ету мен жабдықтар үшін шығындарды есептеу 2-кестеде келтірілген.

2-кесте-жабдықтар мен бағдарламалық қамтамасыз етуге арналған шығындар

Атауы	Бірлік.өлшем	Саны	Бірлік бағасы, тг	Суммасы, тг
Ноутбук Lenovo Ideapad 330	шт.	1	149 990	149 990

OC Windows 10	Лицензиялық көшірме	1	0	0
MS Office 2016	Лицензиялық көшірме	1	0	0
Android Studio 6.1.0	Лицензиялық көшірме	1	0	0
Принтер Epson L-120 СНПЧ А4	шт.	1	49 990	49 990
Қорытынды				199 980

Материалдық ресурстарға жұмсалатын шығындарды есептеу 3-кестеде келтірілген.

3-кесте-материалдық ресурстардың шығындары

Атауы	Саны	Жұмсалған материалдың саны	Бірлік бағасы, тг	Жалпы сома, тг
Қағаз А4		1	1200	1200
USB накопитель	шт.	1	1800	1800
Қалам	шт.	5	50	250
Карандаш	шт.	3	40	120
Қорытынды				3370

Материалдық ресурстарға (ЖМ) шығындардың жалпы сомасы мынадай формула бойынша анықталады:

$$Z_M = \sum_{i=1}^n P_i \times C_i$$

мұндағы P_i – материалдық ресурстың i түрінің шығыны, тг;

C_i – материалдық ресурстың i түрінің бірлігінің бағасы, тг;

i – материалдық Ресурстың түрі;

n – материалдық ресурстар түрлерінің саны.

4.2.2 Электр энергиясына жұмсалған сомаларды есептеу

Мобильді қосымшаны әзірлеу кезінде техникалық жабдықтар пайдаланылды, сәйкесінше электр энергиясы жеткілікті.

Белгілі бір уақыт аралығында электр энергиясын тұтынуды есептеу кезінде мынадай сәттерді ескеру қажет:

- құралдың қуаты: қозғалтқыш немесе қуат блогы;
- аспаптың жұмыс істеу уақытының саны.

Электр энергиясына шығындарды есептеу 4-кестеде келтірілген нысан бойынша жүргізіледі.

4-кесте-электр энергиясына шығындар

Атауы	Жабдықтың қуаты, кВт	Қуатты пайдалану коэффициенті	Жабдықтың жұмыс уақыты, сағ	Жабдықтың жұмыс уақыты, күндер	Электр энергиясының бағасы, $\frac{\text{Тг}}{\text{ч} \times \text{кВт}}$	Соммасы, тг
Ноутбук Lenovo Ideapad 330	0,065	0,9	591	74	28,76	994
Принтер Epson L-120 СНПЧ А4	0,04	0,9	40	6	28,76	41

Электр энергиясына шығындар сомасы мынадай формула бойынша есептеледі:

$$Z_3 = M_i \times K_i \times T_i \times C$$

мұндағы M_i - I электр жабдығының паспорттық қуаты, кВт;

K_i - электр жабдығының қуатын пайдалану коэффициенті ($K_i=0.7$; 0.9 қабылданады);

T_i - қосымшаны әзірлеудің барлық кезеңіндегі i -ші жабдықтың жұмыс уақыты, сағ.;

C - электр энергиясының бағасы, тг/кВт×сағ.;

I - электр жабдығының түрі;

n - электр жабдықтарының саны.

Әрбір жабдық үшін есептеулерді келтіреміз:

Lenovo Ideapad 330 Ноутбук

$$Z_3 = 0,065 \times 0,9 \times 591 \times 28,76 = 994 \text{ тг}$$

Принтер Epson L-120 АЕМПЧ А4

$$Z_3 = 0,04 \times 0,9 \times 40 \times 28,76 = 41 \text{ тг}$$

Электр энергиясына жұмсалған қорытынды сома 1035 теңгені құрады.

4.2.3 Еңбекақы төлеу шығындарын есептеу

Еңбекке ақы төлеу шығындары мобильді қосымшаны әзірлейтін барлық қызметкерлердің еңбегіне ақы төлеу жөніндегі шығыс

Қызметкердің лауазымы	Айлық жалақы, тг	Сағаттық ставка, тг / сағ	Жұмысты орындаудың еңбек сыйымдылығы, адам*сағ	Сумма, тг
Программист	100 000	595	480	285 600
Маркетолог	70 000	417	12	5004
Еңбекақы төлеудің жиынтық сомасы				290 604

Еңбекақы төлеу шығындарының сомасы мына формула бойынша есептеледі:

$$З_{\text{тр}} = \sum_{i=1}^n ЧС_i \times T_i$$

мұндағы $ЧС_i$ - қызметкердің сағаттық мөлшерлемесі, тг;

T_i орындаудың еңбек сыйымдылығы, адам×сағ;

I -қызметкердің санаты;

n – қосымшаны әзірлеумен айналысатын қызметкерлердің саны.

Бір сағаттық табысы қызметкердің мынадай формула бойынша есептеледі:

$$ЧС_i = \frac{ЗП_i}{ФРБ_i}$$

онда $ЗП_i$ - бір айлық жалақысы i -ші қызметкердің тг;

i -айлық жұмыс уақытының қоры I -ші қызметкердің жұмыс уақыты, ГФР жағы = 168 сағ.;

Әр қызметкер үшін есеп жүргіземіз:

$$ЧС_1 = \frac{100\,000}{168} = 595 \text{ тг/ч}, \quad З_{\text{тр}1} = 595 \times 480 = 285\,600 \text{ тг.}$$

$$ЧС_2 = \frac{70\,000}{168} = 417 \text{ тг/ч}, \quad З_{\text{тр}1} = 417 \times 12 = 5004 \text{ тг.}$$

4.2.4 Есеп шығару әлеуметтік қажеттіліктері

Осы бөлім бойынша шығындар бірыңғай әлеуметтік салық (ЕСН) бойынша аударымдар құрайды.

Қазақстандағы әлеуметтік салық ставкасы 9,5% - ды құрайды. Салық салынатын сомасына кірмейді міндетті зейнетақы жарна (МЗЖ). 6-кестеде әлеуметтік қажеттіліктер бойынша аударымдар есептелген.

6-кесте – әлеуметтік қажеттіліктерді аудару

Заңды тұлға төлеген салықтар			Сумма, тг.
ӘШ(Әлеуметтік Шегерім)	3,5	(Жалақы-МЗЖ-)*3,5%	9 154,03
Аударымдар мен жарналар міндетті әлеуметтік медициналық сақтандыру(АЖМӘМС шегерімі)	2,0	Жалақы*2%	5 812,08
ӘС(Әлеуметтік салық)	9,5	Жалақы-МЗЖ-АЖМӘМС*9,5%- ӘШ	15 416,54
Барлық төленген салықтар			30 382,65

4.2.5 Амортизациялық аударымдарды есептеу

Амортизациялық аударымдардың жалпы сомасы мынадай формула бойынша анықталады:

$$Z_{AM} = \sum_{i=1}^n \frac{\Phi_i \times H_{Ai} \times T_{НИРi}}{100 \times T_{ЭФi}}$$

мұнда Φ_i НҚ құны, тг;

H_{Ai} -НҚ амортизациясының жылдық нормасы, %;

$T_{НИРi}$ -жыл ішіндегі i -НҚ жұмыс уақытының тиімді қоры, сағ / жыл;

I -НҚ түрі;

n – НҚ саны.

Есептеу кезінде құны ҚҚ ескеру керек, сондай-ақ шығындар, жеткізу және монтаждау, орнату бойынша Бұл шығындар НҚ сатып алу шығындарының 10-25% мөлшерінде қабылдануы мүмкін. НҚ амортизациясының жылдық нормалары ҚР Салық кодексі бойынша қабылданады немесе НҚ пайдалы пайдаланудың ықтимал мерзімін ескере отырып анықталады:

$$H_{Ai} = \frac{100}{T_{Ni}}$$

мұнда T_{Ni} - НҚ пайдаланудың ықтимал мерзімі, жыл;
 Әрбір жабдық үшін есептеулер жүргіземіз:

a) Lenovo Ideapad 330 Ноутбук

$$H_{Ai} = \frac{100}{4} = 25\%$$

b) Принтер Epson L-120 АЕМПЧ А4

$$H_{Ai} = \frac{100}{5} = 20\%$$

ҚҚ пайдалы пайдаланудың болжамды мерзімі 3 жылдан 10 жылға дейін қабылдануы мүмкін.

$$Z_{AM1} = \frac{149990 * 25 * 591}{100 * 1730} = 12\,809,84$$

$$Z_{AM1} = \frac{49990 * 20 * 40}{100 * 1730} = 230,75$$

6.7-кестеде негізгі қорлардың амортизациясы берілген.

6.7-кесте-негізгі қорлардың амортизациясы

Атауы	Құны,тг	Амортизацияның жылдық нормасы,%	Жұмыс уақытындағы тиімді фонд	Жабдықтың жұмыс уақыты, сағ	Құны,тг
Ноутбук Lenovo Ideapad 330	149 990	20	1730	591	12 809,84
Принтер Epson L- 120 СНПЧ А4	49 990	25	1730	40	230,75
Амортизацияның қорытынды сомасы					13 039,75

4.2.6 Өзге де шығындарды есептеу

Басқа шығындар әртүрлі қызметтерге арналған шығыстарды қамтиды.

Бағдарламаны әзірлеу үшін Tele2 желісі пайдаланылады. СУПЕР архивный" тарифтік жоспары таңдап алынды, оның құны мен жалпы соманың есебі 8-кестеде көрсетілген.

8 – кесте-интернетті пайдалану шығындары

1 айдағы құны, тг	Мерзімі, ай	Сомма, тг
1990	3	5970

Басқа шығындар 5970 теңгені құрайды.

Сметаны жасау

Барлық деректер негізінде мобильді қосымшаны әзірлеуге арналған шығындар сметасы жасалды (9-кесте).

9 – кесте-қосымшаны әзірлеуге арналған шығындар сметасы

Шығындар	Сомма, тг
Материалдық шығындар, сонымен қатар:	3370
Материалы	1035
Электрэнергиясы	
Еңбекақы төлеу шығындары	290 604
Әлеуметтік салық	30 382,65
Негізгі қорлардың амортизациясы	13 039,75
Басқада шығындар	5970
Жалпы сомасы	344 401,4

Материалдық шығындарды, оның ішінде материалдар мен энергия тұтынуды, еңбекақы төлеуге, әлеуметтік қажеттіліктерді аударуға, негізгі қордың амортизациясы мен басқа да шығындарды есептеуден кейін, нәтижесінде орындауға 344 401,4 теңге жұмсалды.

4.3 Мобильді қосымшаның ықтимал бағасын анықтау

Мобильді қосымшаның ықтимал бағасының шамасы Тапсырыс берушінің немесе тұтынушының және Орындаушының экономикалық мүдделеріне жауап беретін деңгейде оның орындалу тиімділігі, сапасы мен мерзімдері ескеріле отырып белгіленуі тиіс.

Қолданбалы мобильді қосымша үшін шарттық баға (Цд) мынадай формула бойынша есептеледі:

$$Ц_d = Z_{\text{НИР}} \times \left(1 + \frac{P}{100}\right),$$

мұнда $Z_{\text{НИР}}$ ҒЗЖ орындауға арналған шығындар (8-кестеден), тг.;

P - ҒЗЖ рентабельділігінің орташа деңгейі, % (экономикалық бөлім бойынша консультанттың келісімі бойынша 20% мөлшерінде қабылданады).

Одан кейін өткізу бағасы қосымша құн салығын (ҚҚС) есепке ала отырып есептеледі, ҚҚС ставкасы ҚР заңнамалық Салық кодексімен белгіленеді. 2020 жылға ҚҚС ставкасы мөлшерінде белгіленді 12%. ҚҚС есебімен өткізу бағасы мынадай формула бойынша есептеледі:

$$Ц_p = Ц_d + Ц_d \times НДС.$$

Барлық алдыңғы есептеулерді ескере отырып, мобильді қосымшаның ықтимал бағасын анықтаймыз:

$$Ц_d = 344\,401,4 \times \left(1 + \frac{20}{100}\right) = 413\,281,68 \text{ тг.}$$

ҚҚС есебімен баға:

$$Ц_p = 413\,281,68 + 413\,281,68 \times 0,12 = 462\,875,482 \text{ тг.}$$

4.4 Мобильді қосымшаның нәтижелілігі мен әлеуметтік тиімділігін бағалау

Кәсіпорындар үшін компанияның пайда көрсеткіштерінің өсуі өте маңызды.

Қосымшаларды әзірлеудің мақсаты тапсырыстарды көбейту және тапсырыс үшін дүкендердің разъезіне уақытты үнемдеу болып табылады. Экономикалық тиімділікті анықтаудың негізгі әдістерінің бірі қосымшаны пайдаланғанға дейін және кейін салыстыру әдісі болып табылады.

Осы әдіс бойынша экономикалық тиімділікті есептеу қосымшаны пайдаланғанға дейін және одан кейін қабылданған тапсырыстардың саны туралы деректер негізінде жүргізілді.

Орташа деректер 2020 жылғы 30 наурыздан 15 сәуірге дейінгі кезеңде алынған.

Қосымшаны пайдаланғанға дейін алынған кезең ішінде орташа сомасы 4 275 000 теңге болатын 95 тапсырыс жасалды. Қосымшаны қолданғаннан кейін тапсырыстар орташа сомасы – 5 650 000 болатын 113-ке дейін артты.

Келтірілген мәліметтерден экономикалық тиімділік қосымшадан сол шығындардан құралады деген қорытынды жасауға болады:

$$Э_n = 5\,650\,000 - 4\,275\,000 = 1\,375\,000 \text{ тг.}$$

Мобильді қосымшаны құру шығындары (күрделі салымдар) 462 875,482 теңгені құрады, сондықтан p рентабельділігі:

$$P = \frac{\text{Э}_\Pi}{\text{Ц}} * 100\% = \frac{1\,375\,000}{462\,875,482} * 100\% = 297\%$$

Жобаның өтімділік мерзімі:

$$T_{\text{ок}} = \frac{\text{Ц}}{\text{Э}_\Pi} = \frac{462\,875,482}{1\,375\,000} = 0,33 \text{ жыл немесе 4 ай}$$

Демек, шешімнің бұл жолы экономикалық тиімді болып саналады.

Осылайша, есептеулерді орындау барысында келесі нәтиже алынды:

-мобильді қосымшаны құру үшін 462 875,482 тг жұмсалды.;

-экономикалық тиімділік 1 375 000 тг. бұл әзірлеу шығындарынан 3 есе артық.;

- жобаның рентабельділігі 297% құрады.

- Келтірілген деректер қосымшаны қолданудан елеулі экономикалық

5 Өміртіршілік қауіпсіздігі бөлімі

Менің істеп жатқан дипломдық жұмысым «Uniprice.kz» ЖШС Компаниясының сұранысы бойынша Сауда орталығына жасалған мобильді қосымша.

Сауда орталығы (галерея, пассаж, сауда-ойын — сауық кешені, Сауда кешені, молл) - біртұтас ретінде басқарылатын және бір ғимаратта немесе ғимараттар кешенінде орналасқан сауда кәсіпорындарының тобы.

Сауда орталығы қаланың ортасында орналасқан және 5 қабатты құрайды, биіктігі 19 м, тәулігіне 12 сағат жыл бойы істейді, ішінде істейтін адам саны 500 құрайды.

Сауда орталығының жоспарына сәйкес кешеннің цокольды қабатында: өнім дүкендері бар аймақ, химиялық тазалау, әкімшілік кабинеттер, электр қалқанша, қойма, банкоматтар орналасқан. Бірінші қабат сауда нүктелеріне, қызметтік бөлмелерге, дәретханаларға арналған. Үшінші және төртінші деңгейде: food соrd аймағы, қызметтік және техникалық Үй-жайлар, дәретханалар, балаларға арналған ойын-сауық орталығы жобаланған. Бесінші қабатта кинотеатр орналасқан. Баспалдақ торларының шатыры мен кеңістігі әкімшілік кабинеттермен, желдеткіш камералармен жабдықталған. Барлық павильондар жеке кіреберістермен жабдықталған.(5 сурет)



5 сурет – Сауда Орталығынан сызбасы

Сауда орталығының интерактивті схемасының ең үлкен бөлігі-бұл әр қабаттық жоспарлау. Ол толық болсын, бірақ көзбен жеңіл. Кіру, тұрақ және басқа да нысандар орналасқан бөлімдер қалай орналасатыны түсінікті болуы тиіс.

5.1 Кәсіби қызметкердің жұмыс орнында еңбек талдауын сипаттау

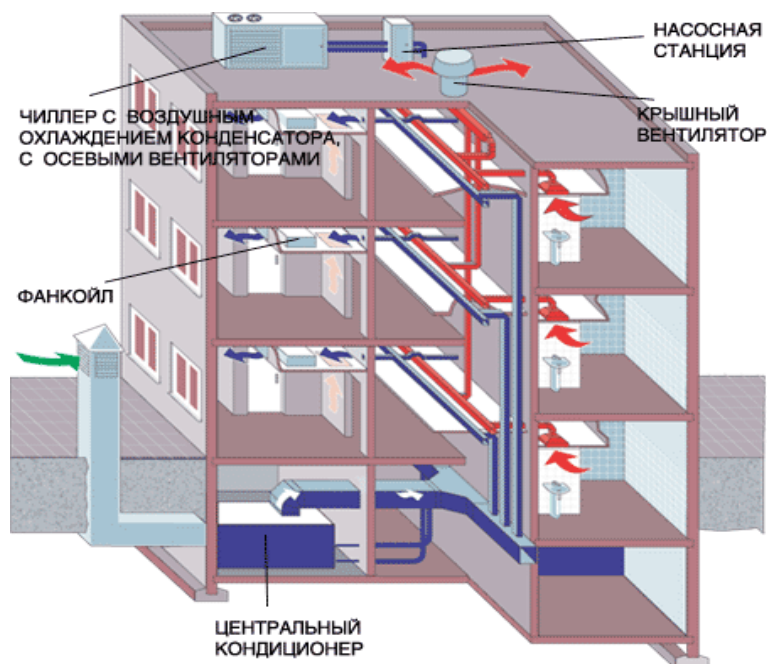
СО жабық мекеме болғандықтан және де адам санының көп болуына байланысты санитарлық-гигиеналық талаптарға сәйкес температураны, ылғалдылықты және таза ауаны ұстап тұру керек.

Өндірістік объектілерге қойылатын санитарлық-гигиеналық талаптарға сәйкес температураны, ылғалдылықты және таза ауаны ұстап тұру процесі ауаны баптау деп аталады. Кондиционерлік жүйеге қойылатын негізгі талаптардың бірі төрт айнымалы арасындағы белгілі бір арақатынасты реттеу: ауа температурасы; қоршаулардың ішкі беттерінің (қабырғалар, едендер, төбелер) орташа өлшенген температуралық мәні; ауа ылғалдылығы үй ішіндегі ауаның орташа жылдамдығы мен біркелкілігі. Сонымен қатар, ауаны баптау жүйесі бөлмедегі газдар, булар мен шаңдардың шоғырлануын реттеуі керек. Егер жүйе адамдарға қолайлы жағдай туғызуға арналған болса, онда адам ағзасынан бөлінетін иістерді де азайту керек.

Сауда орталығын кондиционерлеу үшін ең танымал шешім - чиллер-фанкойл жүйесі. Ол үй-жайлардың көп мөлшерде бір уақытта температураны тәуелсіз реттеу мүмкін болатын Суықпен және жылумен жабдықтау жүйесі болып табылады.

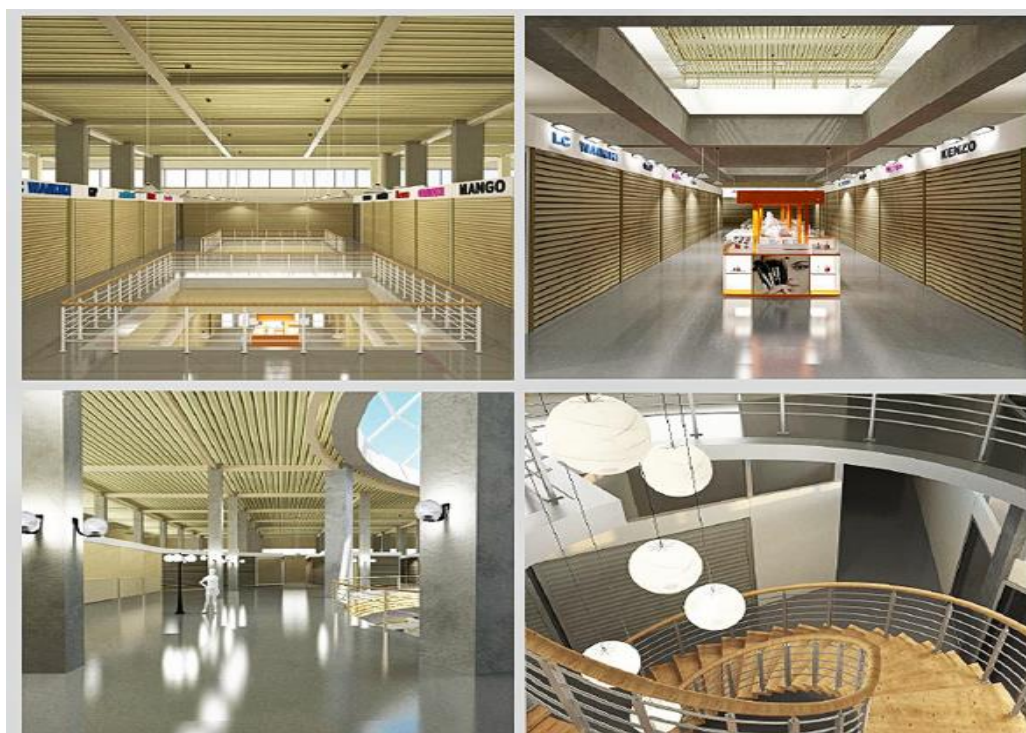
Сауда орталықтарына арналған кондиционерлер кешендердің жұмысы басталғанға дейін құрылыс кезеңінде жобаланады. Климаттық техника сатып алушылардың, компанияға пайда әкелетін және дүкеннің жұмысын қамтамасыз ететін қызметкерлердің жайлылығын қамтамасыз етеді, сондықтан онда үнемдеудің қажеті жоқ. Егер жабдықтың нақты түрі туралы

айтатын болсақ, онда салыстырмалы түрде шағын алаңның бөлмелері үшін қазіргі заманғы - чиллер-фанкойл жүйесі өте жақсы.



5.1.1 сурет- Сауда Орталығындағы кондиционердің орналасу схемасы.

Суық мезгілде сауда орталығында ауа температурасын қалыпқа келтіру үшін және сонымен бірге ауа ылғалдылығын реттеу үшін, ол жергілікті және орталық болып табылатын (әсер ету радиусы бойынша) жылытуға арналған. Сауда Орталығындағы бір помещениыдағы ауа алмасуды есептейміз. Учаскенің ауданы: 0,3180 га, ғимараттың ауданы: 4928,5 м²



5.1.2 сурет – Сауда орталығының ішкі көрінісі

Салқындатудың Q мөлшерін есептеу (киловаттпен) келесі тәртіп бойынша жүргізіледі:

$$Q = Q_1 + Q_2 + Q_3, \text{ мұндағы}$$

Q_1 - терезеден, қабырғалардан, еденнен және төбеден жылу ағыны.

S - бөлменің ауданы (м^2);

h - бөлменің биіктігі (м);

q - коэффициенті $30 - 40 \text{ Вт} / \text{м}^3$:

көлеңкелі бөлме үшін $q = 30$;

$q = 35$ орташа жарықта;

$q = 40$ күн сәулесінің көп түсетін бөлмелері үшін.

Егер күн сәулесі тікелей бөлмеге кірсе, онда терезелерде жеңіл перделер немесе перделер болуы керек.

Q_2 - бұл адамдардан алынатын жылу мөлшері.

Ересек адамнан жылу алу:

$0,1 \text{ кВт}$ - тыныш күйде;

$0,13 \text{ кВт}$ - жарық қозғалысымен;

$0,2 \text{ кВт}$ - физикалық жүктеме кезінде;

Q_3 - Техникадан алынатын жылу мөлшері.

$0,3 \text{ кВт}$ - компьютерден;

$0,2 \text{ кВт}$ - теледидардан;

Басқа құрылғылар үшін олар жылу түрінде максималды қуат тұтынудың 30% шығарады деп болжауға болады (яғни, орташа қуат тұтыну максимумның 30% құрайды деп болжанады).

Төменгі биіктігі 3 м болатын, ені 15 м , 150 м^2 СО бір бөлмесінің кондиционерлердің қуатын есептейміз, сонымен қатар компьютер, камера, теледидар және максималды қуат тұтынуы 165 ватт сорғышты желдету (вытяжной вентиляции) бар. Жасанды сәулеленуденде жылу бөлінеді. Соны бәрін ескеру керек.

Біріншіден, біз терезеден, қабырғалардан, еденнен және төбеден жылу шығынын анықтаймыз. Q коэффициенті 40-қа тең, өйткені бөлме күншығыс жақта орналасқан;

$$Q_1 = S * h * q / 1000 = 150^2 * 3 \text{ м} * 40 / 1000 = 18 \text{ кВт.}$$

Тыныш күйде бір адамның жылу ағыны 0,1 кВт құрайды. 32 адамды есептейтын болсақ - $Q_2 = 3,2 \text{ кВт}$

Әрі қарай, компьютер мен теледидар бір уақытта жұмыс істейтіндіктен, есептеулерде осы құрылғылардың тек біреуін, атап айтқанда көп жылу шығаратын қондырғыны да ескеру қажет. Бұл жылу шығысы 0,3 кВт болатын 40 компьютер 12кВт тең. Сорғышты желдету (вытяжной вентиляции) жылу түрінде электр қуатын тұтынудың шамамен 30% -ын құрайды, яғни $0,165 \text{ кВт} * 30\% / 100\% \approx 0,05 \text{ кВт}$.

$$Q_3 = 12 \text{ кВт} + 0,05 \text{ кВт} = 12,5 \text{ кВт}$$

Енді кондиционердің есептік қуатын анықтай аламыз.

$$Q = Q_1 + Q_2 + Q_3 = 18 \text{ кВт} + 3,2 \text{ кВт} + 12,5 \text{ кВт} = 33,7 \text{ кВт}$$

Кондиционердің күші Q_{range} диапазонында номиналды Q-дан -5% -дан + 15% дейін болуы керек.

Ұсынылатын қуат диапазоны Q_r (номиналды қуаттың -5% -дан + 15% дейін): $33,7 \text{ кВт} < Q_{range} < 3.80 \text{ кВт}$



5.1.3 сурет-СО дағы кондиционер түрлері

Бұл есептеу барысында Сауда орталығындағы бір бөлменің есептедік және ауа алмасуы нормаға сәйкес емес. Себебі СО жабық мекеме болған соң, біріншіден терезелері аз болғандықтан, күн сәулесі дұрыс кірмейді, жасанды сәулендіреді. Сонда бізге кондиционердің күші мен Сорғышты желдету бөлмедегі ауа, ауа алмасу шкаласымен жеткілікті деңгейде емес.

5.2 Төтенше жағдайда сауда орталығының эвакуациялау уақытын есептеу

СО–ның басты міндеті адам қауіпсіздікте болуы, әсіресе төтенше жағдайда. Төтенше жағдайдан қорғаудың негізгі әдістерінің бірі - қызметкерлерді уақытылы эвакуациялау және қауіпті аймақтар мен апат аймақтарынан шығару.

Эвакуация - адамдардың өмірін сақтау және өндірістің жұмыс істеуіне жағдай жасау мақсатында халық пен материалдық құндылықтарды төтенше жағдайлар аймақтарынан және осы заманғы зақымдау құралдары қолданылуы мүмкін аудандардан ұйымдасқан түрде әкету (шығару).

Ғимараттан сыртқа шығуға дейін адамдарды көшіру ұзақтығын көшіру жолдарының созылмалылығы және есіктер мен баспалдақтардың өткізу қабілеттері бойынша анықтайды. Есептеу ағындар тығыздығының біргелілігі және максималды мәніне жетуі жағдайы үшін жүргізіледі.

МСТ 12.1.004-91, адамдарды көшірудің жалпы уақыты «өрттің басталуынан адамдарды көшірудің басталуына дейінгі» $m_{ЭБ}$, аралықтан және көшірунің есептік уақытынан, t_p , құралады, ол көшірунің басталу кезінен бастап, оның маршрутының жеке бөліктері бойынша адамдардың тұрған орындарынан кабинетден, қабаттан, ғимараттан көшірулік шығуларға дейінгі адам ағынының қозғалыс уақытының қосындысын (t_c) білдіреді.

Біздің елде алғаш рет көшірунің басталу уақытын есепке алу керектігі МСТ 12.1.004-91 орнатылған. Әр елдерде жүргізілген зерттеулер көрсеткендей, өрт туралы хабар алған кезде адам жағдайды зерттеп, өрт туралы хабарлап, өртпен күресуге тырысып, заттарын жинауға, көмек көрсетуге және т.б. істеуге кіріседі. Көшіруді бастауды созу уақытының (хабарлау жүйесі болған кезде) орташа мәні үлкен болмауы мүмкін, бірақ кейде үлкен мәнге жетуі мүмкін.

Сонымен бірге, бұл кезеңнің ұзақтығы жалпы көшіру уақытына едәуір әсерін білдіреді, қандай факторлар (осы факторлардың көбі барлық көшіру үрдісіне әсер ететінін ескеру керек) оның мәнін анықтайтынын білген өте маңызды. Осы тұрғыдан бар жұмыстарға сүйене отырып келесілерді атап көрсетуге болады:

- адам жағдайы: тұрақты факторлар (сезім органдарының шектелуі, физикалық шектеу, уақытша факторлар (ұйқы/сергектік), шаршау, ширығу, сонымен бірге маскүнемдік);
- хабарлау жүйесі;
- қызметшілердің әрекеттері;
- адамның әлеуметтік және туыстық байланыстары;
- өртке қарсы тренинг жүргізілуі;
- ғимарат түрі.

Көшіруді бастаудың созылу уақыты D қосымшасына сәйкес алынады.

Адамдарды көшірудің есептік уақытын (t_p) жолдың жеке бөліктері бойынша адам ағыны қозғалысының қосынды уақыты ретінде анықтау керек t_f :

$$t_p = t_{э Б} + t_1 + t_2 + t_3 + \dots + t_i \quad (5.1)$$

мұндағы $t_{э Б}$ – көшіруді бастаудың созылу уақыты;
 t_1 – бірінші бөліктегі адам ағынының қозғалыс уақыты, мин.;
 t_2, t_3, t_i – біріншіден кейінгі келесі әрбір жол бөліктеріндегі адам ағыны қозғалысының уақыты, мин.

Есептеу кезінде адам ағыны қозғалысының барлық жолы бөліктерге (өткел, дәліз, есік аралығы, баспалдақ басқышы, тамбур) ұзындығы, l , және ені, b_j болып бөлінеді. Бастапқы бөліктер болып жұмыс орындары арасындағы өткелдер, жабдықтар және т.б. жатады.

Есептік уақытты анықтаған кезде көшіру жолының әрбір бөлігінің ұзындығы мен ені жоба бойынша қабылданады. Баспалдақ сатысы бойынша, сонымен бірге пандус бойынша жол ұзындығы саты ұзындығы бойынша өлшенеді. Есік аралығындағы жол ұзындығы нөлге тең деп қабылданады. Қалыңдығы 0,7 м қабырғада орналасқан аралықты, сонымен қатар тамбурды ұзындығы жоқ жазық жолдың жеке бөлігі деп санау керек.

Адам ағынының қозғалыс уақыты (t), мин бірінші бөлік жолы бойынша мына теңдеумен есептеледі:

$$t_1 = \frac{L_1}{V_1} \quad (5.2)$$

Мұндағы, L_1 – бірінші бөлік жолының ұзындығы, м;
 V_1 – бірінші бөлікте жазық жол бойынша адам ағыны қозғалысының жылдамдығының мәні D , m^2/m^2 , салыстырмалы тығыздыққа байланысты анықталады.

Бірінші бөліктегі адам ағынының тығыздығы (D), м/м, теңдеу бойынша анықталады:

$$D_1 = \frac{N_1 * f}{L_1 * b_1} \quad (5.3)$$

Мұндағы, N_1 – бірінші бөліктегі адамдар саны, адам;
 f – адамның жазық сұлбесінің орташа ауданы, E , $m^2/адам$, 4.2 кесте бойынша қабылданады;

5.2 кесте – Адамның көлденең проекциясының ауданы

Жасы, адам киімі және жүк түрі	адамның проекциясының көлденең ауданы f , m^2
--------------------------------	---

Ересек адам:	
жазғы киімде	0,10
маусымдық киімде	0,113,
қысқы киімде	0,125
қолында нәресте бар	0,285
рюкзакпен	0,315
жеңіл жиналуымен	0,235
Жасөспірім	0,07
Бала	0,04—0,05

L_1 және b_1 – жолдың бірінші бөлігінің ұзындығы және ені, м.

Біріншіден кейінгі жол бөліктеріндегі адам ағынының жылдамдығы V/E қосымшасының Е.2 кестесі бойынша жолдың осы бөліктерінің әрбірі бойынша адам ағынының қозғалыс қарқындығының мәндеріне байланысты қабылданады, оларды жолдың барлық бөліктері үшін, соның ішінде есік аралықтары үшін де теңдеу бойынша есептеледі:

$$q_i = \frac{q_{i-1} - b_{i-1}}{b_i}$$

(5.4)

Мұндағы, b_i, b_{i-1} – қарастырылатын i - және одан кейінгі жол бөліктерінің ені, м;

q_i, q_{i-1} – қарастырылатын i - және одан кейінгі жол бөліктері бойынша адам ағынының қарқындылығының мәні, м/мин.

Егерде (5.4) теңдеу бойынша анықталатын q_i мәні q_{max} , мәнінен аз немесе тең болса, онда минутына жол бөлігі бойынша қозғалыс уақыты (t_i): бұл кезде q_{max} мәнін 5.1 - кесте бойынша қабылдау керек.

5.2 кесте - Адам ағыны қозғалысының қарқындығы

Жол түрі	Қозғалыс қарқындығы, м/мин
жазық	16,5
есік аралығы	19,6
баспалдақ төмен	16

Егерде (5.4) теңдеу бойынша анықталатын q_h мәні q_{max} , үлкен болса, онда жолдың берілген бөлігінің енін b_j белгілі бір мәнге дейін жоғарлатады, ол кезде мына жағдай сақталады:

$$q_i \leq q_{max} \quad (5.5)$$

(5.5) жағдайының орындалмайтын кезінде жолдың i бөлігіндегі адам ағыны қозғалысының қарқындығы және жылдамдығы E қосымшасының Е.2 кестесі бойынша $D = 0,9$ және жоғары мәндері кезінде анықталады. Бұл кезде пайда болған кептелістен адамдар қозғалыстарының кідіру уақыты есептелу керек.

Бөліктің басында i екі және одан көп адамдар ағындарының қосылуы кезінде қозғалыс қарқындығы (q_i), м/мин. теңдеу бойынша есептеледі:

$$q_i = \frac{\sum q_{i-1} * b_{i-1}}{b_i}$$

Мұндағы, q_{i-1} - бөліктің басында қосылатын адам ағыны қозғалысының қарқындығы, м/мин;

b_{i-1} - қосылу жолы бөлігінің ені, м;

b_i - қарастырылатын жол бөлігінің ені, м.

Егерде (5.6) теңдеу бойынша анықталатын берілген жол бөлігінің q_i мәні q_{max} мәнінен үлкен болса, онда b_i - енін (4.5) жағдайы сақталу үшін белгілі бір шамаға дейін үлкейту керек. Бұл жағдайда i бөлігі бойынша қозғалыс уақыты (4.4) теңдеуі бойынша анықталады.

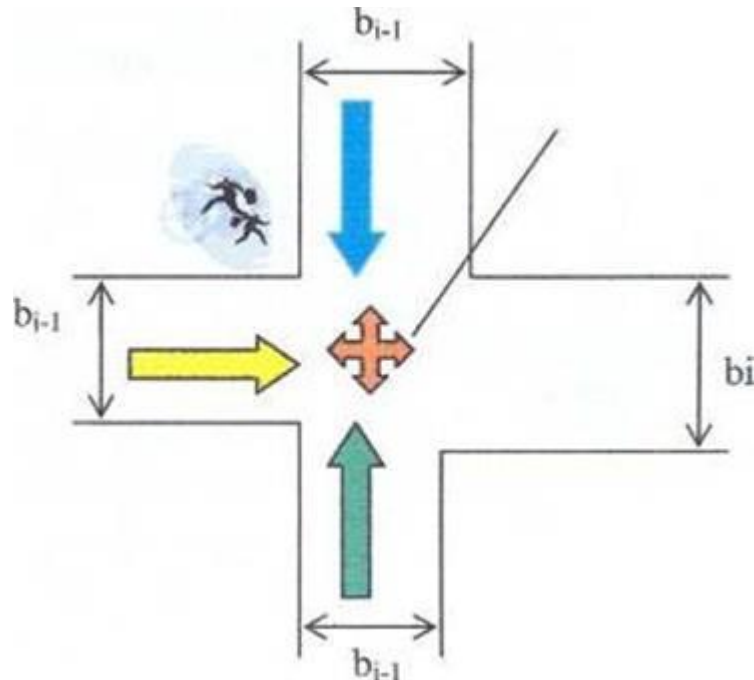
Ені 1,6 м – ден кем есік аралығындағы қозғалыс қарқындылығы теңдеу бойынша анықталады:

$$q_d = 2.5 + 3.75 * b \quad (5.7)$$

Мұндағы, b – аралық ені.

Аралық арқылы қозғалыс уақыты аралықтың өткізгіш қабілетіне ағындағы адам санының жеке бөлігі ретінде анықталады:

$$q_d = \frac{N * f}{q * b} \quad (5.8)$$



5.3.1 сурет - Адам ағындарының қосылуы

Есептеуді жүргізу тәртібі

1. Ғимараттың және кабинеттің санатын және дәрежесін анықтау.

2. $\tau_{n.k} = \sqrt[3]{\frac{w_{\text{каб}} * c * (t_{\text{дағ}} - t_{\text{н}})}{(1-\varphi) * \pi * Q * n * V^2}}$ немесе $\tau_{n.k} = \sqrt[3]{\frac{w_{\text{каб}} * c * (t_{\text{дағ}} - t_{\text{н}})}{(1-\varphi) * \pi * Q * n * V^2}}$ теңдеу

бойынша өрттің дағдарыстық ұзақтығын есептеу.

3. (5.10) теңдеуі бойынша өрттің дағдарыстық ұзақтығын оттегінің құрамының азаюы бойынша есептеу.

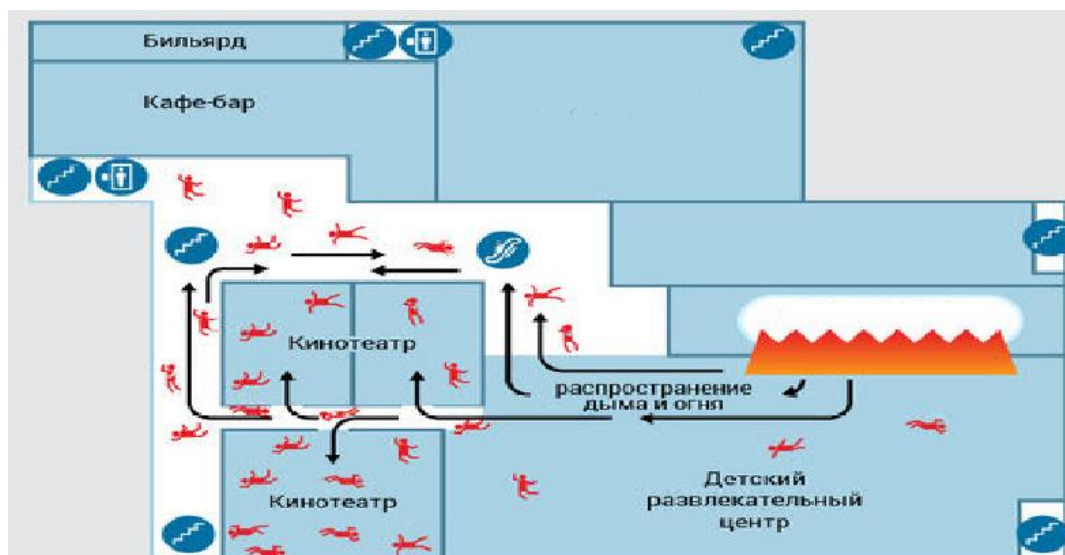
4. Өрттің есептелген дағдарыстық ұзақтығынан ең азын таңдау және сол бойынша $\tau_{\text{доп}}^1 = m * \tau_{n.k}^1$ теңдеумен рұқсат етілген көшіру ұзақтығын есептеу.

5. (5.1) теңдеуді пайдаланып өрт кезінде адамдарды көшірудің есептік уақытын анықтаңдар.

6. Есептік және рұқсат етілген көшіру уақытын салыстыру және қорытынды жасау.

Эвакуациялау уақытын есептеу

Ғимаратта өрт пайда болған кезде Сауда Орталығынан қызметкерлерді көшіру уақытын анықтау керек. Адамдарды ғимараттан көшіру сызбасы 5.3-суретте келтірілген.



5.3.2 сурет – Сауда Орталығынан қызметкерлерді көшіру сызбасы.

Көшіру уақытын есептеу

Сауда орталығы нормалар бойынша өрт қауіпсіздігінің II дәрежесіне сәйкес салынған, олардың конструкциялары мен әрлеу материалдары белгілі бір параметрлерге сәйкес.

- Температурасы бойынша өрттің дағдарыстық ұзақтығы (5.9) теңдеу бойынша кабинеттегі жиһазды ескере отырып есептеледі:

$$\tau_{n.k} = \sqrt[3]{\frac{W_{\text{пом}} * c * (t_{\text{кр}} - t_n)}{(1 - \varphi) * \pi * Q * n * V^2}} = \tau_{n.k} = \sqrt[3]{\frac{1008 * 1009 * (70 - 20)}{(1 - 0.5) * 3.14 * 13800 * 14 * 0.36^2}} = \sqrt[3]{129.36} = 10.5 \text{ мин}$$

- Оттегі құрамы бойынша өрттің дағдарыстық ұзақтығы (5.10) теңдеу бойынша есептеледі:

$$\tau_{n.k}^{O_2} = \sqrt[3]{\frac{(0.01)^{-1} * W_{\text{каб}}}{\pi * n * W_{O_2} * V_2}} = \sqrt[3]{\frac{100 * 100.8}{3.14 * 14 * 4.76 * 0.36^2}} = \sqrt[3]{371.69} = 7.19 \text{ мин}$$

- Температура бойынша өрттің минималды ұзақтығы 5,05 минут. Берілген кабинет үшін рұқсат етілген көшіру ұзақтығы:

$$\tau_{\text{кос}}^1 = m * \tau_{n.k}^1 = 1 * 5.05 = 10.5 \text{ мин}$$

- Көшірудің басталуының бөгелу уақыты D қосымшасының Д.1 кестесі бойынша 4,1 минут деп, өрт туралы автоматты дабылы және хабарлағыш жүйелері жоғы ескеріліп қабылданады.

- Кабинеттің 6×7 м көлемді шамаларын ескере отырып бірінші бөлік бойынша адамдардың қозғалыс уақытын анықтау үшін, (5.3) теңдеу бойынша бірінші бөліктегі адам ағыны қозғалысының тығыздығы анықталады:

$$D_1 = \frac{N_1 * f}{L_1 * b_1} = \frac{7 * 0.1}{6 * 7} = 0,01 \text{ м}^2/\text{м}^2$$

- Қозғалыс жылдамдығы E қосымшаның Е.2 кестесі бойынша 100 м/минутты, қозғалыс қарқындылығы 1 м/минут, сонымен бірінші бөлік бойынша қозғалыс уақыты:

$$t_1 = \frac{L_1}{V_1} = \frac{7}{100} = 0,07 \text{ мин.}$$

- Есік аралағының ұзындығы нөлге тең деп қабылданады. Дұрыс жағдайларда аралықта қозғалыс қарқындылығының мүмкін болатын өте үлкен мәні $g_{\text{mffic}} = 19,6$ м/минут, ені 1.1 м аралықта қозғалыс қарқындылығы (5.7) теңдеу бойынша есептеледі:

$$q_d = 2.5 + 3.75 * b = 2,5 + 3,75 * 1,1 = 6,62 \text{ м/мин. } q_d \leq q_{\text{max}}$$

$q_d \leq q_{\text{max}}$ сол себепті аралық арқылы қозғалыс бөгеліссіз өтеді.

- Аралықтағы қозғалыс уақыты (4.8) теңдеу бойынша анықталады:

$$q_d = \frac{N * f}{q * b} = \frac{7 * 0,1}{6,62 * 1,1} = 0,09 \text{ мин.}$$

- Себебі екінші қабатта 98 адам жұмыс істейді, екінші қабаттағы адам ағынының тығыздығы мынаны құрайды:

$$D_2 = \frac{N_2 * f}{L_2 * b_2} = \frac{98 * 0.1}{28 * 3} = 0,11 \text{ м}^2/\text{м}^2$$

- E қосымшаның Е.2 кестесі бойынша қозғалыс жылдамдығы 80 м/минут, қозғалыс қарқындылығы 8 м/минут, сонымен екінші бөлік (баспалдаққа дәлізден) бойынша қозғалыс уақыты:

$$t_2 = \frac{L_2}{V_2} = \frac{28}{80} = 0,35 \text{ мин.}$$

- Баспалдақ бойынша қозғалыс жылдамдығын анықтау үшін (5.4) теңдеу бойынша үшінші бөлікте қозғалыс қарқындылығы есептеледі:

$$q_i = \frac{q_{i-1} - b_{i-1}}{b_i} = \frac{8 * 3}{1,5} = 16 \text{ м/мин.}$$

- Бұл баспалдақта адам ағынының жылдамдығы 40 м/минутқа дейін төмендейтінін көрсетеді. Баспалдақ бойынша төмен (3-бөлік) қозғалыс уақыты:

$$t_3 = \frac{L_3}{V_3} = \frac{10}{40} = 0,25 \text{ мин.}$$

- Бірінші қабатқа өтер кезде бірінші қабат бойынша қозғалыстағы адамдар ағынымен қосылуы жүреді. Бірінші қабат үшін адам ағынының тығыздығы:

$$D_4 = \frac{N_4 * f}{L_4 * b_4} = \frac{76 * 0,1}{28 * 3} = 0,09 \text{ м}^2/\text{м}^2$$

Бұл кезде қозғалыс қарқындылығы 8 м/минутқа жуықты құрайды.

- 4-бөлікке өткен кезде адамдардың ағындарының қосылуы жүреді, сол себепті қозғалыс қарқындылығы (5.6) теңдеу бойынша анықталады:

$$q_i = \frac{\sum q_{i-1} * b_{i-1}}{b_i} = \frac{(16 * 1,5) + (8 * 3)}{3} = 16 \text{ м/мин.}$$

-Е қосымшаның Е.2 кестесі бойынша қозғалыс жылдамдығы 40 м/минутқа тең, сол себепті қозғалыс жылдамдығы бірінші қабат дәлізі бойынша:

$$t_4 = \frac{L_4}{V_4} = \frac{28}{40} = 0,7 \text{ мин.}$$

- Тамбур көшеге шығу кезіндегі ұзындығы 5 метр болады, бұл бөлікте адам ағынының максималды тығыздығы пайда болады, сол себепті қосымша мәліметтеріне сәйкес жылдамдық 15 м/минутқа дейін төмендейді, ал тамбур бойынша қозғалыс уақыты (4.2) теңдеу бойынша мынаны құрайды:

$$t_5 = \frac{L_5}{V_5} = \frac{5}{15} = 0,3 \text{ мин.}$$

- Көшеге ені 1,6 м аса есік аралығы арқылы адам ағынының максималды тығыздығы кезіндегі қозғалыс қарқындылығы – 8,5 м/минут, ол арқылы қозғалыс уақыты:

$$D_{a2} = \frac{N * f}{q * f} = \frac{174 * 0,1}{8,5 * 2} = 1,02 \text{ мин.}$$

- Көшіруның есептік уақыты (4.1) теңдеу бойынша есептеледі:

$$t_p = t_{э Б} + t_1 + t_2 + t_3 + \dots + t_i = 4,1 + 0,07 + 0,09 + 0,35 + 0,25 + 0,7 + 0,3 + 1,02 = 6,88$$

мин.

Сонымен, СО-да әрбір 6 мың шаршы метрге екі шығу бар. Әсіресе, екінші қабаттарға қатаң талаптар қойылған, онда ереже бойынша әрбір 40 м сайын эвакуациялық шығу жолдары жайғастырылған. Кіру және адам ағынының санына қарай өту жолдарының, дәліздердің (2.5 м-ден кем емес) және баспалдақтың (1.5 м-ден кем емес) ені есептеледі. Эвакуациялық шығу жолдарының конструкциясы, яғни баспалдақ торлары жанбайтын материалдардан (бетон, кірпіш және т.б.) орындалуы тиіс, себебі олар қауіпсіз аймақ болып табылады. Сондай-ақ СОО-да су шымылдығы деп аталатын суландырғыштар орнатыл. Оның рөлін спринклерлік және дренчерлік қондырғылар орындайды. Бірінші үлкен қысымда берілетін су жылу құлыптары немесе терморегуляцияланатын қолбалары бар клапандармен ұсталады. Температураның жоғарылауы кезінде қолба автоматты түрде жарылады, ол шамамен 12 ш. м. (0.08-0.12 л / см² су) сумен суланады. Екінші клапандар жоқ, олар қашықтан немесе қолмен іске қосылады. Өрт кезінде суландырғыштар дабыл бергенге дейін автоматты түрде қосылады.

Қорытынды

Дипломдық жобам толық менім практикамен толықтай байланысты. ЖШС “Uniprice.kz”-тың товарларды көре алатын және товар тапсырыс бере алатын қосымша құрдық. Ол тапсырыс беру формасы тікелей базамен байланысты.

Бірінші бөлімде біз теориялық бөлім болып саналатын, жалпы мобильді қосымша туралы айтып кеттік.

Екінші бөлімде қандай программамен жұмыс істеген туралы айтып кеттік, яғни, деректер қорын қарастырдық және де жүйенің аспектілік модельін жасадық. Деректер қорында бізде 8 кесте құрылды. Әрбір кесте өзіндік тағайындайылуы бар. Тапсырыс барысында клиенттің барлық тапсырысы осы деректер қорында сақталады.

Үшінші бөлімде осы дипломдық жобаның мобильді қосымшасының диаграммасы мен схемасын жасау және оның функционалдық қызметін көрсеттік.

Төртінші бөлімде біз осы ақпараттық жүйенің экономикалық тиімділігі жағын қарастырып есептедік. Сонымен ақпараттық жүйенің іске асыру құны 1 006 402 тенгеге тең болды. Ал экономикалық тиімдігі 45691 тенге, өтімділік мерзімі 1,25 жылға тең болды.

Бесінші бөлім өміртіршілік қауіпсіздік бөлімі. Бұл бөлім еңбек барысында керек нәрсе. Себебі бұл бөлімде программисттің жақсы жұмыс жасауына барлық қолайлы жағдайлар жасалынды.

Қорытындылай келе айтатын болсақ, бұл жобамның мақсаты мекемедегі кезектердің санын қысқарту, кәсіпкерлер мен тұтынушылардың жұмысын жеңілдету және жеделдету болып табылады. “Уақыт–алтынан да қымбат” деп бекер айтылмаған, әсіресе қазіргі уақытта, бұл бағдарлама кез келген адамның уақытын үнемдеуге мүмкіндік береді. Соңғы жылдары мобильді құрылғыларға сұраныс деңгейін сипаттайтын индикатор үнемі өсіп келеді. Мұндай статистикалық мәліметтер мобильді қосымшалардың дамуы өзекті және орынды деген қорытынды жасауға мүмкіндік береді. Ең бастысы, бағдарламалық жасақтама кім үшін және не үшін жасалатынын дұрыс бағалау болды.

Әдебиеттер тізімі

- 1 Маркин, А.В. РНР web-бағдарламалау негіздері: учеб. пособие / А.В. Маркин. – М.: Диалог-МИФИ, 2012 жыл.
- 2 Когтзолл Д. РНР5. Толық жетекшілік. – М.: Вильямс, 2010.
- 3 CSS туралы оқулық – Электронды оқулық:
<http://www.wisdomweb.ru/CSS>.
- 4 HTML туралы оқулық – Электронды оқулық: <https://html5book.ru/>
- 5 С.Е. Мананбаева, Т.С. Санатова, А.С. Бегимбетова, Н.С. Бекмуратова «БЕЗОПАСНОСТЬ ЖИЗНЕДЕЯТЕЛЬНОСТИ» Часть I «Производственная санитария» выполнение раздела БЖД в дипломных проектах Учебно-методическое пособие. Алматы 2016г.
- 6 Базылов К.Б., Алибаева С.А., Бабич А.А. «Методические указания для экономической части выпускной работы». АИЭС - 2008 г.
- 7 Абдимуратов Ж.С., Мананбаева С.Е. Безопасность жизнедеятельности. Методические указания к выполнению раздела «Расчет производственного освещения» в выпускных работах для всех специальностей. Бакалавриат - Алматы: «АИЭС», 2009. - 20 с.
- 8 Методические указания к выполнению экономической части дипломных работ для студентов специальности 5В070400 – Вычислительная техника и программное обеспечение/З.Д.Еркешева, Г.Ш.Боканова. – Алматы: АУЭС, 2013 – 40 с.
- 9 Свободная энциклопедия //WIKIPEDIA.ORG : информация об электронных учебниках. 2005. URL: <https://ru.wikipedia.org/wiki/>
- 10 РНР Объекты, шаблоны и методика программирования. Мэтт Зандстра. 4 издание. Онлайн книга.
- 11 Дженнифер Нидерст Роббинс "HTML5, CSS3 и JavaScript. Исчерпывающее руководство". 4-ое издание (2014).
- 12 Дэвид Макфарланд «Большая книга CSS3» (2014).

А қосымшасы

Бағдарламаның мәтіні

```
<ion-header>
  <ion-toolbar color="primary">

    <ion-buttons slot="start">
      <ion-menu-button></ion-menu-button>
    </ion-buttons>

    <ion-title>{{'COMMON.CATEGORIES' | translate}}</ion-title>

    <ion-buttons slot="end">
      <ion-button class="cartBtn" routerLink="/cart">
        <ion-badge color="danger">{{cartCount}}</ion-badge>
        <ion-icon slot="icon-only" name="cart"></ion-icon>
      </ion-button>
    </ion-buttons>

  </ion-toolbar>
</ion-header>

<ion-content>

  <ion-row>
    <ion-col size="6" *ngFor="let category of categories$ | async">
      <ion-card (click)="openSubCategory(category.id)">
        <img [src]="category.image" alt="">
        <ion-card-content>
          <ion-card-subtitle>{{category.name}}</ion-card-subtitle>
        </ion-card-content>
      </ion-card>
    </ion-col>
  </ion-row>

</ion-content>
```